INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2002; **40**:1237–1273 (DOI: 10.1002/fld.402)

Performance of finite volume solutions to the shallow water equations with shock-capturing schemes

K. S. Erduran*,†, V. Kutija‡ and C. J. M. Hewett§

WRSRL, Department of Civil Engineering, University of Newcastle, Newcastle upon Tyne, NE1 7RU, U.K.

SUMMARY

Numerical methods have become well established as tools for solving problems in hydraulic engineering. In recent years the finite volume method (FVM) with shock capturing capabilities has come to the fore because of its suitability for modelling a variety of types of flow; subcritical and supercritical; steady and unsteady; continuous and discontinuous and its ability to handle complex topography easily.

This paper is an assessment and comparison of the performance of finite volume solutions to the shallow water equations with the Riemann solvers; the Osher, HLL, HLLC, flux difference splitting (Roe) and flux vector splitting. In this paper implementation of the FVM including the Riemann solvers, slope limiters and methods used for achieving second order accuracy are described explicitly step by step. The performance of the numerical methods has been investigated by applying them to a number of examples from the literature, providing both comparison of the schemes with each other and with published results. The assessment of each method is based on five criteria; ease of implementation, accuracy, applicability, numerical stability and simulation time. Finally, results, discussion, conclusions and recommendations for further work are presented. Copyright © 2002 John Wiley & Sons, Ltd.

KEY WORDS: finite volume; shallow water equations; shock-capturing; Riemann solvers; second order accuracy; source terms

1. INTRODUCTION

Numerical methods such as the finite difference, finite element and finite volume methods have become well established tools in physics and engineering. While finite difference schemes such as the Preissmann and Abbott-Ionescu schemes [1] are widely used in hydraulic engineering, they are not suitable for certain types of problem, particularly flows which exhibit discontinuities. These limitations can be dealt with by introducing shock capturing schemes [2, 3]. However, shock capturing capabilities are much easier to implement in the finite volume method

* Correspondence to: K. S. Erduran, WRSRL, Department of Civil Engineering, University of Newcastle, Newcastle-upon-Tyne, NE1 7RU, U.K.
† E-mail: Kutsi.Erduran@ncl.ac.uk
‡ E-mail: Vedrana.Kutija@ncl.ac.uk
§ E-mail: C.J.M.Hewett@ncl.ac.uk

Contract/grant sponsor: Turkish Republic Ministry of National Education

Received April 2001
Revised 6 April 2002

(FVM) because it uses the conservative form of the governing equations [4]. Moreover, the FVM has the advantage over the finite difference method that it can be implemented on any type of grid, structured or unstructured [5], and the advantage over the finite element that it requires significantly less computational effort [6]. The FVM is based on discretization of the computational domain into smaller sub-domains over which the equations are integrated. Shock capturing capabilities are provided by solving a series of one dimensional Riemann problems on the boundary of each sub domain.

FVM with shock-capturing schemes applied to the shallow water equations is thus suitable for modelling a variety of types of flow; subcritical and supercritical; steady and unsteady; continuous and discontinuous and it is thus emerging as a powerful tool in computational hydraulics. In recent years, increased attention has been paid to the use of the finite volume method (FVM) with shock-capturing schemes for hydraulic engineering problems. Most examples given in literature are simulations of discontinuous flows. The simulation of dam break problems is one of the most popular [6], followed by the simulation of an oblique shock wave [7], tidal wave propagation [8], hydraulic jumps [9], and sustained hydraulic jumps [10]. There are also other examples such as the simulation of flow in initially dry areas [11], and in a meandering channel [12, 13]. Recently a river bed deformation due to cut off has been simulated by Zhao *et al*. [14]. Shock capturing schemes have been investigated by a number of authors. Zhao *et al.* [6] compared the accuracy, stability and simulation time of three first order approximate Riemann solvers; flux–vector splitting (FVS), Roe and Osher. Delis *et al.* [15] compared the applicability and validity of four approximate Riemann solvers including the Lax-Friedrichs Riemann solver. In addition, a comparison between first and second order accurate solution of flux difference scheme (Roe) can be found in Kumar Jha *et al.* [16]. In several articles, the application of each scheme using first and second order accurate solutions can be found. For instance, use of the first order Osher scheme is reported by Zhao *et al.* [6], the second order HLL scheme is used by Mingham and Causon [5], Causon *et al*. [17] and Mingham and Causon [18], a first order HLLC scheme is used by Zoppou and Roberts [11], first and second order accurate Osher scheme are used by Erduran *et al*. [13], the second order Roe scheme is applied by Alcrudo and Garcia-Navarro [19] and first order FVS is given by Shen *et al*. [20] as an example of one of the numerical engines in their model RBFVM-2D.

This paper is an assessment and comparison of the performance of finite volume solutions to the shallow water equations with five Riemann solvers found in the literature [21–28], all of which are based on the theory of characteristics; the Osher, HLL, HLLC, Roe (flux difference splitting) and flux vector splitting schemes. In Sections 2, 3, and 4 the finite volume method, Riemann solvers, and methods used for achieving second order accuracy are described, respectively. The theoretical bases are avoided and only main features of the solution schemes are given; the authors intend these three sections to be particularly useful to non-specialists as the method is explained in simple terms and the implementation is described explicitly step by step. In Section 5 the features of SHAW5o (SHAllow water equations solved by finite volume method), a tool developed at Newcastle University, are introduced. In Section 7 the performance of the numerical methods is assessed by applying them to a number of examples from the literature described in Section 6, providing both comparison of the schemes with each other and with published results. The assessment is made according to five criteria; ease of implementation, accuracy, applicability to a variety of problems, numerical stability and simulation time. These cases are chosen with care so that specific goals can be achieved.

The aim is to assess and compare the performance of the schemes with respect to accuracy, applicability, stability and computation time. There are some examples that not all of the schemes presented can handle such as problems involving rarefaction shock waves. The significance of such errors is investigated. Since the solution scheme is explicit there is a requirement for a limit on time step in order to ensure stability. Although stability analysis is based on experimental work, results have been compared with stability conditions defined in the literature. Placing a limit on the time step has implications for how long a simulation takes to run. Therefore, the time taken to complete each simulation is also recorded. Specific examples are used to compare different slope limiters and different second order time integration approaches. Finally, two important problems are discussed; first, the friction computation, which is based on a wide channel approximation; and second, the use of splitting techniques, where the balance between fluxes and source/sink terms cannot be always achieved and can produce considerable errors. Conclusions and suggestions for further work are given at the end of the paper.

2. METHODOLOGY

The two-dimensional form of the shallow water equations can be written as

$$\frac{\partial h}{\partial t}+\frac{\partial (hv_x)}{\partial x}+\frac{\partial (hv_y)}{\partial y}=0$$

$$\frac{\partial (hv_x)}{\partial t}+\frac{\partial (hv_x^2+gh^2/2)}{\partial x}+\frac{\partial (hv_xv_y)}{\partial y}=gh(So_x-Sf_x) \tag{1}$$

$$\frac{\partial (hv_y)}{\partial t}+\frac{\partial (hv_xv_y)}{\partial x}+\frac{\partial (hv_y^2+gh^2/2)}{\partial y}=gh(So_y-Sf_y)$$

where h is the water depth, v_x, v_y represent the depth-averaged velocity components in the x and y directions, g is the acceleration due to gravity, So_x and Sf_y are the bed slopes and friction terms, respectively, in the x direction and similarly for So_y and Sf_y in the y direction (Figure 1).

Denoting $p_1=h$, $p_2=hv_x$, $p_3=hv_y$ and defining the conserved physical vector $\mathbf{p}=[p_1, p_2, p_3]^{\mathrm{T}}$, the conservative vector form of the shallow water equations can be written in vector notation as follows:

$$\frac{\partial \mathbf{p}}{\partial t}+\frac{\partial \mathbf{f}(\mathbf{p})}{\partial x}+\frac{\partial \mathbf{g}(\mathbf{p})}{\partial y}=\mathbf{b}(\mathbf{p}) \tag{2}$$

where $\mathbf{f}(\mathbf{p})$, $\mathbf{g}(\mathbf{p})$ are the flux vectors in the x and y directions, respectively, and $\mathbf{b}(\mathbf{p})$ denotes source/sink terms.

Equation (2) can also be written in the compact–conservative form given below by denoting $\mathbf{F}(\mathbf{p})=[\mathbf{f}(\mathbf{p})\mathbf{g}(\mathbf{p})]^{\mathrm{T}}$

$$\frac{\partial \mathbf{p}}{\partial t}+(\nabla.\mathbf{F})^{\mathrm{T}}=\mathbf{b}(\mathbf{p}) \tag{3}$$

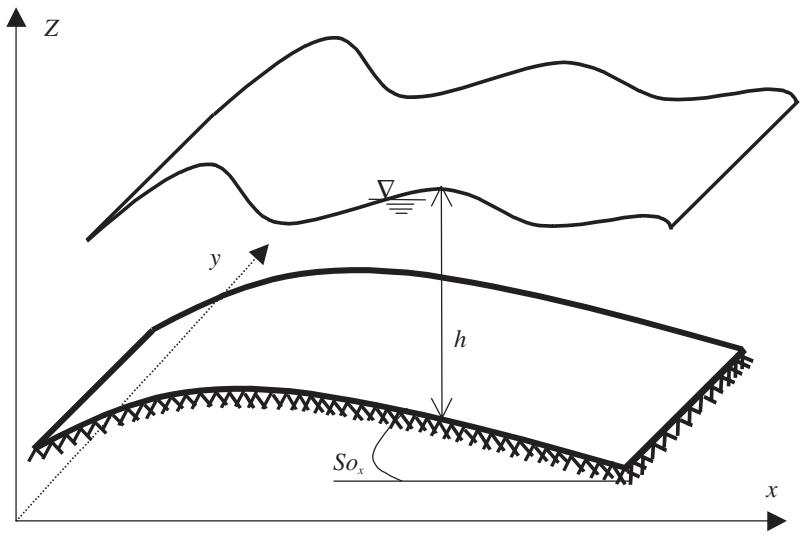

Figure 1. Water depth, channel bottom and slope.

The FVM is based on integration of the equations of interest over each finite volume (cell) covering the computational domain. Thus,

$$\int_V \left(\frac{\partial \mathbf{p}}{\partial t} + (\nabla . \mathbf{F})^{\mathrm{T}} \right) \mathrm{d}V = \int_V \mathbf{b}(\mathbf{p}) \, \mathrm{d}V \tag{4}$$

where V represents the volume over which integration is performed.

Assuming that $\mathbf{p}$ varies with time but is constant over the cell, applying the divergence theorem to the second term on the left hand side of Equation (4), using the rotational invariance property between $\mathbf{f}(\mathbf{p})$ and $\mathbf{g}(\mathbf{p})$ on each side of the cell and resolving $\mathbf{F}(\mathbf{p})$ in the direction of the normal vector $\mathbf{n}$, the 2D equations are reduced to a number of 1D local problems which are solved separately; one across each cell boundary. In discretized form these are given by

$$A \frac{\mathrm{d}\mathbf{p}}{\mathrm{d}t} = -\sum_{k=1}^{m} \mathbf{T}^{-1}(\theta^k) \mathbf{f}^k(\mathbf{q}^k) L^k + A\mathbf{b}(\mathbf{p}) \tag{5}$$

where A is the area of the cell, m is the number of sides of the cell, k is an index that represents the side, θ^k is the angle between the outward normal vector $\boldsymbol{n}$ and the x axis, $\mathbf{T}(\theta^k)$ is the transformation matrix which can be obtained by rotating the co-ordinate axes, $\mathbf{T}^{-1}(\theta^k)$ is the inverse transformation matrix, L^k is the length of the kth cell side, $\mathbf{q}^k$ is the transformed conserved physical vector obtained by multiplying $\mathbf{p}$ by the transformation matrix and $\mathbf{f}^k(\mathbf{q}^k)$ is the transformed numerical flux vector. For convenience, hereafter superscript k will be omitted. It will only be used for L^k as a reminder. We have

$$\mathbf{q} = \mathbf{T}(\theta)\mathbf{p} = [h, hu, hv]^{\mathrm{T}} \quad \text{and} \quad \mathbf{f}(\mathbf{q}) = [hu, hu^2 + gh^2/2, huv]^{\mathrm{T}}$$

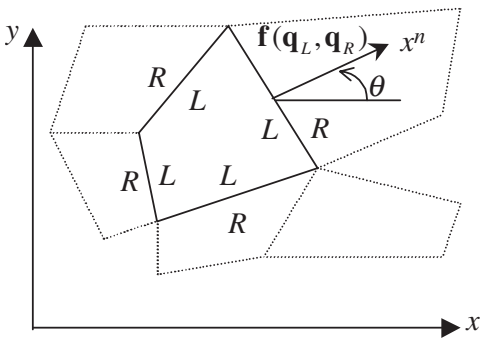

Figure 2. Typical finite volume cell and Riemann interface.

where u, v are local components of velocity in the normal and tangential directions respectively given by

$$u = v_x \cos\theta + v_y \sin\theta, \quad v = -v_x \sin\theta + v_y \cos\theta$$

Each of these 1D local problems is a Riemann problem defined to be an initial value problem which exhibits an abrupt change in the flow variables. The Riemann problem is characterised by two states, for which the conservative variables have known values at the beginning of a time step, separated by a line called the Riemann interface. If we let x^n be a local co-ordinate normal to the cell side the Riemann problem can be written:

$$\frac{\partial \mathbf{q}}{\partial \mathrm{t}} + \frac{\partial[\mathbf{f}(\mathbf{q})]}{\partial x^n} = \mathbf{0} \tag{6}$$

The initial state is given by

$$\mathbf{q}(x^n, 0) = \begin{cases} \mathbf{q}_\mathrm{L}; & x^n < 0 \\ \mathbf{q}_\mathrm{R}; & x^n > 0 \end{cases}$$

where $\mathbf{q}_\mathrm{L}$ and $\mathbf{q}_\mathrm{R}$ denote the values of the transformed conserved physical vector to the left and right of the cell interface, respectively. The inside of the cell under consideration always corresponds to the left hand side of the Riemann interface, and the neighbouring cell to the right hand side as shown in Figure 2.

Equation (6) is solved numerically, giving the normal flux through each cell boundary, denoted $\mathbf{f}(\mathbf{q}_\mathrm{L}, \mathbf{q}_\mathrm{R})$ in Figure 2. For more detailed information about the solution presented so far, see Zhao *et al.* [6], Tan [25], Hirsch [26].

In practice, it is not always possible to use the FVM to solve Equation (5) directly. In such cases the source/sink terms are dealt with separately. This is achieved using a splitting technique. The shallow water equations given in Equation (2) are split into two parts; the homogeneous part, which consists of only the terms on the left-hand side of the equation, and the inhomogeneous part, which is of the form:

$$\frac{\partial \mathbf{p}}{\partial t} = \mathbf{b}(\mathbf{p}) \tag{7}$$

The homogeneous part is solved using Euler's method, giving the following expression:

$$\mathbf{p}^{n+1} = \mathbf{p}^n - \frac{\Delta t}{A} \sum_{k=1}^{m} \mathbf{T}^{-1}(\theta)\mathbf{f}(\mathbf{q})L^k \tag{8}$$

where $\mathbf{p}^{n+1}$ shows values at the next time step and Δt is the length of the time step. This equation shows that, in order to find the conservative variables h, hv_x, hv_y at the next time step, first the numerical fluxes for each cell side are computed, they are then multiplied by the inverse transformation matrix and the length of the corresponding cell side and summed.

The inhomogeneous part, which is the set of ordinary differential equations (ODEs) given in Equation (7), can be solved by any suitable method. Here a fourth order Runge Kutta method is adopted.

Note that the solution to these ODEs uses as initial values the conservative variables obtained from Equation (8). Hence, the solution of Equation (7) gives the corrected values at the next time step. It may be worth mentioning that, although the fourth order Runge Kutta method is used for the solution of ODEs, the solution of the system is still first order accurate as shown in Equation (9).

$$\mathbf{p}^{n+1} = \mathrm{S}^{(\Delta t)}[\mathrm{H}^{(\Delta t)}[\mathbf{p}^n]] \tag{9}$$

where $\mathrm{S}^{(\Delta t)}$ and $\mathrm{H}^{(\Delta t)}$ are operators which correspond to solutions to the inhomogeneous (source/sink terms) and the homogeneous parts, respectively.

Briefly, for a first order accurate solution, the following steps are taken:

(a) Solve Equation (6) numerically to calculate fluxes.
(b) Use Equation (8) to find the new values of the conservative variables (ignoring the source/sink terms).
(c) Solve Equation (7) by taking the values of the conservative variables calculated at the previous time step as initial values in order to find the values at the next time step.

3. RIEMANN SOLVERS

The approximate Riemann solvers presented here are HLL, HLLC, Osher, flux difference scheme (FDS) i.e. Roe, flux vector splitting. The solution of the Riemann problem in these schemes is based on characteristic theory. More information about characteristic theory can be found in Abbott and Minns [1]. In the solution to the 2D shallow water equations there are three characteristic lines corresponding to eigenvalues $\lambda_1, \lambda_2, \lambda_3$ as shown in Figure 3. There are two constant states which have the known values of the conservative variables and which lie below the first and third characteristics, shown as the shaded area in the figure.

