

Hybrid Finite Element / Volume Method for Shallow Water Equations

Shahrouz Aliabadi, Muhammad Akbar and Reena Patel

Jackson State University, School of Engineering

Northrop Grumman Center for High Performance Computing, Jackson, MS

ABSTRACT

A hybrid numerical scheme based on finite element and finite volume methods is developed to solve shallow water equations. In the past recent years, we introduced a series of hybrid methods to solve incompressible low and high Reynolds number flows for single and two fluid flow problems. The presented work extends the application of hybrid method to shallow water equations. In our hybrid shallow water flow solver, we write the governing equations in non-conservation form and solve the nonlinear wave equation using finite element method with linear interpolation functions in space. On the other hand, the momentum equation is solved with highly accurate cell-center finite volume method. Our hybrid numerical scheme is truly a segregated method with primitive variables stored and solved for at both node and element centers. To enhance the stability of the hybrid method around discontinuities, we introduce a new shock capturing which will act only around sharp interfaces without sacrificing the accuracy elsewhere. Matrix-free GMRES iterative solvers are used to solve both the wave and momentum equations in finite element and finite volume schemes. Several test problems are presented to demonstrate the robustness and applicability of the numerical method.

1 INTRODUCTION

The shallow-water equations govern variety of problems of coastal and environmental engineering. These equations are simplifications of the full equations of motion for an incompressible fluid. The key assumption is that the dimensions in the horizontal plane are by far larger than the vertical dimension. Therefore, it is reasonable to assume that flow is homogeneous along the vertical axis. The hydrostatic balance equation states that gravity balances the pressure gradient in the vertical direction resulting in strong 2D hyperbolic (or weak parabolic) equation systems. The hyperbolic characteristics of shallow water equations dictate that the disturbances will propagate with left and right running wave speed namely $u - C$ and $u + C$, where $C = \sqrt{gH}$ is the wave propagation speed and u , H , and g are the speed of water, depth of water and the gravitational force, respectively.

Usually, the shallow water equations are written in conservation form (with conservation variables) and solved using finite difference, finite volume or discontinuous Galerkin finite element methods [1-7] with approximate Riemann solvers [8-9] coupled with flux limiters [10] to enhance numerical stabilities around shocks and discontinuities. These method usually deliver very good results if the water velocity, u , and the wave speed, C , are in the same order. If C is much larger than u , the numerical scheme becomes very stiff and the approximate Riemann solvers will result in inaccurate unwinding. In these circumstances, the governing equations are

usually written in non-conservation form (with primitive variables) and two sets of equations are derived and solved separately.

The earliest finite element formulations for shallow water equations written in non-conservation form were unstable. The wave continuity equations (WCE), introduced in [11] suppressed the spurious oscillations without having to dampen the solution either numerically or artificially. Alternatively, selective lumping [12] method was introduced to stabilize the finite element formulations if the equations were solved in their original format.

The generalized wave continuity equation (GWCE) introduced in [13] further enhanced the stability of finite element methods for shallow water equations. The analysis revealed that in GWCE there was no loss in the wave propagation characteristics if, during the formulation, the bottom friction τ is replaced by a numerical parameter G . Several models have been developed using this approach since its conception twenty years ago, including ADCIRC (an ADvanced CIRCulation model) [14-16].

Recently, we have developed a hybrid finite element/volume (FE/FV) solver for 3D incompressible single and two fluid flow problems at low and high Reynolds numbers [17-20]. Our hybrid flow solver is based on the well-known pressure correction (projection) method [21-22]. The solution procedure follows a segregated approach to decouple the pressure from the velocity. The velocity field is updated by solving the momentum equation provided that a known pressure field is given as a source term, through a element-centered finite volume (FV) discretization. The pressure does not directly enter the momentum equation. Instead, an auxiliary variable, which is closely related to the pressure, takes the place of pressure in the momentum equation, providing pressure gradient information. We put the auxiliary variable on the vertices of elements. This deployment provides a convenient way to evaluate the pressure gradient using the local finite element basis functions. The incremental value of the auxiliary variable is computed by solving a Poisson equation using the Galerkin finite element (FE) method. The auxiliary variable is then used to update the velocity field. After the final velocity field is determined, the pressure can be updated using the auxiliary variable and the velocity divergence field. The pressure is updated in such a way that the pressure field is free of unphysical conditions in the boundary layer.

Our hybrid finite volume/element solver is aimed to take advantage of the merits of both the FV and the FE methods and avoid their shortcomings. For example, highly-stretched elements (also known as high-aspect-ratio elements) are commonly used inside the boundary layer for high Reynolds number flows to resolve the boundary layer and reduce the number of elements. The stabilization parameters in the stabilized FE based flow solvers [23-24] are related to the characteristic element length which is not well defined for high-aspect-ratio mesh elements. Due to this, it is very difficult to control the numerical dissipation of stabilized finite element solvers. By contrast, the finite volume flow solver is very insensitive to the aspect ratio of the mesh elements. For this reason, we use the finite volume method to solve the momentum equation. On the other hand, the classic Galerkin FE method is very suitable for the elliptic typed equations like the pressure Poisson equation emerging from the segregated approach. Therefore, the combination of the FV method and the FE method is expected to perform well in the incompressible flow solvers based on the pressure projection method, which has been confirmed by our earlier work [17].

In this article we will describe the extension of our implicit hybrid finite element/volume solver to shallow water equations written in non-conservative form. In our implementation, we solve the momentum equation using cell-centered finite volume method to obtain intermediate velocity and solve the equivalent of continuity equation using finite element method.

The rest of this paper is organized as follows. Section 2 presents the governing equations. Section 3 provides details of finite volume implementation. Section 4 describes the finite element formulation and matrix-free solution strategy. Finally, in Section 5 we demonstrate the performance of the hybrid method by solving series of test problems.

2 DISCRETIZED GOVERNING EQUATIONS

The Shallow Water Equations (SWEs) are derived by depth-averaging the Reynolds equations for column of fluid with mass and momentum conservation. In SWEs it is assumed that vertical motions are negligible and that pressure is hydrostatic. The velocity and depth of fluid moving in the domain $\mathbf{x}(x, y) \in W$ with boundary $\partial W = \partial W_g + \partial W_b$ during the time interval $t \in (0, T)$ in non-conservation form can be described by

$$\frac{\partial H}{\partial t} + \tilde{\mathbf{N}} \times (\mathbf{H}\mathbf{u}) = 0, \quad (1)$$

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \times \tilde{\mathbf{N}})\mathbf{u} = -g\tilde{\mathbf{N}}H - \tilde{\mathbf{N}} \frac{\partial p_a}{\partial r_0} + gZ \frac{\partial}{\partial t} + \frac{\boldsymbol{\tau}_s}{r_0 H} - \frac{\boldsymbol{\tau}_b}{r_0 H} + \frac{n}{H} \tilde{\mathbf{N}} \times \tilde{\mathbf{N}} (\mathbf{u}H), \quad (2)$$

where H , \mathbf{u} , g , p_a , r_0 , n and Z are the water depth, velocity, gravity, surface pressure, fluid density, kinematic viscosity and surface elevation, respectively. Figure 1 demonstrates the terminology we use to describe H , h , and Z . In addition, in Eq. (2) $\boldsymbol{\tau}_s$ and $\boldsymbol{\tau}_b$ are surface and bed shear stresses defined as

$$\boldsymbol{\tau}_b = r_0 U c_b \mathbf{u}, \quad \boldsymbol{\tau}_s = r_0 U c_s \mathbf{u}, \quad (3)$$

where c_s and c_b are surface and bed friction coefficients and $U = \|\mathbf{u}\|$. Using Eq. (3), we can rewrite the momentum equation as following

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \times \tilde{\mathbf{N}})\mathbf{u} = -g\tilde{\mathbf{N}}H - \tilde{\mathbf{N}} \frac{\partial p_a}{\partial r_0} + gZ \frac{\partial}{\partial t} + \frac{U}{H} (c_s - c_b)\mathbf{u} + n(\tilde{\mathbf{N}} \times \tilde{\mathbf{N}})\mathbf{u} + \frac{n}{H} \mathbf{u}(\tilde{\mathbf{N}} \times \tilde{\mathbf{N}})H \quad (4)$$

