INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 16, 489–505 (1993)

A HIGH-RESOLUTION GODUNOV-TYPE SCHEME IN FINITE VOLUMES FOR THE 2D SHALLOW-WATER EQUATIONS

FRANCISCO ALCRUDO AND PILAR GARCIA-NAVARRO

Departamento de Ciencia y Tecnologia de Materiales y Fluidos, Facultad de Ciencias, Universidad de Zaragoza, Zaragoza 50009, Spain

SUMMARY

A high-order Godunov-type scheme based on MUSCL variable extrapolation and slope limiters is presented for the resolution of 2D free-surface flow equations. In order to apply a finite volume technique of integration over body-fitted grids, the construction of an approximate Jacobian (Roe type) of the normal flux function is proposed. This procedure allows conservative upwind discretization of the equations for arbitrary cell shapes. The main advantage of the model stems from the adaptability of the grid to the geometry of the problem and the subsequent ability to produce correct results near the boundaries. Verification of the technique is made by comparison with analytical solutions and very good agreement is found. Three cases of rapidly varying two-dimensional flows are presented to show the efficiency and stability of this method, which contains no terms depending on adjustable parameters. It can be considered well suited for computation of rather complex free-surface two-dimensional problems.

KEY WORDS Free-surface flow Two-dimensional modelling Finite volumes MUSCL approach Upwind differencing

1. INTRODUCTION

In recent years many advances have been made in the study of hyperbolic partial differential equations and in the theory of one-dimensional difference operators applied to hyperbolic partial differential equations describing fluid flows.[1] Much is known about numerical techniques for solving conservation laws, i.e. equations of the form

$$\frac{\partial u}{\partial t}+\frac{\partial f(u)}{\partial x}=0. \tag{1}$$

Solutions of such equations produce discontinuities in general (unless f is a linear function of u) and numerical methods have been developed that can handle these non-linear features.

The numerical resolution of systems of conservation laws stems from the work of Lax and Wendroff (1960),[10] who formally expressed the importance of the conservative discretizations to capture automatically the discontinuities present in the solution. In the 1970s the concepts of artificial viscosity and modified equation were introduced and the numerical schemes such as Lax–Wendroff or MacCormack became popular.

At the end of the 1970s the original ideas of Godunov about the incorporation of the physical reality to the solution of simple problems were reconsidered. At the same time van Leer analysed

0271–2091/93/060489–17$13.50

Received June 1992
Revised November 1992

the properties of conservation and monotonicity for convection problems in a series of papers. This gave rise to an increase on the interest in the *upwind* schemes due to the appearance of the concept of *approximate Riemann solvers* in the 1980s. These made possible the formal generalization of the upwind schemes to systems of equations and led to two new techniques known as *flux vector splitting* and *flux difference splitting*.

Indeed, in the 1980s the classical schemes have become less widely used following the work of Harten and the advent of adaptive TVD schemes, such as those of Osher, Roe, van Leer, which make use of *flux limiters* and are able to generate successfully sharp solution profiles near fronts, at the same time being accurate in regions of smooth data. It is near these discontinuities that classical schemes show oscillatory behaviour and require the addition of artificial viscosity.

However, all the theory concerning conservation laws and flux limiters has been developed only in one space dimension. What happens when we are faced with a two-dimensional problem in computational fluid dynamics remains an unanswered question, even more if dealing with free-surface flows. There are only a few truly two-dimensional approaches to solving such problems and are generally based in operator splitting. These consist on regarding the two-dimensional problem as a two one-dimensional problem and then using the 1D scheme individually. A rectangular 2D mesh of grid points is required in this approach.

Despite excellent results obtained via operator splitting, certain questions could arise about the convenience of employing it when solving the two-dimensional open channel equations. Since no analysis has been done with respect to 2D non-linear systems of equations, this paper is a contribution to the evidence for and against a genuinely two-dimensional technique.

Assuming the validity of the shallow-water approach, a conservative formulation of the 2D system is applied together with the characteristic formulation necessary to ensure an adequate boundary problem treatment. A finite volume technique based on a high-resolution Godunov-type scheme is described for the discretization of the partial difference equations. The method has been devised to perform integration on body-fitted meshes in an attempt to overcome the difficulty of having curved boundaries.

2. GOVERNING EQUATIONS

One of the more widely used approaches for the description of free-surface flows is that of shallow water, which can be obtained from the depth-averaged Navier–Stokes equations. The resulting system corresponds to the one of a compressible fluid (in which the depth plays the role of density) similar to an isentropic gas with $\gamma = 2$. This system represents the mass and momentum conservation and its resolution may lead to discontinuities. It is a partial differential non-linear system of equations well suited for the application of the above-stated numerical techniques.

In the present work, the terms describing the diffusion of momentum due to turbulence are considered negligible and not included in the equations, which may be written in conservative form as follows:

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \mathbf{F} = \mathbf{S}. \tag{2}$$

Or, more commonly, as

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{E}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} = \mathbf{S}, \tag{3}$$

since $\mathbf{F} = (\mathbf{E},\ \mathbf{G})$.

The vector of unknowns is

$$\mathbf{U}=\begin{pmatrix} h \\ uh \\ vh \end{pmatrix}. \tag{4}$$

And the Cartesian components of the flux

$$\mathbf{E}=\begin{pmatrix} uh \\ u^2h+\dfrac{gh^2}{2} \\ uvh \end{pmatrix}, \qquad \mathbf{G}=\begin{pmatrix} vh \\ uvh \\ v^2h+\dfrac{gh^2}{2} \end{pmatrix}, \tag{5}$$

where h represents the water depth, u and v stand for the depth-averaged velocity components along the x and y directions, respectively, and g is the acceleration due to gravity.

The right-hand side of the equation contains the sources and sinks of momentum. It accounts for the bed slopes and the friction losses along the two co-ordinate directions. In this paper, only the numerical treatment of the homogeneous system will be described.

It will be later on useful to have the homogeneous system of equations expressed in the equivalent non-conservative form

$$\frac{\partial \mathbf{U}}{\partial t}+\mathbf{A}\frac{\partial \mathbf{U}}{\partial x}+\mathbf{B}\frac{\partial \mathbf{U}}{\partial y}=0. \tag{6}$$

The Jacobians of the fluxes are

$$\mathbf{A}=\frac{\partial \mathbf{E}}{\partial \mathbf{U}}=\begin{pmatrix} 0 & 1 & 0 \\ -u^2+c^2 & 2u & 0 \\ -uv & v & u \end{pmatrix}, \qquad \mathbf{B}=\frac{\partial \mathbf{G}}{\partial \mathbf{U}}=\begin{pmatrix} 0 & 0 & 1 \\ -uv & v & u \\ -v^2+c^2 & 0 & 2v \end{pmatrix}, \tag{7}$$

where the wave speed $c=\sqrt{(gh)}$.