Each 1D problem can be written in the form of Equation (6), where the transformed numerical flux vector $\mathbf{f}(\mathbf{q})$ has the form given below:

$$\mathbf{f}(\mathbf{q}) = [hu, hu^2 + gh^2/2, huv]^{\mathrm{T}}$$

Equation (6) can be represented by the quasi-linear equation given below.

$$\frac{\partial \mathbf{q}}{\partial t} + \mathbf{J}\frac{\partial \mathbf{q}}{\partial x} = \mathbf{0} \tag{10}$$

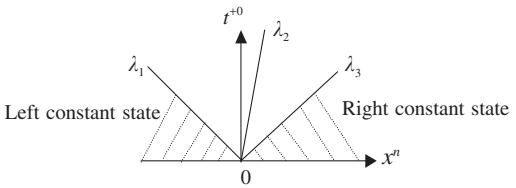

Figure 3. Structure of the Riemann problem in x^n–t^{+0} plane.

where **J** is the Jacobian matrix (also known as the coefficient matrix) given below.

$$\mathbf{J} = \begin{bmatrix} \dfrac{\partial f_1}{\partial q_1} & \dfrac{\partial f_1}{\partial q_2} & \dfrac{\partial f_1}{\partial q_3} \\ \dfrac{\partial f_2}{\partial q_1} & \dfrac{\partial f_2}{\partial q_2} & \dfrac{\partial f_2}{\partial q_3} \\ \dfrac{\partial f_3}{\partial q_1} & \dfrac{\partial f_3}{\partial q_2} & \dfrac{\partial f_3}{\partial q_3} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -u^2 + c^2 & 2u & 0 \\ -uv & v & u \end{bmatrix}$$

Solution of the Riemann problem requires knowledge of the slope of the characteristic lines (eigenvalues). These are found by solving the characteristic equation, $|\mathbf{J} - \lambda \mathbf{I}| = 0$. The solution yields three eigenvalues; $\lambda_1 = u - c,\ \lambda_2 = u,\ \lambda_3 = u + c$ given in order of ascending magnitude, where $c = \sqrt{gh}$. Note that the Jacobian matrix **J** is diagonalizable and the hyperbolic equation (6) can thus be written as a system with real eigenvalues and the diagonalizable coefficient matrix given above. Some Riemann solvers also require the Riemann invariants to be known, which means that the eigenvectors are also required. The eigenvectors, $\boldsymbol{\gamma}$, are obtained from the equation $\mathbf{J}\boldsymbol{\gamma} = \lambda\boldsymbol{\gamma}$ and the solutions for corresponding eigenvalues $\lambda_1,\ \lambda_2,\ \lambda_3$ are $\boldsymbol{\gamma}_1 = [1, u - c, v]^{\mathrm{T}},\ \boldsymbol{\gamma}_2 = [0, 0, 1]^{\mathrm{T}},\ \boldsymbol{\gamma}_3 = [1, u + c, v]^{\mathrm{T}}$, respectively. The Riemann invariants are constant along the characteristics. They can be found by solving the following ordinary differential equations for each eigenvector:

$$\frac{\mathrm{d}q_1}{\gamma_{j1}} = \frac{\mathrm{d}q_2}{\gamma_{j2}} = \frac{\mathrm{d}q_3}{\gamma_{j3}}$$

where j is the number of the eigenvector, γ_{jk} denotes the kth component of $\boldsymbol{\gamma}_j$.

The above equations produce two Riemann invariants for each eigenvector, denoted ψ_j^i, $i = 1, 2$. We have

$$\psi_1^1 = u - 2c, \quad \psi_1^2 = v, \quad \psi_2^1 = u, \quad \psi_2^2 = h, \quad \psi_3^1 = u + 2c, \quad \psi_3^2 = v$$

The aim of solving the Riemann problem is to find the solution in the wedge between the first and third characteristic lines [27]. The Riemann problem must be constructed and solved at each cell interface, thus the number of solutions for each cell equals the number of sides of the cell.

The schemes used here to solve the Riemann problem are upwinded (also known as biased or upstream). It means upwind discretization is applied in the direction of flow. They are monotone (also called positive). In other words, occurrences of local extreme values are

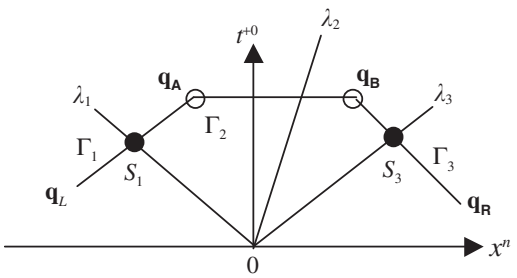

Figure 4. Osher integration paths $(\Gamma_1, \Gamma_2, \Gamma_3)$, intersection points $(\mathbf{q}_A, \mathbf{q}_B)$, sonic points (S_1, S_3) and slope of characteristics.

avoided. More information about these two characteristics can be found in Tan [25]. In the use of the first order accurate solution of the schemes, monotonicity is preserved. However, in the higher order accurate solution, an oscillation (the occurrence of local maximal or minimal value) is observed. One way to preserve monotonicity is to apply the Total Variation Diminishing (TVD) method (see Section 4).

In the following subsections, a step by step solution procedure for the schemes mentioned above is briefly given. For the complete solution, the reader can refer to Toro [21, 27].

3.1. Osher scheme

There are two ways to follow Osher's method. One is the original Osher scheme (sometimes called O-ordering) and the other is known as a physical ordering or P-ordering [21]. In this paper only P-ordering is presented.

In Figure 4, the sonic point (critical point) represents a point where the characteristic speed becomes zero and occurs when the wave is crossed (the sign of the characteristic speed changes). A point where two characteristics intersect is called an intersection point. In order to find these points, the Riemann invariants are used. At points A and B, the velocities and water depths can be given as follows:

$$u_A = u_B = \frac{\psi_{\mathrm{L}} + \psi_{\mathrm{R}}}{2}, \qquad h_A = h_B = \frac{1}{g}\left(\frac{\psi_{\mathrm{L}} - \psi_{\mathrm{R}}}{4}\right)^2$$

and at points S_1, S_3 they can be given as

$$u_{S_1} = \tfrac{1}{3}\psi_{\mathrm{L}}, \quad h_{S_1} = \frac{(u_{S_1})^2}{g}, \quad u_{S_3} = \tfrac{1}{3}\psi_{\mathrm{R}}, \quad h_{S_3} = \frac{(u_{S_3})^2}{g}, \quad v_{\mathrm{L}} = v_A = v_{S_1}, \quad v_{\mathrm{R}} = v_B = v_{S_3}$$

where $\psi_{\mathrm{L}} = u_{\mathrm{L}} + 2\sqrt{gh_{\mathrm{L}}}$, $\psi_{\mathrm{R}} = u_{\mathrm{R}} - 2\sqrt{gh_{\mathrm{R}}}$ are constant across λ_1 and λ_3, respectively.

Each integration path is drawn to be tangential to the corresponding eigenvector. The solution begins with splitting the Jacobian matrix of the shallow water equations into two as one part has only positive eigenvalues and the other has only negative eigenvalues. Similarly, the flux vector can be split. Then, by choosing the direction of integration (left to right or right to left), the integration is completed according to the sign of the eigenvalues. By considering

Table I. Estimation of the normal flux using Osher scheme [4].

Hydraulic Conditions	$u_L - c_L \geqslant 0$ $u_R + c_R \geqslant 0$	$u_L - c_L \geqslant 0$ $u_R + c_R \leqslant 0$	$u_L - c_L \leqslant 0$ $u_R + c_R \geqslant 0$	$u_L - c_L \leqslant 0$ $u_R + c_R \leqslant 0$
$u_A \geqslant 0$ $u_A - c_A \geqslant 0$	$\mathbf{f}(\mathbf{q}_L)$	$\mathbf{f}(\mathbf{q}_L) + \mathbf{f}(\mathbf{q}_R)$ $-\mathbf{f}(\mathbf{q}_{S_3})$	$\mathbf{f}(\mathbf{q}_{S_1})$	$\mathbf{f}(\mathbf{q}_{S_1}) - \mathbf{f}(\mathbf{q}_{S_3})$ $+\mathbf{f}(\mathbf{q}_R)$
$u_A \geqslant 0$ $u_A - c_A \leqslant 0$	$\mathbf{f}(\mathbf{q}_L) - \mathbf{f}(\mathbf{q}_{S_1})$ $+\mathbf{f}(\mathbf{q}_A)$	$\mathbf{f}(\mathbf{q}_L) - \mathbf{f}(\mathbf{q}_{S_1})$ $+\mathbf{f}(\mathbf{q}_A) + \mathbf{f}(\mathbf{q}_R) - \mathbf{f}(\mathbf{q}_{S_3})$	$\mathbf{f}(\mathbf{q}_A)$	$\mathbf{f}(\mathbf{q}_R) + \mathbf{f}(\mathbf{q}_A)$ $-\mathbf{f}(\mathbf{q}_{S_3})$
$u_B \leqslant 0$ $u_B + c_B \geqslant 0$	$\mathbf{f}(\mathbf{q}_L) - \mathbf{f}(\mathbf{q}_{S_1})$ $+\mathbf{f}(\mathbf{q}_B)$	$\mathbf{f}(\mathbf{q}_L) - \mathbf{f}(\mathbf{q}_{S_1}) + \mathbf{f}(\mathbf{q}_B)$ $-\mathbf{f}(\mathbf{q}_{S_3}) + \mathbf{f}(\mathbf{q}_R)$	$\mathbf{f}(\mathbf{q}_B)$	$\mathbf{f}(\mathbf{q}_B) - \mathbf{f}(\mathbf{q}_{S_3})$ $+\mathbf{f}(\mathbf{q}_R)$
$u_B \leqslant 0$ $u_B + c_B \leqslant 0$	$\mathbf{f}(\mathbf{q}_L) - \mathbf{f}(\mathbf{q}_{S_1})$ $+\mathbf{f}(\mathbf{q}_{S_3})$	$\mathbf{f}(\mathbf{q}_L) - \mathbf{f}(\mathbf{q}_{S_1})$ $+\mathbf{f}(\mathbf{q}_R)$	$\mathbf{f}(\mathbf{q}_{S_3})$	$\mathbf{f}(\mathbf{q}_R)$

only the negative eigenvalues, the integral equation from left to right for the general case can be given as follows:

$$\mathbf{f}(\mathbf{q}_L, \mathbf{q}_R) = \mathbf{f}(\mathbf{q}_L) + \int_{\mathbf{q}_L}^{\mathbf{q}_{S_1}} \mathbf{J}^-(\mathbf{q})\,d\mathbf{q} + \int_{\mathbf{q}_{S_1}}^{\mathbf{q}_A} \mathbf{J}^-(\mathbf{q})\,d\mathbf{q} + \int_{\mathbf{q}_A}^{\mathbf{q}_B} \mathbf{J}^-(\mathbf{q})\,d\mathbf{q} + \int_{\mathbf{q}_B}^{\mathbf{q}_{S_3}} \mathbf{J}^-(\mathbf{q})\,d\mathbf{q} + \int_{\mathbf{q}_{S_3}}^{\mathbf{q}_R} \mathbf{J}^-(\mathbf{q})\,d\mathbf{q} \tag{11}$$

where $\mathbf{J}^-$ is a Jacobian matrix, which has only negative or zero eigenvalues, $\mathbf{f}(\mathbf{q}_L, \mathbf{q}_R)$ denotes the normal flux through the cell interface from left to right.

The integration results in the 16 different cases shown in Table I and each one determines the numerical flux through each cell interface. Each case represents a hydraulic condition [4].

An outline of the steps required to solve the Riemann problem using the Osher scheme are given below. For more detail the reader is referred to the literature.

(a) Prepare the left and right initial data for the initial value Riemann problem for each cell interface by computing u_L, v_L and h_L for the left interface and u_R, v_R and h_R for the right interface.
(b) Compute wave speeds for both interfaces c_L, c_R and the Riemann invariants ψ_L, ψ_R.
(c) Compute $u_A, u_B, h_A, h_B, v_A, v_B, c_A, c_B, u_{S_1}, h_{S_1}, u_{S_3}, h_{S_3}, v_{S_1}, v_{S_3}$
(d) Estimate the normal flux at each cell interface by taking the appropriate normal flux definition given in Table I.

3.2. HLL scheme

The HLL scheme was developed by Harten *et al.* [23]. Unlike the other Riemann solvers, the solution takes account only of the left and right characteristics. This results in three states that are separated by two characteristics as shown in Figure 5.

In Figure 5, $\lambda_{\min}, \lambda_{\max}$ refer to the minimum and maximum wave speeds, respectively. Estimation of these is given later.

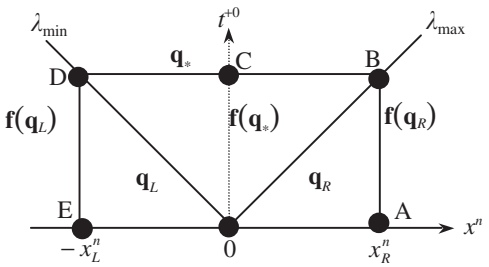

Figure 5. The structure of HLL Riemann solver.

As can be seen from Figure 5, there are three regions; the left region, which lies below and to the left of OD, the right region, which lies below and to the right of OB and the star region, which is between OB and OD. Again the flow variables are known in the left and right regions and they set up the initial values for the Riemann problem. However, in the star region, the flow variables are unknown. Therefore, the target is to find the normal fluxes in the star region.

The HLL solution scheme starts with the application of Green's theorem to Equation (6).

$$\oint (\mathbf{q}\,\mathrm{d}x^n - \mathbf{f}(\mathbf{q})\,\mathrm{d}t) \tag{12}$$

By knowing that $-x_{\mathrm{L}}^n = -\lambda_{\min}\Delta t$, and $x_{\mathrm{R}}^n = \lambda_{\max}\Delta t$, Equation (12) is integrated over the control volume EOCD yielding;

$$\mathbf{q}_* = \mathbf{q}_{\mathrm{L}} + \frac{\mathbf{f}(\mathbf{q}_*) - \mathbf{f}(\mathbf{q}_{\mathrm{L}})}{\lambda_{\min}} \tag{13}$$

Equation (12) is also integrated over OABC and that gives

$$\mathbf{q}_* = \mathbf{q}_{\mathrm{R}} + \frac{\mathbf{f}(\mathbf{q}_*) - \mathbf{f}(\mathbf{q}_{\mathrm{R}})}{\lambda_{\max}} \tag{14}$$

Eliminating $\mathbf{q}_*$ from Equations (13) and (14) gives the normal flux equation for the star region, which is given in Equation (15).

$$\mathbf{f}(\mathbf{q}_*) = \frac{\lambda_{\max}\mathbf{f}(\mathbf{q}_{\mathrm{L}}) - \lambda_{\min}\mathbf{f}(\mathbf{q}_{\mathrm{R}})}{\lambda_{\max} - \lambda_{\min}} + \frac{\lambda_{\max}\lambda_{\min}(\mathbf{q}_{\mathrm{R}} - \mathbf{q}_{\mathrm{L}})}{\lambda_{\max} - \lambda_{\min}} \tag{15}$$

The full solution for the estimation of the normal flux using the HLL scheme is given by

$$\mathbf{f}(\mathbf{q}_{\mathrm{L}}, \mathbf{q}_{\mathrm{R}}) = \begin{cases} \mathbf{f}(\mathbf{q}_{\mathrm{L}}) & \text{if } \lambda_{\min} \geqslant 0, \\ \mathbf{f}(\mathbf{q}_{\mathrm{R}}) & \text{if } \lambda_{\max} \leqslant 0, \\ \mathbf{f}(\mathbf{q}_*) & \text{Otherwise} \end{cases} \tag{16}$$

Now, the solution requires the estimation of minimum and maximum wave speeds. There are different approaches for estimating them [21]. Here, two-rarefaction wave approximation is

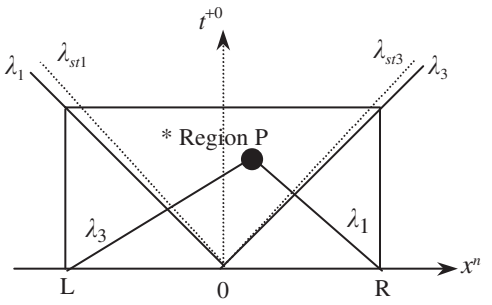

Figure 6. Wave speed estimation.

adopted (there is a smooth transition between the initial constant states). The wave speeds can be computed using the Riemann invariants (Figure 6).

In Figure 6, let $\lambda_{st1}, \lambda_{st3}$ be the negative and the positive characteristic in the star region, respectively (known also as C^-, C^+) and let P be a point in the star region. The task is to find $\lambda_{\min}$, which could be either λ_1 or λ_{st1} and $\lambda_{\max}$, which could be either λ_3 or λ_{st3}. Therefore, the velocity u_{st} and the water depth h_{st} are needed in order to compute the slope of the characteristic in the star region.