Here we expanded the viscous term from conservation form to non-conservation form. In our hybrid method which will be introduced later, we store the variable H at the node and the velocity \mathbf{u} at the element center. Using linear interpolation function will result in constant gradient and zero Laplacian for working variable H . As a result, we can drop the last term in Eq (4). Furthermore the time discretization of Eq (4) using backward difference formula will yield

$$\frac{a_1 \mathbf{u} + a_0 \mathbf{u}^n + a_{-1} \mathbf{u}^{n-1}}{\Delta t} + (\mathbf{u} \times \tilde{\mathbf{N}})\mathbf{u} - \frac{U}{H} (c_s - c_b)\mathbf{u} - n(\tilde{\mathbf{N}} \times \tilde{\mathbf{N}})\mathbf{u} = -g\tilde{\mathbf{N}}H - \tilde{\mathbf{N}} \frac{\partial p_a}{\partial r_0} + gZ \frac{\partial}{\partial t}. \quad (5)$$

where both \mathbf{u} and H are unknowns at time step $n+1$. For first order time accurate scheme, $a_1 = 1.0$, $a_0 = -1.0$ and $a_{-1} = 0.0$ and for second order time accurate scheme $a_1 = 1.5$, $a_0 = -2.0$ and $a_{-1} = 0.5$. The hybrid FE/FV scheme evolves by perturbing H such that

$$H \approx H + H\phi, \quad (6)$$

where $H\phi$ is very small compared to H . Using Eq (6), the time-discretized momentum equation will lead to:

$$\begin{aligned} & \frac{a_1 \mathbf{u} - a_1 \mathbf{u}}{Dt} + \frac{a_1 \mathbf{u} + a_0 \mathbf{u}^n + a_{-1} \mathbf{u}^{n-1}}{Dt} + (\mathbf{u} \times \tilde{\mathbf{N}}) \mathbf{u} - \frac{U}{H} (c_s - c_b) \mathbf{u} - n (\tilde{\mathbf{N}} \times \tilde{\mathbf{N}}) \mathbf{u} \\ & = -g \tilde{\mathbf{N}} H - g \tilde{\mathbf{N}} H \phi - \tilde{\mathbf{N}} \left(\frac{\partial p}{\partial r_0} + g Z \frac{\partial}{\partial z} \right) \end{aligned} \quad (7)$$

Here, we introduced \mathbf{u} which is the final velocity at time $n+1$ in the iterative nonlinear scheme. In this context \mathbf{u} will be intermediate velocity field during the nonlinear iteration and the balance between \mathbf{u} and \mathbf{u} will be enforced through the gradient of $H\phi$. This process is similar to the projection methods commonly used to solve incompressible Navier Stokes equations [17-18, 22]. Note that as $H\phi \approx 0$, then $\tilde{\mathbf{N}} H \phi \approx 0$ which ensures $\mathbf{u} \approx \mathbf{u}$. We use the fractional time splitting method to first compute an intermediate velocity (predictor step) from Eq (5) and then use the results in the correction phase described as:

$$g \frac{Dt}{a_1} \tilde{\mathbf{N}} H \phi = \mathbf{u} - \mathbf{u}. \quad (8)$$

Remark¹. We can observe that Equation (5) used in the predictor phase is time discretized momentum equation in its original form. Clearly, the predictor phase satisfies consistency criteria and conserves the momentum.

To derive the continuity wave equation, we multiply Eq (8) by H and then take the divergence to obtain

$$g \frac{Dt}{a_1} \tilde{\mathbf{N}} \times (H \tilde{\mathbf{N}} H \phi) = \tilde{\mathbf{N}} \times (H \mathbf{u}) - \tilde{\mathbf{N}} \times (H \mathbf{u}). \quad (9)$$

Since the last term in Eq (9) include the final velocity at time step $n+1$, we can replace it by its equivalent in continuity equation. The time discretization of continuity equation with perturbed H is

$$\tilde{\mathbf{N}} \times (H \mathbf{u}) = \frac{a_1 H + a_1 H \phi + a_0 H^n + a_{-1} H^{n-1}}{Dt}. \quad (10)$$

Combine Equation (9) and (10) to obtain time-discretized wave equation. The results can be written as

$$H \phi + \frac{Dt}{a_1} \tilde{\mathbf{N}} \times (H \mathbf{u}) - \frac{Dt^2}{a_1^2} \tilde{\mathbf{N}} \times (C^2 \tilde{\mathbf{N}} H \phi) = - \frac{Dt}{a_1} \frac{a_1 H + a_0 H^n + a_{-1} H^{n-1}}{Dt} + \tilde{\mathbf{N}} \times (H \mathbf{u}) - \frac{Dt}{a_1} \tilde{\mathbf{N}} \times (H \mathbf{u}) \quad (11)$$

where $C = \sqrt{gH}$ is the wave speed.

Remark². We can clearly see that the right hand side of Equation (11) is weighted by time discretized continuity equation. Therefore, as $H \rightarrow 0$, Eq (11) will yield zero residual for continuity equation. Clearly Eq (11) satisfies consistency criteria and conserves the mass.

3 FINITE VOLUME METHOD FOR MOMENTUM EQUATION

The time-discretized momentum equation represented in Eq (5) is a nonlinear advection-diffusion equation system with gradients of H , p_a , Z as source terms in the right hand side. We use the finite volume (FV) method to discretize this system in space and solve for velocity field using Newton-Raphson iterative method [17-18]. After the computational domain is discretized into conforming (without hanging nodes) elements (hybrid triangular/quadrilateral elements are allowed), the finite volume formulation can be written for each control volume. Since the current FV solver is element-centered, i.e. the control volume is the element itself, the velocity unknowns are stored at the element centers. Let's rearrange Eq (5) in a format appropriate for FV application. Using partial derivative for advection part, and gradient of velocity to represent the viscous term, we can re-write Eq (5) as following

$$M\mathbf{u} + \tilde{N} \times (\mathbf{u}\mathbf{u}) - n\tilde{N} \times (\tilde{N}\mathbf{u})^T = S \quad (12)$$

where

$$M = \frac{\partial a_1}{\partial t} - \frac{U}{H}(c_s - c_b) - \tilde{N} \times \frac{\partial \mathbf{u}}{\partial t} \quad S = - \frac{a_0 \mathbf{u}^n + a_{-1} \mathbf{u}^{n-1}}{Dt} - g\tilde{N}H - \tilde{N} \frac{\partial p_a}{\partial r_0} + gZ \frac{\partial \mathbf{u}}{\partial t} \quad (13)$$

Following the standard finite volume discretization, we can integrate Eq (12) over the i^{th} element volume and use divergence theorem to obtain

$$\oint_{W_i} M \mathbf{u} dW + \oint_{W_i} u_n \mathbf{u} dG - \oint_{W_i} n \hat{\mathbf{n}} \times (\tilde{N}\mathbf{u})^T dG = \oint_{W_i} S dW, \quad (14)$$

where $\hat{\mathbf{n}}$ is the unit vector normal to the face (pointing outward) and u_n is $\hat{\mathbf{n}} \times \mathbf{u}$. Using average quantities over small element volumes and faces, we can write approximation of Eq (14) in this form

$$M^{(i)} \mathbf{u}^{(i)} dV^{(i)} + \sum_{k=1}^{nef} u_n^{(ik)} \mathbf{u}^{(ik)} dA^{(ik)} - \sum_{k=1}^{nef} n \hat{\mathbf{n}}^{(ik)} \times (\tilde{N}\mathbf{u}^{(ik)})^T dA^{(ik)} = S dV^{(i)} \quad (15)$$

where ik refers to the k^{th} face of i^{th} element and nef stands for total number of element faces. The coefficient M and source S include H , p_a , Z which are stored at each node. Using standard linear finite element functions, these terms can easily be evaluated at element centers. The divergence of velocity in M can be reconstructed by Green's theorem.

In Eq (15) the second and third terms represent inviscid and viscous fluxes, respectively. Since each face is shared with two adjoined elements (left and right elements), unique values for $u_n^{(ik)}$,

$\mathbf{u}^{(ik)}$, and $\tilde{\mathbf{N}}\mathbf{u}^{(ik)}$ need to be defined. The inviscid flux takes the general form of the upwind Steger-Warming flux widely used as an approximate Riemann solver in compressible flow solvers.

$$\mathbf{u}_n^{(ik)}\mathbf{u}^{(ik)} = \frac{1}{2} \left(\left(u_n^{(ik)} + |u_n^{(ik)}| \right) \mathbf{u}_L^{(ik)} + \left(u_n^{(ik)} - |u_n^{(ik)}| \right) \mathbf{u}_R^{(ik)} \right) \hat{\mathbf{n}}^{(ik)} \quad (16)$$

Here unique solution, $u_n^{(ik)}$, at the element interface is the averaged nodal values of each face. The nodal values are obtained using inverse distance weighted averaging from element center values.