The respective eigenvalues are

$$\mathbf{A}\begin{cases} a^1=u+c, \\ a^2=u, \\ a^3=u-c, \end{cases} \qquad \mathbf{B}\begin{cases} b^1=v+c, \\ b^2=v, \\ b^3=v-c, \end{cases} \tag{8}$$

and the eigenvectors are

$$\mathbf{A}\begin{cases} \mathbf{e}^1=(1, u+c, v), \\ \mathbf{e}^2=(0, 0, c), \\ \mathbf{e}^3=(1, u-c, v), \end{cases} \qquad \mathbf{B}\begin{cases} \mathbf{f}^1=(1, u, v+c), \\ \mathbf{f}^2=(0, -c, 0), \\ \mathbf{f}^3=(1, u, v-c). \end{cases} \tag{9}$$

They correspond to the characteristic speeds and directions and, therefore, inform about the propagation of information, providing a means of building up approximations which transmit information only in the correct directions.

3. 1D NUMERICAL SCHEME

For the sake of clarity, the numerical scheme used to discretize (2) is described first in the scalar one-dimensional case, that is, taking a domain of integration (x, t) characterized by points (x_i, t^n), where $x_i=i\Delta x$ and $t^n=n\Delta t$.

When solving a one-dimensional equation of the form (1), the solution at time $(n+1)\Delta t$ for the computational point i is obtained by application of a conservative second order in space and time predictor–corrector sequence as follows:

$$u_i^{\mathrm{P}} = u_i^n - \frac{\Delta t}{2\Delta x}\left(f_{i+1/2}^{*n} - f_{i-1/2}^{*n}\right), \tag{10}$$

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{\Delta x}\left(f_{i+1/2}^{*\mathrm{P}} - f_{i-1/2}^{*\mathrm{P}}\right). \tag{11}$$

The quantity $f_{i+1/2}^*$ represents the numerical flux through the wall between cells i and $i+1$ and is calculated as follows:

$$f_{i+1/2}^* = \tfrac{1}{2}\left[f_{\mathrm{R}}^n + f_{\mathrm{L}}^n - |\tilde{a}_{i+1/2}|(u_{\mathrm{R}} - u_{\mathrm{L}})\right], \tag{12}$$

u_{R} and u_{L} being two states on both sides of the wall at the time level $n\Delta t$ obtained from the initial distribution of the discrete function u_i^n at the grid points and the local slopes of this distribution, following a MUSCL[2] procedure, which is illustrated in Figure 1:

$$\begin{aligned} u_{\mathrm{R}} &= u_{i+1}^n - \tfrac{1}{2}(\delta u)_{i+1}^n \\ u_{\mathrm{L}} &= u_i^n + \tfrac{1}{2}(\delta u)_i^n. \end{aligned} \tag{13}$$

The local slopes of the discrete function are calculated as

$$(\delta u)_i^n = (u_i^n - u_{i-1}^n)\,\varphi\left(\frac{u_{i+1}^n - u_i^n}{u_i^n - u_{i-1}^n}\right). \tag{14}$$

The presence of these two intermediate states in the numerical flux is equivalent to having a piecewise linear initial distribution of the function and, accordingly, renders the Godunov-type scheme second-order-accurate in space. On the other hand, it is important to remark that second-order accuracy in time is achieved by the predictor integration over one-half time step.

The limiter φ in (14) is a non-linear function of the ratio of adjacent gradients and is responsible for obtaining non-oscillatory solutions despite the application of a second-order scheme to problems containing discontinuities. Several forms of the function can be found in the literature,[1] their effect being basically the same, adding sufficient dissipation to the scheme to guarantee monotonicity when there is a steep gradient, while retaining second-order accuracy in regions of smooth flow.

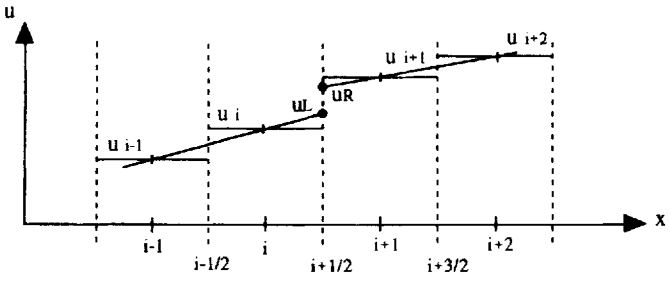


Figure 1. Linear one-sided variable extrapolation of interface values.

On the other hand, $f_R = f(u_R)$, $f_L = f(u_L)$ and $\tilde{a}_{i+1/2} = \tilde{a}(u_R, u_L)$ is an approximation to the characteristic speed at the wall defined as follows:

$$\tilde{a}_{i+1/2} = \begin{cases} \dfrac{f_R - f_L}{u_R - u_L} & \text{if } u_R \neq u_L, \\ a_R = a_L & \text{if } u_R = u_L, \end{cases} \tag{15}$$

where $a = \partial f / \partial u$.

The quantity $f^{*P}_{i+1/2}$ present in the case of the corrector step, represents the numerical flux function evaluated at the predictor values of the functions u^P_R, u^P_L:

$$f^{*P}_{i+1/2} = \tfrac{1}{2}[f^P_R + f^P_L - |a^P_{i+1/2}|(u^P_R - u^P_L)]. \tag{16}$$

These states can now be calculated from the predictor values of the function. It is worth noting that the values of the local slopes evaluated at time level $n\Delta t$ should be used in these definitions:

$$\begin{aligned} u^P_R &= u^P_{i+1} - \tfrac{1}{2}(\delta u)^n_{i+1} \\ u^P_L &= u^P_i + \tfrac{1}{2}(\delta u)^n_i. \end{aligned} \tag{17}$$

4. 2D GENERALIZATION

In this section the extension of the algorithm introduced in Section 3 to the two-dimensional case is presented and the finite volume integration procedure applied is described.

The first problem under consideration is the discretization of the homogeneous part of the system represented by equation (2):

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \mathbf{F} = 0. \tag{18}$$

This is done in a spatially 2D domain of integration divided into a set of cells labelled (i, j), each of them identified by the corresponding centre point. The grid is not forced to be rectangular; on the contrary, the shape and size of the cells can be non-uniform, it being nevertheless advisable for the sake of accuracy to follow a certain degree of smoothness when defining the mesh variation. In Figure 2, the elemental cell (i, j) and its immediate neighbours are sketched.

