

A CIP/multi-moment finite volume method for shallow water equations with source terms

R. Akoh¹, S. Ii¹ and F. Xiao^{1, 2, *, †}

¹*Department of Energy Sciences, Tokyo Institute of Technology, 4259 Nagatsuta, Midori-ku, Yokohama 226-8502, Japan*

²*LHD, Institute of Mechanics, Chinese Academy of Sciences, 15 Beisihuanxi Road, Beijing 100080, People's Republic of China*

SUMMARY

A novel finite volume method has been presented to solve the shallow water equations. In addition to the volume-integrated average (VIA) for each mesh cell, the surface-integrated average (SIA) is also treated as the model variable and is independently predicted. The numerical reconstruction is conducted based on both the VIA and the SIA. Different approaches are used to update VIA and SIA separately. The SIA is updated by a semi-Lagrangian scheme in terms of the Riemann invariants of the shallow water equations, while the VIA is computed by a flux-based finite volume formulation and is thus exactly conserved. Numerical oscillation can be effectively avoided through the use of a non-oscillatory interpolation function. The numerical formulations for both SIA and VIA moments maintain exactly the balance between the fluxes and the source terms. 1D and 2D numerical formulations are validated with numerical experiments. Copyright © 2007 John Wiley & Sons, Ltd.

Received 2 February 2007; Revised 15 August 2007; Accepted 16 August 2007

KEY WORDS: shallow water equations; source terms; finite volume method; high-order accuracy; CIP; multi-moment; method of characteristics

1. INTRODUCTION

The shallow water equations not only serve as a set of governing equations for large-scale water waves where the wave amplitude is quite small compared with the wavelength, but also provide a good mathematical model for nonlinear hyperbolic differential equations that may have solutions like shock wave and expansion fan. Numerical simulations for shallow water equations are of practical importance in hydraulic and coastal engineering.

*Correspondence to: F. Xiao, LHD, Institute of Mechanics, Chinese Academy of Sciences, 15 Beisihuanxi Road, Beijing 100080, People's Republic of China.

†E-mail: xiao@imech.ac.cn

A great number of studies on the numerical solutions to the shallow water equations are found in the literature. The nonlinear shock solutions in the shallow water equations bring about some extra difficulties in the construction of numerical methods. High-resolution schemes of either finite difference method (FDM) or finite volume method (FVM) with exact or approximate Riemann solvers prove very successful in solving the shallow water equations. Some representative numerical algorithms for shallow water equations are found in [1]. Implementations of some other more sophisticated numerical methods, e.g. discontinuous Galerkin (DG) [2] and spectral volume (SV) [3], are getting an increasing attention from researchers for both theoretical and practical interests.

We have recently developed a novel numerical framework, namely constrained interpolated profile/multi-moment finite volume method or shortly CIP/MM FVM, for fluid dynamics [4–10]. The underlying idea of a CIP/MM FVM is to utilize more than one type of moments of a physical field as the model variables. Different from the DG formulation, the moments in a CIP/MM FVM are defined straightforwardly as the volume-integrated averages (VIA) over a control volume, the surface-integrated averages (SIA) over a plane segment or the point value (PV) at any specified point. Different moments are temporally updated by different ways in CIP/MM FVM. For example, the conserved moment VIA is computed by a flux-based formulation, while the PV or the SIA, which needs not be exactly conserved, can be updated in time by a semi-Lagrangian method or an Eulerian method that may not necessarily be conservative. Thus, CIP/MM FVM provides a more flexible framework for building numerical models.

This paper presents an implementation of the CIP/MM FVM to the shallow water equations. Two types of moments, which are the VIA and the SIA, are used as the model variables. The spatial interpolation reconstruction is based on both VIA and SIA to get high accuracy with small grid stencil. SIA and VIA are separately advanced in time. Making use of the hyperbolicity of the shallow water equations, SIA is updated by a semi-Lagrangian solution along the characteristic lines, whereas VIA is advanced through a flux-form formulation. The numerical formulation is devised so that the source terms are well balanced to the fluxes on the discretized level for both VIA and SIA moments. The resulting numerical algorithm is accurate and robust. The numerical results are competitive to other existing numerical methods.

In Section 2, we briefly describe the CIP-CSL3 reconstruction [11] that uses two kinds of moments and effectively suppresses numerical oscillations. The CIP/multi-moment finite volume formulations for one and two dimensions are presented in Section 3 with details. Section 4 gives numerical results including some widely used benchmark tests where shock waves are involved to validate the numerical method. Finally, we end up with some concluding remarks in Section 5.

2. CIP-CSL METHOD

In this section, we introduce the basic idea of a type of advection schemes, namely the CIP-CSL methods [11–14], with some details of the CIP-CSL3 reconstruction that is employed in this paper.

The 1D computational domain along the x -axis is partitioned into mesh cells of finite width. The VIA and the SIA degenerate to the line-integrated average and the PV, respectively. Assuming that the i th cell is bounded by $[x_{i-1/2}, x_{i+1/2}]$, we defined the VIA of a field variable $\phi(x, t)$ over cell i as

$$\overline{V}\phi_i = \frac{1}{\Delta x_i} \int_{x_{i-1/2}}^{x_{i+1/2}} \phi(x, t) dx \quad (1)$$

with $\Delta x_i = x_{i+1/2} - x_{i-1/2}$. The SIA is correspondingly the PV at the cell boundary

$$\overline{P\phi}_{i+1/2} = \phi(x_{i+1/2}, t) \quad (2)$$

With one cell-integrated average ($\overline{V\phi}_i$) and two PVs at the cell boundaries ($\overline{P\phi}_{i-1/2}$, $\overline{P\phi}_{i+1/2}$) known, the CIP-CSL reconstructions can be made. To get rid of numerical oscillations associating the discontinuous solutions, like shock wave, we used CIP-CSL3 schemes to enforce the monotonicity.

2.1. CIP-CSL3

The CIP-CSL3 scheme makes use of a cubic interpolation function constructed over a single cell under the constraints of two PVs at the cell boundaries, VIA over the cell and the first-order derivative at the middle point of the cell. A cubic interpolation function then reads

$$\begin{aligned} \Phi_i(x) &= a_1(x - x_{i-1/2})^3 + a_2(x - x_{i-1/2})^2 + a_3(x - x_{i-1/2}) + \overline{P\phi}_{i-1/2} \\ a_1 &= \frac{4(\overline{P\phi}_{i+1/2} - \overline{P\phi}_{i-1/2} - \Delta x_i d_i)}{\Delta x_i^3} \\ a_2 &= \frac{3(-2\overline{V\phi}_i - \overline{P\phi}_{i+1/2} + 3\overline{P\phi}_{i-1/2} + 2\Delta x_i d_i)}{\Delta x_i^2} \\ a_3 &= \frac{2(3\overline{V\phi}_i - 3\overline{P\phi}_{i-1/2} - \Delta x_i d_i)}{\Delta x_i} \end{aligned} \quad (3)$$

where the first-order derivative d_i is not treated as a prognostic variable but approximated by other moments, i.e. PV and VIA. Some practical candidates of the approximations for d_i can be found in [11].

We approximate d_i in this paper using the formula in [15], which effectively eliminates numerical oscillations in the presence of discontinuities,

$$d_i^* = \begin{cases} \frac{\min(|\delta\phi_i|, 2|\phi_{i+1} - \phi_i|, 2|\phi_i - \phi_{i-1}|) \operatorname{sgn}(\delta\phi_i)}{\Delta x} & \text{if } (\phi_{i+1} - \phi_i)(\phi_i - \phi_{i-1}) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

where

$$\delta\phi_i = \frac{1}{2}(\phi_{i+1} - \phi_{i-1}) \quad (5)$$

and ϕ_i is the PV at the center of the i th cell obtained by (3). Moreover, we can modify the numerical solution by using

$$d_i = \beta d_i^* \quad (6)$$

with β being a slope modification parameter. It is clear that if β is bigger than 1, the numerical solution becomes sharper, while if β is smaller than 1, the numerical solution tends to be flattened (see Reference [5] for more details).

2.2. Update of the moments

The original CIP-CSL schemes were devised to solve the conservative transport equation as follows:

$$\frac{\partial \phi}{\partial t} + \frac{\partial(u\phi)}{\partial x} = 0 \quad (7)$$

where ϕ is the advected quantity and u is the velocity.

The conservative moment VIA is solved by a flux-form formulation derived straightforwardly from (7), while the non-conservative moment PV is computed by a semi-Lagrangian approach based on the advection form of the transport equation:

$$\frac{\partial \phi}{\partial t} + u \frac{\partial \phi}{\partial x} = -\phi \frac{\partial u}{\partial x} \quad (8)$$

In the semi-Lagrangian method, the departure point X for cell boundary $x_{i+1/2}$ is obtained by solving the following initial value problem:

$$\begin{aligned} \frac{dX}{dt'} &= -u(x, t^n) \\ X(t'=0) &= X_0 \equiv x_{i+1/2} \end{aligned} \quad (9)$$

In this paper, we use the third-order TVD Runge–Kutta method [16] to solve (9) as follows:

$$\begin{aligned} X_1 &= X_0 - u(X_0, t^n) \Delta t \\ X_2 &= \frac{3}{4} X_0 + \frac{1}{4} X_1 - \frac{1}{4} u(X_1, t^n) \Delta t \\ X_3 &= \frac{1}{3} X_0 + \frac{2}{3} X_2 - \frac{2}{3} u(X_2, t^n) \Delta t \end{aligned} \quad (10)$$

where X_l are the upwind points at each Runge–Kutta sub-step $l=0, 1, 2, 3$.

Therefore, the PV is updated by (8), (10) as

$$\overline{P\phi}_{i+1/2}^{(l)} = \Phi(X_l) - \int_{\tau} \phi \frac{\partial u}{\partial x} d\tau \quad (11)$$

where $l=1, 2, 3$ denotes the Runge–Kutta sub-step, Φ is the interpolation function in the cell including X_l , and τ represents the trajectory. The PV at the cell boundary of step $n+1$ is obtained as

$$\overline{P\phi}_{i+1/2}^{n+1} = \overline{P\phi}_{i+1/2}^{(3)} \quad (12)$$

The conservative moment VIA is updated by the following finite volume formulation in flux form:

$$\frac{\partial \overline{V\phi}_i}{\partial t} = -\frac{1}{\Delta x_i} (\overline{\mathcal{F}}_{i+1/2}(u, \phi) - \overline{\mathcal{F}}_{i-1/2}(u, \phi)) \quad (13)$$

where $\overline{\mathcal{F}}(u, \phi)$ denotes the numerical flux.