The second order accurate reconstruction is used to compute the left state and the right state of the solution at the interface as required in Eq (16).

$$\begin{aligned} \mathbf{u}_L^{(ik)} &= \mathbf{u}^{(i)} + (\mathbf{x}^{(ik)} - \mathbf{x}^{(i)}) \times \tilde{\mathbf{N}}\mathbf{u}^{(i)} \\ \mathbf{u}_R^{(ik)} &= \mathbf{u}^{(j)} + (\mathbf{x}^{(ik)} - \mathbf{x}^{(j)}) \times \tilde{\mathbf{N}}\mathbf{u}^{(j)} \end{aligned} \quad (17)$$

where j refers to element j across the face ik of element i . To evaluate the viscous flux, we need to define a unique gradient of velocity across the interface of two elements. We follow the work presented in [25] to compute the gradient at across the face the face ik .

$$\tilde{\mathbf{N}}\mathbf{u}^{(ik)} = \frac{\mathbf{u}^{(j)} - \mathbf{u}^{(i)}}{(\mathbf{x}^{(j)} - \mathbf{x}^{(i)}) \times \hat{\mathbf{n}}^{(ik)}} \hat{\mathbf{n}}^{(ik)} + \frac{1}{2} \left(\tilde{\mathbf{N}}\mathbf{u}^{(i)} + \tilde{\mathbf{N}}\mathbf{u}^{(j)} \right) - \frac{(\tilde{\mathbf{N}}\mathbf{u}^{(i)} + \tilde{\mathbf{N}}\mathbf{u}^{(j)}) \times (\mathbf{x}^{(j)} - \mathbf{x}^{(i)})}{(\mathbf{x}^{(j)} - \mathbf{x}^{(i)}) \times \hat{\mathbf{n}}^{(ik)}} \hat{\mathbf{n}}^{(ik)} \quad (18)$$

The idea behind Eq. (18) is that the gradient at the face is divided into two components. The first component is the gradient in the direction normal to the face and computed by the first term of Eq. (18). The second component is computed by averaging the gradients at element centroids and removing the component normal to the face.

We used Newton-Raphson iteration technique to solve the nonlinear system of equation described in Eq 15. At each nonlinear iteration, a linear system of equation is solved using Generalized Minimal RESidual method (GMRES) [26]. Because the GMRES algorithm involves only matrix-vector multiplication, it is unnecessary to form the Jacobian matrix explicitly. However, in the matrix-free version, we can approximate the matrix-vector product as

$$\mathbf{J}^{(i,m)}\mathbf{v}^{(m)}; \frac{\mathbf{R}^{(i)}(\mathbf{u} + e\mathbf{v}) - \mathbf{R}^{(i)}(\mathbf{u})}{e}, \quad (19)$$

where

$$\mathbf{R}^{(i)}(\mathbf{u}) = M^{(i)}\mathbf{u}^{(i)} dV^{(i)} + \sum_{k=1}^{nef} u_n^{(ik)}\mathbf{u}^{(ik)} dA^{(ik)} - \sum_{k=1}^{nef} n \hat{\mathbf{n}}^{(ik)} \times (\tilde{\mathbf{N}}\mathbf{u}^{(ik)})^T dA^{(ik)}. \quad (20)$$

In this approximation, the choice of e is a balance between the approximation accuracy and the floating point rounding error. We use the following formula to obtain ε :

$$e = d \frac{\|\mathbf{u}\|_2}{\|\mathbf{v}\|_2} \quad (21)$$

where δ is usually taken as the square root of the machine zero, which is about 10^{-8} to 10^{-7} on most platforms. Obviously, this approximation has the following advantages: (i) avoid the difficulty and cost in forming the Jacobian matrix. For high order finite volume solvers, the analytic evaluation of the Jacobian matrix is not readily available because of the big stencil involved; (ii) save a significant amount of memory for storing the Jacobian matrix. Even though the sparse Jacobian matrix could be stored in a compressed way [17-18], the storage saving is still significant.

The convergence performance of the GMRES algorithm is highly related to the preconditioning. In our hybrid incompressible flow solver [17-19], we adopt the Lower-Upper Symmetric Gauss Seidel (LU-SGS) method [27] as a preconditioning technique. As mentioned in the previous subsection, it is not easy to form the Jacobian matrix analytically in high-order finite volume solvers. By contrast, the Jacobian matrix of the low-order (assuming the solution is constant inside the control volume, i.e. first order accurate) flux function can be trivially obtained and is more diagonally dominant and more compact than the high-order Jacobian matrix. Therefore, the low-order Jacobian matrix is a good candidate as preconditioner to the high-order Jacobian matrix. Using lower order approximation, we can write the inviscid flux as

$$u_n^{(ik)} \mathbf{u}^{(ik)}; \frac{1}{2} \left(\mathbf{u}^{(i)} \times \hat{\mathbf{n}}^{(ik)} + \left| \mathbf{u}^{(i)} \times \hat{\mathbf{n}}^{(ik)} \right| \right) \mathbf{u}^{(i)} + \left(\mathbf{u}^{(i)} \times \hat{\mathbf{n}}^{(ik)} - \left| \mathbf{u}^{(i)} \times \hat{\mathbf{n}}^{(ik)} \right| \right) \mathbf{u}^{(j)} \frac{\Delta x}{\Delta y} \quad (22)$$

where the unique solution at face $, u_n^{(ik)}$, is assumed to be $\mathbf{u}^{(i)} \times \hat{\mathbf{n}}^{(ik)}$. Therefore, the upwind flux can be written as

$$\mathring{\mathbf{a}}_{k=1}^{nef} u_n^{(ik)} \mathbf{u}^{(ik)} dA^{(ik)} = \mathring{\mathbf{a}}_{k=1}^{nef} \frac{1}{2} \left(\mathbf{u}^{(i)} \times \hat{\mathbf{n}}^{(ik)} + \left| \mathbf{u}^{(i)} \times \hat{\mathbf{n}}^{(ik)} \right| \right) \mathbf{u}^{(i)} + \left(\mathbf{u}^{(i)} \times \hat{\mathbf{n}}^{(ik)} - \left| \mathbf{u}^{(i)} \times \hat{\mathbf{n}}^{(ik)} \right| \right) \mathbf{u}^{(j)} \frac{\Delta x}{\Delta y} dA^{(ik)} \quad (23)$$

Considering the fact that $\mathring{\mathbf{a}}_{k=1}^{nef} \mathbf{n}_i^k dA_i^k = 0$ for all elements, the first term in the right hand side of Eq (23) can be dropped.

$$\mathring{\mathbf{a}}_{k=1}^{nef} u_n^{(ik)} \mathbf{u}^{(ik)} dA^{(ik)} = \mathring{\mathbf{a}}_{k=1}^{nef} \frac{1}{2} \left(\mathbf{u}^{(i)} \times \hat{\mathbf{n}}^{(ik)} \right) \mathbf{u}^{(i)} + \left(\mathbf{u}^{(i)} \times \hat{\mathbf{n}}^{(ik)} - \left| \mathbf{u}^{(i)} \times \hat{\mathbf{n}}^{(ik)} \right| \right) \mathbf{u}^{(j)} \frac{\Delta x}{\Delta y} dA^{(ik)} \quad (24)$$

To increase the order of accuracy, we can replace back $\mathbf{u}^{(i)} \times \hat{\mathbf{n}}^{(ik)}$ by $u_n^{(ik)}$ (the unique solution at face) and summarize

$$\mathring{\mathbf{a}}_{k=1}^{nef} u_n^{(ik)} \mathbf{u}^{(ik)} dA^{(ik)} = \mathring{\mathbf{a}}_{k=1}^{nef} \frac{1}{2} \left(u_n^{(ik)} \right) \mathbf{u}^{(i)} + \left(u_n^{(ik)} - \left| u_n^{(ik)} \right| \right) \mathbf{u}^{(j)} \frac{\Delta x}{\Delta y} dA^{(ik)} \quad (25)$$