As stated previously, along the negative characteristic λ_1 the Riemann invariant is found to be $\psi_1^1 = u - 2c$, and along the positive characteristic λ_3, the Riemann invariant is given by $\psi_3^1 = u + 2c$. Hence, at the point P, the following relationships can be written as

$$u_{\mathrm{L}} + 2\sqrt{gh_{\mathrm{L}}} = u_{st} + 2\sqrt{gh_{st}}, \qquad u_{\mathrm{R}} - 2\sqrt{gh_{\mathrm{R}}} = u_{st} - 2\sqrt{gh_{st}} \tag{17}$$

By adding these two relationships together, the velocity in the star region can be found as follows:

$$u_{st} = \frac{u_{\mathrm{L}} + u_{\mathrm{R}}}{2} + \sqrt{gh_{\mathrm{L}}} - \sqrt{gh_{\mathrm{R}}}$$

and by substracting one from the other, it can be shown that:

$$\sqrt{gh_{st}} = \frac{u_{\mathrm{L}} - u_{\mathrm{R}}}{4} + \frac{(\sqrt{gh_{\mathrm{L}}} + \sqrt{gh_{\mathrm{R}}})}{2}$$

Now, the wave speeds can be given as follows:
$\lambda_{\min} = \min(\lambda_1, \lambda_{st1}), \quad \lambda_{\max} = \max(\lambda_3, \lambda_{st3})$ where $\lambda_1 = u_{\mathrm{L}} - \sqrt{gh_{\mathrm{L}}}$, $\lambda_3 = u_{\mathrm{R}} + \sqrt{gh_{\mathrm{R}}}$, $\lambda_{st1} = u_{st} - \sqrt{gh_{st}}$ and $\lambda_{st3} = u_{st} + \sqrt{gh_{st}}$.

The steps required for solution by the HLL method can be summarized as follows:

(a) Prepare the left and right initial data for the initial value Riemann problem for the each cell interface by computing the transformed velocities u_{L} and h_{L} for the left interface and u_{R} and h_{R} for the right interface.
(b) Compute $\lambda_1, \lambda_3, \lambda_{st1}, \lambda_{st3}$ and find the minimum and maximum wave speeds $\lambda_{\min}$ and $\lambda_{\max}$.
(c) Use Equation (16) to estimate the normal fluxes.

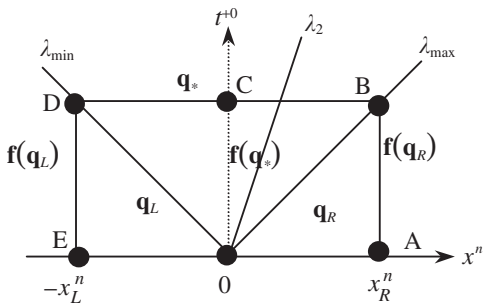

Figure 7. The structure of the Riemann problem in the HLLC scheme.

3.3. HLLC scheme

This scheme is a modified version of the HLL scheme. It was first introduced by Toro *et al.* [28] for solution to the Euler equations. Unlike HLL, the middle characteristic is also considered. The middle wave is also known as a Contact wave, which is why the scheme is named HLLC. The Riemann structure has the form shown in Figure 7.

The normal fluxes are estimated as in the HLL scheme. Computation of the third component ($f_3(\mathbf{q}) = huv$) of the normal flux vector is computed as follows:

If $\lambda_2 \geqslant 0$ then $f_3(\mathbf{q}) = f_1(\mathbf{q})v_L$ if $\lambda_2 \leqslant 0$ then $f_3(\mathbf{q}) = f_1(\mathbf{q})v_R$

More explicitly, the case $\lambda_2 \leqslant 0$ can be given as follows:

$$f_3(\mathbf{q}) = \left[\frac{\lambda_{\max}(h_L u_L) - \lambda_{\min}(h_R u_R)}{\lambda_{\max} - \lambda_{\min}} + \frac{\lambda_{\max}\lambda_{\min}(h_R - h_L)}{\lambda_{\max} - \lambda_{\min}} \right] v_R$$

Fraccorollo and Toro [29] have also given more reliable wave speed estimates in the case of a dry bed.

The basic steps in the HLLC solution are given below:

(a) Prepare the left and right initial data for the initial value Riemann problem for the each cell interface by computing u_L, v_L and h_L for the left interface and u_R, v_R and h_R for the right interface.
(b) Compute wave speeds by considering also dry bed conditions, $\lambda_1, \lambda_3, \lambda_{st1}, \lambda_{st3}, \lambda_2$ and take the maximum and minimum wave speeds and the speed of contact wave.
(c) Use Equation (16) to compute the normal fluxes.

3.4. Roe (flux difference splitting) Scheme

This method, introduced by Roe [24], is sometimes referred to as the flux difference splitting (FDS) method. First, a modified linear coefficient matrix is introduced [19]. There are three conditions that this matrix must satisfy; hyperbolicity, consistency and conservation (see Reference [21]). Then, Equation (6) can be rewritten as

$$\frac{\partial \mathbf{q}}{\partial t} + \mathbf{J}' \frac{\partial \mathbf{q}}{\partial x} = \mathbf{0} \tag{18}$$

where $\mathbf{J}'$ denotes the modified Jacobian matrix. This matrix, its eigenvalues and eigenvectors are written in terms of average values, which are calculated using the variables h, u and v on either side of the Riemann interface. These average values given below are found during the construction of the modified Jacobian matrix providing that three conditions mentioned above are satisfied.

$$u' = \frac{h_R^{1/2}u_R + h_L^{1/2}u_L}{h_R^{1/2} + h_L^{1/2}}, \quad v' = \frac{h_R^{1/2}v_R + h_L^{1/2}v_L}{h_R^{1/2} + h_L^{1/2}} \quad \text{and} \quad c' = \sqrt{\frac{g(h_R + h_L)}{2}}$$

The next step is to write the difference in the conservative variable vector $\mathbf{q}$ across the Riemann interface in terms of the eigenvectors, which gives:

$$\Delta\mathbf{q} = \mathbf{q}_R - \mathbf{q}_L = \sum_{i=1}^{3} \alpha_i' \gamma_i' \tag{19}$$

where the coefficients α_i', $i = 1, 2, 3$ are called the wave strengths.

This equality can also be written as follows:

$$\Delta h = \alpha_1' + \alpha_3', \quad \Delta(hu) = \alpha_1'(u' - c') + \alpha_3'(u' + c'), \quad \Delta(hv) = \alpha_1' v' + \alpha_2' + \alpha_3' v$$

Rearranging these equations to obtain α_i' explicitly in terms of the element $\Delta\mathbf{q}$, we have

$$\alpha_1' = \frac{\Delta h}{2} + \frac{u'\Delta h - \Delta(hu)}{2c'}, \quad \alpha_2' = \Delta(hv) - v'\Delta h, \quad \alpha_3' = \Delta h - \alpha_1'$$

Then the normal fluxes are found by

$$\mathbf{f}(\mathbf{q}_L, \mathbf{q}_R) = \tfrac{1}{2}(\mathbf{f}(\mathbf{q}_L) + \mathbf{f}(\mathbf{q}_R)) - \tfrac{1}{2}\sum_{i=1}^{3} \lambda_i' \alpha_i' \gamma_i' \tag{20}$$

The definition of normal fluxes can also be expressed in alternative ways (see Reference [25]).

Note that the eigenvalues and eigenvectors are of the same form as those given earlier because of the similarity between the Jacobian matrices $\mathbf{J}$ and $\mathbf{J}'$ i.e. $\lambda_1' = u' - c'$, $\lambda_2' = u'$, $\lambda_3' = u' + c'$.

Under some conditions the solution procedure given here violates the entropy condition given below.

$$\lambda_{iL}' > S_i > \lambda_{iR}'$$

where λ_{iL}' and λ_{iR}' denote the average value of the ith characteristic for the left and right hand side of the wave. S_i is a shock speed expressed by the Rankine Hugoniot jump condition given below:

$$\mathbf{f}(\mathbf{q}_R) - \mathbf{f}(\mathbf{q}_L) = S_i(\mathbf{q}_R - \mathbf{q}_L)$$

In the test cases we have investigated that this only causes a problem in the dam break case and this is due to the occurrence of a rarefaction shock i.e. the left wave is a rarefaction ($\lambda_{1L} \leqslant \lambda_{1R}$) and is sonic ($\lambda_1 \approx 0$). The solution produces an unphysical discontinuity. Detailed information about the entropy violation and how to fix it can be found in Toro [21].

As can be seen in the above reference, there are several ways to fix this problem. Here, Harten–Hyman entropy fix [30] is applied. Briefly, in the case of a sonic rarefaction wave, the

corresponding wave speed (if λ_1 corresponds to the left sonic rarefaction, or if λ_2 corresponds to the right sonic rarefaction) is redefined. This solves the problem of the unphysical jump.

The steps in FDS solution are given below:

(a) Prepare the left and right initial data for the initial value Riemann problem for the each cell interface by computing u_L, v_L and h_L for the left interface and u_R, v_R and h_R for the right interface.
(b) Compute the average velocities and celerity u', v', c'.
(c) Compute λ_i' and α_i', $i = 1, 2, 3$.
(d) Fix the entropy violation if it exists.
(e) Compute the normal fluxes using Equation (20).

3.5. *Flux vector splitting scheme*

The flux vector splitting (FVS) method, first introduced by Steger and Warming [31] has a lot in common with the FDS approach. However, as is indicated by the name of the method, the flux vector is split rather than the flux differences. This method generates the same entropy violation as the Roe scheme, which is not fixed in the SHAW5o model but it can be [26]. Tan [25] and later Zhao *et al.* [4] applied the FVS method to the shallow water equations, which required the introduction of an additional equation; an energy equation, in order to ensure flux homogeneity.

Inclusion of the energy equation changes the character of the problem, leading to a different Jacobian matrix to the one discussed earlier [4]. The method proceeds by decomposing the Jacobian according to the sign of the eigenvalues into $\mathbf{J} = \mathbf{J}^+ + \mathbf{J}^-$. The flux vector is also decomposed and written in the split form given below:

$$\mathbf{f}(\mathbf{q}_L, \mathbf{q}_R) = \mathbf{f}^+(\mathbf{q}_L) + \mathbf{f}^-(\mathbf{q}_R)$$

where $\mathbf{q} = (h, hu, hv, e)^T$, $e = gh^2/2 + h((u^2 + v^2)/2)$ and $\mathbf{f}(\mathbf{q}) = (hu, hu^2 + gh^2/2, huv, u(e + gh^2/2))^T$.

The $+$ sign on top of the split flux indicates that it has only positive eigenvalues. Similarly, the $-$ sign represents the flux with negative eigenvalues: $\lambda^+ = \max(\lambda, 0)$ and $\lambda^- = \min(\lambda, 0)$

In computing the normal fluxes, only the first three component of $\mathbf{f}(\mathbf{q}_L, \mathbf{q}_R)$ are required, since the value of the fourth component is not of interest in the solution to the shallow water equations.

$$\mathbf{f}(\mathbf{q}_L, \mathbf{q}_R) = \frac{h}{4}\begin{bmatrix} 2\lambda_2^+ + \lambda_3^+ + \lambda_1^+ \\ 2\lambda_2^+ u_L + \lambda_3^+(u_L + c_L) + \lambda_1^+(u_L - c_L) \\ 2\lambda_2^+ v_L + \lambda_3^+ v_L + \lambda_1^+ v_L \end{bmatrix} + \frac{h}{4}\begin{bmatrix} 2\lambda_2^- + \lambda_3^- + \lambda_1^- \\ 2\lambda_2^- u_R + \lambda_3^-(u_R + c_R) + \lambda_1^-(u_R - c_R) \\ 2\lambda_2^- v_R + \lambda_3^- v_R + \lambda_1^- v_R \end{bmatrix} \tag{21}$$

The steps in the FVS solution method are given below:

(a) Prepare the left and right initial data for the initial value Riemann problem for the each cell interface by computing u_L, v_L and h_L for the left interface and u_R, v_R and h_R for the right interface.
(b) Compute the eigenvalues for the left and right interfaces. Check whether the eigenvalues are greater than or less than 0. If the eigenvalue on the left is greater than 0, take it as it is. Otherwise, take 0 instead. If the eigenvalue on the right is greater than 0, take 0, otherwise take it as it is.
(c) Use expression (21) to find the normal fluxes.

4. SECOND ORDER ACCURACY

In some cases second order accuracy is required, particularly when modelling problems which exhibit significant discontinuities in the flow. Obviously, they generally give better results than first order schemes.

As our problem is also a time dependent problem, second order accuracy in both time and space must be achieved. It will be shown later that second order accuracy in space is achieved by a so-called variable extrapolation approach. Meanwhile, second order accuracy in time can be achieved either using van Leer to Hancock method [32], or a predictor–corrector method given by Hirsch [26].

4.1. Second order accuracy in space

In order to achieve second order accuracy in space, the cell boundary values are used instead of cell centre values as in the first order accurate solution. We use a second order fully one-sided scheme [26]. In other words, one-sided extrapolation at the interface between average values at the two upstream cells is carried out giving the cell boundary values, immediately adjacent to the Riemann interface:

$$\mathbf{p}_L^{i+1/2} = \mathbf{p}_i + \tfrac{1}{2}(\mathbf{p}_i - \mathbf{p}_{i-1}) \tag{22}$$

$$\mathbf{p}_L^{i-1/2} = \mathbf{p}_i - \tfrac{1}{2}(\mathbf{p}_{i+1} - \mathbf{p}_i) \tag{23}$$

where the superscript refers to the position of interest and the subscript refers to the side of the Riemann interface under consideration. Figure 8 shows the left and right interface Riemann data for the one-dimensional case.

Therefore, for this particular one-dimensional problem we have two Riemann interfaces each with a left and right conserved physical vector. Since the values of $\mathbf{p}$ inside the cell are defined earlier, we can define $\mathbf{p}_R$ for both interfaces using:

$$\mathbf{p}_R^{i+1/2} = \mathbf{p}_{i+1} - \tfrac{1}{2}(\mathbf{p}_{i+2} - \mathbf{p}_{i+1})$$

and

$$\mathbf{p}_R^{i-1/2} = \mathbf{p}_{i-1} + \tfrac{1}{2}(\mathbf{p}_{i-1} - \mathbf{p}_{i-2})$$

In the beginning of Section 3, it is said that the second order accurate solution does produce unwanted local maxima (overshoot) or minima (undershoot). In order to avoid this behaviour,

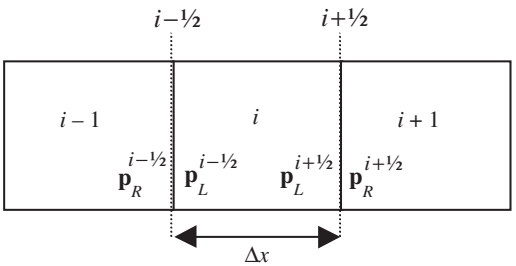

Figure 8. Second order accurate representation of Riemann problem in 1D space.

so called limiters are used. There are two types; flux limiters and slope limiters. Their name comes from the fact that they limit either the fluxes themselves or slope of the Riemann data. Here, we have briefly explained the slope limiters given in the literature and used in our model development.

The numerical generation of oscillations occurs because of the numerical treatment of the second-order approximation. They, in fact, are produced when the slope in a cell becomes larger than the difference between adjacent mean values [26].

The reason limiters are used is to limit the slope of variables in a cell so that there will be no overshoots or undershoots. This is achieved by multiplying the slope of variables by a non-linear function, which varies with the type of limiter. Definition of a suitable non-linear function can be found in the literature [21, 26].

Here, four slope limiters are introduced. The reader is referred to the literature for more detail. These four slope limiters are van Leer, Superbee, Minmod, and van Albada. They are expressed in the literatures as follows:

(a) van-Leer limiter

$$\varphi(r) = \frac{r + |r|}{1 + r}$$

where r is a gradient ratio (gradient of the slopes) for one of the components of $\mathbf{p}$ and $\varphi(r)$ is a non-linear function of r. To avoid an over or undershoot at $i + 1/2$, the ratio for the jth component of $\mathbf{p}$ is given as; $r_{j\mathrm{L}}^{i+1/2} = (\mathbf{p}_{i+1})_j - (\mathbf{p}_i)_j / (\mathbf{p}_i)_j - (\mathbf{p}_{i-1})_j$. For the interface $i - 1/2$, it is expressed as $r_{j\mathrm{L}}^{i-1/2} = (\mathbf{p}_i)_j - (\mathbf{p}_{i-1})_j / (\mathbf{p}_{i+1})_j - (\mathbf{p}_i)_j$. Note that these limited data are inside a cell under consideration.