In this paper, we approximate the numerical flux by

$$\overline{\mathcal{F}}_{i+1/2}(u, \phi) \approx \hat{\mathcal{F}}_{i+1/2} = \frac{\mathcal{F}(u, \overline{P\phi}_{i+1/2}^{(0)}) + \mathcal{F}(u, \overline{P\phi}_{i+1/2}^{(1)}) + 4\mathcal{F}(u, \overline{P\phi}_{i+1/2}^{(2)})}{6} \quad (14)$$

with the third-order Runge–Kutta method. Finally, the VIA is predicted by

$$\overline{v\phi}_i^{n+1} = \overline{v\phi}_i^n - \frac{\Delta t}{\Delta x_i} (\hat{\mathcal{F}}_{i+1/2} - \hat{\mathcal{F}}_{i-1/2}) \quad (15)$$

It is noted that we compute the numerical flux $\mathcal{F}(u, \phi)$ by averaging the PVs sampled at the Runge–Kutta sub-steps rather than integrating the interpolation function as in the original CIP-CSL schemes.

3. CIP/MM FVM FOR SHALLOW WATER EQUATIONS

We consider the shallow water equations with source terms:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} = \mathbf{S} \quad (16)$$

with

$$\mathbf{U} = \begin{bmatrix} h \\ hu \\ hv \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \\ huv \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} hv \\ huv \\ hv^2 + \frac{1}{2}gh^2 \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} 0 \\ S_x \\ S_y \end{bmatrix} \quad (17)$$

where h is the water depth, u and v are the x and y components of flow velocity, respectively, and g is the gravitational acceleration. \mathbf{S} is the source term including the bottom effects of topography.

In this section, we firstly describe the numerical procedure of the CIP/MM FVM for the shallow water equations without source term and then present a well-balanced formulation including the source terms in one dimension. The 2D scheme is constructed by simply implementing the 1D formulation with a splitting.

3.1. Formulation for 1D shallow water equations without the source term

Without the source term, the 1D shallow water equations read as

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = 0 \quad (18)$$

with

$$\mathbf{U} = \begin{bmatrix} h \\ hu \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \end{bmatrix} \quad (19)$$

being the conservative variables and the fluxes, respectively.

In the 1D case, the PVs of the primitive variables h and u at each cell boundary and VIAs of the conservative variables h and uh over each mesh cell are defined as the moments and hence predicted variables.

We at first describe how to update the PVs at the cell boundary. Equation (18) can be recast in its non-conservative form with the primitive variables \mathbf{W} as

$$\frac{\partial \mathbf{W}}{\partial t} + \mathbf{A} \frac{\partial \mathbf{W}}{\partial x} = 0, \quad \mathbf{W} = \begin{bmatrix} h \\ u \end{bmatrix} \quad (20)$$

and \mathbf{A} is the Jacobian matrix of the flux function and defined as

$$\mathbf{A} = \frac{\partial \mathbf{F}}{\partial \mathbf{W}} = \begin{bmatrix} u & h \\ g & u \end{bmatrix} \quad (21)$$

Owing to the hyperbolicity, the characteristic form of the shallow water equations can be derived as

$$\mathbf{L} \frac{\partial \mathbf{W}}{\partial t} + (\mathbf{L} \mathbf{A} \mathbf{R}) \mathbf{L} \frac{\partial \mathbf{W}}{\partial x} = 0 \quad (22)$$

where the matrix of the left eigenvectors \mathbf{L} and the right eigenvectors \mathbf{R} are

$$\mathbf{L} = -\frac{1}{2ch} \begin{bmatrix} -c & -h \\ -c & h \end{bmatrix}, \quad \mathbf{R} = \mathbf{L}^{-1} = \begin{bmatrix} h & h \\ c & -c \end{bmatrix} \quad (23)$$

with $c = \sqrt{gh}$ being the speed of the gravity wave. The diagonalized matrix is made up of the two characteristic speeds as

$$\mathbf{\Lambda} = \mathbf{L} \mathbf{A} \mathbf{R} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} = \begin{bmatrix} u+c & 0 \\ 0 & u-c \end{bmatrix} \quad (24)$$

Consequently, the following set of relations of the characteristic variables (or the Riemann invariants) can be derived:

$$\begin{aligned} \frac{g}{c} dh + du = 0 \quad \text{on } \mathcal{C}_1(X_0): \quad \frac{dx}{dt'} = \lambda_1 = u+c, \quad x(t'=0) = X_0 \\ \frac{g}{c} dh - du = 0 \quad \text{on } \mathcal{C}_2(X_0): \quad \frac{dx}{dt'} = \lambda_2 = u-c, \quad x(t'=0) = X_0 \end{aligned} \quad (25)$$

where $\mathcal{C}_m(X_0)$ denote the characteristic curve corresponding to characteristic speed λ_m and passing through cell boundary point X_0 .

Hence, the primitive variables at X_0 can be found by the following relations:

$$\begin{aligned} \frac{g}{c}(h^* - h^{(1)}) + (u^* - u^{(1)}) &= 0 \\ \frac{g}{c}(h^* - h^{(2)}) - (u^* - u^{(2)}) &= 0 \end{aligned} \quad (26)$$

with

$$\begin{aligned} \phi^* &= \phi(X_0) \\ \phi^{(m)} &= \phi(X(\mathcal{C}_m)) \end{aligned} \quad \text{for } m = 1, 2 \quad (27)$$

where $X(\mathcal{C}_1)$ and $X(\mathcal{C}_2)$ indicate the points on the characteristic curves \mathcal{C}_1 and \mathcal{C}_2 , respectively. Consequently, we have

$$\begin{aligned} h^* &= \frac{1}{2} \left[h^{(1)} + h^{(2)} + \frac{c}{g} (u^{(1)} - u^{(2)}) \right] \\ u^* &= \frac{1}{2} \left[u^{(1)} + u^{(2)} - \frac{g}{c} (h^{(1)} - h^{(2)}) \right] \end{aligned} \quad (28)$$

The upwind points $X(\mathcal{C}_m)$ of cell boundary $x_{i+1/2}$ for characteristic lines \mathcal{C}_m are obtained by solving the following initial value problem for the characteristic speeds λ_m :

$$\begin{aligned} \frac{dX}{dt'} \Big|_{i+1/2} &= -\lambda_m(x, t) \quad \text{for } m = 1, 2 \\ X(t'=0) &= X_0 \equiv x_{i+1/2} \end{aligned} \quad (29)$$

We use the third-order Runge–Kutta method given in the previous section:

$$\begin{aligned} X_1(\mathcal{C}_m) &= X_0 - \lambda_m(X_0, t^n) \Delta t \\ X_2(\mathcal{C}_m) &= \frac{3}{4} X_0 + \frac{1}{4} X_1(\mathcal{C}_m) - \frac{1}{4} \lambda_m(X_1, t^n) \Delta t \quad \text{for } m = 1, 2 \\ X_3(\mathcal{C}_m) &= \frac{1}{3} X_0 + \frac{2}{3} X_2(\mathcal{C}_m) - \frac{2}{3} \lambda_m(X_2, t^n) \Delta t \end{aligned} \quad (30)$$

Given both the PVs and the VIAs of the primitive variables $\overline{P}h_{i+1/2}^n$, $\overline{P}u_{i+1/2}^n$, $\overline{V}h_i^n$ and $\overline{V}u_i^n$ for all i at step n ($t=t^n$), the CIP-CSL3 reconstruction of the primitive variables can be made. We solve $\overline{P}h_{i+1/2}^{(l)}$ and $\overline{P}u_{i+1/2}^{(l)}$ for each Runge–Kutta sub-step ($l=1, 2, 3$), using the semi-Lagrangian solutions as

$$\overline{P}h_{i+1/2}^{(l)} = \frac{1}{2} \left[\mathcal{H}(X_l(\mathcal{C}_1)) + \mathcal{H}(X_l(\mathcal{C}_2)) + \frac{c}{g} \{ \mathcal{U}(X_l(\mathcal{C}_1)) - \mathcal{U}(X_l(\mathcal{C}_2)) \} \right] \quad (31)$$

$$\overline{P}u_{i+1/2}^{(l)} = \frac{1}{2} \left[\mathcal{U}(X_l(\mathcal{C}_1)) + \mathcal{U}(X_l(\mathcal{C}_2)) + \frac{g}{c} \{ \mathcal{H}(X_l(\mathcal{C}_1)) - \mathcal{H}(X_l(\mathcal{C}_2)) \} \right] \quad (32)$$

where \mathcal{H} and \mathcal{U} represent the CIP-CSL3 reconstruction interpolation functions for h and u , respectively.

The PVs of the primitive variables for step $n+1$ are found directly using

$$\begin{aligned} \overline{P}h_{i+1/2}^{n+1} &= \overline{P}h_{i+1/2}^{(3)} \\ \overline{P}u_{i+1/2}^{n+1} &= \overline{P}u_{i+1/2}^{(3)} \end{aligned} \quad (33)$$

Concerning the updating of the cell-integrated average values of the conservative variables, we consider the following finite volume formulation in flux form:

$$\frac{\partial \overline{\mathbf{U}}_i}{\partial t} = -\frac{1}{\Delta x_i} (\mathcal{F}_{i+1/2}(h, u) - \mathcal{F}_{i-1/2}(h, u)) \quad (34)$$

where $\mathcal{F}(h, u)$ denotes the numerical flux.