In addition, the viscous flux can be approximated as

$$n \hat{\mathbf{n}}^{(ik)} (\tilde{\mathbf{N}} \mathbf{u}^{(ik)})^T ; \frac{n (\mathbf{u}^{(j)} - \mathbf{u}^{(i)})}{(\mathbf{x}^{(j)} - \mathbf{x}^{(i)}) \times \hat{\mathbf{n}}^{(ik)}} \quad (26)$$

Using Eq (25) and Eq (26), the approximation of Eq (20) is

$$\mathbf{R}^{(i)}(\mathbf{u}^{(i)});$$

$$M^{(i)} \mathbf{u}^{(i)} dV^{(i)} + \sum_{k=1}^{nef} \frac{1}{2} \left(|u_n^{(ik)}| \right) \mathbf{u}^{(i)} + \left(u_n^{(ik)} - |u_n^{(ik)}| \right) \mathbf{u}^{(j)} \frac{\dot{\mathbf{u}}}{u} dA^{(ik)} - \sum_{k=1}^{nef} n \frac{(\mathbf{u}^{(j)} - \mathbf{u}^{(i)})}{(\mathbf{x}^{(j)} - \mathbf{x}^{(i)}) \times \hat{\mathbf{n}}^{(ik)}} dA^{(ik)} \quad (27)$$

Therefore the diagonal, lower and upper part of LU-SGS will be

$$D^{(i,i)} = \frac{\dot{\mathbf{c}} a_1}{\dot{\mathbf{c}} \mathbf{D} t} - \frac{U}{H} (c_s - c_b) - \tilde{\mathbf{N}} \times \mathbf{u}^{(i)} \frac{\dot{\mathbf{u}}}{u} dV^{(i)} + \sum_{k=1}^{nef} \frac{\dot{\mathbf{c}}}{\dot{\mathbf{c}} 2} |u_n^{(ik)}| + \frac{n}{(\mathbf{x}^{(j(ik)}) - \mathbf{x}^{(i)}) \times \hat{\mathbf{n}}^{(ik)}} \frac{\dot{\mathbf{c}}}{\dot{\mathbf{c}} \theta} dA^{(ik)} \quad (28)$$

$$L^{(i,j)} = \sum_{k=1}^{nef} \frac{\dot{\mathbf{c}}}{\dot{\mathbf{c}} 2} |u_n^{(ik)}| - \frac{1}{2} |u_n^{(ik)}| - \frac{n}{(\mathbf{x}^{(ik)} - \mathbf{x}_i) \times \hat{\mathbf{n}}^{(ik)}} \frac{\dot{\mathbf{c}}}{\dot{\mathbf{c}} \theta} dA^{(ik)} \quad \text{for all } j(ik) < i \quad (29)$$

$$U^{(i,j)} = \sum_{k=1}^{nef} \frac{\dot{\mathbf{c}}}{\dot{\mathbf{c}} 2} |u_n^{(ik)}| - \frac{1}{2} |u_n^{(ik)}| - \frac{n}{(\mathbf{x}^{(ik)} - \mathbf{x}_i) \times \hat{\mathbf{n}}^{(ik)}} \frac{\dot{\mathbf{c}}}{\dot{\mathbf{c}} \theta} dA^{(ik)} \quad \text{for all } j(ik) > i \quad (30)$$

4 FINITE ELEMENT FORMULATION

The time discretized wave equation presented in Eq (11) is solved using Galerkin Finite Element Method. In the finite element formulation, we first define appropriate sets of trial solution space S^h and weighing function space V^h . The finite element formulation of Eq (11) can then be written as follows: for all $w \in V^h$ find $H \phi \in S^h$ such that:

$$\int_{\mathbb{W}} \dot{\mathbf{K}}_0 w H \phi dW + \int_{\mathbb{W}} \dot{\mathbf{K}}_0 \mathbf{u} \times \tilde{\mathbf{N}} H \phi dW + \int_{\mathbb{W}} \dot{\mathbf{K}}_0 \tilde{\mathbf{N}} w \times \tilde{\mathbf{N}} H \phi dW = - \frac{D t}{a_1} \int_{\mathbb{W}} \dot{\mathbf{K}}_0 w dW + \int_{\mathbb{W}_b} \dot{\mathbf{K}}_0 w (\mathbf{n} \times \tilde{\mathbf{N}} H \phi) dG \quad (31)$$

where

$$\dot{\mathbf{K}}_0 = 1 + \frac{D t}{a_1} \tilde{\mathbf{N}} \times \mathbf{u}, \quad \dot{\mathbf{K}}_0 = \frac{D t}{a_1}, \quad \dot{\mathbf{K}}_0 = \frac{D t^2}{a_1^2} C^2, \quad \dot{\mathbf{K}}_0 = \frac{\dot{\mathbf{c}} a_1 H + a_0 H^n + a_{-1} H^{n-1}}{D t} + \tilde{\mathbf{N}} \times (\mathbf{H} \mathbf{u}) - \frac{\dot{\mathbf{c}}}{\dot{\mathbf{c}} \theta} \quad (32)$$

Since $w = 0|_{\mathbb{W}_g}$ where $H \phi$ is imposed ($H \phi = 0$), the boundary integral is performed only over \mathbb{W}_b . From Eq (8), we can also conclude that $\mathbf{n} \times \tilde{\mathbf{N}} H \phi = 0$ when normal component of velocity is imposed (inflow and symmetry boundaries). Therefore,

$$\int_w \tilde{K} \tilde{w} H dW + \int_w \tilde{K} \tilde{u} \tilde{N} H dW + \int_w \tilde{K} \tilde{N}_w \tilde{N} H dW = - \frac{Dt}{a_1} \int_w \tilde{K} \tilde{w} dW. \quad (33)$$

We can observe that the right hand side of Eq (33) is the weighted residual of continuity equation. The finite element formulation in Eq (33) is discretized using linear functions for triangles or bilinear functions for quadrilaterals.

We can enhance the stabilization of finite element formulation for problems with discontinuities by adding residual-based shock capturing. Let's define

$$u^* = \left| \tilde{K} (\tilde{N} H \tilde{N} H) \right|^{1/2} \quad (34)$$

to be velocity scale derived from the residual of continuity equation. The finite element formulation with enhanced shock capturing is formulated such that

$$\int_w \tilde{K} \tilde{w} H dW + \int_w \tilde{K} \tilde{u} \tilde{N} H dW + \int_w \tilde{K} \tilde{N}_w \tilde{N} H dW + \frac{Dt}{a_1} \int_w \frac{1}{2} Du^* \tilde{N}_w \tilde{N} H dW = - \frac{Dt}{a_1} \int_w \tilde{K} \tilde{w} dW + \int_w \frac{1}{2} Du^* \tilde{N}_w \tilde{N} H dW, \quad (35)$$

where D is the element face length.

Remark³. The shock capturing adds numerical diffusion to the finite element formulation. However, the coefficient of numerical diffusion is residual-based which satisfies consistency criteria. In addition, the shock capturing adds nonlinearities to the formulation which is a desirable feature to control the over shoot and undershoot around discontinuities.

We also use exact GMRES matrix-free iterative technique [28] to solve either Eq (33) or Eq (35). The finite element formulations in these equations are already in incremental form (linearized). Using finite element basis functions, the discretization of these equations yield

$$\mathbf{B} \begin{bmatrix} \mathbf{a}^{n-ng} \\ \mathbf{a}^{nn-ng} \end{bmatrix} = \mathbf{a}^{n-ng} \quad (36)$$

where $\mathbf{B}(\cdot, \cdot)$ is a linear binary operator. Here ng is the total number of nodes with Dirichlet boundary condition and $nn-ng$ is the total number of nodes with unknown values. Note that the right hand side of Eq (36) is the residual, which in the discrete form can be calculated directly as a vector. In the GMRES matrix-free iterative scheme, the result of the matrix-vector multiplication can be computed similar to residual computation as a vector. Let's assume the minimization vector in GMRES algorithm can be expanded similar to the trial solution

$$\mathbf{v} = \begin{bmatrix} \mathbf{a}^{nn-ng} \\ \mathbf{a}^{ng} \end{bmatrix} N_j v_j. \quad (37)$$

Therefore the result of matrix-vector multiplication, \mathbf{w} , can be computed directly as

$$\mathbf{w} = \mathbf{B} \begin{matrix} \mathbf{a} \\ \mathbf{c} \\ \mathbf{c} \end{matrix} \begin{matrix} n-ng \\ i=1 \end{matrix} c_i N_i, \begin{matrix} \mathbf{a} \\ \mathbf{a} \end{matrix} \begin{matrix} n-ng \\ j=1 \end{matrix} N_j v_j \frac{\partial}{\partial x} \quad (38)$$

For more detail on matrix-free GMRES in finite element formulation, see [28].