The equations are integrated by a finite volume technique on each of these cells covering the whole domain, so that it is convenient to rewrite equation (18) in integral form:

$$\int_V \left(\frac{\partial \mathbf{U}}{\partial t} + \nabla \mathbf{F} \right) \mathrm{d}V = 0. \tag{19}$$

The application of Gauss theorem to the integral of the second term gives

$$\int_V \frac{\partial \mathbf{U}}{\partial t} \, \mathrm{d}V = \oint_S (\mathbf{F} \cdot \mathbf{n}) \, \mathrm{d}s = 0, \tag{20}$$

where the volume integral actually represents integral of the time evolution of the function over the area of the cell, whilst the surface integral is the total normal flux through the cell boundaries. The normal vectors to the cell walls are defined as shown in Figure 3. The scalar product can be expressed in terms of the Cartesian components:

$$\mathbf{F} \cdot \mathbf{n} = \mathbf{E} n_x + \mathbf{G} n_y. \tag{21}$$

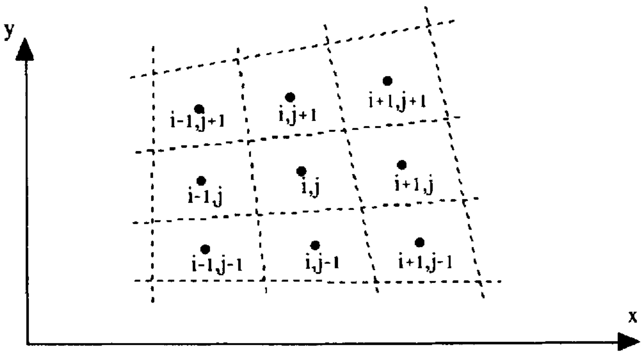


Figure 2. Two-dimensional cells in arbitrary co-ordinates.

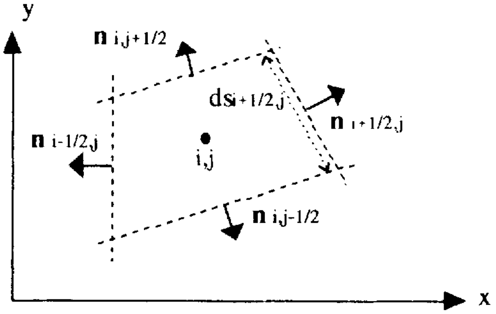


Figure 3. Normal vectors and side length in a cell of the two-dimensional grid.

Assuming $\mathbf{U}$ to be variable with t but constant over the cell, equation (20) becomes

$$\frac{\partial \mathbf{U}}{\partial t}\Delta V+\oint_S (\mathbf{F}\cdot\mathbf{n})\,\mathrm{d}s=0. \tag{22}$$

Equation (22) can be discretized now provided that the surface integral is approximated by a sum over the four walls of a numerical flux in the following way:

$$\oint_S (\mathbf{F}\cdot\mathbf{n})\,\mathrm{d}s \approx \sum_{r=1}^{4} (\mathbf{F}_r^*\cdot\mathbf{n}_r)\,\mathrm{d}s_r, \tag{23}$$

where $\mathrm{d}s_r$ are, respectively, the lengths of the four walls which contour the cell i, j.

In this work, the applied numerical flux function is the one corresponding to the high-order Godunov-type scheme described in Section 3. It can be generalized for the 2D system case so that, for the wall corresponding to $r=i+1/2, j$ has the following form:

$$(\mathbf{F}^*\cdot\mathbf{n})_{i+1/2,j}=\tfrac{1}{2}[\mathbf{F}_\mathrm{R}+\mathbf{F}_\mathrm{L}-|\mathbf{A}_{i+1/2,j}|(\mathbf{U}_\mathrm{R}-\mathbf{U}_\mathrm{L})]\cdot\mathbf{n}_{i+1/2,j}, \tag{24}$$

where $\mathbf{A}_{i+1/2,j}$ is the Jacobian matrix of the projection of the flux $\mathbf{F}$ in the normal direction evaluated at some average of the variables at states $\mathbf{U}_\mathrm{R}$, $\mathbf{U}_\mathrm{L}$:

$$\mathbf{A}=\frac{\partial(\mathbf{F}\cdot\mathbf{n})}{\partial \mathbf{U}}=\begin{pmatrix} 0 & n_x & n_y \\ (c^2-u^2)n_x-uvn_y & 2un_x-vn_y & un_y \\ -uvn_x+(c^2-v^2)n_y & vn_x & un_x+2vn_y \end{pmatrix}. \tag{25}$$

Following the technique proposed by Roe for constructing a suitable coefficient matrix to define a modified, linear but equivalent system of conservation laws, we have searched for a matrix $\tilde{\mathbf{A}}$ with eigenvalues of the form[3]

$$\begin{aligned}\tilde{a}^1 &= \tilde{\omega}\cdot\mathbf{n}+\tilde{c}=\tilde{u}n_x+\tilde{v}n_y+\tilde{c},\\ \tilde{a}^2 &= \tilde{\omega}\cdot\mathbf{n}=\tilde{u}n_x+\tilde{v}n_y,\\ \tilde{a}^3 &= \tilde{\omega}\cdot\mathbf{n}-\tilde{c}=\tilde{u}n_x+\tilde{v}n_y-\tilde{c},\end{aligned} \tag{26}$$

where $\tilde{\omega}=(\tilde{u},\tilde{v})$, and eigenvectors of the form

$$\tilde{\mathbf{e}}^1=\begin{pmatrix}1\\ \tilde{u}+\tilde{c}n_x\\ \tilde{v}+\tilde{c}n_y\end{pmatrix},\qquad \tilde{\mathbf{e}}^2=\begin{pmatrix}0\\ -\tilde{c}n_y\\ \tilde{c}n_x\end{pmatrix},\qquad \tilde{\mathbf{e}}^3=\begin{pmatrix}1\\ \tilde{u}-\tilde{c}n_x\\ \tilde{v}-\tilde{c}n_y\end{pmatrix}. \tag{27}$$