(b) Superbee limiter

$$\varphi(r) = \max[0, \min(2r, 1), \min(r, 2)]$$

(c) Minmod limiter

$$\varphi(r) = \max[0, \min(1, r)]$$

(d) van Albada limiter

$$\varphi(r) = \frac{r + r^2}{1 + r^2}$$

Finally, Equations (22) and (23) can be rewritten for the first component h as follows:

$$h_{\mathrm{L}}^{i+1/2} = h_i + \tfrac{1}{2}\,\varphi(r_{1\mathrm{L}}^{i+1/2})(h_i - h_{i-1}) \tag{24}$$

$$h_{\mathrm{L}}^{i-1/2} = h_i - \tfrac{1}{2}\,\varphi(r_{1\mathrm{L}}^{i-1/2})(h_{i+1} - h_i) \tag{25}$$

The principle underlying the use of limiters is that there is a total variation diminishing (TVD) property for non-linear equations see Toro [21], Leveque [22], Hirsch [26], Jeng and Payne [33], Arora and Roe [34], Delis and Skeels [35]. The reader can look at these publications for more details about the concept of TVD. The differences between the slope and flux limiters and also between the slope limiters presented here can be found in these publications.

As can clearly be seen there are five cells involved in the solution. The second order accurate solution presented is only valid for structured and uniform grids. Recently, Causon *et al.* [17] presented an approach, which is also valid for structured non-uniform grids. This can be implemented quite simply. For instance, for the first component of $\mathbf{p}$, h, the data on either side of the right cell interface $(i+1/2, j)$ can be expressed for two dimensional problem as follows:

$$h_{\mathrm{L}}^{i+1/2} = h_{i,j}^{n} + \left| \frac{X[i,j]}{X\mathrm{centre}[i+1,j] - X\mathrm{centre}[i,j]} \right| \varphi(r_{1\mathrm{L}}^{i+1/2})(h_{i,j}^{n} - h_{i-1,j}^{n}) \tag{26}$$

$$h_{\mathrm{R}}^{i+1/2} = h_{i+1,j}^{n} + \left| \frac{X[i+1,j]}{X\mathrm{centre}[i+1,j] - X\mathrm{centre}[i,j]} \right| \varphi(r_{1\mathrm{R}}^{i+1/2})(h_{i+2,j}^{n} - h_{i+1,j}^{n}) \tag{27}$$

where i, j define the cell position in two dimensional space, X is a distance normal vector, normal to the interface under consideration, Xcentre is the x co-ordinate of the central point of the cell.

It can be seen that, for uniform rectangular grids, the terms

$$\left| \frac{X[i,j]}{X\mathrm{centre}[i+1,j] - X\mathrm{centre}[i,j]} \right| \quad \text{and} \quad \left| \frac{X[i+1,j]}{X\mathrm{centre}[i+1,j] - X\mathrm{centre}[i,j]} \right|$$

reduce to $1/2$.

Similarly in the other interfaces i.e. $(i-1/2, j), (i, j+1/2)$, the Riemann interface data are constructed.

It should be mentioned that in the literature a variable extrapolation method is also known as Monotone Upstream-centered Scheme for Conservation Laws (MUSCL). This approach indeed preserves the monotonicity thanks to the use of a limiter, and is upwinded and centred.

4.2. Second order accuracy in time

In this section we will present two approaches for the time integration which provide second order accuracy. One is the Hancock two stage time integration method [32], and another is a predictor–corrector method given by Hirsch [26].

(a) Hancock method: This method requires a half time step solution initially and then these values are updated in the following stage. The two stages are described as the predictor and corrector stage and they are given as follows:

Predictor stage:

$$\mathbf{p}_{i,j}^{n+1/2} = \mathbf{p}_{i,j}^{n} - \frac{\Delta t}{2A_{i,j}} \sum_{k=1}^{m} \mathbf{T}^{-1}(\theta)\mathbf{f}(\mathbf{q})L^{k} \tag{28}$$

It is important to say that the predictor stage does not require a conservative solution because it defines only an intermediate update for $\mathbf{p}_{i,j}^{n}$ over a half time step [21]. In other words, there is no need to solve the Riemann problem at this stage and the numerical fluxes are simply computed using the constructed and transformed conservative variables inside the cell.

This can be achieved by multiplying the cell interface data by the inverse transformation matrix and then computing the fluxes using these data. It is also worth noting that this computation is carried out within the same cell; no neighbouring cells are involved.

Corrector stage:

Unlike the predictor stage, this stage is fully conservative. Thus it requires use of one of the Riemann problem solvers to compute the numerical fluxes. Hence, the transformed data at both Riemann interfaces are used i.e one inside the interface of a cell under consideration the other at the interface of the neighbouring cell.

Before giving the expression for this stage, it is necessary to explain that the data reconstruction procedure needs to be repeated. Fortunately, the same gradient computed in the predictor stage is used. For example, repeated data reconstruction at a point $(i+1/2, j)$ can be given for the first component as follows:

$$h_{\mathrm{L}}^{i+1/2} = h_{i,j}^{n+1/2} + \left| \frac{X[i,j]}{X\mathrm{centre}[i+1,j] - X\mathrm{centre}[i,j]} \right| \varphi(r_{1\mathrm{L}}^{i+1/2})(h_{i,j}^{n} - h_{i-1,j}^{n}) \tag{29}$$

$$h_{\mathrm{R}}^{i+1/2} = h_{i+1,j}^{n+1/2} + \left| \frac{X[i+1,j]}{X\mathrm{centre}[i+1,j] - X\mathrm{centre}[i,j]} \right| \varphi(r_{1\mathrm{R}}^{i+1/2})(h_{i+2,j}^{n} - h_{i+1,j}^{n}) \tag{30}$$

Similarly, all data in the other interfaces can be reconstructed.

Now, the corrector stage can be written as

$$\mathbf{p}_{i,j}^{n+1} = \mathbf{p}_{i,j}^{n} - \frac{\Delta t}{A_{i,j}} \sum_{k=1}^{m} \mathbf{T}^{-1}(\theta)\mathbf{f}(\mathbf{q}^{\mathrm{p}})L^{k} \tag{31}$$

where $\mathbf{q}^{\mathrm{p}}$ refers to predictor data which is transformed and reconstructed. The Riemann problem is set up according to this.

(b) Hirsch method: This method also has two stages; predictor and corrector. This approach differs from the Hancock method in that the solution of the Riemann problem is repeated in both stages, which means that they both have a conservative solution, and data construction is performed only at the corrector stage.

Predictor stage:

The same expression can be used as that given in Equation (28). Then the fluxes are computed by solving the Riemann problems at the cell interfaces at both the cell under consideration and at its neighbouring cells. Moreover, the conservative variables do not need to be constructed. This stage uses only the cell centre data, not the cell interface data.

Corrector stage:

Before the corrector stage, the conservative variables in both Riemann interfaces are constructed in order to provide second order accuracy in space. This can be done by using the

same expressions (Equations (29) and (30)). This is usually described as data construction (see Hancock method). Now, we can use Equation (31) to evaluate the next time step values.

The steps to be taken in both approaches can be summarized as follows:

(a) Hancock method:

  (a) Construct overall data by using MUSCL method i.e. use Equations (26) and (27).
  (b) Compute the numerical fluxes without solving Riemann problems but using the constructed data at the cell interfaces inside the cell under consideration.
  (c) Use Equation (28) to complete the predictor stage.
  (d) Reconstruct overall data, this provides the cell interface data to be used for the solution of Riemann problems at the cell interfaces e.g. use Equations (29) and (30) at the interface $(i+1/2, j)$.
  (e) Use Equation (31) to evaluate variables at the next time step.

(b) Predictor–Corrector by Hirsch:

  (a) Compute the numerical fluxes as done for the first order accurate flux computations i.e. use Equation (8) but take $\Delta t/2$ instead of Δt for a half time step solution.
  (b) Construct overall data, e.g. use Equations (29) and (30) at the interface $(i+1/2, j)$.
  (c) Use Equation (31) to evaluate variables at the next time step.

As before, the source terms have not been taken into account yet. In order to provide a second order accuracy including the source terms, the overall solution reads:

$$\mathbf{p}^{n+1} = \mathrm{S}^{(\Delta t/2)}[\mathrm{H}^{(\Delta t)}[\mathrm{H}^{(\Delta t/2)}[\mathrm{S}^{\Delta t/2}[\mathbf{p}^n]]]] \tag{32}$$

5. MODEL (SHAW5o)

In order to assess and compare the performance of the numerical methods presented, a program called SHAW5o (SHAllow water equations solved by finite volume method) is used. It is a hydrodynamic model developed by the first author at Newcastle University [Erduran, KS. 1999; Internal Report, University of Newcastle]. The code is written in the object-oriented programming language DELPHI 5 which is a rapid application development tool (RAD), provides pre-programmed objects for the graphical user interface. Programs written in Delphi exceed the speed and efficiency of programs written in C, C++, Borland Pascal 7.0 and Visual Basic [36].

The main features of the model are:

(a) Five Riemann solvers; HLL, HLLC, Osher, Roe and FVS.
(b) Suitability for both first and second order accurate computations.
(c) Two alternative time integration methods for second order accuracy in time; Predictor–Corrector and van Leer to Hancock.
(d) Four limiters; Superbee, Minmod, van Leer and van Albada,
(e) Ability to handle complex topography i.e. use of a variety of cell shapes. In particular, any grid type such as unstructured, non-uniform can be used if first order accuracy is chosen. Second order accurate solution is restricted to structured grids but they can be non-uniform. For the slope computations, a surface slope is defined. As long as the

co-ordinates (x, y, z) of the corner of the cells are known the model can compute the slope in the x and y directions.

(f) Ability to handle different boundary conditions. The boundary conditions available in the program include; rating curves, time dependant discharge boundary, given discharge and water depth or both, closed or open boundaries.

The model has been used in three previous studies [10, 13, 37]. The performance of the finite volume schemes implemented for a variety of conditions is assessed and comparison is made between the different schemes.

One limitation of SHAW5o at present is the absence of an automatic grid generation facility. In our experience the most difficult part of setting up a simulation is the preparation of data. Use of an automatic grid generator, which also produces grids dynamically, would greatly speed up the process of running simulations.

6. TEST PROBLEMS

Several test cases from the literature have been selected to assess the schemes including a one-dimensional (1D) dam break, a two-dimensional (2D) partial dam break, a 2D circular dam break, a shock reflection and drying and wetting processes in a channel. Since the examples are from the literature, not all the details of how they are set up are given. All of the tests are performed using SHAW5o.

6.1. 1D complete failure of a frictionless dam break

This problem was first reported by Garcia and Kahawita [38] and has been used by several authors such as Zhao *et al.* [4, 6]. Similar examples can be found elsewhere in the literature ([2, 7, 39]). This example is generally used to illustrate the shock capturing capabilities of the schemes but has also been used to compare three Riemann solvers with first order accuracy [4] and to compare results obtained with first and higher order accuracy [16].

The problem is set up as follows; the computational domain is divided into 300 cells. The size of each cell is 1×1 m. A dam is initially situated in the middle of the domain. The initial flow condition is given as 5 m water depth upstream (u/s) of the dam and 0.3m water depth downstream (d/s). The dam is removed instantaneously and the model is run for 10 seconds.

6.2. 2D partial dam break

This is one of the most popular examples in the literature. Authors who have presented this problem include Glaister [2], Zhao *et al.* [4, 6], Mingham and Causon [5], Alcrudo and Garcia-Navarro [19], Fennema and Chaudhry [40], Anastasiou and Chan [41], and Louaked and Hanich [42]. This example is of particular interest because it is 2D and the solution is characterized by one wave (a shock wave) travelling d/s with a sudden increase in water depth and a second wave (a rarefaction wave) travelling u/s with a decrease in water depth, often described as a rarefaction shock.

The computational domain is divided into a 40×40 rectangular grid. Each cell has a length of 5 m in both the x and y directions. A dam is again located in the middle of the domain and it is assumed that there is a 75 m breach. Water depths of 10 and 5 m are taken as u/s and d/s initial flow conditions, respectively. The run is completed 7.2 s after failure of the dam.

6.3. 2D circular dam break

A column of water in the middle of the computational domain is suddenly released and a shock wave propagates in radial directions. It is again a frictionless computation. References include [5, 19, 41]. As Alcrudo and Garcia-Navarro [19] note, this problem becomes 1D in the radial direction. They set up this problem in order to demonstrate the ability of the method to conserve symmetries and to illustrate the importance of grid selection.

Both a rectangular grid and a circular grid are tested. While the number of cells we use is 100×100, each of which has a size of 0.5×0.5 m, the number of circular bands and their size are taken exactly as reported in the literature. They are: 25 cells of 1 m length along the radial direction and 50 cells in the tangential direction.

Initial conditions are the same in both cases and 10 m water depth up to a radius of 11 and 1 m water depth from 11 to 25 m. The run is completed after 0.69 s.

6.4. Oblique shock (oblique jumps, oblique bore reflection)

An oblique shock occurs when supercritical flow hits a wall at an angle to the direction of flow and thus the flow is deflected. This example can be found in the same references as for Test 6.3 as well as in References [4, 39], and there is a similar problem tested by Chippada *et al.* [9].

The computational domain is divided into a grid of 60×80 cells, which are composed of rectangular and non- rectangular grids. The angle of reflection is taken as 8.95°. The initial condition is a water depth of 1 m, and a velocity of 8.57 m/s in the x direction. The upstream boundary has a 1 m water depth and a velocity of 8.57 m/s, resulting in supercritical flow. The downstream boundary is assumed to be open.

6.5. Irrigation channel

Zoppou and Roberts [11] used this example to illustrate the wetting and drying capabilities of their model. Once the model begins to run, the slope of the channel and the differences in water depth cause the water to start to flow, which in turn drives the wetting and drying processes.

An irrigation channel, 400 m long and 50 m wide, has a slope of 0.02 and Manning's roughness coefficient of 0.02. Initially the part of the channel 0–150 m and 250–400 m is dry and between 150 and 250 m there is a 10 m water depth. 5×40 rectangular cells are used.

7. RESULTS AND DISCUSSION

7.1. Accuracy and applicability

Our aim is to highlight the differences between the solution methods presented rather than to demonstrate the features of the methods such as their shock-capturing capabilities. Such features are already proven and can be found in the literature.

7.1.1. Frictionless 1D dam break. In the literature, Test 6.1 is used to demonstrate the shock capturing capabilities of the schemes. Once the dam fails a rarefaction wave travels u/s causing a decrease in water depth, and a shock wave travels d/s with an abrupt increase in water depth.

Table II. Results of 1D dam break problem.

First order accurate in time and space		
Schemes	Velocity at dam side (m/s)	Water depth at dam side (m)
Exact†	4.670	2.220
Osher	4.661	2.222
Fvs	4.463	2.316
HLL	4.459	2.285
HLLC	4.459	2.285
Roe	4.717	2.212
Roe entropy fixed	4.639	2.233

† Taken from Zhao *et al.* [10].

It has been observed that the schemes capture the shock. However, it has also been seen that the Roe and FVS schemes produce an unphysical jump at the dam location. This problem is discussed in Section 3.4 and is known as an entropy violation. For both of these schemes there are techniques that are used to fix this problem (see for example Reference [26]). Table II shows that for the Roe scheme this jump results in considerable errors in both water depth and velocity. This problem is seen in dam break problems in the literature (see for example Reference [39]). Thus, application of the Roe and FVS schemes without fixing the entropy violation can produce errors where the sonic (critical) point occurs.

The exact results given in Reference [4] were used to compare with the results of the numerical schemes. The results show that the Osher scheme is the most accurate. No difference is observed between HLL and HLLC for this test; their results are the least accurate. The value given in Table II for the Roe scheme is an average value over the unphysical discontinuity and therefore appears to be more accurate than it actually is. The Roe scheme is greatly improved by fixing the entropy violation. Therefore for this problem placing the schemes in order of increasing accuracy gives: HLL/HLLC, FVS, Roe, Roe with entropy fixed (REF), Osher, and exact.

We have extended the 1D dam break test case with a grid range of 0.25, 0.5, 1, 2, 4, 5, 8, 10 m. The results for second order accuracy are given in Figure 9 for different grid sizes. The time step is 0.05s for all cases.

The grid size is an important factor in determining the accuracy of a scheme. Figure 9 shows how the accuracy improves as the grid size decreases for both velocity and water depth at the location where the dam was initially. The results are very close to the exact results when the grid size is 0.25m. It is noted that, although Figure 9 shows the accuracy changing linearly with grid size, this happens due to results obtained by averaging of data points in front and behind the dam. The shock height and speed of the shock are found to be 1.655 m and 7.27 m/s for all cases. In order to assess the effect of slope limiters on accuracy, Test 6.1 was repeated using the HLLC scheme with four limiters and the results show that, except for the superbee limiter which gives a sharper profile, most limiters produce similar effects. These results agree with those reported by Hu *et al.* [7]. It has been observed that there is no significant difference between the accuracy of predictor-corrector time integration and van-Leer to Hancock approaches (Figure 10).