5 NUMERICAL EXAMPLES

Dam Break on a Flat Riverbed. This test problem demonstrates the robustness of the numerical method in the presence of discontinuity. Although our hybrid shallow water equation solver is not intended to solve such problem, but it is robust and accurate enough to do so. In this problem, initially

$$H(x,0) = \begin{cases} 1.0 & \text{if } 0.0 \leq x \leq 0.5 \\ 0.5 & \text{if } 0.5 < x \leq 1.0 \end{cases}, \quad u(x,0) = 0, \quad (39)$$

where the discontinuity at $x = 0.5$ represents a barrier which separates the two initial river heights. The river bed is flat and does not change with respect to x . At $t = 0$, the barrier is removed. For this particular problem, the down stream flow is subcritical with eigenvalue of $u - c$ being negative. The expansion fans will be developed as the water height drop. Contrary, the upstream flow is supercritical with eigenvalue $u + c$ being positive. In this region, the discontinuity will advance as water height increase. We may obtain an exact solution of this problem by using the analysis in [29-31]. The following exact solution is illustrated in Figures 2 and 3.

$$H(x,t) = \begin{cases} 1 & x < \frac{1}{2} - t\sqrt{g} \\ \frac{1}{9g} \left(2\sqrt{g} - \frac{2x - 1}{2t} \right)^2 & \frac{1}{2} - t\sqrt{g} \leq x \leq (u_2 - c_2)t + \frac{1}{2} \\ \frac{1}{4} \sqrt{1 + \frac{16C_s^2}{g}} - \frac{1}{C_s} & (u_2 - c_2)t + \frac{1}{2} < x \leq C_s t + \frac{1}{2} \\ \frac{1}{2} & C_s t + \frac{1}{2} < x \end{cases}, \quad (40)$$

and

$$u(x,t) = \begin{cases} 0 & x < \frac{1}{2} - t\sqrt{g} \\ \frac{1}{3t}(2x - 1 + 2t\sqrt{g}) & \frac{1}{2} - t\sqrt{g} \leq x \leq (u_2 - c_2)t + \frac{1}{2} \\ u_2 & (u_2 - c_2)t + \frac{1}{2} < x \leq C_s t + \frac{1}{2} \\ 0 & C_s t + \frac{1}{2} < x \end{cases} \quad (41)$$

where

$$u_2 = C_s - \frac{g}{8C_s} + \sqrt{1 + \frac{16C_s^2}{g}}, \quad c_2 = \sqrt{\frac{g}{4} + \frac{16C_s^2}{g}}, \quad C_s = 2.957918120187525, \quad (42)$$

The numerical solution compared to the exact solution at $t = 0.01s$ and $t = 0.1s$ using our hybrid scheme with shock capturing for first order accurate in space and time and second order accurate in space and time are shown in Figures 2 and 3, respectively.

Dam-Break on a Variable Depth Riverbed. This test problem is similar to the previous problem but the riverbed is no longer of constant depth. For this test problem, the riverbed is defined as

$$Z(x) = \begin{cases} \frac{1}{8} \cos(10\pi x - \frac{1}{2}) + \frac{1}{2} & \text{if } \frac{2}{5} \leq x \leq \frac{3}{5}, \\ 0 & \text{otherwise} \end{cases} \quad (43)$$

and the initial conditions are

$$u(x,0) = 0 \quad \text{and} \quad H(x,0) = \begin{cases} 1.0 - Z(x) & \text{if } 0 \leq x \leq \frac{1}{2}, \\ 0.5 - Z(x) & \text{if } \frac{1}{2} < x \leq 1 \end{cases}, \quad (44)$$

which is illustrated in Figures 4 and 5. Here the discontinuity at $x = 0.5$ represents a barrier, which separates the two initial river heights and is removed at $t = 0$. Walls are again present at $x = 0$ and at $x = 1$ giving reflection at these boundaries. Since a source term is now present, difficulties can arise when numerically approximating this problem. Computed water depth and velocity of water using our hybrid scheme with shock capturing second order accurate in both space and time are compared to the target solution reported in [31] at $t = 0.01s$ and $t = 0.1s$ in Figures 4 and 5, respectively.

Tidal Wave Propagation on a Variable Depth Bed. This test problem was discussed by Bermudez and Vazquez[1]. Here, the river bed is given as

$$Z(x) = 10 + \frac{40x}{L} - 10 \sin\left(\frac{4\pi}{L}x + \frac{p\dot{Q}}{2\dot{\theta}}\right) \quad (45)$$

with initial conditions described as

$$u(x,0) = 0 \quad \text{and} \quad H(x,0) = 60.5 - Z(x). \quad (46)$$

which is shown in Figure 6. The variable water depth at inflow is imposed such that

$$H(0,t) = \begin{cases} 64.5 + 4 \sin\left(\frac{4\pi}{86400}t - \frac{p\dot{Q}}{2\dot{\theta}}\right) & \text{if } t \leq 43200 \\ 60.5 & \text{if } t > 43200 \end{cases} \quad (47)$$

In addition we impose $u(L,t) = 0$. This problem represents a tidal wave propagating on a variable depth riverbed. The maximum height of the wave is 8m which will occur at $t = 21,600s$. Also, since the waves propagate at speed of \sqrt{gH} , the tidal wave should only reach as far as 216,000m at $t = 10,800s$ when $L = 648,000m$.

The graphs in Figures 7 and 8 show the computed water height and velocity of water at two instances compared with numerical results reported in [31].

Subcritical Flow Over a Bump. A 1D steady flow in a 25m long channel with a bump is considered [32]. The bump profile is defined by

$$Z(x) = \begin{cases} 0.2 - 0.05(x-10)^2 & \text{if } 8 \leq x \leq 12 \\ 0 & \text{otherwise} \end{cases} \quad (48)$$

The water elevation $h = 2.0$ m is imposed at the downstream boundary, and a discharge $Q = 4.42m^3/s$ is imposed at the upstream boundary. A rectangular domain of 1000×1 uniform four-node quadrilateral elements is used for computing. The initial condition is assumed to be

$$H(x,0) = 2.0 - Z(x), \quad u(x,0) = 2.21 \text{ m/s} \quad (49)$$

Note that according to our convention, as displayed in Figure 1, $H = h - Z$.

Bernoulli's equation can be applied between the inlet (*i.e.*, $x = 0$) and any x location of the domain) under the assumption that the pressure distribution is hydrostatic. Therefore, the analytical solution of the case can be found as

$$\frac{Q^2}{2g b^2 h(0)^2} + h(0) - Z(0) = \frac{Q^2}{2g b^2 h(x)^2} + h(x) - Z(x) \quad (50)$$

where b is the width of the domain. The value of $h(x)$ can be calculated iteratively. Note that the top surface pressure is assumed to be the same everywhere. Comparison of the computed steady-state water surface with the analytic solution plotted in Figure 9, shows an excellent agreement with analytical solution. The global percentage of error of the numerical solution is less than 0.7%, calculated by

$$R = \sqrt{\sum_i \left(\frac{h_{analytical,i} - h_{numerical,i}}{h_{analytical,i}} \right)^2} \times 100\%. \quad (51)$$

2D Dam Break. We have modeled a partial dam-break or rapid opening of a sluice gate with non-symmetric breach. This problem was previously modeled and discussed in [32-34]. The computational domain is 200×200 m with a flat horizontal bottom. A dam with 10m in thickness is located in the middle of the domain. The initial upstream and downstream water heights are 10m and 5m, respectively. At $t = 0$, the dam fails and water is released through the 75m wide non-symmetric breach. Water rushes by forming a bore wave that propagates downstream while spreading laterally and a depression wave that moves upstream.