They are written in terms of the average values $\tilde{u},\tilde{v}$, and $\tilde{c}$, which have been found to be

$$\tilde{u}=\frac{u_{\mathrm{R}}\sqrt{h_{\mathrm{R}}}+u_{\mathrm{L}}\sqrt{h_{\mathrm{L}}}}{\sqrt{h_{\mathrm{R}}}+\sqrt{h_{\mathrm{L}}}},\qquad \tilde{v}=\frac{v_{\mathrm{R}}\sqrt{h_{\mathrm{R}}}+v_{\mathrm{L}}\sqrt{h_{\mathrm{L}}}}{\sqrt{h_{\mathrm{R}}}+\sqrt{h_{\mathrm{L}}}} \tag{28}$$

$$\tilde{c}=\sqrt{\frac{g(h_{\mathrm{R}}+h_{\mathrm{L}})}{2}}. \tag{29}$$

If the approximate Jacobian matrix is used instead of $\mathbf{A}$, the numerical flux (24) can be transformed as

$$(\mathbf{F}^*\cdot\mathbf{n})_{i+1/2,j}=\frac{1}{2}\left[\mathbf{F}_{\mathrm{R}}+\mathbf{F}_{\mathrm{L}}-\sum_{k=1}^{3}\tilde{\alpha}^k|\tilde{a}^k|\tilde{\mathbf{e}}^k\right]\cdot\mathbf{n}_{i+1/2,j}, \tag{30}$$

where the $\tilde{\alpha}^k$ are the coefficients of the decomposition in the basis of eigenvectors of $\tilde{\mathbf{A}}$,

$$\mathbf{U}_{\mathrm{R}}-\mathbf{U}_{\mathrm{L}}=\sum_{k=1}^{3}\tilde{\alpha}^k\mathbf{e}^k, \tag{31}$$

and are dependent on the jumps $\Delta=(\ )_{\mathrm{R}}-(\ )_{\mathrm{L}}$ in the functions:

$$\tilde{\alpha}^{1,3}=\frac{\Delta h}{2}\pm\frac{1}{2\tilde{c}}\left[\Delta(hu)n_x+\Delta(hv)n_y-\tilde{\mathbf{u}}\cdot\hat{n}\,\Delta h\right] \tag{32}$$

$$\tilde{\alpha}^2=\frac{1}{\tilde{c}}\left\{[\Delta(hv)-\tilde{v}\Delta h]n_x-(\Delta(hu)-\tilde{u}\Delta h)n_y\right\}. \tag{33}$$

The intermediate states $\mathbf{U}_{\mathrm{R}}$, $\mathbf{U}_{\mathrm{L}}$ are defined at the cellwalls. They are extrapolated from the centrepoints of the neighbouring cells by using slope limiters in a similar way as was done in the 1D case. They are responsible for the high-order accuracy of the scheme. Their expressions when dealing, for instance, with wall $(i+1/2,j)$ are

$$\begin{aligned}\mathbf{U}_{\mathrm{R}}&=\mathbf{U}^n_{i+1,j}-\tfrac{1}{2}(\delta\mathbf{U})^n_{i+1,j}\\ \mathbf{U}_{\mathrm{L}}&=\mathbf{U}^n_{i,j}+\tfrac{1}{2}(\delta\mathbf{U})^n_{i,j},\end{aligned} \tag{34}$$

$$(\delta\mathbf{U})^n_{i,j}=(\mathbf{U}^n_{i,j}-\mathbf{U}^n_{i-1,j})\cdot\varphi\left(\frac{\mathbf{U}^n_{i+1,j}-\mathbf{U}^n_{i,j}}{\mathbf{U}^n_{i,j}-\mathbf{U}^n_{i-1,j}}\right), \tag{35}$$

where formula (35) should be interpreted componentwise for the vector $\mathbf{U}$.

4.1. Numerical stability

The stability of an unsplit two-dimensional scheme when used to solve a linear advection equation of the form

$$\frac{\partial u}{\partial t}+a\frac{\partial u}{\partial x}+b\frac{\partial u}{\partial y}=0, \quad a,b>0 \tag{36}$$

is restricted by the CFL-like condition[4]

$$\Delta t\leq\left(\frac{a}{\Delta x}+\frac{b}{\Delta y}\right)^{-1}. \tag{37}$$

A somewhat different criterion, better suited for the finite volume formulation, has been adopted for the actual computation:

$$\Delta t\leq\frac{\min\{\mathrm{d}r_{i,j}\}}{2\max\{(c+\sqrt{[u^2+v^2]})_{i,j}\}}. \tag{38}$$

In the above inequality, the quantities $\mathrm{d}r_{i,j}$ represent the whole set of distances between every centrepoint (i,j) and those of its four adjacent cells.

4.2. Initial and boundary conditions

The information required to start the time evolution computation on the two-dimensional domain has been provided by specifying the values of the three dependent variables h, u, v at every grid point (i,j) for time $t=0$, and assuming them to be uniform over the cell represented by that point.

The numerical treatment of the boundary problem has been intended to be as correctly as possible and, hence, the theory of characteristics in two dimensions has been followed. As is well known, a consistent set of boundary conditions is required to complement the flow equations. The local value of the Froude number is what determines the flow regime and, accordingly, the correct number of boundary conditions to be applied. For subcritical flow, two external conditions must be specified at inflow boundaries, whereas only one is required at the outflow boundary. Two-dimensional supercritical flow requires the imposition of three inflow boundary conditions and none at the downstream side, where the flow remains free and only influenced by the information coming from the interior points.

In the particular case of solid walls limiting the flow field, the velocity is projected into the tangential and normal directions to the wall. Then the latter is set equal to zero in order to represent no flux through the solid boundary.

Use is made of the information carried by the outgoing bicharacteristics (Riemann invariants) in order to calculate the remaining unknown variables at the borders.

5. NUMERICAL RESULTS

5.1. Oblique hydraulic jump

A fundamental aspect when dealing with numerical schemes is to be able to check their predictions against suitable test problems, preferably ones for which an exact solution is available. Such is the case for the first example presented, in which an oblique hydraulic jump is induced by means of an interaction between a supercritical flow and a converging wall deflected

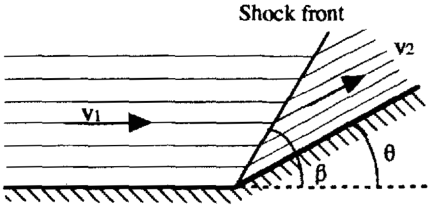


Figure 4. Planar view of oblique shock front in supercritical flow.