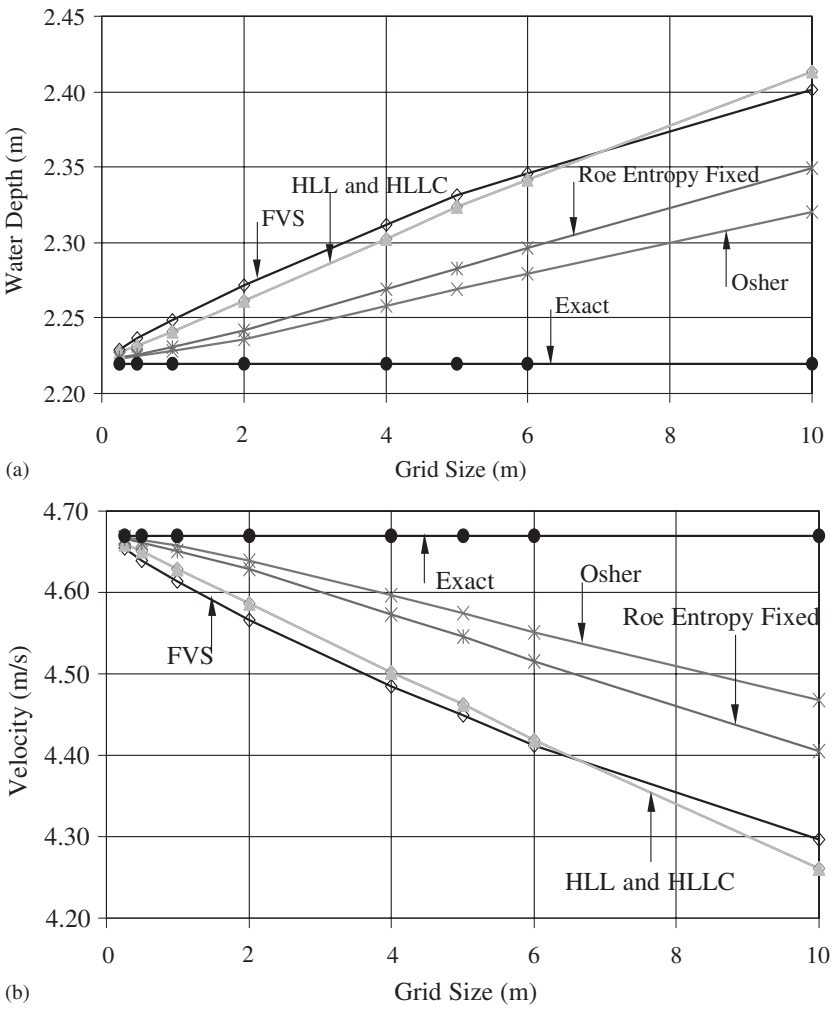

Figure 9. Second order results for 1D dam break problem at dam side for different grid sizes obtained using van-Leer to Hancock time integration approach and Minmod limiter: (a) water depth; (b) velocity.

7.1.2. 2D partial dam break. This problem is also used to illustrate the shock capturing capabilities of the schemes. Again the flow is characterised by a rarefaction wave travelling u/s and a shock wave travelling d/s. Unlike the 1D dam break problem, it shows the formation of a 2D bore. Plate 1 shows the results of the HLL, HLLC and Osher schemes for comparison. Discrepancies between the results obtained from the HLL and HLLC schemes arise; the maximum differences are observed u/s of the dam. The results of the Osher scheme show good agreement with the HLLC scheme indicating that the HLL scheme is the one that has a problem, perhaps because of the omission of the second characteristic.

Plate 2 shows the results of the same problem but with the grid size reduced to half the previous one. Again the maximum differences are seen in the region just u/s of the

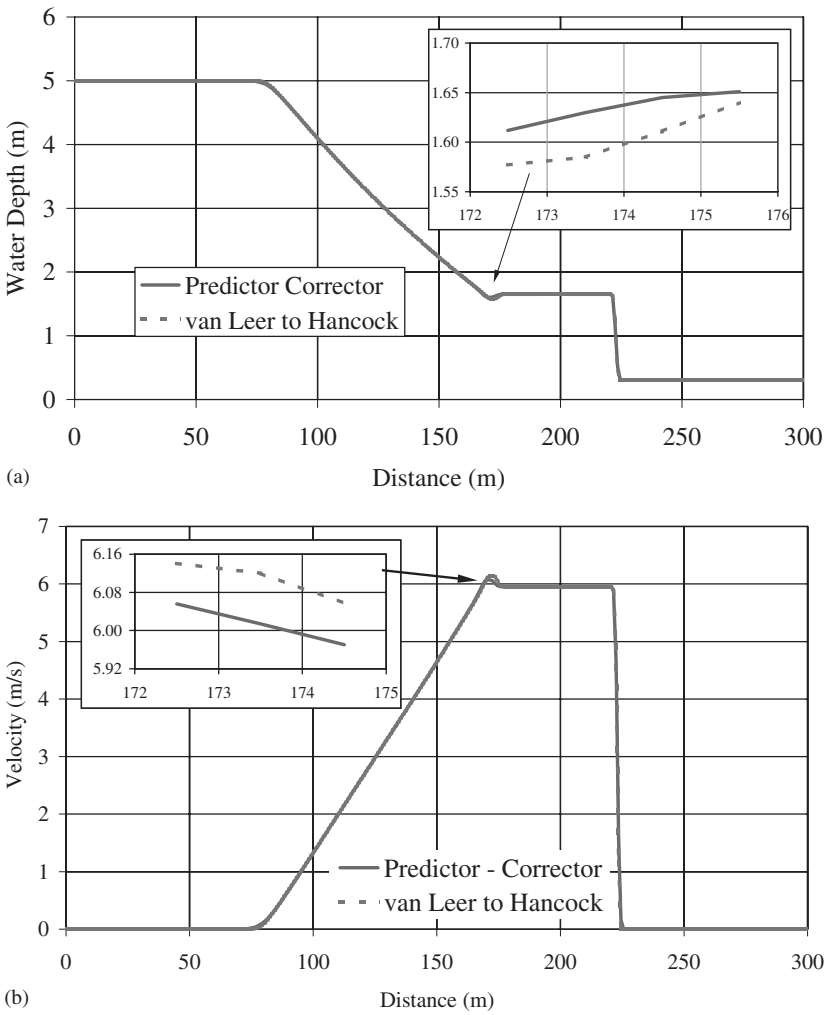

Figure 10. Results for 1D dam break problem obtained by Predictor–Corrector and van-Leer to Hancock time integration methods using the Osher scheme and superbee limiter: (a) water depth; (b) velocity.

dam. The HLL and HLLC schemes exhibit differences between the results only in the 2D case. Since there are no exact results for this test, accuracy cannot be discussed. However, Plate 2(b) shows that while the velocities obtained from all the schemes in y direction are almost identical, in the x direction the velocities obtained from the HLL and FVS schemes are quite different from those of the others.

7.1.3. Circular dam break. A 2D circular dam break problem is also found in the literature. Alcrudo and Garcia Navarro [19] use it to check the ability of the method to preserve cylindrical symmetry. Alcrudo and Garcia Navarro [19] and Mingham and Causon [5] reported

that this test also demonstrates the effectiveness of the mesh selected to represent the problem at hand. In order to show the grid effectiveness, this test is also applied to a domain which is covered by a rectangular grid. Unlike other authors, we have selected 100×100 rectangular cells in order to represent the circular initial flow condition more closely. In the circular grid case, the number of cells in the radial and tangential directions is taken to be the same as that used in the literature. When the column of water is released, the shock wave results in a dramatic increase in the water depth in the lower region and propagation in the radial direction, producing a circular propagation path. Plate 3 shows the sectional view (a cut along one direction) of the 2D circular dam break with a rectangular grid.

Plate 3(a) shows that, while the shock is captured well by the Osher, Roe entropy fixed and HLLC schemes, the HLL and FVS schemes again smear the results at the beginning of the shock. Although second order accurate solutions capture the discontinuous nature of the problem better than first order schemes, their results do not show such large differences, see Plate 3(b).

In the rectangular grid case (Plate 4), it is clear that the cylindrical propagation of the shock front is not perfect. Even in the case of a second order accurate solution with a rectangular grid, the cylindrical propagation of the shock front is poor. This could be because of the use of rectangular grids, with which it is not possible to set up a perfect cylindrical flow condition initially. It may also be concluded that the use of rectangular grid leads to noticeable oscillations in water depth between the shock front and the depression wave. Hence, small kidney shaped contours occur.

Results show that each limiter produces similar but slightly different results (Plate 5). The superbee limiter tends to produce a steeper profile and a greater depth. Thus this limiter is particularly well suited to problems involving a sharp cornered shock front such as Test 6.1. Results obtained using triangular grids show a similar lack of cylindrical symmetry to the rectangular case [41]. However, use of circular grids for both first and second order accurate solutions results in the required cylindrical symmetry. Sectional water profiles for first and second order results are shown in Plate 6.

These tests show that there are small discrepancies between the results obtained using different schemes and that increasing the order of accuracy makes very little difference to the results. In the circular grid case, much better cylindrical and symmetrical shock propagation are achieved (see Plate 7). Results obtained using the predictor–corrector and van Leer to Hancock time integration methods produce very similar results (Plate 8).

Finally, this test demonstrates the importance of mesh selection on the simulation. As in other applications, it has been seen that second order accuracy is generally better than first order accuracy. It should be noted that the software used to draw these figures also affects the results.

7.1.4. Oblique shock. This example demonstrates the use of a second order accurate solution with a non-uniform grid. The computational domain is composed of quadrilateral and rectangular cells.

With regard to accuracy, it is noted that the results agree well with those of the published [4, 19, 39, 41]. As an example we provide one of the second order results (see Figure 11).

The resulting oblique shock heights for first order Osher, FVS, REF, HLLC and HLL are 1.498, 1.497, 1.498, 1.498 and 1.498 m, respectively. However, by using a second order

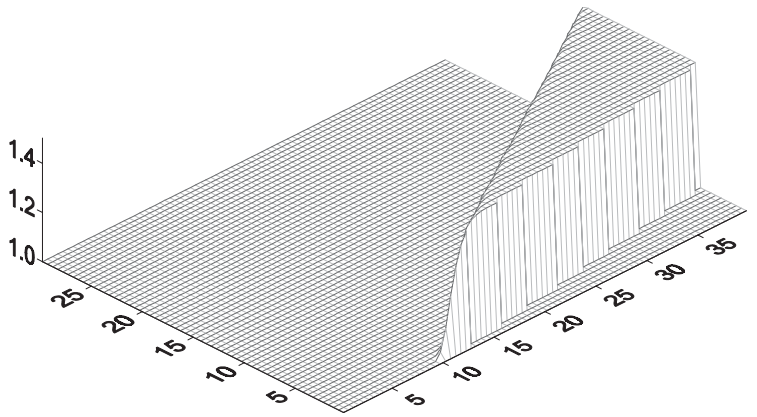

Figure 11. Water depth for Second order oblique shock test results obtained using the Osher scheme, van-Leer limiter and van Leer to Hancock time integration approach.

accurate solution, the exact result (1.5 m) reported by Zhao *et al.* [4] can be obtained. The most important conclusion drawn from Test 6.4 is that the method presented is applicable to non-uniform grids.

7.1.5. Wetting and drying processes. In flood events, flood plains are usually dry initially. Some time after the flood, they start drying again. In order to simulate such events, the model should be able to deal with wetting and drying processes. Test 6.5, reported by Zoppou and Roberts [11], was used to assess how well the numerical schemes can model these. Our experience shows that there is no problem with wetting the initially dry areas for all of the schemes tested provided that flow conditions in dry cells are as follows: velocities are zero and water depths are very small (0.00001 m). This small initial depth must be given to avoid zero division occurring during the computation of velocities. Although this numerical trick stops the model from failing, it changes the character of the problem, which may cause inaccuracies [27].

Results obtained using SHAW5o show that, apart from the REF scheme, all of the schemes tested can handle a drying process following a wetting process. However, the REF scheme exhibits numerical failure just after drying process starts. During the drying process, the water depth decreases below the initially given number for dry cells. This makes the REF scheme fail. It may be said that the REF scheme is too sensitive to handle such a small number. This is suggested by the fact that failure is sudden; even just before failure there is no noise. On the other hand, the results obtained using other schemes agree well with published results (Figure 12). For this test, the wave speed estimates for dry bed conditions given by Fraccarollo and Toro [29] are used for the HLL and HLLC schemes.

Drying and wetting processes are two important processes occurring during flooding. Any of the schemes except the REF scheme can be used for such a simulation. The HLLC scheme with the wave speed for dry bed conditions is preferred. Since we have not got exact results accuracy is not discussed further.

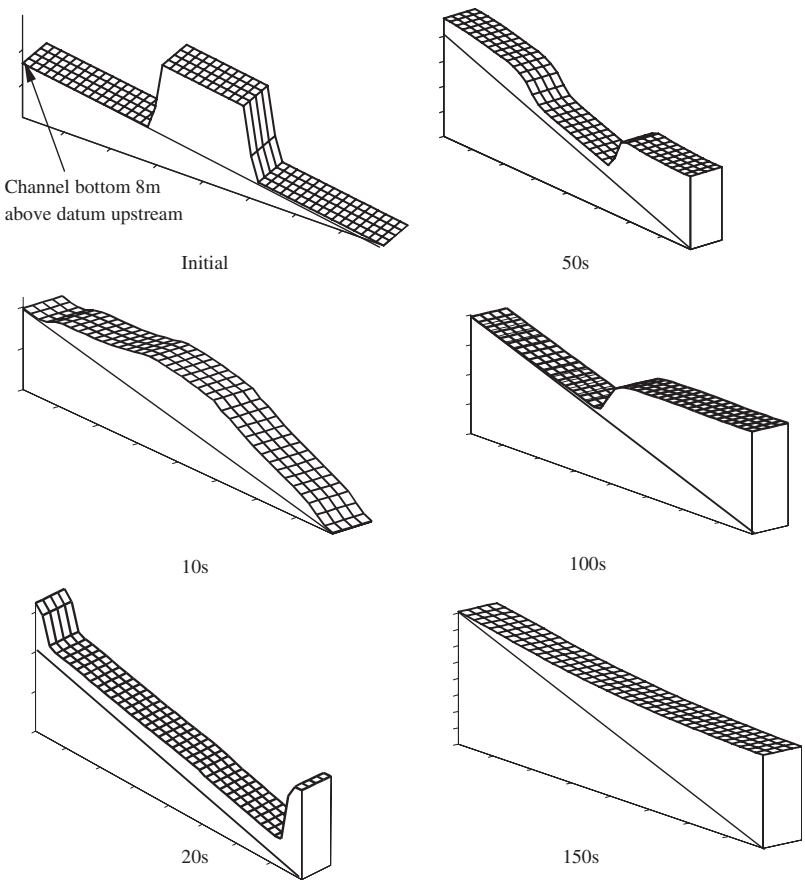

Figure 12. Elevation for wetting and drying test in a 400 m long, 50 m wide channel up to 150 s obtained using the Osher scheme, van-Leer to Hancock time integration approach and van-Leer limiter.

7.2. Simulation time

When the solution method is explicit the size of the time step must be limited in order to ensure stability, which can result in long simulation times. Although the examples presented are not real engineering problems, which are normally set up on considerably larger grids, allowing a larger time step to be used, the time step limitation can still be severe. Therefore, simulation time is a crucial consideration for modelling. In our analysis, we use SHAW5o run on a Pentium III, 500MHz PC, Windows 98.

As shown in Plate 9, the FVS scheme (both first and second order accurate) provides the quickest results, and the Roe scheme with Entropy Fixed (REF) takes the longest. Taking a time step of 0.08 and 0.1 s for the simulation of 10 s results in such a fast simulation that no differences are seen by the user. However, for the schemes tested, the following list runs from slowest to fastest simulation; REF, Osher, HLL/HLLC, FVS.

Table III. Requirement computer time ratio for 2D circular dam break simulation.

Time Ratio for 2D circular dam	FVS	OSHER	REF	HLLC	HLL
First order, circular grid	1.00	1.27	1.60	1.20	1.20
Second order, circular grid	1.00	1.04	1.13	1.01	1.04
First order, rectangular grid	1.00	1.28	1.53	1.13	1.15
Second order, rectangular grid	1.00	1.03	1.07	1.05	1.05

These results agree with those reported by Zhao *et al.* [4]. The simulation time ratio of first order to second order solution is given in Plate 10. Use of second order solution (van Leer to Hancock) results in a considerable increase in the simulation time. While the maximum increase is seen in the use of FVS, the minimum increase is estimated to correspond to the REF. The increase varies between 2 and 6 times.

Plate 11 shows a comparison of simulation times for the predictor-corrector and van Leer to Hancock time integration methods with the Osher scheme for the 1D dam break problem. Except for the case of 0.1s time step, the van Leer to Hancock solution results in a longer simulation time than the predictor–corrector approach. However, this exception may be caused by a measurement error, which is based on rounding to the nearest second.

The fact that the van Leer to Hancock method takes longer to run despite the repeated solution of the Riemann problem required by the predictor–corrector approach, leads to the conclusion that reconstruction of conservative variables (repetition of data construction) takes longer than solution of the Riemann problem twice. However, the time required to solve the Riemann problem varies with the scheme. Simulation times for these two time integration methods were also compared for the circular dam break problem with the FVS scheme. It is again seen that the predictor–corrector method, which uses the conservative solution at both stages, requires a shorter simulation time than the van Leer to Hancock approach does. The ratio between them is found to be approximately 0.89.