It is obvious that (34) exactly guarantees the conservation of $\overline{V\mathbf{U}}$. Similar to the scalar conservative transport, the numerical flux in (34) at each cell boundary is approximated as

$$\begin{aligned} \mathcal{F}_{i+1/2}(h, u) &\approx \hat{\mathcal{F}}_{i+1/2} \\ &= \frac{\mathcal{F}(\overline{P}h_{i+1/2}^{(0)}, \overline{P}u_{i+1/2}^{(0)}) + \mathcal{F}(\overline{P}h_{i+1/2}^{(1)}, \overline{P}u_{i+1/2}^{(1)}) + 4\mathcal{F}(\overline{P}h_{i+1/2}^{(2)}, \overline{P}u_{i+1/2}^{(2)})}{6} \end{aligned} \quad (35)$$

with the PVs sampled at different sub-steps of the third-order Runge–Kutta method. Finally, the VIAs of the conservative variables are updated by

$$\overline{V\mathbf{U}}_i^{n+1} = \overline{V\mathbf{U}}_i^n - \frac{\Delta t}{\Delta x_i} (\hat{\mathcal{F}}_{i+1/2} - \hat{\mathcal{F}}_{i-1/2}) \quad (36)$$

3.2. Formulation for 1D shallow water equations with the source term

We include the effect of the bottom topography as the source term in the 1D shallow water equations:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = \mathbf{S} \quad (37)$$

with the source term expressed as

$$\mathbf{S} = \begin{bmatrix} S^{(1)} \\ S^{(2)} \end{bmatrix} = \begin{bmatrix} 0 \\ -gh \frac{\partial z}{\partial x} \end{bmatrix} \quad (38)$$

where z represents the bottom topography.

A numerical scheme in this case is required to give numerically the exact balance between the flux and the source term. It simply means that the numerical solutions should remain stationary if one starts from a balanced still state

$$H(x, t) = h(x, t) + z(x) = \text{constant}, \quad u(x, t) = 0 \quad (39)$$

This is the so-called C-property [17] which states that the fundamental balance between source term and flux gradient

$$\frac{\partial}{\partial x} \left(\frac{1}{2} gh^2 \right) = -gh \frac{\partial z}{\partial x} \quad (40)$$

should be satisfied on a discretized level.

In a CIP/MM FVM, we have to update the PVs and VIAs of the physical variables. Next we present the numerical formulations that assure the numerical solutions for both the PV and the VIA moments satisfying the C-property.

We first describe the updating of PVs at the cell boundary. Just like the previous section, Equation (37) can be recast in its non-conservative form of the primitive variables \mathbf{W} as

$$\frac{\partial \mathbf{W}}{\partial t} + \mathbf{A} \frac{\partial \mathbf{W}}{\partial x} = \mathbf{C} \quad (41)$$

where \mathbf{W} and \mathbf{A} are the vector of primitive variables and the Jacobian matrix given before, and \mathbf{C} are the reduced source terms

$$\mathbf{C} = \begin{bmatrix} 0 \\ -g \frac{\partial z}{\partial x} \end{bmatrix} \quad (42)$$

In an analogous way, the characteristic form can be derived by multiplying (41) with the matrix of the left eigen vectors,

$$\mathbf{L} \frac{\partial \mathbf{W}}{\partial t} + \mathbf{A} \mathbf{L} \frac{\partial \mathbf{W}}{\partial x} = \mathbf{L} \mathbf{C} \quad (43)$$

From (43) we obtain the following set of equations for the Riemann invariants:

$$\begin{aligned} \frac{g}{c} dh + du &= -g \int_{\tau_1} \frac{\partial z}{\partial x} d\tau \quad \text{on } \mathcal{C}_1(X_0): \quad \frac{dx}{dt'} = \lambda^{(1)} = u + c, \quad x(t'=0) = X_0 \\ \frac{g}{c} dh - du &= g \int_{\tau_2} \frac{\partial z}{\partial x} d\tau \quad \text{on } \mathcal{C}_2(X_0): \quad \frac{dx}{dt'} = \lambda^{(2)} = u - c, \quad x(t'=0) = X_0 \end{aligned} \quad (44)$$

where τ_1 and τ_2 denote the line segments along the characteristic curves $\mathcal{C}_1(X_0)$ and $\mathcal{C}_2(X_0)$, respectively. We approximate these terms by

$$\begin{aligned} \int_{\tau_1} \frac{\partial z}{\partial x} d\tau &\approx -\frac{1}{u+c} dz \\ \int_{\tau_2} \frac{\partial z}{\partial x} d\tau &\approx -\frac{1}{u-c} dz \end{aligned} \quad (45)$$

The primitive variables at X_0 can then be evaluated as follows:

$$h^* = \frac{1}{2} \left\{ h^{(1)} + h^{(2)} + \frac{c}{g} (u^{(1)} - u^{(2)}) - \frac{c}{u+c} (z^* - z^{(1)}) + \frac{c}{u-c} (z^* - z^{(2)}) \right\} \quad (46)$$

$$u^* = \frac{1}{2} \left\{ u^{(1)} + u^{(2)} - \frac{g}{c} (h^{(1)} - h^{(2)}) - \frac{g}{u+c} (z^* - z^{(1)}) + \frac{g}{u-c} (z^* - z^{(2)}) \right\} \quad (47)$$

In order to enforce the C-property, we introduce the total elevation of the water surface $H = h + z$ and re-write (46), (47) into

$$h^* = \frac{1}{2} \left\{ \frac{c}{g} (u^{(1)} - u^{(2)}) + \frac{1}{u+c} (cH^{(1)} + uh^{(1)} - cz^*) - \frac{1}{u-c} (cH^{(2)} - uh^{(2)} - cz^*) \right\} \quad (48)$$

$$u^* = \frac{1}{2} \left\{ u^{(1)} + u^{(2)} - \frac{g}{c(u+c)} (cH^{(1)} + uh^{(1)} - cz^*) - \frac{g}{c(u-c)} (cH^{(2)} - uh^{(2)} - cz^*) \right\} \quad (49)$$

where $H^{(1)} = h^{(1)} + z^{(1)}$ and $H^{(2)} = h^{(2)} + z^{(2)}$. It is straightforward to show that (48) and (49) satisfy the C-property [18].

Similarly, after the upstream departure points for each characteristic lines are obtained by the third-order Runge–Kutta method (30), the PVs of the primitive variables are computed by the semi-Lagrangian solutions as

$$\begin{aligned} \overline{P}h_{i+1/2}^{(l)} = & \frac{1}{2} \left[\frac{c}{g} \{ \mathcal{U}(X_I(\mathcal{C}_1)) - \mathcal{U}(X_I(\mathcal{C}_2)) \} \right. \\ & + \frac{1}{u+c} \{ c\mathcal{F}\mathcal{H}(X_I(\mathcal{C}_1)) + u\mathcal{H}(X_I(\mathcal{C}_1)) - cz_{i+1/2} \} \\ & \left. - \frac{1}{u-c} \{ c\mathcal{F}\mathcal{H}(X_I(\mathcal{C}_2)) - u\mathcal{H}(X_I(\mathcal{C}_2)) - cz_{i+1/2} \} \right] \end{aligned} \tag{50}$$

$$\begin{aligned} \overline{P}u_{i+1/2}^{(l)} = & \frac{1}{2} \left[\mathcal{U}(X_I(\mathcal{C}_1)) + \mathcal{U}(X_I(\mathcal{C}_2)) \right. \\ & + \frac{g}{c(u+c)} \{ c\mathcal{F}\mathcal{H}(X_I(\mathcal{C}_1)) + u\mathcal{H}(X_I(\mathcal{C}_1)) - cz_{i+1/2} \} \\ & \left. + \frac{g}{c(u-c)} \{ c\mathcal{F}\mathcal{H}(X_I(\mathcal{C}_2)) - u\mathcal{H}(X_I(\mathcal{C}_2)) - cz_{i+1/2} \} \right] \end{aligned} \tag{51}$$

where $\mathcal{F}\mathcal{H}$ denotes the interpolation function for the total elevation H . For the third-order Runge–Kutta time integration scheme, we finally have the PVs at the new time step as

$$\overline{P}h_{i+1/2}^{n+1} = \overline{P}h_{i+1/2}^{(3)} \quad \text{and} \quad \overline{P}u_{i+1/2}^{n+1} = \overline{P}u_{i+1/2}^{(3)} \tag{52}$$

Concerning the updating of VIA, we consider the semi-discretization formulation of the balance law as follows:

$$\overline{V}\mathbf{U}_i^{n+1} = \overline{V}\mathbf{U}_i^n - \frac{\Delta t}{\Delta x_i} (\hat{\mathcal{F}}_{i+1/2} - \hat{\mathcal{F}}_{i-1/2}) + \frac{\Delta t}{\Delta x_i} \hat{\mathcal{S}}_i \tag{53}$$

where

$$\hat{\mathcal{S}}_i = \begin{bmatrix} \hat{\mathcal{S}}_i^{(1)} \\ \hat{\mathcal{S}}_i^{(2)} \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} \int_{x_{i-1/2}}^{x_{i+1/2}} \left(-gh \frac{\partial z}{\partial x} \right) dx dt \end{bmatrix} \tag{54}$$

In order to maintain the C-property of the resulting numerical scheme, the source term $S^{(2)}$ is split into the following two terms [19]:

$$S^{(2)} = -gh \frac{\partial z}{\partial x} = -g(h+z) \frac{\partial z}{\partial x} + \frac{1}{2}g \frac{\partial(z^2)}{\partial x} \tag{55}$$

The second element in (54) is then expressed as

$$\begin{aligned} \hat{\mathcal{G}}_i^{(2)} &= \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} \int_{x_{i-1/2}}^{x_{i+1/2}} \left(-g(h+z) \frac{\partial z}{\partial x} + \frac{1}{2} g \frac{\partial(z^2)}{\partial x} \right) dx \\ &= -\frac{g}{2} (\hat{H}_{i+1/2} + \hat{H}_{i-1/2})(z_{i+1/2} - z_{i-1/2}) + \frac{g}{2} ((z_{i+1/2})^2 - (z_{i-1/2})^2) + \hat{\mathcal{G}}_i \\ &= -\frac{g}{2} (\hat{h}_{i+1/2} + \hat{h}_{i-1/2})(z_{i+1/2} - z_{i-1/2}) + \mathcal{G}_i \end{aligned} \tag{56}$$

where $\hat{\phi}$ denotes a weighted averaging of the values of ϕ at the Runge–Kutta sub-steps:

$$\hat{\phi}_{i+1/2} = \frac{\overline{P}\phi_{i+1/2}^{(0)} + \overline{P}\phi_{i+1/2}^{(1)} + 4\overline{P}\phi_{i+1/2}^{(2)}}{6} \tag{57}$$

and

$$\mathcal{G}_i = -g \int_{x_{i-1/2}}^{x_{i+1/2}} \left\{ \left[\frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} H(x, t) dt - \frac{1}{2} (\hat{H}_{i+1/2} + \hat{H}_{i-1/2}) \right] \frac{\partial z}{\partial x} \right\} dx \tag{58}$$

In (58), integral $\int_{t^n}^{t^{n+1}} \int_{x_{i-1/2}}^{x_{i+1/2}} H(x, t) (\partial z / \partial x) dt dx$ is evaluated by a fourth order Gauss quadrature with the CIP-CSL3 reconstruction $\mathcal{T}\mathcal{H}(x)$ based on the corresponding moments at step $n + 1$. It is clear that \mathcal{G}_i equals zero when the total height H is constant. A simpler approximation

$$\int_{t^n}^{t^{n+1}} \int_{x_{i-1/2}}^{x_{i+1/2}} H(x, t) \frac{\partial z}{\partial x} dt dx \approx \int_{x_{i-1/2}}^{x_{i+1/2}} \frac{1}{2} \left((\hat{H}_{i+1/2} + \hat{H}_{i-1/2}) \frac{\partial z}{\partial x} \right) dx$$

directly vanishes \mathcal{G} if $H = \text{constant}$.