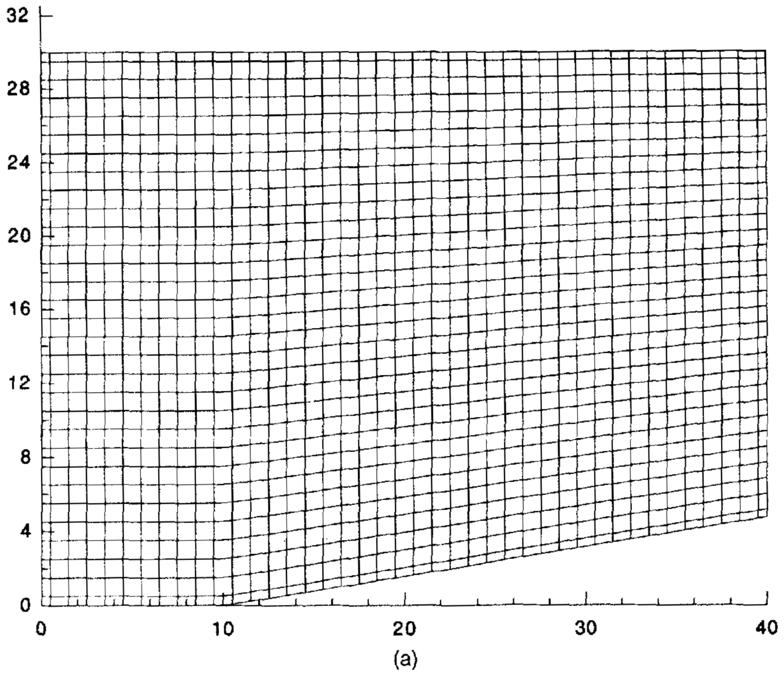


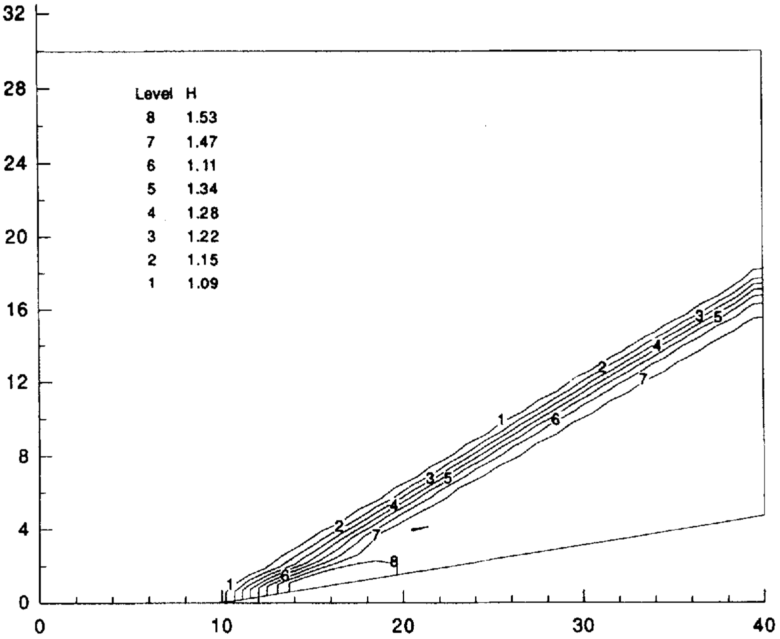
Figure 5(a). Grid used in the oblique hydraulic jump test case.

through an angle θ. As reported in the literature, the equation for the angle formed by the shock wave is defined in Figure 4.

A 41×31 non-rectangular mesh, displayed in Figure 5(a), was used to reproduce the discontinuous flow in a non-prismatic channel where $\theta = 8{\cdot}95^\circ$ for the converging side. The chosen initial conditions were $h_0(i,j) = 1$ m, $u_0(i,j) = 8{\cdot}57$ m/s and $v_0(i,j) = 0$, that is, a uniform supercritical flow with $Fr_0 = 2{\cdot}74$. Supercritical flow boundary conditions were applied both upstream and downstream.

The exact solution corresponding to the upstream flow and geometry imposed was calculated. The predicted values were $h_2 = 1{\cdot}5$ m, $|\mathbf{u}_2| = 7{\cdot}9556$, $Fr_2 = 2{\cdot}075$ for the downstream variables and $\beta = 30^\circ$ for the angle of the jump connecting them to the given upstream conditions.

Using the proposed scheme, the computation converged to the steady state from the given initial situation. As can be verified from Figure 5(b), the agreement of the numerical results with the correct solution was very good despite the fact that the grid could have been improved. The



(b)

Figure 5(b). Depth contour plot showing oblique hydraulic jump.

angle formed by the oblique hydraulic jump was closely reproduced (the error found was of 1·56 per cent) as well as the values of the flow variables on both sides of it ($h_2 = 1{\cdot}5049$ m, $|\mathbf{u}_2| = 7{\cdot}9419$, $Fr_2 = 2{\cdot}068$) and a discontinuous water surface devoid of oscillations was obtained.

5.2. *Dambreak problem*

Various solutions of the one-dimensional shallow-water equations for the initial boundary value of dambreak flow have recently become available. When the assumptions underlying these models were justified, the results were excellent. Some numerical results on the mathematical modelling of the two-dimensional dambreak problem have also been reported.[5–9]

The high-order Godunov-type scheme together with the *minmod* slope limiter is applied to the test case computed by Fennema *et al.*[5] for partial failure of a dam in a 200 × 200 m basin of simplified geometry depicted in Figure 6. The initial height ratio is 2, with values of $h_{01} = 10$ m and $h_{02} = 5$ m on both sides of an idealized dam that has been represented just as a mathematical discontinuity on the water surface. Water is released into the downstream side through a breach 75 m wide, forming a wave that propagates while spreading laterally. At the same time, a negative wave spreads into the reservoir, with its speed depending on the local undisturbed depth of water.

A 40 × 40 cell rectangular grid was chosen in this case taking into account the regular shape of the domain of integration. Figure 7 shows a 3D view of the water surface elevation as obtained at a time $t = 7{\cdot}2$ s in which the waves have not yet reached all the boundaries. It has to be remarked that the method needs no tuning of any adjustable parameter such as artificial viscosity coefficients as other methods do. Since the computation is inviscid, strong rarefactions involving large velocity gradients appear around the edges of the breach and represent a demanding test for the numerical procedure. They manifest as abrupt depressions in the water surface elevation in

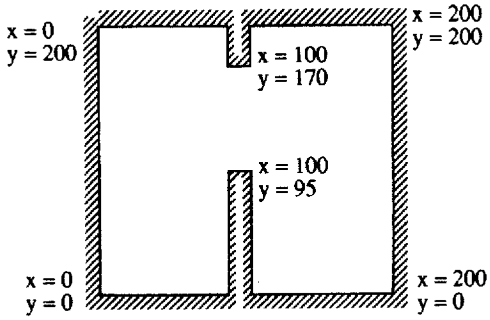


Figure 6. Reservoir geometry for the example of partial failure of a dam.