Table III shows the computer time required to run first and second order accurate solutions of circular dam break problems using rectangular and circular grids. Second order results are obtained using van Leer to Hancock time integration approach and van Leer limiter. The ratio of the simulation time required by four schemes to that required by the FVS method (the fastest solver) is shown in this table.

Again the Roe scheme with Entropy Fixed results in the longest simulation, and the FVS gives the shortest simulation. The schemes in order of speed, from slowest to fastest are: REF, Osher, HLLC/HLL, FVS. Looking at the ratio of simulation time of first order to second order (Figure 13), it is again clearly seen that second order accuracy is computationally expensive.

The maximum increase in simulation time is seen for the FVS method, and the minimum is seen using the REF scheme. This is quite acceptable, the FVS gives the quickest solution in all cases. In other words, it is found to be the fastest Riemann solver. However, the problem is not only solution of the Riemann problem but also the reconstruction of data and this reconstruction step affects the simulation time seriously. Figure 13 shows that the use of the slowest Riemann solver is the least affected. Similar results are observed for other cases (Table IV). Again the ratio is defined in the same way as that in Table III.

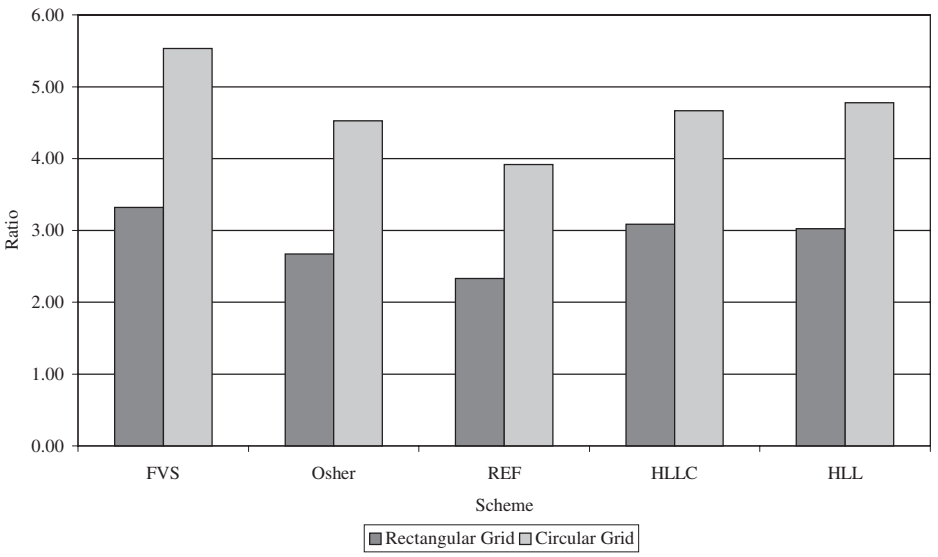

Figure 13. 2D circular dam break problem. The ratio of simulation times taken by first and second order schemes.

Table IV. Ratio of simulation time for Oblique shock and 2D partial dam break test case.

Test name	FVS	Osher	REF	HLLC	HLL
Oblique shock	1.00	1.34	1.73	1.25	1.25
2D Partial dam break	1.00	1.14	1.29	1.00	1.00

7.3. Experimental stability analysis

Explicit schemes are those in which values are calculated at the new time step in terms of values known at the previous time step. Although the solution and implementation of explicit schemes is much easier than that of implicit schemes, which require a system of equations to be solved, they require careful time step selection to fulfil stability requirements. This restriction on the selection of time step can result in expensive computation. Stability analysis for the algorithms presented requires non-linear analysis. The non-linear analysis is normally simplified to a linearised solution. Courant, Friedrichs and Lewy [43] defined a stability criterion for fully explicit schemes given by $\mathrm{CFL} < 1$ where CFL is known as the Courant number.

However, different authors use a different definition of the Courant number, thus giving different time step limitations. For instance:

Zhao *et al.* [4] defined the Courant number as

$$\mathrm{CFL1} = \frac{\Delta t}{\Delta s} \left| \frac{\Delta f(\mathbf{q})}{\Delta \mathbf{q}} \right|_{\max} \leqslant 1$$

where Δs is a distance between the centres of two adjacent cells.

Table V. Maximum allowable time steps.

Distance from cell centre to the centre of adjacent cell or Δx (m)	Expression by Toro [21]	Expression by Mingham and Causon [5]	Expression by Zhao *et al.* [4]
0.25	0.029	0.029	0.039
0.50	0.057	0.057	0.078
1.00	0.115	0.115	0.157
2.00	0.229	0.229	0.314
4.00	0.458	0.458	0.627
6.00	0.573	0.573	0.784
8.00	0.687	0.687	0.941
10.00	1.146	1.146	1.568

Toro's definition [21] is

$$\mathrm{CFL2} = \frac{\Delta t}{\Delta x}(\lambda_{\max}) \leqslant 1$$

This expression is more suitable for finite difference type 1D solutions. However, if uniform grids are used, this condition applies to the finite volume method if Δx is taken to be the distance between the cell centre and the centre of the adjacent cell. Mingham and Causon [5] give an expression which applies to 2D finite volumes:

$$\Delta t = v \min(\Delta t_x, \Delta t_y), \quad \Delta t_x = \min\left\{ \frac{A_{ij}}{|u_{ij} S_{i+1/2,j}| + \sqrt{g h_{ij}}|S_{i+1/2,j}|} \right\},$$

$$\Delta t_y = \min\left\{ \frac{A_{ij}}{|u_{ij} S_{i,j+1/2}| + \sqrt{g h_{ij}}|S_{i+j+1/2}|} \right\}$$

where $S_{i+1/2,j}$ and $S_{i,j+1/2}$ are presumably the distance between cell centre and the centre of the adjacent cell in the x and y directions respectively and v is taken to be 0.9. Note that for one dimension with a uniform grid, it reduces to the expression given by Toro above. Note that these expressions are only valid in the absence of source terms. The stability also depends on the stability of the split part [21].

For the 1D frictionless dam break problem, the maximum time step $\Delta t_{\max}$ as calculated by different stability criteria is given in Table V. Note that the maximum time steps in this table is computed by setting $\mathrm{CFL1} = 1$ and $\mathrm{CFL2} = 1$.

As anticipated the stability criteria given by both Toro [21] and Mingham and Causon [5] produce the same maximum time step. On the other hand, using the criteria given by Zhao *et al.* [4] results in a higher value.

Our experimental analysis is carried out for both first and second order accurate solutions for each scheme for each Δx. The maximum allowable time step is calculated not according to complete numerical failure but to the first occurrence of significant noise in the solution. For instance, the second order Osher scheme is to be unstable for $\Delta t \geqslant 0.106$ s for a grid size of 1 m. However, complete numerical failure is observed only when Δt reaches 0.108 s.

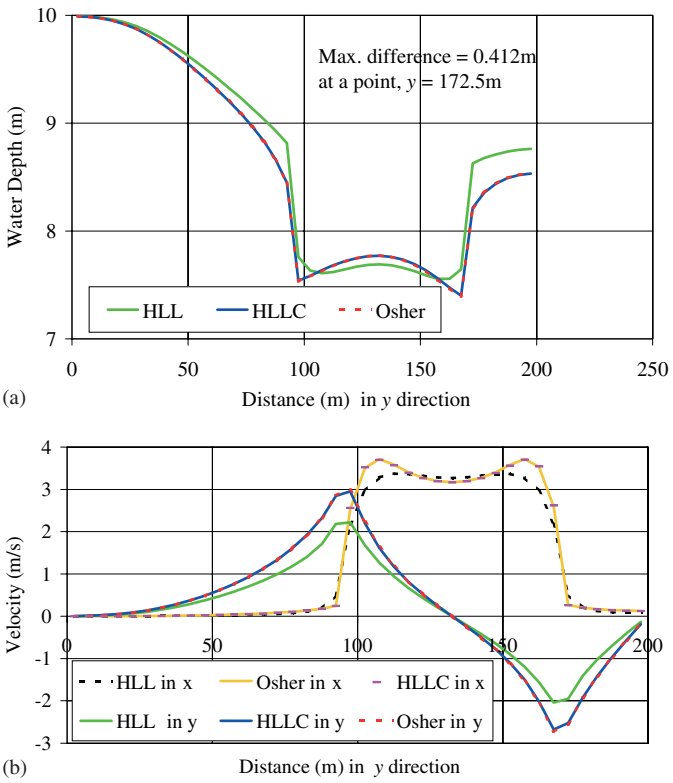

Plate 1. First order results for 2D partial dam break problem obtained along a section at $x = 92.5$ m: (a) water depth; (b) velocity in the x and y directions.

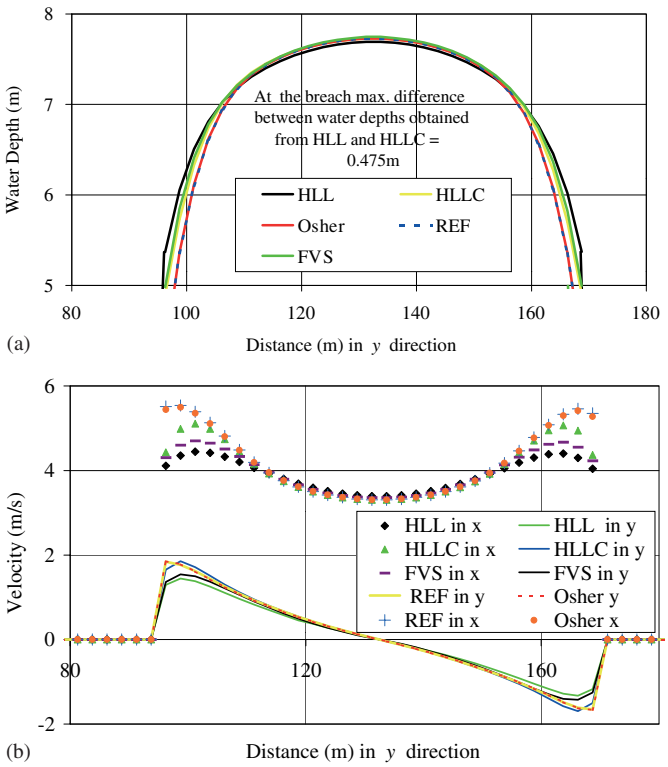

Plate 2. First order results for 2D partial dam break problem obtained along a section at $x = 92.5$ m: (a) water depth; (b) velocity in the x and y directions.

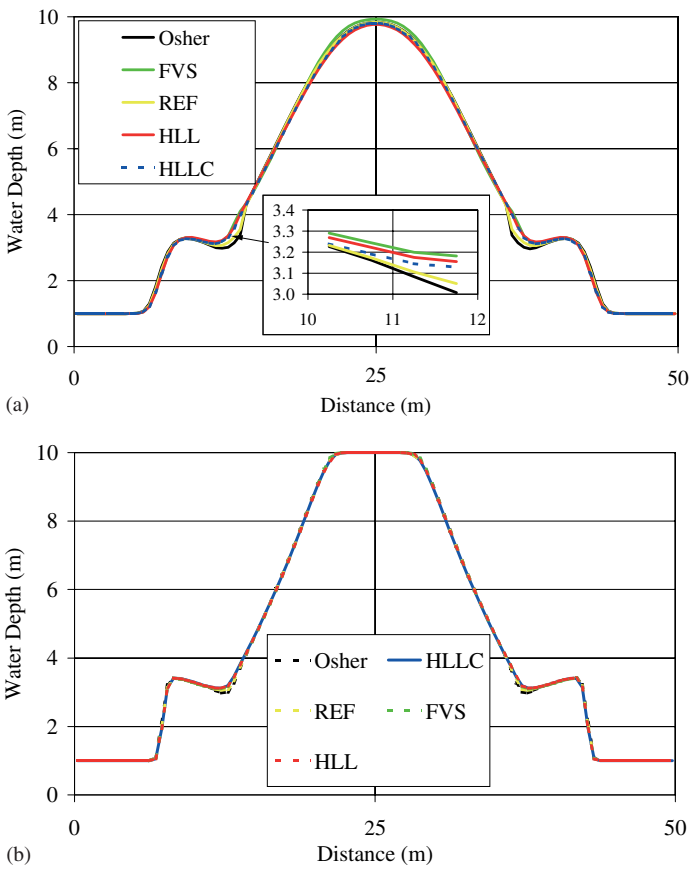

Plate 3. Water depth for 2D circular dam break problem using a rectangular grid: (a) first order; (b) second order obtained using predictor–corrector approach and van-Leer limiter.

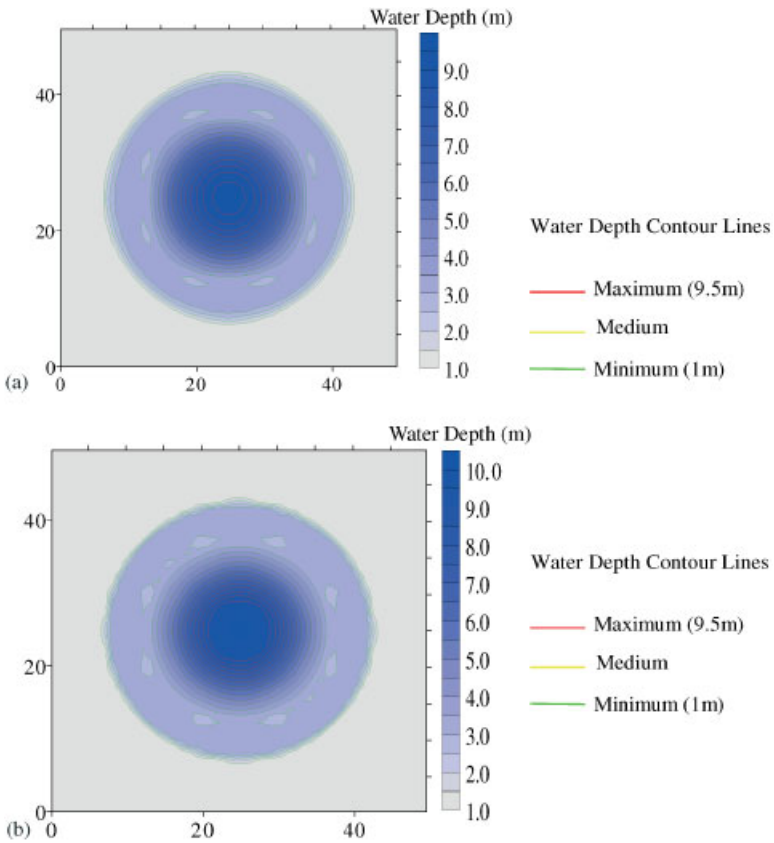

Plate 4. Water depth contours for 2D circular dam break problem using REF and a rectangular grid: (a) first order; (b) second order obtained using predictor–corrector approach and van-Leer limiter.

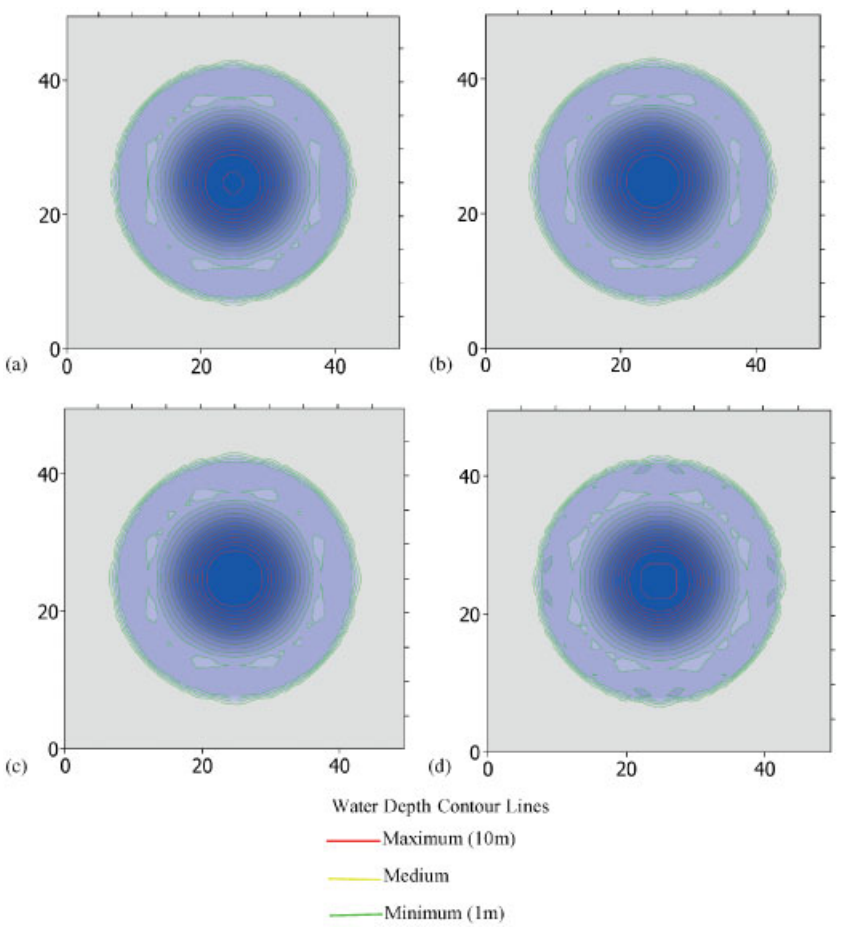

Plate 5. Water depth contours for 2D circular dam break problem obtained using the Osher scheme, predictor–corrector time integration approach and a rectangular grid: (a) van-Leer; (b) Minmod; (c) van Albada; (d) Superbee.