Consequently, we have, for $H = \text{constant}$, the exactly balanced formulation as follows:

$$\hat{\mathcal{F}}_{i+1/2}^{(2)} - \hat{\mathcal{F}}_{i-1/2}^{(2)} + \hat{\mathcal{S}}_i^{(2)} = -\frac{g}{2} (\hat{h}_{i+1/2} + \hat{h}_{i-1/2})(\hat{H}_{i+1/2} - \hat{H}_{i-1/2}) = 0 \tag{59}$$

This again states the C-property for the finite volume formulation used to update the VIAs.

3.3. Formulation for 2D shallow water equations

Extending the previous procedure to two dimensions can be simply implemented by dimensional splitting.

The 2D version of the shallow water equations without the source terms in conservative form reads

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}^x}{\partial x} + \frac{\partial \mathbf{F}^y}{\partial y} = 0 \tag{60}$$

where

$$\mathbf{U} = \begin{bmatrix} h \\ hu \\ hv \end{bmatrix}, \quad \mathbf{F}^x = \begin{bmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \\ huv \end{bmatrix}, \quad \mathbf{F}^y = \begin{bmatrix} hv \\ huv \\ hv^2 + \frac{1}{2}gh^2 \end{bmatrix} \quad (61)$$

The quasi-linear form of the shallow water equations of primitive variables is expressed as

$$\frac{\partial \mathbf{W}}{\partial t} + \mathbf{A} \frac{\partial \mathbf{W}}{\partial x} + \mathbf{B} \frac{\partial \mathbf{W}}{\partial y} = 0 \quad (62)$$

where

$$\mathbf{W} = \begin{bmatrix} h \\ u \\ v \end{bmatrix}, \quad \mathbf{A} = \frac{\partial \mathbf{F}^x}{\partial \mathbf{W}} = \begin{bmatrix} u & g & 0 \\ h & u & 0 \\ 0 & v & 0 \end{bmatrix}, \quad \mathbf{B} = \frac{\partial \mathbf{F}^y}{\partial \mathbf{W}} = \begin{bmatrix} v & 0 & h \\ 0 & 0 & u \\ g & 0 & v \end{bmatrix} \quad (63)$$

Introducing an arbitrary unit vector $\mathbf{n} = (n_x, n_y)$, we project the Jacobian matrix onto \mathbf{n} as

$$\mathbf{C} = \mathbf{A}n_x + \mathbf{B}n_y = \begin{bmatrix} un_x + vn_y & gn_x & hn_y \\ hn_x & un_x & un_y \\ gn_y & vn_y & vn_y \end{bmatrix} \quad (64)$$

The eigenvalues of \mathbf{C} are

$$\begin{aligned} \lambda_1 &= un_x + vn_y + c \\ \lambda_2 &= un_x + vn_y \\ \lambda_3 &= un_x + vn_y - c \end{aligned} \quad (65)$$

The left eigen matrices are correspondingly

$$\mathbf{L} = -\frac{1}{2ch} \begin{bmatrix} -c & -hn_x & -hn_y \\ 0 & n_y & n_x \\ -c & hn_x & hn_y \end{bmatrix} \quad (66)$$

Letting n_y and n_x be alternatively zero, we decompose the solution procedure into sequential steps and solve

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}^\beta}{\partial \beta} = 0 \quad (67)$$

$$\mathbf{L}^\beta \frac{\partial \mathbf{W}}{\partial t} + \mathbf{\Lambda}^\beta \mathbf{L}^\beta \frac{\partial \mathbf{W}}{\partial \beta} = 0 \quad (68)$$

with β separately being x and y . In (68), the matrix of the eigenvalues

$$\Lambda^\beta = \begin{bmatrix} \lambda_1^\beta & 0 & 0 \\ 0 & \lambda_2^\beta & 0 \\ 0 & 0 & \lambda_3^\beta \end{bmatrix} \quad (69)$$

are computed by

$$\Lambda^\beta = \mathbf{L}^\beta \mathbf{U} \mathbf{R}^\beta \quad (70)$$

with

$$\mathbf{L}^x = \mathbf{L}(n_y=0), \quad \mathbf{L}^y = \mathbf{L}(n_x=0) \quad (71)$$

We denote SIAs and VIA of any physical variable $\phi(x, y, t)$ by $\overline{S^x \phi}$, $\overline{S^y \phi}$ and $\overline{V \phi}$ in a 2D Cartesian coordinate (x, y) .

In a Cartesian coordinate, the control volume is defined by a volume element (VE) and four surface elements (SE) as

$$\begin{aligned} V_{ij} &= [x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}] \\ S_{i\pm 1/2j}^x &= x_{i\pm 1/2} \cap [y_{j-1/2}, y_{j+1/2}] \\ S_{ij\pm 1/2}^y &= y_{j\pm 1/2} \cap [x_{i-1/2}, x_{i+1/2}] \end{aligned} \quad (72)$$

The volume of the VE and the areas of the SEs are consequently

$$\begin{aligned} |V_{ij}| &= \Delta x_i \Delta y_j \\ |S_{i\pm 1/2j}^x| &= \Delta y_j \\ |S_{ij\pm 1/2}^y| &= \Delta x_i \end{aligned} \quad (73)$$

with $\Delta x_i = x_{i+1/2} - x_{i-1/2}$ and $\Delta y_j = y_{j+1/2} - y_{j-1/2}$. The SIAs and the VIA of any physical variable $\phi(x, y, t)$ are, respectively, defined on $S_{i+1/2j}^x$, $S_{ij+1/2}^y$ and V_{ij} for all computational cells by

$$\overline{S^x \phi}_{i+1/2j} = \frac{1}{|S_{i+1/2j}^x|} \int_{y_{j-1/2}}^{y_{j+1/2}} \phi(x_{i+1/2}, y, t) dy \quad (74)$$

$$\overline{S^y \phi}_{ij+1/2} = \frac{1}{|S_{ij+1/2}^y|} \int_{x_{i-1/2}}^{x_{i+1/2}} \phi(x, y_{j+1/2}, t) dx \quad (75)$$

and

$$\overline{V \phi}_{ij} = \frac{1}{|V_{ij}|} \int_{x_{i-1/2}}^{x_{i+1/2}} \int_{y_{j-1/2}}^{y_{j+1/2}} \phi(x, y, t) dx dy \quad (76)$$

Making use of the dimensional splitting, we solve the 1D problem in the x direction with the following Riemann invariant solutions derived from (68):

$$\begin{aligned} \frac{g}{c} dh + du = 0 \quad \text{on } \mathcal{C}_1(X_0): \frac{dx}{dt'} = \lambda_1^x = u + c, \quad x(t'=0) = X_0 \\ dv = 0 \quad \text{on } \mathcal{C}_2(X_0): \frac{dx}{dt'} = \lambda_2^x = u, \quad x(t'=0) = X_0 \\ \frac{g}{c} dh - du = 0 \quad \text{on } \mathcal{C}_3(X_0): \frac{dx}{dt'} = \lambda_3^x = u - c, \quad x(t'=0) = X_0 \end{aligned} \tag{77}$$

Thus, the primitive variables at X_0 can be found by the following relations:

$$\begin{aligned} h^* &= \frac{1}{2} \left[h^{(1)} + h^{(3)} + \frac{c}{g} (u^{(1)} - u^{(3)}) \right] \\ u^* &= \frac{1}{2} \left[u^{(1)} + u^{(3)} - \frac{g}{c} (h^{(1)} - h^{(3)}) \right] \\ v^* &= v^{(2)} \end{aligned} \tag{78}$$

We compute the SIAs of the primitive variables at cell boundary, $\overline{S^x h}_{i+1/2j}^{(l)}$, $\overline{S^x u}_{i+1/2j}^{(l)}$ and $\overline{S^x v}_{i+1/2j}^{(l)}$, for each Runge–Kutta sub-step ($l = 1, 2, 3$), using the semi-Lagrangian solutions as

$$\overline{S^x h}_{i+1/2j}^{(l)} = \frac{1}{2} \left[\mathcal{H}(X_l(\mathcal{C}_1)) + \mathcal{H}(X_l(\mathcal{C}_3)) + \frac{c}{g} \{ \mathcal{U}(X_l(\mathcal{C}_1)) - \mathcal{U}(X_l(\mathcal{C}_3)) \} \right] \tag{79}$$

$$\overline{S^x u}_{i+1/2j}^{(l)} = \frac{1}{2} \left[\mathcal{U}(X_l(\mathcal{C}_1)) + \mathcal{U}(X_l(\mathcal{C}_3)) + \frac{g}{c} \{ \mathcal{H}(X_l(\mathcal{C}_1)) - \mathcal{H}(X_l(\mathcal{C}_3)) \} \right] \tag{80}$$

$$\overline{S^x v}_{i+1/2j}^{(l)} = \mathcal{V}(X_l(\mathcal{C}_2)) \tag{81}$$

where \mathcal{H} , \mathcal{U} and \mathcal{V} are the CIP-CSL3 reconstruction functions for u , h and v , respectively. The departure points $X_l(\mathcal{C}_m)$ in the characteristic fields \mathcal{C}_m are found from an initial value problem similar to (29) but with the characteristic speeds $\lambda_x^{(1)}$, $\lambda_x^{(2)}$ and $\lambda_x^{(3)}$ given above. The third-order Runge–Kutta method (30) is again used here.