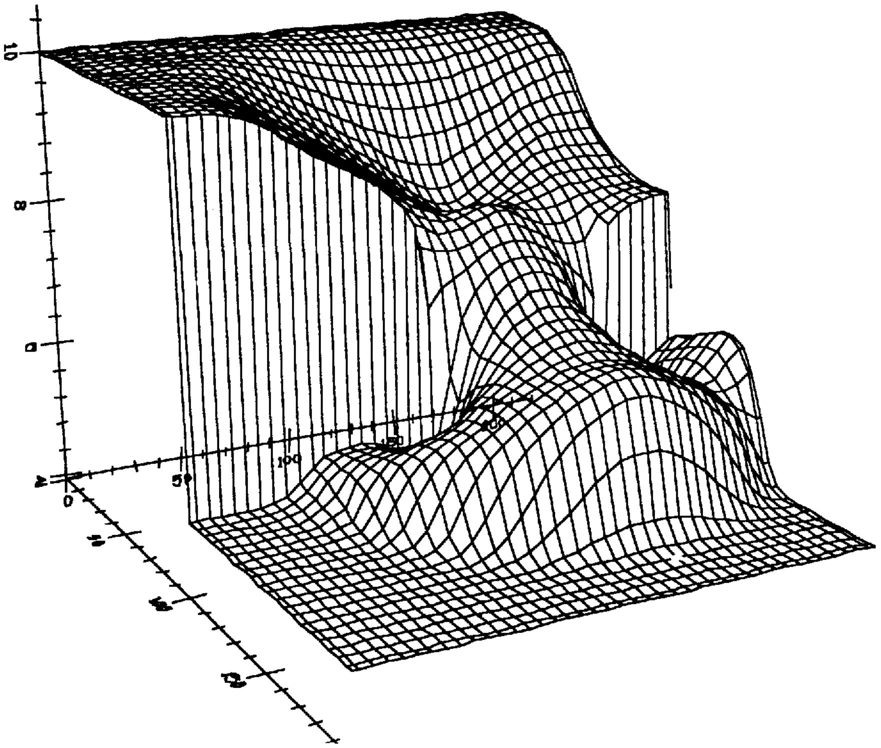


Figure 7. Water surface elevation at $t = 7{\cdot}2$ s after breaking of the dam.

these regions but, as can be observed, the numerical solution shows up as being stable and well-behaved everywhere.

Figure 8(a) provides a map of level lines for the depth, and the velocity field for this test is plotted in Figure 8(b) making plain the strongly two-dimensional character of the flow.

5.3. *Breaking of a circular dam*

Another interesting test case for the analysis of the algorithm performance is that of the breaking of a dam of cylindrical geometry and the time evolution of the subsequent waves. The

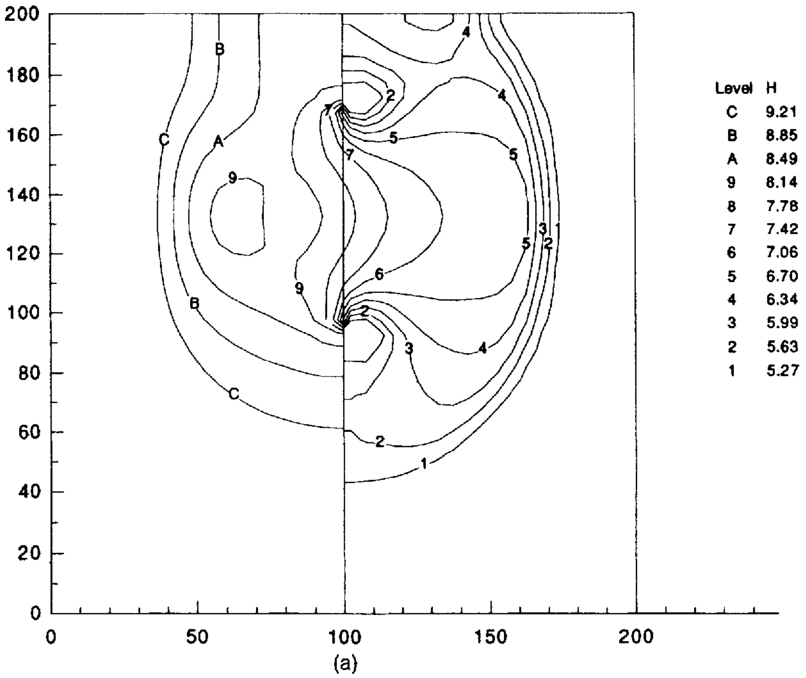


Figure 8(a). Contour plot showing depth distribution for the partial dambreak test case.

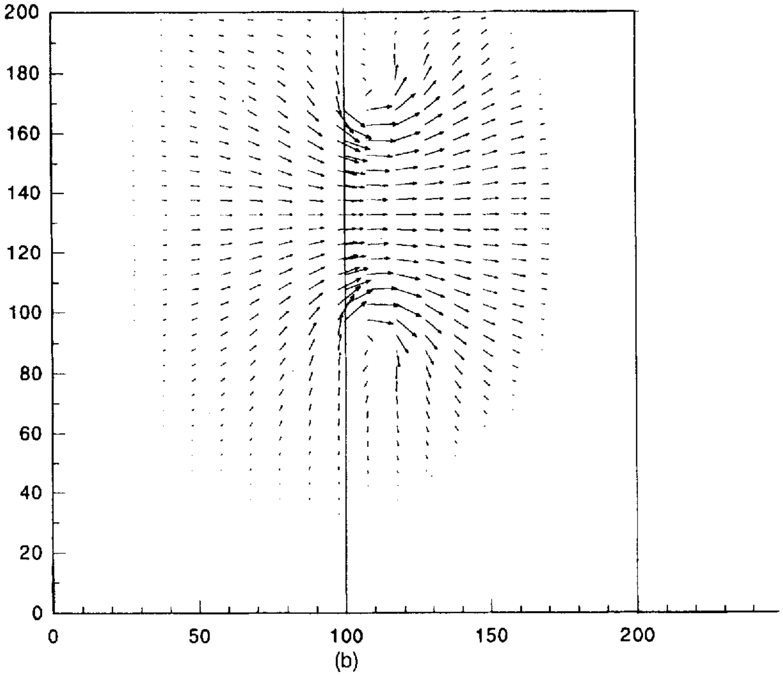


Figure 8(b). Velocity field for the partial dambreak test case.

high symmetry of the problem renders it an adequate example to evidence the advantages of taking body-fitted co-ordinates instead of a rectangular mesh for the numerical integration.