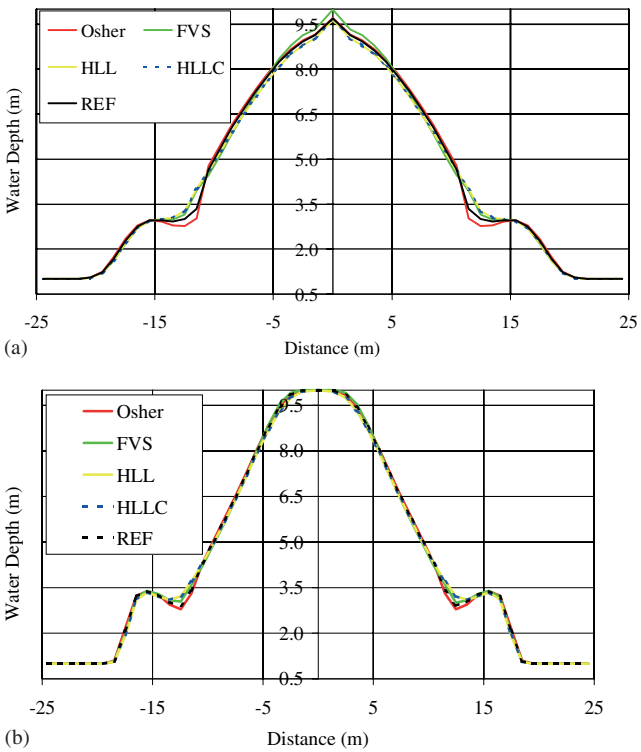

Plate 6. Water depth for 2D circular dam break problem using a circular grid: (a) first order; (b) second order obtained using van-Leer to Hancock time integration approach and van-Leer limiter.

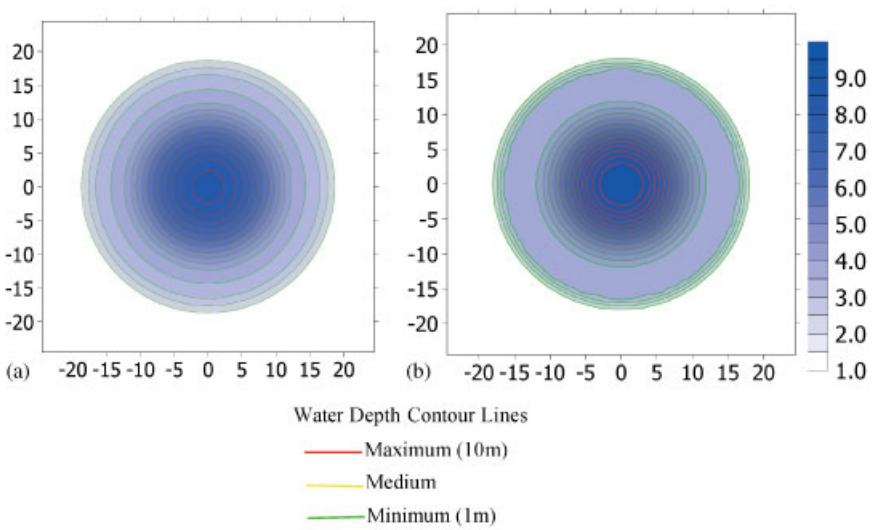

Plate 7. Water depth for 2D circular dam break problem obtained using the HLLC scheme and a circular grid: (a) first order; (b) second order obtained using van-Leer to Hancock approach and van-Leer limiter.

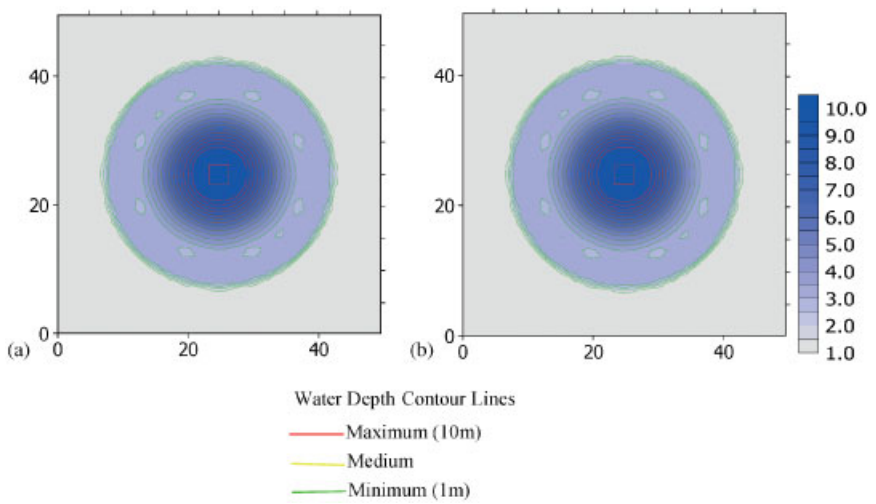

Plate 8. Water depth for 2D circular dam break problem obtained using a rectangular grid, the FVS scheme and van-Leer limiter: (a) predictor–corrector; (b) van-Leer to Hancock.

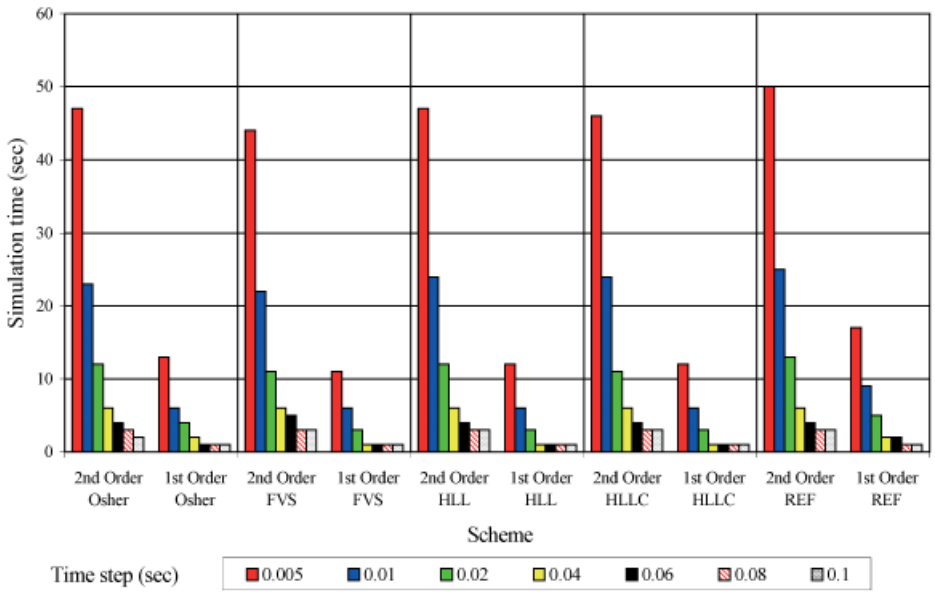

Plate 9. Comparison of simulation times for 1D dam break problem, second order results obtained using van-Leer to Hancock time integration approach and Minmod slope limiter.

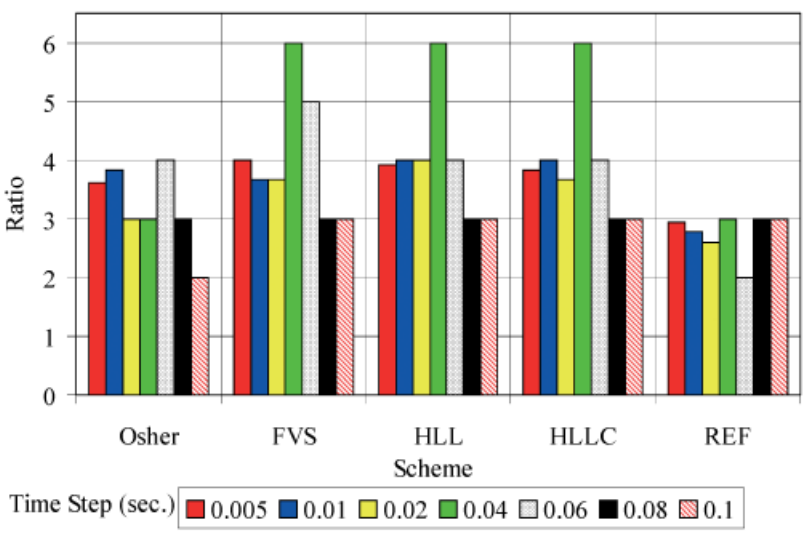

Plate 10. Ratio of simulation times for 1D dam break problem taken by first and second order solutions.

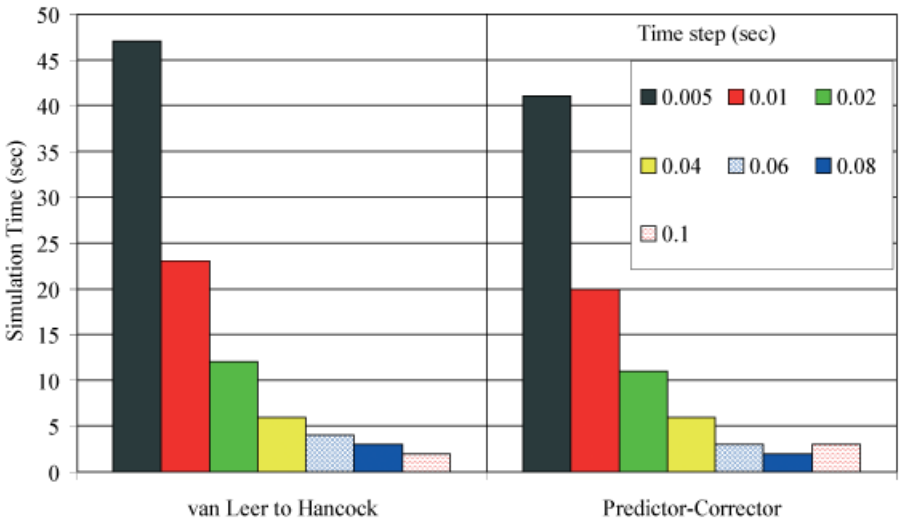

Plate 11. Requirement of simulation time for time integration approaches obtained using the Osher scheme and minmod limiter from 1D dam break test.

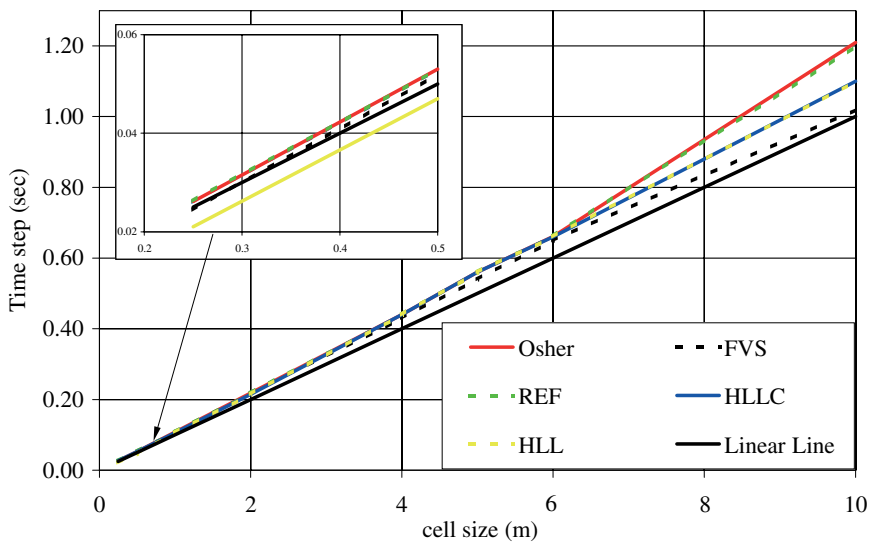

Plate 12. Maximum allowable time step for 1D dam break problem.

Table VI. Maximum time step allowable for 2D partial dam break problem.

Test name:	FVS	Osher	REF	HLLC	HLL
5 m cells	0.31	0.29	0.29	0.28	0.27
2.5 m cells	0.15	0.13	0.13	0.13	0.12

Note that the maximum allowable time step is estimated from the numerical experiment whereas the maximum time step is computed as described earlier.

It has been found that the stability criterion using CFL2 and that given by Mingham and Causon [5] agree and is more reliable. The criterion, CFL1 overestimates the maximum allowable time step.

Our results show that the relationship between Δx and Δt is non-linear (Plate 12). There are slight differences between the maximum time step allowable for each scheme. In particular Osher and REF schemes remain stable for a fractionally larger time step. Second order schemes require a smaller time step than first order schemes. This may be connected with the slightly higher velocities and water depths, generated by second order schemes. However, these differences are not considered to be large enough to be significant.

Comparing with the literature, Zhao *et al.* [4] note that the Osher and FVS scheme become unstable when the time step $\Delta t > 0.1$ s and an unstable result is observed when $\Delta t > 0.075$ s for the REF scheme. Our results also show that the Osher and FVS schemes become unstable when $\Delta t > 0.107$ s for first order solution with a grid size of 1 m. However, we also observe that the REF scheme becomes unstable when $\Delta t > 0.107$ s.

Stability analysis is performed for the 2D partial dam break problem, where two simulations were run using grid sizes of 5 m$\times$5 m and 2.5 m$\times$2.5 m. The overall maximum time steps providing stable results are given in Table VI. When CFL2 $= 1$, $\Delta t_{\max} = 0.396$ s for a 5 m grid and $\Delta t_{\max} = 0.198$ s for a 2.5 m grid.

Table VI shows that the schemes are less stable when Δx is reduced. But reduction of Δx does have not the same effect on stability as the reduction of Δt. The relationship between Δx and Δt is again non-linear. Surprisingly the FVS scheme is found to be the most stable. However, it may be wrong to conclude that FVS is more stable since it produces lower velocities (Plate 2), which plays a significant role in stability. Other results are similar to that of previous analyses. Again the experimental stable region is equated with the beginning of significant noise.

The stability test is extended to the oblique shock problem. Using CFL2 we find the maximum time step, $\Delta t_{\max} = 0.0427$ s. However, if you use CFL1, $\Delta t_{\max} = 0.0846$ s. Our experimental results showed that all schemes fail for $\Delta t > 0.039$ s. This result agrees well with that reported by Mingham and Causon [5]. Our criterion for experimental stability is again the beginning of noise, see Figure 14.

7.4. Splitting technique and source terms

As mentioned in Sections 2 and 3, our solutions are based on the solution to the Riemann problem, which is found by splitting the shallow water equations and solving the homogenous part only. This technique is known as the fractional step or splitting method in the literature. Many authors such as Hu *et al.* [7], Zoppou and Roberts [11], Erduran and Kutija [10], Erduran *et al.* [13] have used it. The advantage of the technique is that it is simple and easy

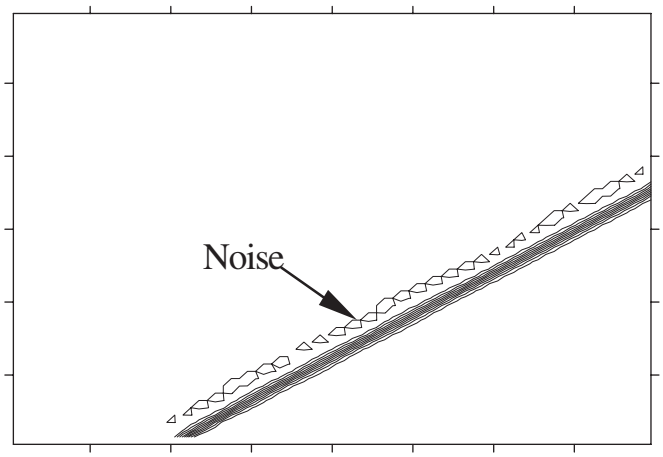

Figure 14. 2D oblique shock test, experimental stability criterion, beginning of noise in front of shock.

to implement. In addition, the solution method for the split part is independent of the solution to the homogenous part. Hence, any solution method suitable for ODEs can be used.

However it does not always work well, in particular in the case of static or steady flows in a channel with variable bottom topography. In such cases, errors are generated and arise when there is significant variation in the bottom of the channel, particularly at the boundaries. More information about the problems associated with the splitting technique can be found in Toro [21] and Leveque [44]. The concept of stiff source terms given by Toro is important for the stability criteria. The time step cannot be chosen by considering the homogeneous part only since this does not guarantee the stability of the source part.