Dam layout for a 501×501 four-node quadrilateral elements mesh is shown in Figure 10. Along the dam thickness, odd number (25) elements are used for better initial condition implementation across the water height discontinuity. Therefore, the discontinuity line at $x = 100m$ falls along the center of relevant elements. The boundaries 1-4 in Figure 10a have symmetry boundary conditions. The dam walls, boundaries 5 and 6 in Figure 10a, have no-slip boundary condition. For water height, H , no flux boundary condition is used for all boundaries. A time step of 0.02s is used for all simulation. Note that the solutions are non-dimensionalized using reference length equal to dam span (75m) and reference velocity equal to square root of gravity (9.81 m/s^2) multiplied by dam span.

Simulation starts with flow at rest initially. As simulation progresses, a bore wave travels to the right, while a depression wave travels to the left. As flow separates from the truncated dam walls just downstream of the breach, it forms counter-rotating eddy, as depicted in Figure 10b. As displayed in Figure 10c, the water depth at the center of these eddies is less than 5m (or, 0.067 normalized), the initial water depth of the lower basin.

We have computed this problem using our hybrid scheme with shock capturing for first and second order accurate in space and time. The computed water elevations after 4s and 8s are shown in series of images in Figure 11. The form and propagating speed of the waves and flow field agree well with those obtained by Liang et al. [32]; Mingham and Causon [33]; and Fennema and Chaudhry [34].

ACKNOWLEDGMENTS

Work is funded by US Department of Homeland Security and Northrop Grumman Ship Building.

REFERENCES

- [1] Bermudez A, Vazquez ME. Upwind Methods for Hyperbolic Conservation Laws with Source Terms. *Computers and Fluids* Vol. 23, No. 8, 1049 – 1071 (1994).
- [2] R. J. LeVeque. *Finite Volume Methods For Hyperbolic Problems*. Cambridge Texts in Applied Mathematics. Cambridge University Press, Cambridge, United Kingdom, 2002.

- [3] Bale D, LeVeque RJ, Mitran S, Rossmanith J. A Wave-propagation Method for Conservation Laws or Balance Laws with Spatially Varying flux functions. *SIAM J. Sci. Comp.*, 24: 266-278, 2002.
- [4] Buard T, Gallouet T, Herard JM. A naive Godunov scheme to solve the shallow water equations. *CR Acad Sci Paris*, 1(326): 885-890, 1998.
- [5] Gallouet T, Jean-Marc H. Some Approximate Godunov Schemes to Compute Shallow Water Equations with Topography. *Computers and Fluids*, 32: 479-513, 2003.
- [6] Kurganov A, Levy D. Central Upwind Schemes for the St. Venant System. *Mathematical Modeling and Numerical Analysis*, 36: 397-425, 2002.
- [7] Botta N, Klein R, Langenberg S, Lutzenkirchen S. Well Balanced Finite Volume Methods for Nearly Hydrostatic Flows. *J. Comp. Phys.*, 196 (2004), pp. 539-565.
- [8] Roe PL. Approximate Riemann Solvers, Parameter Vectors and Difference Schemes. *J. Comput. Phys.* 43, 357 – 372 (1981).
- [9] Harten A, Lax PD, van Leer B. On Upstream differencing and Godunov-type Schemes for Hyperbolic Conservation Laws. *SIAM Review*, 25: 235-261, 1983.
- [10] Sweby PK. High Resolution Schemes Using Flux Limiters for Hyperbolic Conservation Laws. *SIAM J. Num. Anal.* 21, 995 (1984).
- [11] Lynch DR, Gray WG. A Wave Equation Model for Finite Element Tidal Computations. *Computers and Fluids*. 7(3): 207-228, (1979).
- [12] Kawahara M, Kashiyama K. Selective Lumping Finite Element Method for Near Shore Current. *International Journal of Numerical Methods in Fluids* 4:71-97 (1984).
- [13] Kinnmark IPE. The Shallow Water Wave Equations: Formulations, Analysis, and Application. Lecture notes in engineering, Vol. 15, Springer-Verlag, Berlin (1986).
- [14] Luettich RA, Westerink JJ, Scheffner NW. ADCIRC: An Advanced Three-dimensional Circulation Model for Shelves, Coasts and Estuaries; Report 1: Theory and Methodology of ADCIRC-2DDI and ADCIRC-3DL, Technical Report DRP-92-6, Dept. of the Army, USACE. Washington, D.C. (1991).
- [15] Westerink JJ, Luettich RA, Blain CA, Scheffner NW. An Advanced Three-dimensional Circulation Model for Shelves, Coasts and Estuaries; Report 2: Users Manual for ADCIRC-2DDI, Dept. of the Army, USACE. Washington, D.C. (1992).
- [16] Kolar RL, Gray WG, Westerink JJ, Luettich RA. Shallow Water Modeling in Spherical Coordinates: Equation Formulation, Numerical Implementation and Application. *Journal of Hydraulic Research*. 32(1): 3-24 (1994).
- [17] Tu S, Aliabadi S. Development of a Hybrid Finite Volume/Element Solver for Incompressible Flows. *International Journal of Numerical Methods in Fluids*. 2007; 55: 177-203.
- [18] Tu S, Aliabadi S, Patel R, Watts M. An Implementation of the Spalart-Allmaras DES Model in an Implicit Unstructured Hybrid Finite Volume/Element Solver for Incompressible Turbulent Flow. *International Journal of Numerical Methods in Fluids*. 2009; 59: 1051-1062.
- [19] Wan T, Aliabadi S, Bigler C. A Hybrid Scheme Based on Finite Element / Volume Methods for Two Immiscible Fluid Flows. It will appear in the *International Journal of Numerical Methods in Fluids*.
- [20] Wan T., Aliabadi S, Bigler C. A Parallel Hybrid Finite Element / Volume Methods for Two Immiscible Fluid Flows. 19th AIAA Computational Fluid Dynamics Conference, June 22-25, 2009, San Antonio, TX. AIAA Paper Number: 2009-3887.

- [21] Guermond JL, Minev P, Shen J. An Overview of Projection Methods for Incompressible Fows. *Computer Methods in Applied Mechanics and Engineering* 2006; 195:6011–6045.
- [22] Timmermans LJP, Minev PD, van de Vosse FN. An Approximate Projection Scheme for Incompressible Flow Using Spectral Elements. *International Journal of Numerical Methods in Fluids* 1996; 22:673–688.
- [23] Aliabadi S, Johnson AA, Abedi J. Stabilized-finite-element/interface-capturing Technique for Parallel Computation of Unsteady Flows with Interfaces. *Computer & Fluids* 2003; 32:535–545.
- [24] Aliabadi S, Tu S, Watts MD, Ji A, Johnson AA. Integrated High Performance Computational Tools for Simulations of Transport and Diffusion of Contaminants in Urban Areas. *International Journal of Computational Fluid Dynamics* 2006; 20(3–4):253–267.
- [25] Mathur SR, Murphy JY. A pressure-based method for unstructured meshes. *Numerical Heat Transfer, Part B* 1997; **31**:195-215.
- [26] Saad Y. *Iterative methods for sparse linear systems*, PWS Publishing Company, 1996.
- [27] Luo H, Baum J, LÄohner R. A Fast Matrix-free Implicit Method for Compressible Flows on Unstructured Grids. *J. Comput. Phys.* 1998; 146:664-690.
- [28] Aliabadi S. *Parallel Finite Element Computations in Aerospace Applications*. Ph.D. thesis, University of Minnesota, 1994.
- [29] Stoker JJ. *Water Waves*, Wiley Interscience (1957).
- [30] Glaister P. *Difference Schemes for the Shallow Water Equations*, Numerical Analysis Report 9/87, University of Reading (1987).
- [31] Hudson J. *Numerical Techniques for Morphodynamic Modeling*. PhD Thesis, Department of Mathematics, University of Reading, October 2001.
- [32] Liang SJ, Tang JH, Wu MS. Solution of Shallow Water Equations Using Least Squares Finite Element Method. *Acta Mech. Sin.* 24, 523–532 (2008)
- [33] Mingham CG, Causon DM. High-resolution Finite-volume Method for Shallow Water Flows. *ASCE J. Hydraul. Eng.* 124(6), 605–614 (1998)
- [34] Fennema RT, Chaudhry MH. Explicit Methods for 2D Transient Free-surface Flows. *ASCE J. Hydraul. Eng.* 116(11), 1013– 1034 (1990)

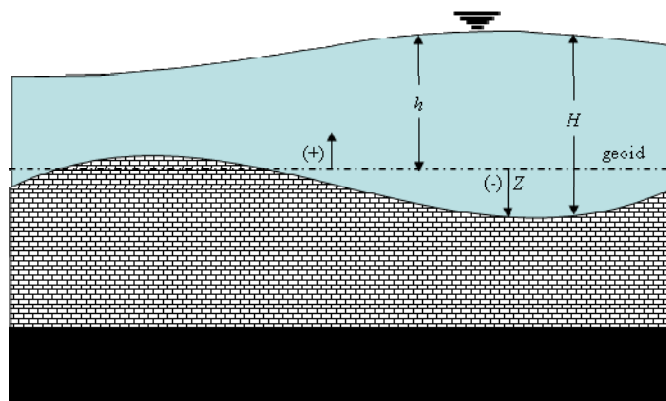


Figure 1. Problem Description.