It is well known that, when transforming the flow equations to polar co-ordinates, the mathematical problem of the breaking of a circular dam becomes one-dimensional along the radial direction, that is, independent of the polar angle. Even though there is not an exact solution to compare with, this enables the possibility of checking the ability of the method to conserve symmetries.

For the sake of comparison, two calculations were made, one using a rectangular grid (following the Cartesian co-ordinates) and another one in a circular mesh (following the r, θ polar co-ordinates). The rectangular grid was constructed with 50×50 cells of individual size 1×1 m. The circular grid was 50×25 (thus, half the number of points of the previous one), that is, 50 cells following the tangential direction versus 25 cells of 1 m length along the radial direction. They are displayed in Figures 9(a) and 9(b), respectively.

The initial conditions were the same for both cases, two regions of still water separated by a cylindrical wall (radius = 11 m) so that, on the inner side $h_{01} = 10$ m, whilst $h_{02} = 1$ m outside the dam. All the boundaries were supposed to be rigid.

The numerical results obtained with the high-order Godunov scheme on each of the grids for time $t = 0{\cdot}69$ s are shown in Figures 10–12. Figures 10(a) and 10(b) are plots of the velocity fields as computed in the rectangular and the circular meshes, respectively. Figures 11(a) and 11(b) represent the respective level line maps of the depth. The differences between both calculations can be clearly observed in both sets of pictures. The results produced in the rectangular mesh are somewhat *squared* though a perfect circular symmetry is expected. The advancing front is not perfectly circular, and the zone between it and the depression wave shows small but unexpected

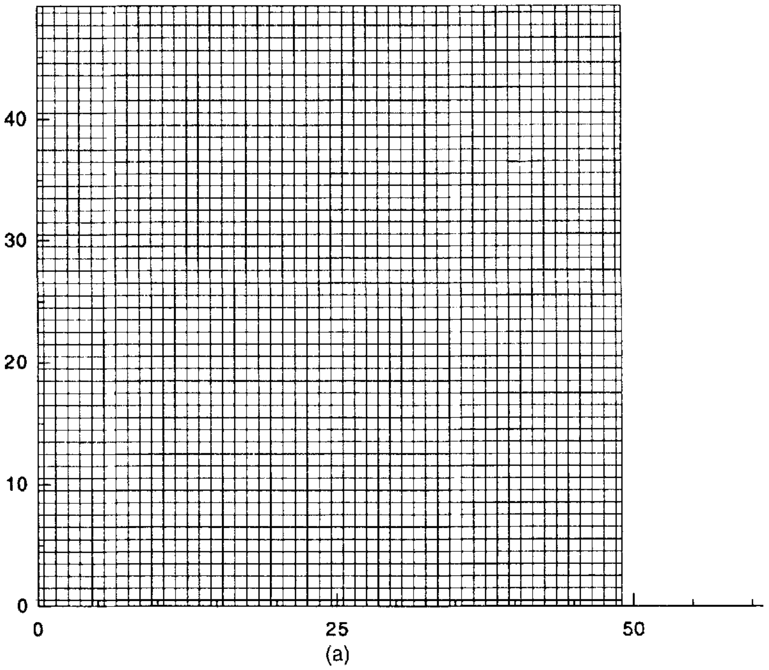


Figure 9(a). Rectangular mesh used in the circular dambreak test case.

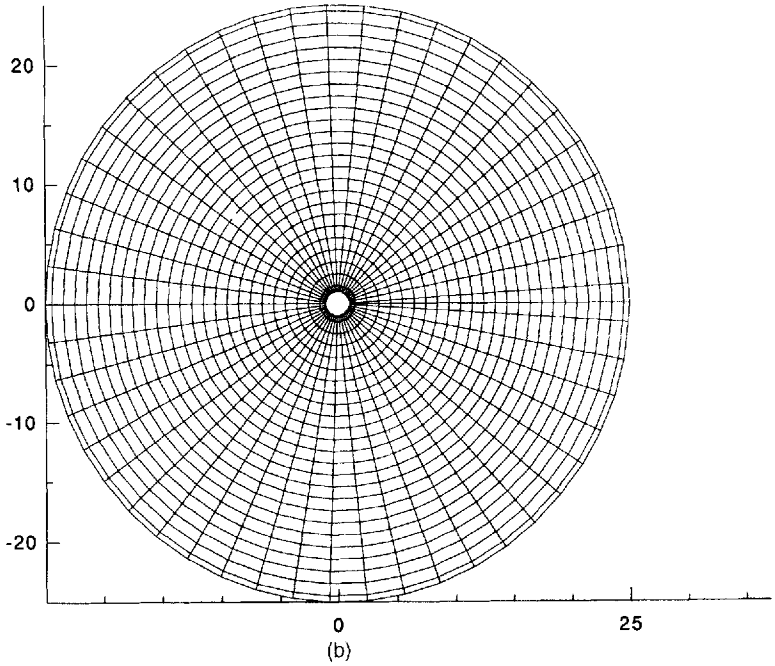


Figure 9(b). Circular mesh used in the circular dambreak test case.

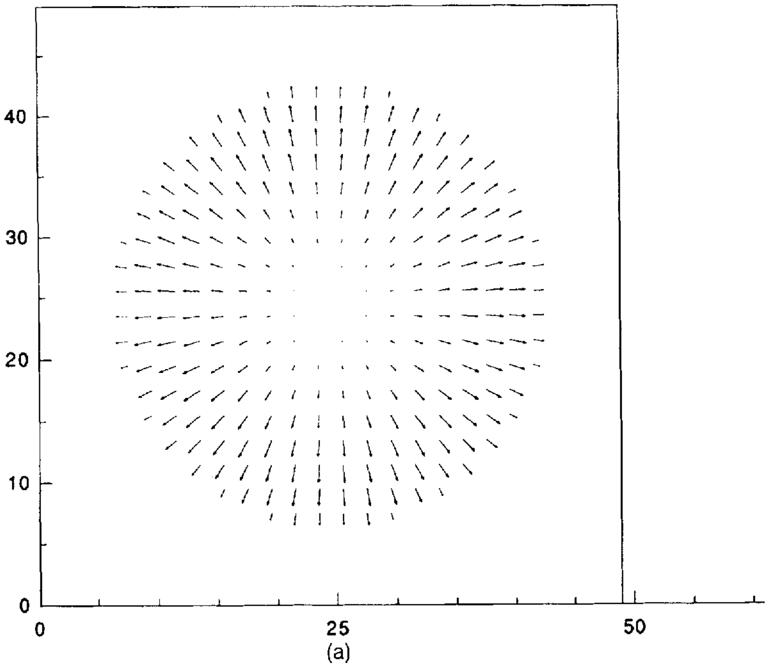


Figure 10(a). Velocity field computed using the rectangular mesh.