In fact, the reason for the occurrence of errors caused by using the splitting technique is well known. It is due to the imbalance between the fluxes and source terms. For instance, in the 1D static flow case, the following balance must be provided.

$$\frac{\partial}{\partial x}(\tfrac{1}{2}\, gh^2) = gh(So_x - Sf_x)$$

or in the steady state case

$$\frac{\partial}{\partial x}(hu^2 + \tfrac{1}{2}\, gh^2) = gh(So_x - Sf_x)$$

The left hand side of this expression shows the flux in the x direction and the right hand side shows the source terms.

In order to balance fluxes with source terms, there are several alternative techniques introduced in the literature. One approach is to upwind the source terms [8, 45]. Roe [46] has also shown that the source terms can be upwinded in the same way as the fluxes. Another technique is the quasi-steady Wave-propagation algorithm given by Leveque [44].

In order to show the occurrence of errors we have set up an example for which we know the exact solution. Initially the flow is at rest in a channel with 1 m width and 1 m water depth. On the channel bottom there is a hump, whose height is 0.5 m. The water should stay

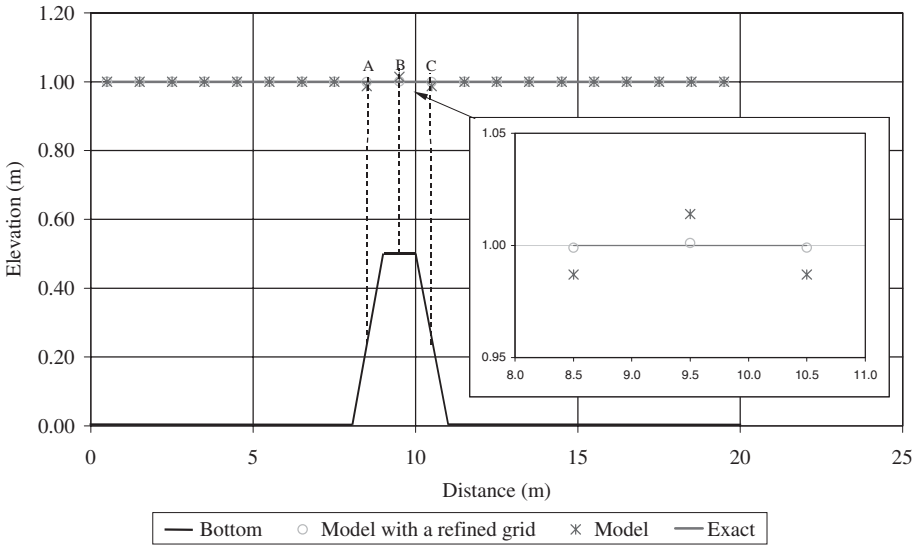

Figure 15. Occurrence of errors in the use of the splitting technique.

Table VII. Error estimation in splitting technique.

Point at	Absolute errors	Absolute errors with refined grid	Absolute errors in in total mass	Absolute errors total mass with refined grid
A,C	0.017333	0.001333		
B	0.028	0.002	0.000146	0.00002512

at rest and there should be no flow. When we run the model, it is seen that errors occur at the surface above the hump. The absolute errors computed at points A, B and C in Figure 15 are given in Table VII. This result is obtained using $1\,\mathrm{m}\times 1\,\mathrm{m}$ cells everywhere. Then, we refine the grid over the hump. The size of cells used is $0.2\,\mathrm{m}\times 1\,\mathrm{m}$.

Our experiences show that the splitting technique works well when the bottom slope of the channel is constant. Moreover it has two additional advantages over other methods; the solution is independent of the solution to the homogeneous part and it is easy to implement.

7.5. Friction computation

Friction computation is also one of the drawbacks of the algorithms presented. This is because friction computation is based on a wide channel assumption (that the channel width is much greater than the water depth). In a wide channel, the hydraulic radius has a value close to that of the water depth. In fact, the wider the channel, the closer the water depth is to the hydraulic radius. Therefore, it can be assumed that $h \approx R$ for this kind of channel, where h is water depth and R is the hydraulic radius.

Table VIII. Performance of overall schemes according to the specified criteria given earlier.

Ease of implementation	Accuracy	Applicability	Stability	Simulation time
HLL	Osher	Problems	Almost	FVS
HLLC	REF	with REF.	identical	HLL
FVS	HLLC			HLLC
REF	FVS			Osher
Osher	HLL			REF
Easiest to most difficult	Least to most accurate			Fastest to slowest
Top to bottom	Top to bottom			Top to bottom

In the algorithms presented, friction computation is carried out using Manning's formula with a wide channel approximation. That is

$$Sf_x = \frac{n^2 v_x \sqrt{v_x^2 + v_y^2}}{h^{4/3}} \quad \text{in the } x \text{ direction}, \qquad Sf_y = \frac{n^2 v_y \sqrt{v_x^2 + v_y^2}}{h^{4/3}} \quad \text{in the } y \text{ direction}$$

where n is Manning's resistance coefficient.

However, in the algorithms presented and in many of the examples in the literature, these expressions are used for all computations. Therefore, if the ratio of water depth to channel width increases, the validity of the numerical solutions decreases.

Erduran and Kutija [10] demonstrate the validity of this assumption. They applied the finite volume method with the Osher scheme to several flow conditions in the same channel. Then, they compared the steady state results with the results obtained by Manning's formula with and without a wide channel approximation. As expected, their model results are almost the same as the results obtained using Manning's formula with the wide channel approximation. However, it is clear from the results that this assumption leads to errors and that these errors increase when the water depth increases.

8. CONCLUSIONS

As discussed above the performance of the schemes was evaluated according to the five criteria; ease of implementation, accuracy, applicability, simulation time and stability. Table VIII summarises the results of the numerical experiments in terms of these criteria.

8.1. Accuracy

The Osher scheme gives the most accurate results of the schemes tested. The Roe and FVS scheme are unsuitable for problems involving rarefaction shock unless the entropy violation is fixed. Otherwise they produce an unphysical jump, resulting in errors. The HLL scheme becomes diffusive in some cases. The HLLC scheme is found to be very efficient and sufficiently accurate to handle all of the flow conditions tested. Second order schemes give better results, but in some cases first order accurate solution works well. We have not found any significant differences between the predictor-corrector and van Leer to Hancock time integration methods in terms of accuracy. Slope limiters play a minor role in determining accuracy. We found the Minmod and van Leer limiters to be the most suitable. Second order or higher order accuracy

may be preferred for the simulation of large discontinuities. However, we found second order solutions accurate enough even for some extreme cases.

Accuracy of the finite volume solution depends on the shape and size of grids. Refining the grids gives better results. Introduction of more grid points is achieved by a static method in the examples discussed, although it would be much better if a dynamic grid generator was used. In particular, it may be needed for problems involving a moving shock wave. The shape of the grid is important for both the representation of the computational domain and the flow conditions. The better the representation, the better the results.

The splitting technique introduces errors, which can be significant in some problem. A wide channel assumption can cause significant errors in some cases and its use should thus be considered carefully.

8.2. Applicability

Regarding applicability, of all the schemes tested only the REF scheme fails in some conditions—those involving drying processes. This can be avoided by introducing a lower limit for water depth in a cell. First order accurate solutions are more flexible, the algorithms can be applied to both structured and unstructured grids without any problems. However, second order accurate solution algorithms are valid only for structured grids, which can be uniform or non-uniform. Use of the splitting technique provides some advantages since the solution of ODEs does not depend on the solution of the homogeneous hyperbolic part. Hence, introduction of additional source/sink terms does not increase the solution complexity. For instance, we introduce rainfall as an additional discharge in the continuity equation by use of the splitting technique. On the other hand, an additional split equation means additional stability restrictions. Therefore, selection of the time step for stable computation is influenced by two criteria.

8.3. Simulation time

In assessing simulation time, the FVS scheme was found to be the fastest and the REF scheme was found to be the slowest Riemann solver. Second order accurate solutions result in longer simulations, which can be twice or more of the computer time needed for a first order accurate solution. One interesting conclusion is that the van Leer to Hancock method takes significantly longer to run than the predictor–corrector approach does.

8.4. Stability

Stability criteria given in the literature are useful for the beginning of a run, but are found to be just an indicator. This is probably because the stability analysis applies to the linearized equations only. In practice some of the algorithms fail at a Courant number of 0.7 using condition CFL2. The stability regions of all the schemes presented are found to be almost identical. It is observed that the relationship between the time step and the distance between two neighbouring cells is non-linear. Therefore, to provide a stable solution, it is observed that a reduction in time step is not proportional to a decrease in the grid size.

Finally, we would like to highlight that a first order accurate solution algorithm using either the Osher or HLLC scheme can be recommended for simulation of all kinds of applications.

FURTHER WORK

The splitting technique is not always the best way to treat source terms and other possible terms such as lateral discharge. It produces errors in certain well-known cases. Other alternatives are shown to exhibit good results in the existence of variable topography but none of them have yet been proven to be the best. Each has its individual drawbacks and advantages. This requires further investigation.

ACKNOWLEDGEMENTS

The authors would like to thank Professor E.F. Toro for providing a draft copy of his book and his valuable comments on our results. The first author is sponsored by Turkish Republic Ministry of National Education, and he would like to thank them for their financial support. The third author would also like to thank University of Newcastle for their financial support.

REFERENCES

1. Abbott MB, Minns AW. *Computational Hydraulics* (2nd edn). Ashgate: USA, 1998.
2. Glaister PA. Numerical scheme for two-dimensional open channel flows with non-rectangular geometries. *International Journal of Engineering Science* 1993; **31**:1003–1011.
3. Glaister PA. Prediction of steady, supercritical, free-surface flow. *International Journal of Engineering Science* 1995; **33**:845–854.
4. Zhao DH, Shen HW, Lai JS, Tabios III GQ. Approximate Riemann Solvers in FVM for 2D hydraulic shock wave problems. *Journal of Hydraulic Engineering* 1996; **122**:692–702.
5. Mingham CG, Causon DM. High-resolution finite volume method for shallow water flows. *Journal of Hydraulic Engineering* 1998; **124**:605–614.
6. Zhao DH, Shen HW, Tabios III GQ, Lai JS, Tan WY. Finite-volume two-dimensional unsteady-flow model for river basins. *Journal of Hydraulic Engineering* 1994; **120**:863–882
7. Hu K, Mingham CG, Causon DM. A bore-capturing finite volume method For open-channel flows. *International Journal for Numerical Methods in Fluids* 1998; **28**:1241–1261.
8. Bermudez A, Dervieux A, Desideri J-A, Vazquez ME. Upwind schemes for the two-dimensional shallow water equations with variable depth using unstructured meshes. *Computer Methods in Applied Mechanics and Engineering* 1998; **155**:49–72.
9. Chippada S, Dawson CN, Martinez ML, Wheler MF. A Godunov-type finite volume method for the system of shallow water equations. *Computer Methods in Applied Mechanics and Engineering* 1998; **151**:105–129.
10. Erduran KS, Kutija V. Applications of finite volume method with Osher scheme and split technique on different types of flow in channel. In *Proceedings of the International Conference on Godunov Methods*: *Theory and Applications*. Oxford, UK, 1999.
11. Zoppou C, Roberts S. Catastrophic collapse of water supply reservoirs in urban areas. *Journal of Hydraulic Engineering* 1999; **125**:686–695.
12. Ye J, McCorquodale JA. Depth-averaged hydrodynamic model in curvilinear collocated grid. *Journal of Hydraulic Engineering* 1997; **123**:380–388.
13. Erduran KS, Kutija V, Hewett CJM. An investigation of a straight channel assumption for meandering channels. In *Proceedings of the 4th International Conference on Hydroinformatics*, Iowa, USA, 2000, p. 126.
14. Zhao DH, Chen Q, Lai JS, Yang J, Zhang L, Shen HW. 2D-FVM model for modelling riverbed deformation due to cutoff. In *Proceedings of the 4th International Conference on Hydroinformatics*, Iowa, USA, 2000, p. 255.
15. Delis AI, Skeels CP, Ryrie SC. Evaluation of some approximate Riemann solvers for transient open channel flows. *Journal of Hydraulic Research* 2000; **38**:217–231.
16. Kumar Jha A, Akiyama J, Ura M. First and second-order flux-difference splitting schemes for dam break problems. *Journal of Hydraulic Engineering* 1995; **121:**877–884.
17. Causon DM, Mingham CG, Ingram DM. Advances in calculation methods for supercritical flow in spillway channels. *Journal of Hydraulic Engineering* 1999; **125**:1039–1049.
18. Mingham CG, Causon DM. Calculation of unsteady bore diffraction using a high resolution finite volume method. *Journal of Hydraulic Research* 2000; **38**:49–56.
19. Alcrudo F, Garcia-Navarro P. A high-resolution Godunov-type scheme in finite volumes for the 2D shallow-water equations. *International Journal for Numerical Methods in Fluids* 1993; **16**:489–505.

20. Shen HW, Zhao DH, Tabios III GQ, Loftin K, Sculley S, Chamberlain J. Application of RBFVM-2D to kissimmee river restrotation in Florida state of USA. In the 27th *IAHR Congress*, Wang SSY (ed.), vol. 2. ASCE: San Francisco, 1997; 474–479.
21. Toro EF. *Rieman Solvers and Numerical Methods for Fluid Dynamics*. Springer: Berlin, 1997.
22. LeVeque RJ. *Numerical Methods for Conservation Laws* (2nd edn). Birkhauser: Berlin, 1999.
23. Harten A, Lax P, van Leer B. On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM Review* 1983; **25**:35–61.
24. Roe PL. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics* 1981; **43**:357–372.
25. Tan W. *Shallow Water Hydrodynamics*. Elsevier: Amsterdam, Netherlands, 1992.
26. Hirsch C. *Numerical Computation of Internal and External Flows*, vol. 2. Wiley: England, U.K., 1990.
27. Toro EF. *Shock-Capturing Methods for Shallow Flows*. Wiley: New York, 2001.
28. Toro EF, Spruce M, Spares W. Restoration of the contact surface in the HLL-Riemann solver. *Shock Waves* 1994; **4**:25–34.
29. Fraccarollo L, Toro EF. Experimental and numerical assessment of the shallow water model for two-dimensional dam-break type problems. *Journal of Hydraulic Research* 1995; **33**:843–864.
30. Harten A. High resolution schemes for hyperbolic conservation laws. *Journal of Computational Physics* 1983; **49**:357–393.
31. Steger JL, Warming RF. Flux vector splitting of the inviscid gas dynamic equations with application to finite—difference methods. *Journal of Computational Physics* 1981; **40**:263–293.
32. Van Leer B. On the relation between the upwind-differencing schemes of Godunov, Engquist-Osher, and Roe. *SIAM Journal of Scientific and Statistical Computing* 1984; **5**:1–20.
33. Jeng YN, Payne UJ. An adaptive TVD limiter. *Journal of Computational Physics* 1995; **118**:229–241.
34. Arora M, Roe PL. A well-behaved TVD limiter for high-resolution calculations of unsteady flow. *Journal of Computational Physics* 1997; **132**:3–11.
35. Delis AI, Skeels CP. TVD Schemes for open channel flow. *International Journal for Numerical Methods in Fluids* 1998; **26**:791–809.
36. Swan, T. *Delhi 4 Bible*. IDG Books Worldwide: Inc., U.S.A., 1998.
37. Jeong C. Comparison of the finite-difference and finite volume method in open channel. *M.Sc. Thesis*, University of Newcastle upon Tyne, U.K., 1999.
38. Garcia R, Kahawita RA. Numerical solution of the St. Venant equations with the MacCormack finite difference scheme. *International Journal for Numerical Methods in Fluids* 1986; **6**:507–527.
39. Tseng M-H. Explicit finite volume non-oscillatory schemes for 2D transient free-surface flows. *International Journal for Numerical Methods in Fluids* 1999; **30**:831–843.
40. Fennema RJ, Chaudhry MH. Explicit methods for 2-D transient free-surface flows. *Journal of Hydraulic Engineering* 1990; **116**:1013–1034.
41. Anastasiou K, Chan CT. Solution of the 2D shallow water equations using the finite volume method on unstructured triangular meshes. *International Journal for Numerical Methods in Fluids* 1997; **24**:1225–1245.
42. Louaked M, Hanich L. TVD scheme for the shallow water equations. *Journal of Hydraulic Research* 1998; **36**:363–378.
43. Courant R, Friedrichs K, Lewy H. Uber die pertieller Differenzen–Glichungen der mathematischen Physik. *Mathematische Annalar* 1928; **100**:32–74.
44. 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.
45. Vazquez-Cendon ME. Improved treatment of source terms in upwind schemes for the shallow water equations in channels with irregular geometry. *Journal of Computational Physics* 1999; **148**:497–526.
46. Roe PL. Upwind differenced schemes for hyperbolic conservation laws with source terms. In *Proceedings of the Conference on Hyperbolic Problems*, Carasso, Raviart, Serre (eds). Springer: Berlin, 1986; 41–51.