The intermediate SIAs of the primitive variables (superscripted by ‘ \diamond ’) after updating in the x direction are computed directly using

$$\overline{S^x \mathbf{W}}_{i+1/2j}^{\diamond} = \begin{bmatrix} \overline{S^x h}_{i+1/2j}^{\diamond} \\ \overline{S^x u}_{i+1/2j}^{\diamond} \\ \overline{S^x v}_{i+1/2j}^{\diamond} \end{bmatrix} = \begin{bmatrix} \overline{S^x h}_{i+1/2j}^{(3)} \\ \overline{S^x u}_{i+1/2j}^{(3)} \\ \overline{S^x v}_{i+1/2j}^{(3)} \end{bmatrix} \tag{82}$$

The conservative moments $\overline{V\mathbf{U}}$ are updated in the x direction by the flux-form formulation as

$$\overline{V\mathbf{U}}_{ij}^{\diamond} = \overline{V\mathbf{U}}_{ij}^n - \frac{\Delta t}{\Delta x_i} (\hat{\mathcal{F}}_{xi+1/2j} - \hat{\mathcal{F}}_{xi-1/2j}) \tag{83}$$

with the numerical fluxes in (83) approximated by averaging the SIAs at the Runge–Kutta sub-step obtained from (79)–(81):

$$\hat{\mathcal{F}}_{xi+1/2j} = \frac{\mathcal{F}_x(\overline{S^x \mathbf{U}}_{i+1/2j}^{(0)}) + \mathcal{F}_x(\overline{S^x \mathbf{U}}_{i+1/2j}^{(1)}) + 4\mathcal{F}_x(\overline{S^x \mathbf{U}}_{i+1/2j}^{(2)})}{6} \quad (84)$$

Above process applies to the computations in the y direction by starting with the intermediate values of SIA and VIA moments, $\overline{S^x \mathbf{U}}^\diamond$ and $\overline{V \mathbf{U}}^\diamond$, obtained from the sweep in the x direction.

The dimensional splitting in the 2D formulation requires a so-called time evolution converting (TEC) formula to update the SIAs which are not computed in the 1D algorithm in the x - or y directions, respectively. For example, $\overline{S^x \phi}$ is updated from the time change of $\overline{V \phi}$ due to the 1D computation in the y direction, while $\overline{S^y \phi}$ is updated in the same way with the time change of $\overline{V \phi}$ due to the 1D computation in the x direction. One may refer to [7, 8, 14] for details.

When the source term is included, the dimensional splitting with the 1D building block presented in Section 3.2 can be implemented in an analogous manner. It is obvious that one can obtain the C-property in 2D through a splitting only if the 1D formulation possesses the C-property.

Shown above, the splitting technique provides an easier way to make use of the formulations developed in one dimension, like the limiting projection in the CIP-CSL3 reconstruction and the numerical formulation for the C-property which appear more complicated in a fully multi-dimensional framework. The numerical results given in the next section show that the splitting formulation used in the present paper is able to produce competitive numerical accuracy. When implemented on unstructured grid, however, a multi-dimensional reconstruction is required. We should note that the dimensional splitting technique is not essential to extend a multi-moment formulation to multi-dimensions. As a matter of fact, some formulations based on unstructured triangular mesh have also been developed in our group and will be reported separately.

4. NUMERICAL RESULTS

In this section, we present numerical results for several typical test problems to verify the presented numerical methods.

4.1. 1D dam breaking

In the first numerical test [20], the 1D dam-breaking problem is solved in a domain of $0 \leq x \leq 200$. The mesh number is 201 and the time step is 0.1 s. The initial conditions are

$$h_0(x) = \begin{cases} h_L(\text{m}), & 0.0 \leq x \leq 100.0 \\ h_R(\text{m}), & 100.0 < x \leq 200.0 \end{cases} \quad (85)$$

$$u_0(x) = 0.0$$

Two ratios of the initial water depths are separately specified as $h_L/h_R = 10$ (mild shock) and $h_L/h_R = 100$ (strong shock). In both the cases, a shock wave moving right and a rarefaction fan extending left were generated after the instantaneous collapse of the dam. We computed the numerical solutions until $t = 20.0$ s.

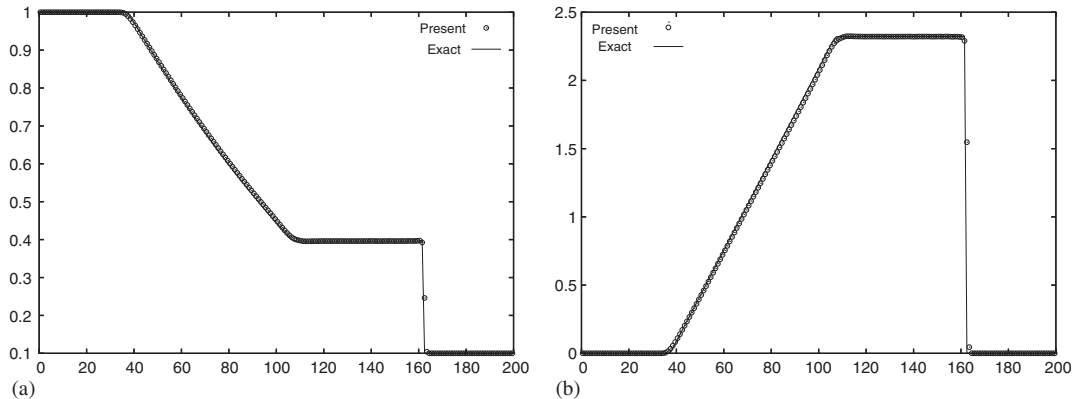


Figure 1. Numerical results (mild shock) of water depth (a) and velocity (b) at $t=20.0$ s by the present method (circles) and exact solution (solid line).

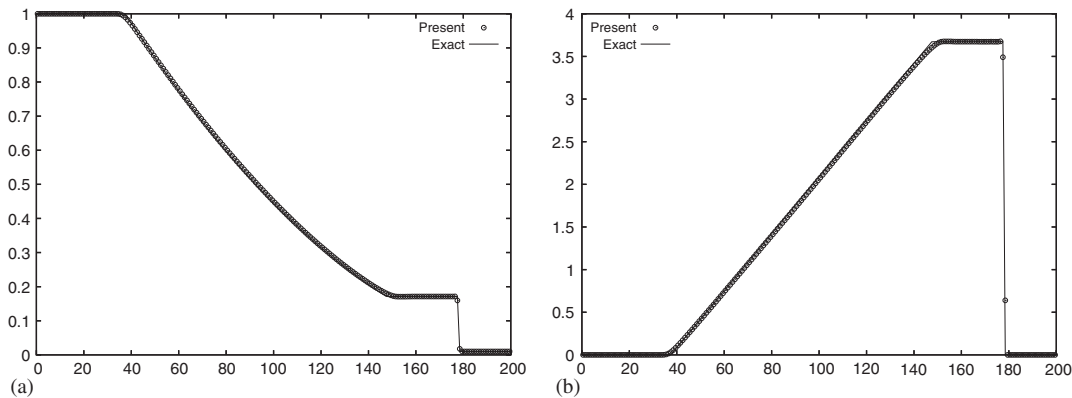


Figure 2. Same as Figure 1, but for the strong shock problem.

The simulated water depths h and the velocity u are depicted against the exact solutions in Figures 1 and 2. Both the shock wave and the rarefaction wave are accurately resolved. The conservativeness of the VIA quantities guarantees the correct location of the shock wave. The CIP-CSL3 reconstruction eliminates the spurious oscillation and the numerical diffusion is effectively controlled.

Before dealing with more numerical examples, we make a few remarks on the computational cost of the current scheme. In a multi-moment formulation, although extra memory space is required to accommodate the newly introduced moment(s), the increase in CPU time is not necessarily much significant. Using the 1D dam-break test shown above, we compared the CPU time for one-step updating needed by the current scheme and a TVD scheme that uses the MUSCL reconstruction [21] and the flux difference scheme of Roe [22]. We ran both schemes on a Linux PC with a Pentium(R) 2.8 GHz CPU. Table I shows the comparison of the run-time in seconds of the two schemes for different grid sizes.

Table I. The comparison of the calculation run-time between a conventional TVD scheme and the present scheme.

Mesh number	Run-time (s)	
	TVD scheme	Present scheme
200	0.069	0.097
400	0.146	0.207
600	0.212	0.300
800	0.276	0.381

It is found that compared with a conventional TVD scheme the average-increase in CPU time of the present scheme is less than 30%.

4.2. 1D symmetrical rarefaction waves

As given in [1], a symmetrically diverging velocity creates two strong rarefaction waves traveling in opposite directions. The initial condition is given by

$$\begin{aligned}
 h_0(x) &= 1.0 \text{ m} \\
 u_0(x) &= \begin{cases} -5.0 \text{ m/s}, & 0.0 \leq x \leq 25.0 \\ 5.0 \text{ m/s}, & 25.0 < x \leq 50.0 \end{cases} \quad (86)
 \end{aligned}$$

The mesh number is 200 and the time step is 0.02 s. As commented in [1], this test results in a very thin water layer in the domain center that may cause a negative height in the numerical outputs of some schemes. The numerical solutions of the water depth h , and the velocity u at $t=2.5$ s are plotted in Figure 3. The numerical outputs of both the height and the velocity from our scheme are accurate.