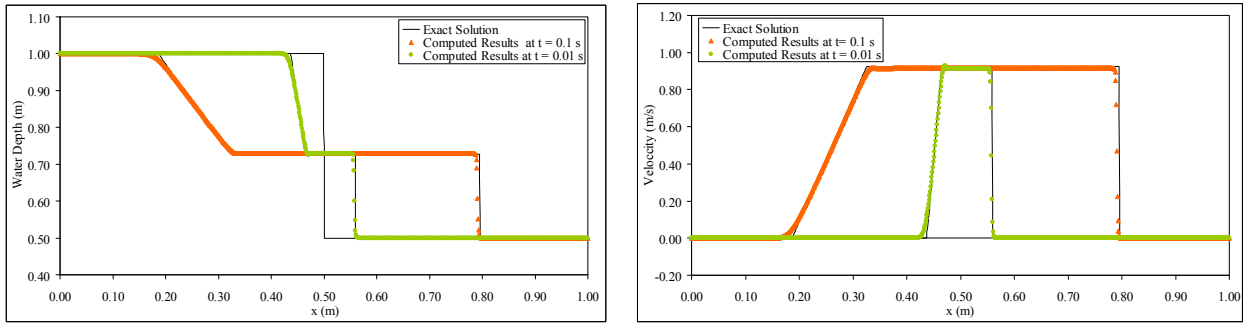


Figure 2. Dam Break on a Flat Riverbed. Numerical solution compared to the exact solution at $t = 0.01s$ and $t = 0.1s$ using hybrid scheme first order accurate in space and time.

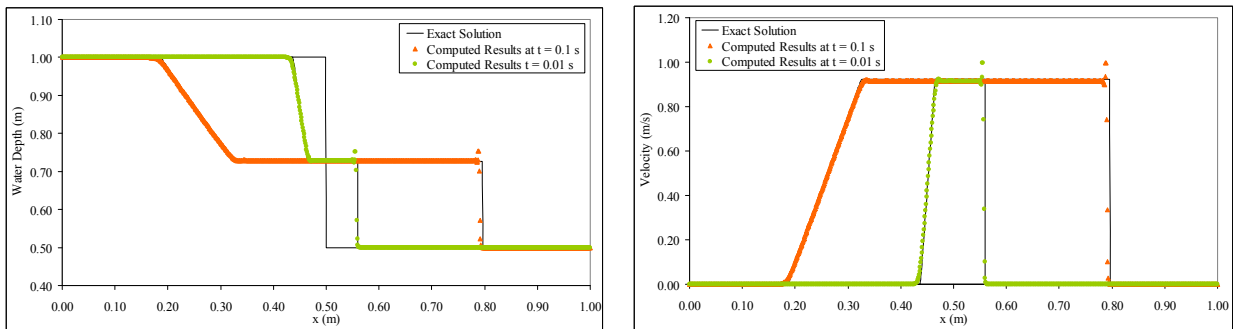


Figure 3. Dam Break on a Flat Riverbed. Numerical solution compared to the exact solution at $t = 0.01s$ and $t = 0.1s$ using hybrid scheme second order accurate in space and time.

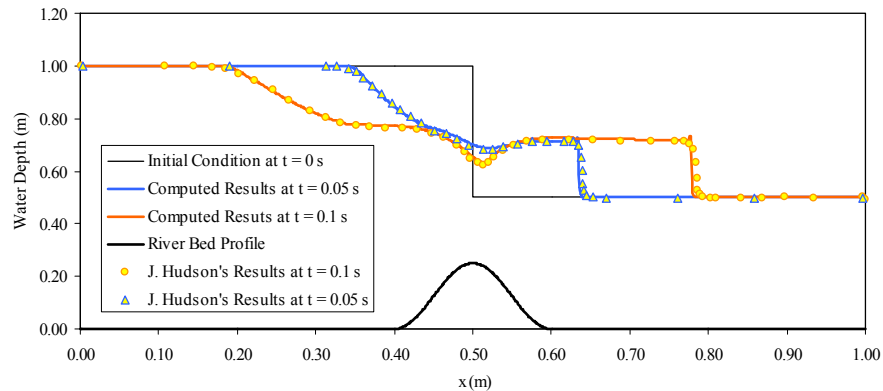


Figure 4. Dam-Break on a Variable Depth Riverbed. Computed water depth compared to the target solution reported in [31] at $t = 0.01s$ and $t = 0.1s$ using hybrid scheme second order accurate in both space and time.

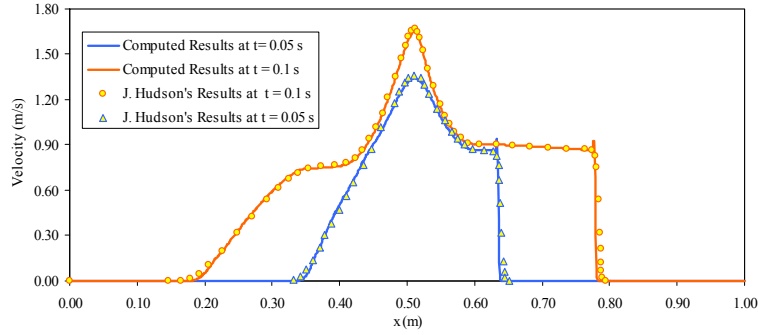


Figure 5. Dam-Break on a Variable Depth Riverbed. Computed velocity compared to the target solution reported in [31] at $t = 0.01s$ and $t = 0.1s$ using hybrid scheme second order accurate in both space and time.

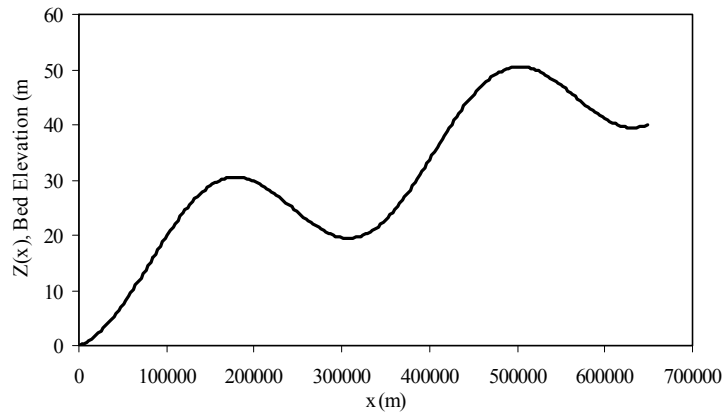


Figure 6. Tidal Wave Propagation on a Variable Depth Bed. The graph shows the bed elevation.

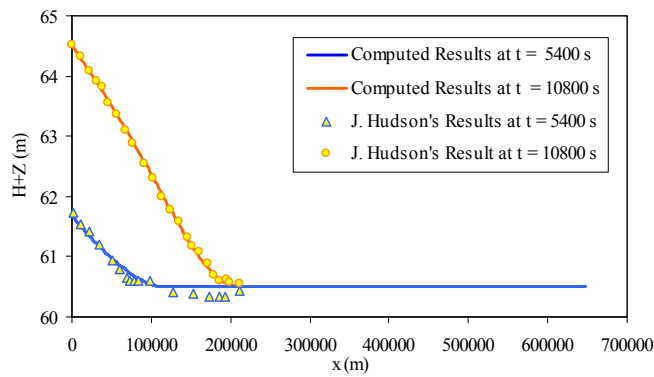


Figure 7. Tidal Wave Propagation on a Variable Depth Bed. The image shows the computed water height ($H + Z$) at two instances compared with numerical results reported in [31].