4.3. 1D test for the exact C-property

The purpose of this test [19, 23] is to verify that the formulation maintains the exact C-property over a non-flat bottom. In this paper, we computed with two different profiles of the bottom topography separately given by

$$z(x) = 5.0 \exp\left(-\frac{2}{5}(x-5.0)^2\right), \quad 0.0 \leq x \leq 2.0 \quad (87)$$

and

$$z(x) = \begin{cases} 4.0 & \text{if } 4.0 \leq x \leq 8.0 \\ 0.0 & \text{otherwise} \end{cases} \quad (88)$$

We refer to (87) as the smooth bottom and (88) as the discontinuous bottom. The initial conditions are specified as a stationary state:

$$h + z = 10.0, \quad hu = 0.0 \quad (89)$$

This state should be exactly maintained physically.

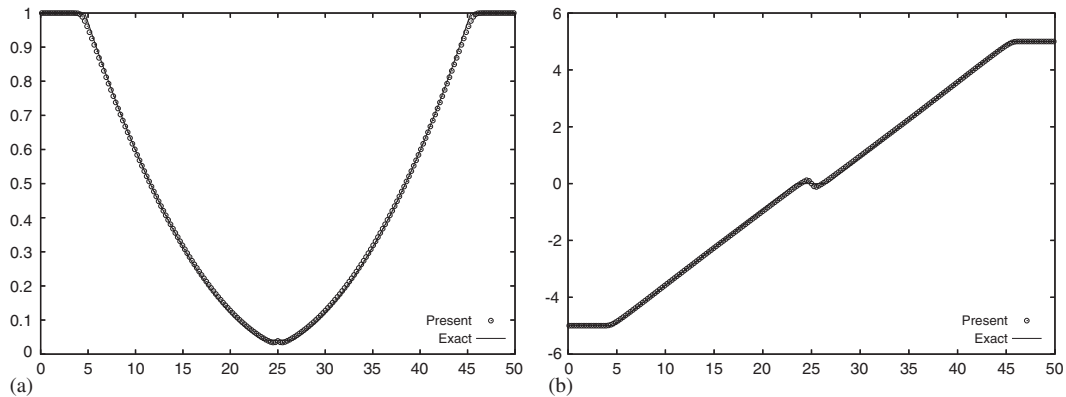


Figure 3. Numerical results of water depth (a) and velocity (b) at $t=2.5$ s by the present method (circles) and exact solution (solid line).

Table II. L^1 and L^∞ errors for the stationary solution with a smooth bottom (87) and a discontinuous bottom (88).

Bottom	L^1 error		L^∞ error	
	h	hu	h	hu
Smooth	1.59E-15	2.07E-14	6.22E-15	8.08E-14
Discontinuous	1.30E-15	1.22E-14	7.11E-15	6.11E-14

We used a mesh of 200 cells and computed the solutions until $t=0.5$. The L^1 and the L^∞ errors for the water height h and the discharge hu are given in Table II for the two bottom profiles. It is clear that the L^1 and the L^∞ errors are of a magnitude of the round-off error, which manifests the exact C-property of the proposed numerical method.

4.4. 1D perturbation of a lake at rest

The purpose of the fourth problem [19, 23, 24] is to test the present formulation regarding the numerical source term. A small perturbation is imposed to a water mass at rest with variable bottom topography. The bottom topography is given with

$$z(x) = \begin{cases} 0.25(1.0 + \cos(10.0\pi(x-0.5))) & \text{if } 1.4 \leq x \leq 1.6 \\ 0.0 & \text{otherwise} \end{cases} \quad (90)$$

where $x \in [0, 2]$. The initial conditions are

$$H_0(x) = \begin{cases} 1.0 + \Delta H & \text{if } 1.1 \leq x \leq 1.2 \\ 1.0 & \text{otherwise} \end{cases} \quad (91)$$

$$u_0(x) = 0.0$$

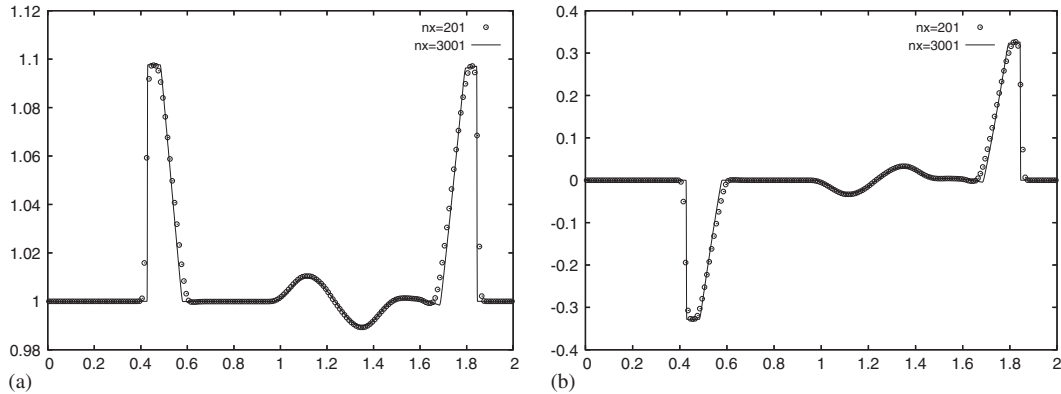


Figure 4. Numerical results (big pulse) of total height (a) and momentum (b) at $t=0.2$ s by the present method with a 201 mesh (circle) and a 3001 mesh (solid line).

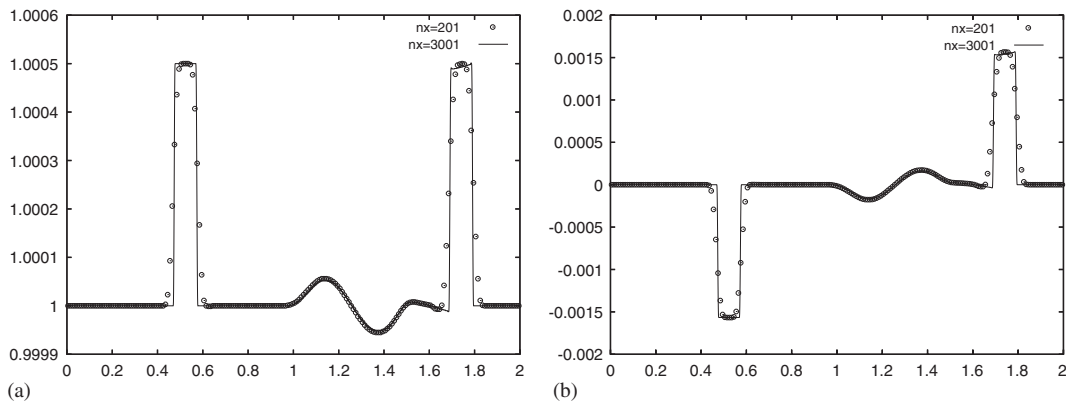


Figure 5. Numerical results (small pulse) of total height (a) and momentum (b) at $t=0.2$ s by the present method with a 201 mesh (circle) and a 3001 mesh (solid line).

where ΔH is a non-zero perturbation constant. In the presented example, two cases with the initial perturbations, respectively being $\Delta H=0.2$ (big pulse) and $\Delta H=0.001$ (small pulse) have been examined. The solutions at $t=0.2$ s for the big pulse $\Delta H=0.2$ over 201 mesh cells and 3001 mesh cells are shown in Figure 4. The results for small pulse $\Delta H=0.001$ are shown in Figure 5. The numerical results are accurate and free of spurious oscillations and look very competitive to those found in other existing literature [19, 23–25].

4.5. 1D dam-breaking problem over a rectangular bump

In the fifth numerical test, we compute the dam-breaking problem over a rectangular bump involving a rapidly varying flow over discontinuous bottom topography. It presents a more serious test to evaluate the robustness and accuracy of the balanced numerical treatment of the source term.

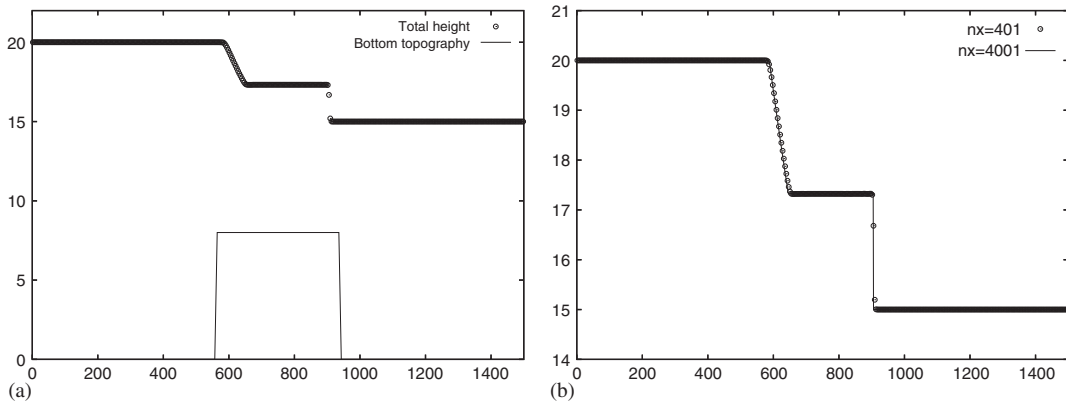


Figure 6. Numerical results of total height at $t = 15$ s: (a) the numerical solution using 401 mesh (circle) with the bottom topography (solid line) and (b) the numerical solution using 401 mesh (circle) and a 4001 mesh (solid line).

The bottom topography is given by

$$z(x) = \begin{cases} 8.0 \text{ m} & \text{if } |x - 750.0| \leq 1500.0/8.0 \\ 0.0 \text{ m} & \text{otherwise} \end{cases} \quad (92)$$

where $x \in [0, 1500]$.

The initial condition is

$$H_0(x) = \begin{cases} 20.0 \text{ m} & \text{if } x \leq 750.0 \\ 15.0 \text{ m} & \text{otherwise} \end{cases} \quad (93)$$

$$u_0(x) = 0.0 \text{ m/s}$$

The numerical results obtained with 401 mesh cells and with 4001 mesh cells are shown in Figures 6 and 7 for different ending time $t = 15$ and 60 s. The numerical results show that even for completely discontinuous bottom topography our scheme is still able to reproduce well-balanced and accurate numerical solutions. Again our numerical results are competitive to other existing methods.