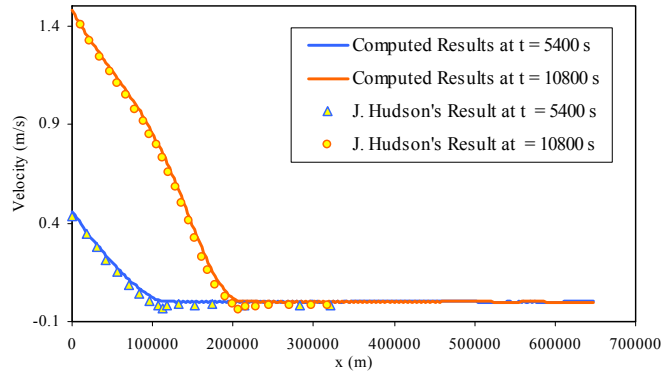


Figure 8. Tidal Wave Propagation on a Variable Depth Bed. The image shows the computed velocity at two instances compared with numerical results reported in [31].

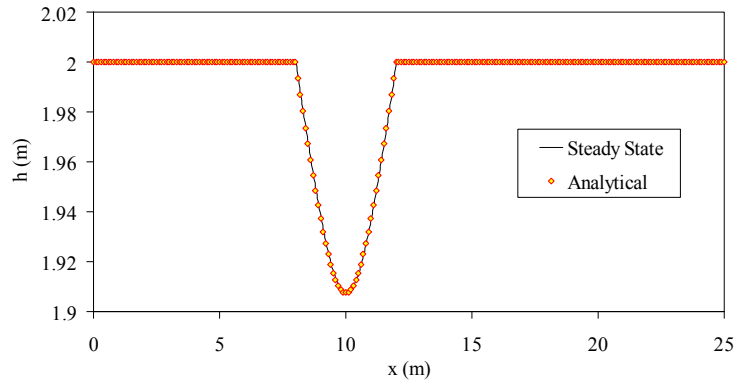
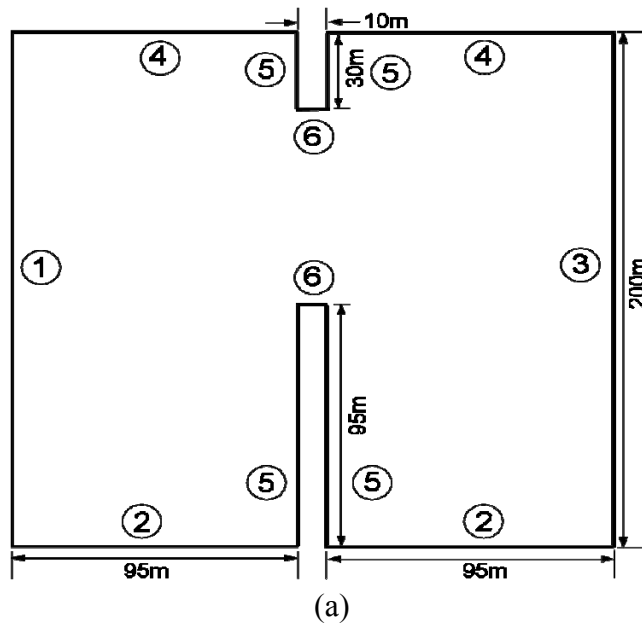
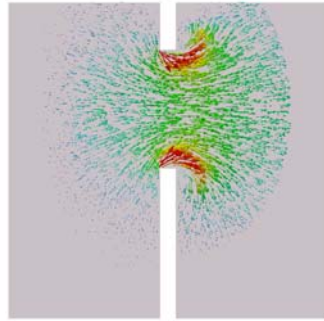
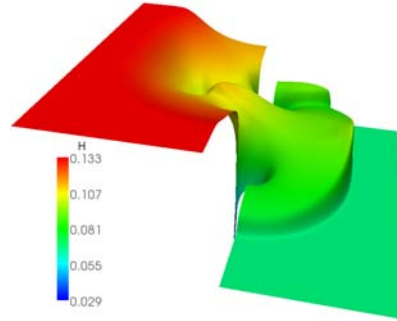


Figure 9. Subcritical Flow Over a Bump. Comparison of simulated and analytical solutions of water surface profile at steady-state.





(b)



(c)

Figure 10. 2D Dam Break. (a) Problem description and boundary conditions, (b) Velocity vector showing the formation of eddies after 7.2s, (c) Water height after 7.2s showing height plunge in the eddy locations.

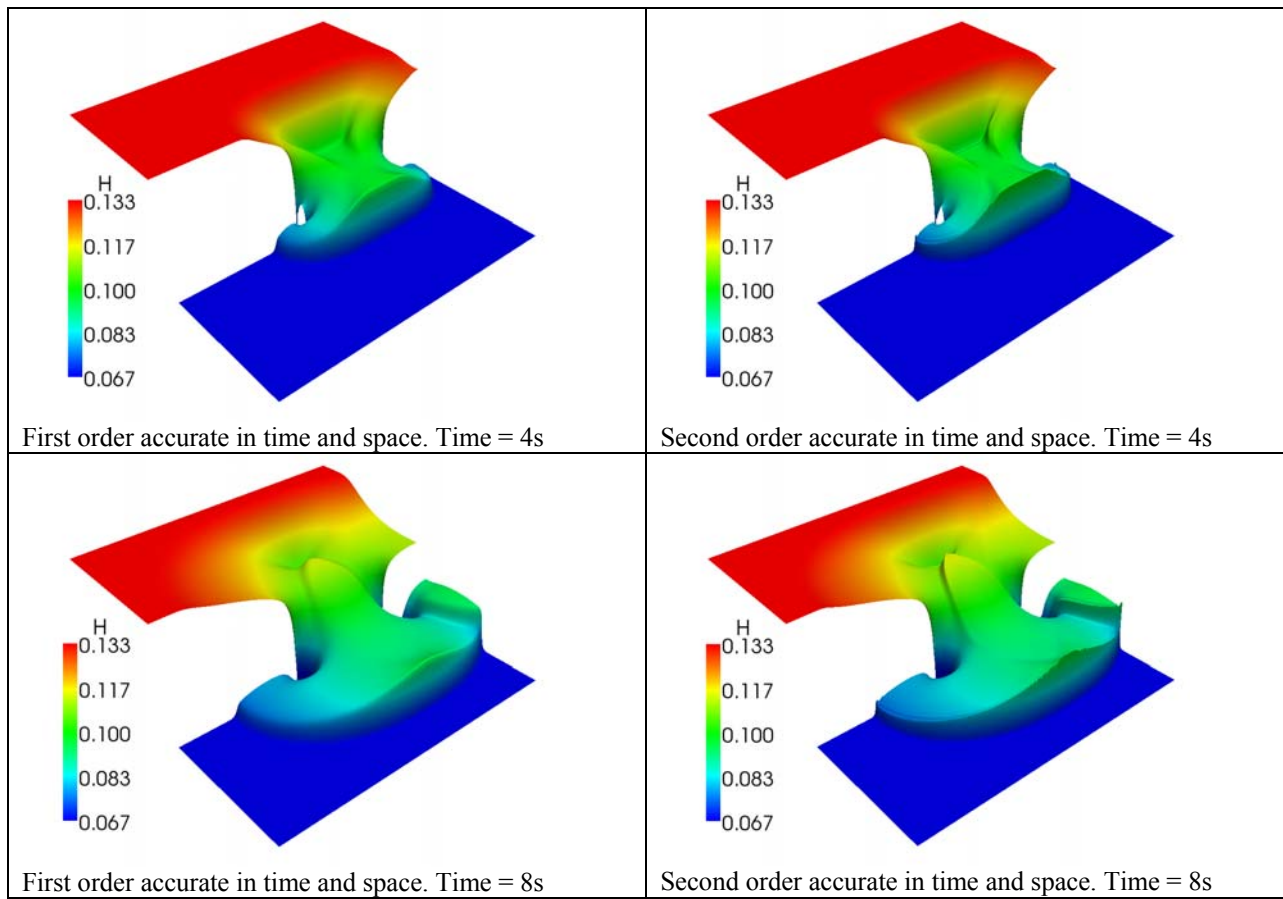


Figure 11. 2D Dam Break. The images on the left and right show the computed water height after 4s and 8s using hybrid scheme first and second order accurate in time and space, respectively.