4.6. 2D circular dam-break problem

A 2D dam-break [26, 27] is solved in the domain of $[0, 200] \times [0, 200]$. The purpose of this problem is to test the implementation of the present formulation in two dimensions. The initial conditions are

$$h_0(x, y) = \begin{cases} 10.0 \text{ m} & (x - 25.0)^2 + (y - 25.0)^2 \leq (10.5)^2 \\ 1.0 \text{ m} & \text{otherwise} \end{cases} \quad (94)$$

$$u_0(x, y) = 0.0 \text{ m/s}$$

$$v_0(x, y) = 0.0 \text{ m/s}$$

The mesh number is 101×101 and the time step is 0.02 s.

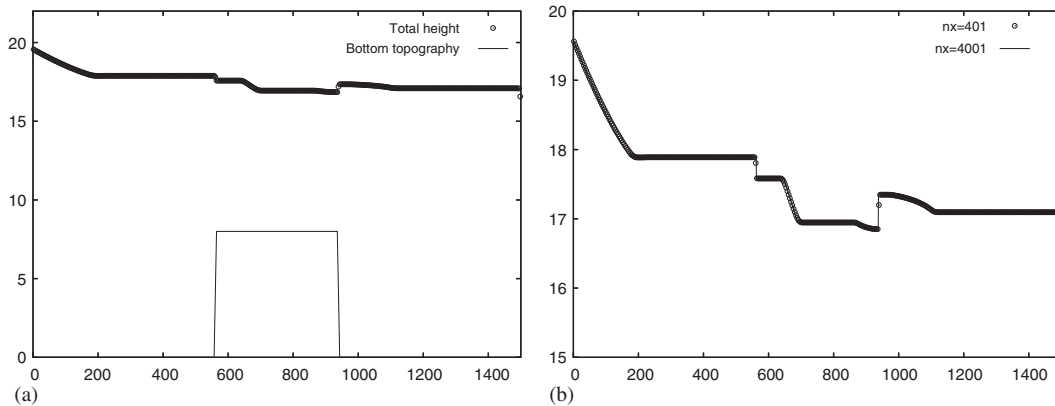


Figure 7. Numerical results of total height at $t=60$ s: (a) the numerical solution using 401 mesh (circle) with the bottom topography (solid line) and (b) the numerical solution using 401 mesh (circle) and a 4001 mesh (solid line).

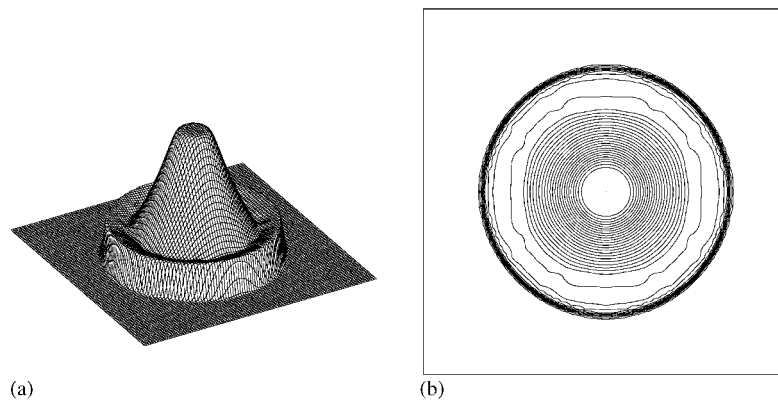


Figure 8. Numerical results of water depth at $t=0.69$ s, bird's eye view (a) and an contour view (b).

Figure 8 shows the bird's eye view and contour lines of h [m] at $t=7.2$ s. The symmetry of the numerical solution is well preserved. Figure 9 displays the profiles of h on cross-sections of the x -axis, y -axis and $y=x$. The shock wave and the expansion fan are computed with a direction-independent resolution although a splitting technique has been used on a Cartesian grid.

4.7. 2D partial dam-break problem

Being one of the most popular benchmark test used to evaluate numerical schemes for shallow water equations, a 2D partial dam break was computed. A dam is located in the middle of a

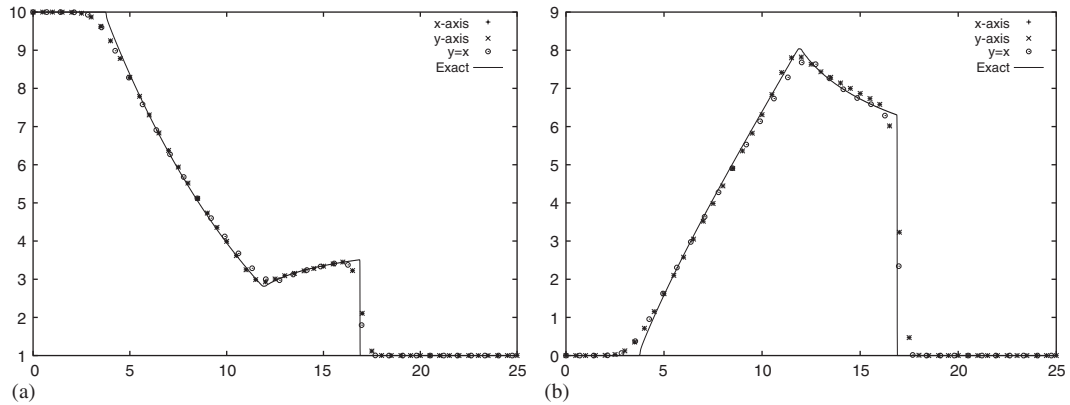


Figure 9. Numerical results at $t=0.69$ s of water depth (a) and total velocity (b).

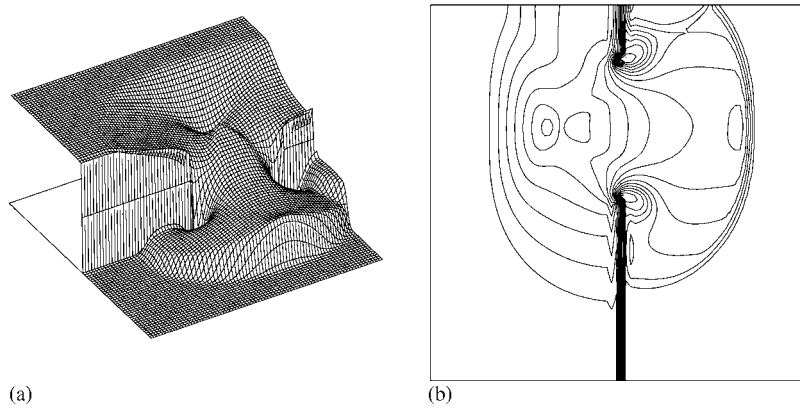


Figure 10. Numerical results of water depth at $t=7.2$ s. Plotted are the bird's eye view (a) and the contour map (b).

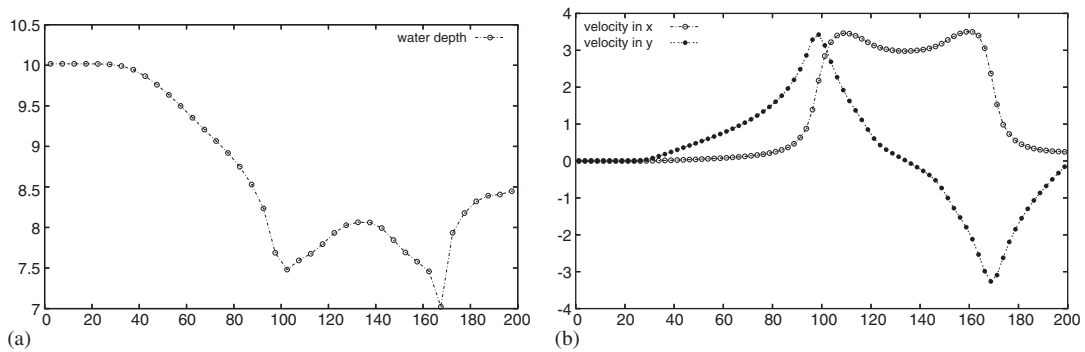


Figure 11. Numerical results of (a) water depth and (b) velocity components at $t=7.2$ s.

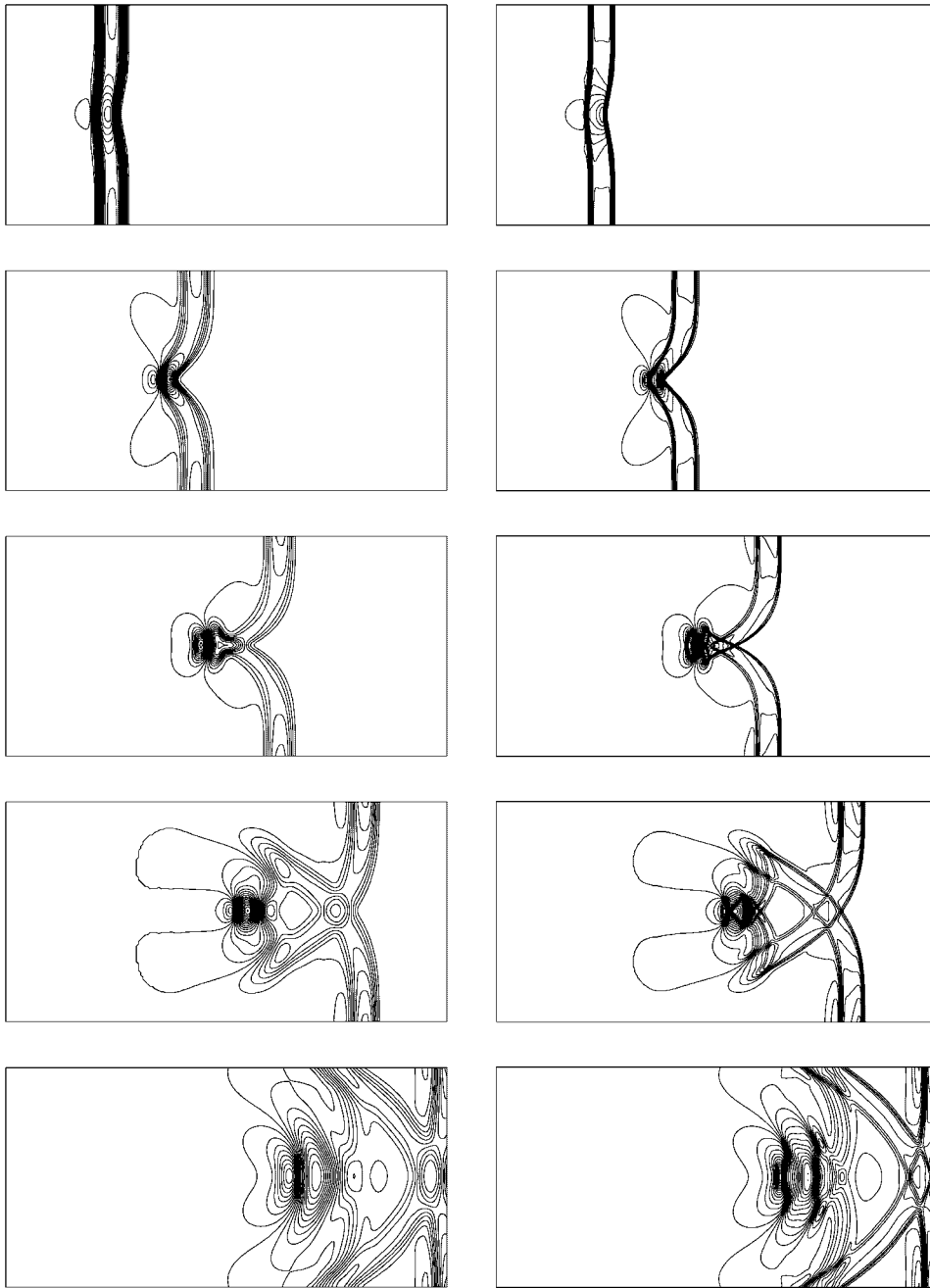


Figure 12. Contour view of the surface level at $t=0.12, 0.24, 0.36, 0.48, 0.60$ s with 201×101 grid (left) and 601×301 grid (right). The contours range from 0.999727 to 1.006286 at $t=0.12$, from 0.994892 to 1.016554 at $t=0.24$, from 0.988693 to 1.011509 at $t=0.36$, from 0.990372 to 1.004979 at $t=0.48$ and from 0.995114 to 1.005568 at $t=0.60$.

200m × 200m square. The initial conditions are

$$h_0(x, y) = \begin{cases} 10.0\text{m}, & 0.0 \leq x \leq 100.0 \\ 5.0\text{m}, & 100.0 < x \leq 200.0 \end{cases} \quad (95)$$

$$u_0(x, y) = 0.0\text{m/s}, \quad v_0(x, y) = 0.0\text{m/s}$$

A 75-m-wide breach allows the partial collapse of the dam and generates a 2D bore wave as plotted in Figure 10 where a 81×81 mesh is used.

To compare with the numerical results given in [28] where numerical tests of shallow water equations by several well-established Riemann solvers are reported, we ran again the test with a 41×41 mesh and give the profiles of h , u and v on the cross-section of $x = 92.5\text{m}$ in Figure 11. It is observed that the present model has competitive numerical accuracy in simulating 2D bore shock wave.

4.8. 2D test for the source terms of bottom effects

We examined the 2D formulation with the effects of the bottom topography using a numerical test widely used in the literature [19, 23–25]. The computational domain is $[0, 2] \times [0, 1]$, and the bottom topography is given by

$$z(x, y) = 0.8 \exp(-5.0(x - 0.9)^2 - 50.0(y - 0.5)^2) \quad (96)$$

The initial surface level is almost flat. Only in the region $0.05 \leq x \leq 0.15$ it is perturbed upward by the displacement 0.01 and the initial momentum in the x and y directions is zero:

$$H_0(x, y) = \begin{cases} 1.01\text{m} & \text{if } 0.05 \leq x \leq 0.15 \\ 1.0\text{m} & \text{otherwise} \end{cases} \quad (97)$$

$$hu_0(x, y) = 0.0\text{m/s}$$

$$hv_0(x, y) = 0.0\text{m/s}$$

We used two different uniform meshes with $[201 \times 101]$ and $[601 \times 301]$. Figure 12 shows 30 uniformly spaced contour lines of the surface level H at $t = 0.12, 0.24, 0.36, 0.48, 0.60$. The results obtained with the coarse grid appear on the left side, while on the right half we find the numerical solutions obtained with the fine grid. The initial perturbation propagates right and is affected by the bottom. The reflections and interactions of the surface wave result in complex but symmetrical structures. It is observed that our 2D scheme exactly simulated the symmetrical structure with the fine structures well resolved.

5. CONCLUSION

We presented a well-balanced numerical formulation for the shallow water equations by using the CIP/multi-moment finite volume method. Two kinds of moments, namely the volume-integrated average (VIA) and the surface-integrated average (SIA) are treated as the model variables and predicted forward in time separately. The VIA moments are computed through a finite volume formulation in flux form, while the SIAs on each cell boundary are updated by a semi-Lagrangian

procedure in the characteristic fields. The source terms are discretized so as to exactly balance the numerical fluxes in terms of both VIA and SIA. The proposed scheme is local, accurate and robust. The numerical results presented in this paper are quite competitive. The presented numerical formulation can provide a practical and accurate framework for hydraulic and oceanic applications.

REFERENCES

1. Toro EF. *Shock-Capturing Methods for Free-Surface Shallow Flows*. Wiley: New York, 2001.
2. Aizinger V, Dawson C. A discontinuous Galerkin method for two-dimensional flow and transport in shallow water. *Advances in Water Resources* 2002; **25**:67–84.
3. Choi BJ, Iskandarani M, Levin J, Haidvogel DB. A spectral finite-volume method for the shallow water equations. *Monthly Weather Review* 2004; **132**:1777–1791.
4. Xiao F. Profile-modifiable conservative transport schemes and a simple multi integrated moment formulation for hydrodynamics. *Computational Fluid Dynamics* 2002. Springer: Berlin, 2003; 106.
5. Xiao F, Ikebata A. An efficient method for capturing free boundary in multi-fluid simulations. *International Journal for Numerical Methods in Fluids* 2003; **42**:187–210.
6. Xiao F. Unified formulation for compressible and incompressible flows by using multi-integrated moments I: one-dimensional inviscid compressible flow. *Journal of Computational Physics* 2004; **195**:629–654.
7. Xiao F, Akoh R, Ii S. Unified formulation for compressible and incompressible flows by using multi-integrated moments II: multi-dimensional version for compressible and incompressible flows. *Journal of Computational Physics* 2006; **213**:31–56.
8. Xiao F, Ikebata A, Hasegawa T. Numerical simulations of free-interface fluids by a multi integrated moment method. *Computers and Structures* 2005; **83**:409–423.
9. Ii S, Shimuta M, Xiao, F. A 4th-order and single-cell-based advection scheme on unstructured grids using multi-moments. *Computer Physics Communications* 2005; **173**:17–33.
10. Ii S, Xiao F. CIP/multi-moment finite volume method for Euler equations: a semi-Lagrangian characteristic formulation. *Journal of Computational Physics* 2007; **222**:849–871.
11. Xiao F, Yabe T. Completely conservative and oscillation-less semi-Lagrangian schemes for advection transportation. *Journal of Computational Physics* 2001; **170**:498–522.
12. Tanaka R, Nakamura T, Yabe T. Constructing exactly conservative scheme in a non-conservative form. *Computer Physics Communications* 2000; **126**:232–243.
13. Yabe T, Tanaka R, Nakamura T, Xiao F. Exactly conservative semi-Lagrangian scheme (CIP-CSL) in one dimension. *Monthly Weather Review* 2001; **129**:332–344.
14. Xiao F, Yabe T, Peng X, Kobayashi H. Conservative and oscillation-less atmospheric transport schemes based on rational functions. *Journal of Geophysical Research* 2002; **107**:4609.
15. Colella P, Woodward P. The piecewise parabolic method for gas-dynamical simulations. *Journal of Computational Physics* 1984; **54**:174–201.
16. Shu CW. Total variation diminishing time discretizations. *SIAM Journal on Scientific and Statistical Computing* 1988; **9**:1073–1084.
17. Bermudez A, Vazquez ME. Upwind method for hyperbolic conservation laws with source terms. *Computers and Fluids* 1994; **23**:1049–1071.
18. Zhou JG, Causon DM, Mingham CG, Ingram DM. The surface gradient method for the treatment of source terms in the shallow-water equations. *Journal of Computational Physics* 2001; **168**:1–25.
19. Xing Y, Shu CW. High order finite difference WENO schemes for a class of hyperbolic systems with source terms. *Journal of Computational Physics* 2005; **208**:206–227.
20. Burguete J, Navarro PG. Efficient construction of high-resolution TVD conservative schemes for equations with source terms: application to shallow water flows. *International Journal for Numerical Methods in Fluids* 2001; **37**:209–248.
21. van Leer B. Toward the ultimate conservative difference scheme. Part V: a second order sequel to Godunov's method. *Journal of Computational Physics* 1979; **32**:101–136.
22. Roe PL. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics* 1981; **43**:357–372.

23. Xing Y, Shu CW. High order well-balanced finite volume WENO schemes and discontinuous Galerkin methods for a class of hyperbolic systems with source terms. *Journal of Computational Physics* 2006; **214**:567–598.
24. Noelle S, Pankratz N, Puppo G, Natvig JR. Well-balanced finite volume schemes of arbitrary order of accuracy for shallow water flows. *Journal of Computational Physics* 2006; **213**:474–499.
25. LeVeque RJ. Balancing source terms and flux gradients in high-resolution Godunov methods: the quasi-steady wave-propagation algorithm. *Journal of Computational Physics* 1998; **146**:346–365.
26. Lin GF, Lai JS, Guo WL. Finite-volume component-wise TVD schemes for 2D shallow water equations. *Advances in Water Resources* 2003; **26**:861–873.
27. Wang JW, Liu RX. The composite finite volume method on unstructured meshes for the two-dimensional shallow water equations. *International Journal for Numerical Methods in Fluids* 2001; **37**:933–949.
28. Erduran KS, Kutija V, Hewett CJM. Performance of finite volume solutions to the shallow water equations with shock-capturing schemes. *International Journal for Numerical Methods in Fluids* 2002; **40**:1237–1273.