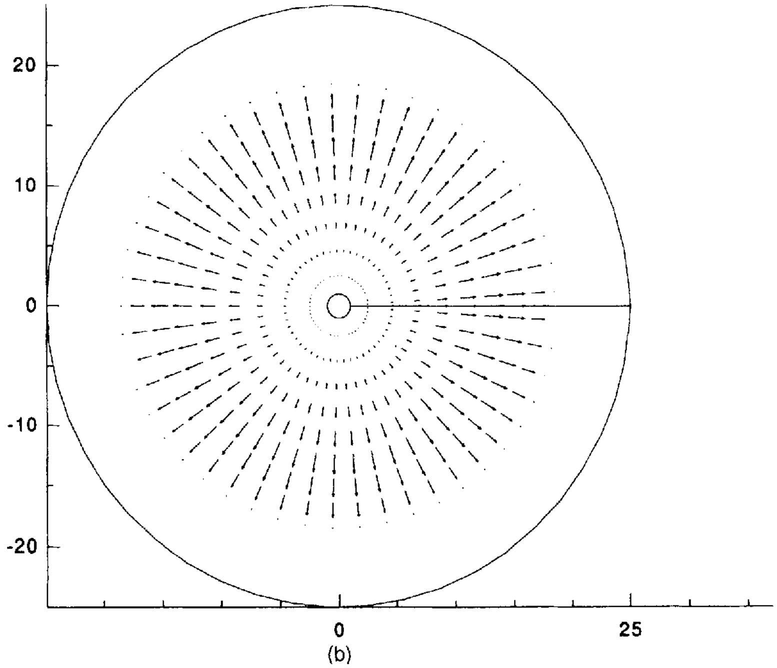


Figure 10(b). Velocity field computed using the circular mesh.

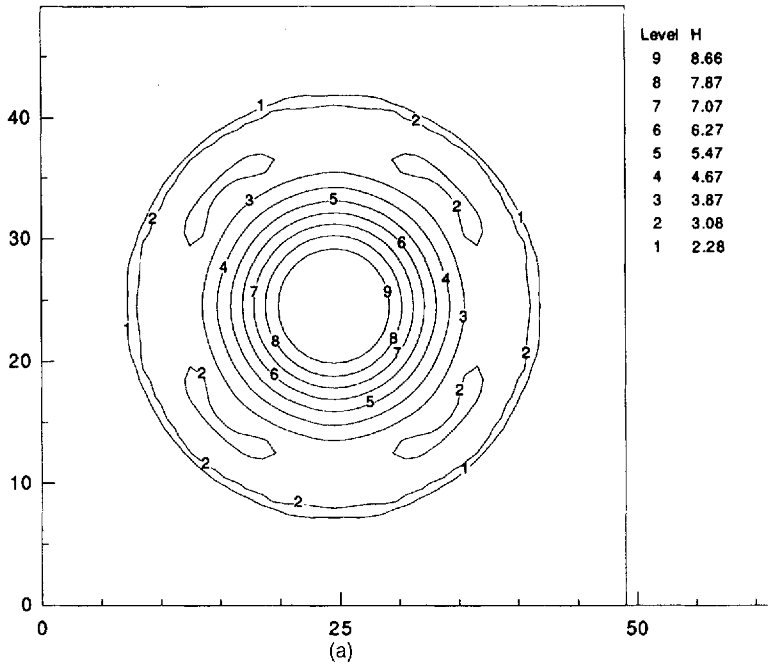


Figure 11(a). Contour plot of the depth computed using the rectangular mesh.

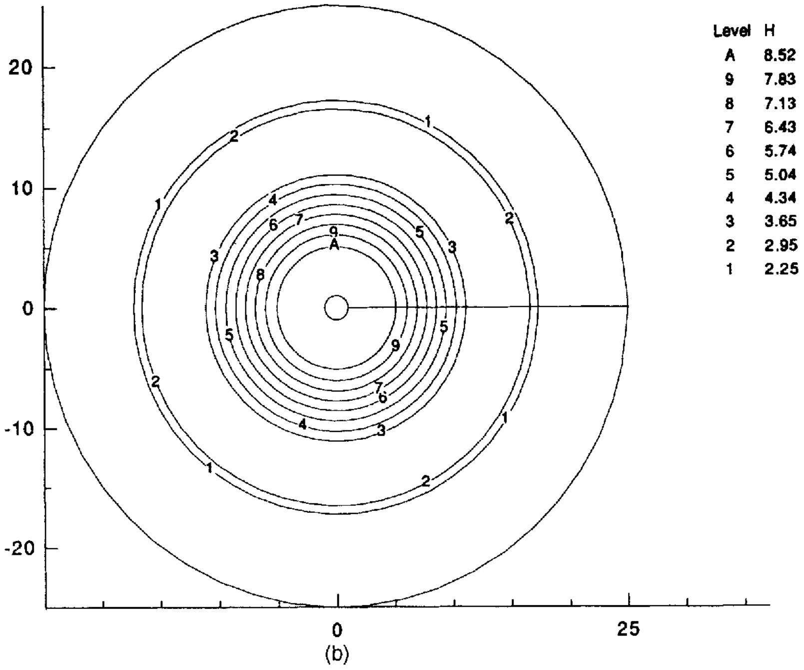


Figure 11(b). Contour plot of the depth computed using the circular mesh.

variations of the depth in the form of four symmetrically distributed kidney-shaped isles. All this negative features are not present when the circular grid is used. This simple test case points out the mesh dependence of the numerical solutions and strongly recommends the use of grids adapted as close as possible to the geometry of the problem. In this sense the finite volume formulation used in this paper shows an advantage over classical finite difference methods.

Finally, a perspective view plot of the free-surface results computed in the rectangular grid is shown in Figure 12. The irregularities around the propagating front and in the zone between it and the depression wave mentioned above are visible on this picture. It seems that the two co-ordinate directions are considered by the scheme as privileged lines.

6. CONCLUSIONS

A high-order Godunov-type scheme based on MUSCL variable extrapolation and slope limiters is presented for the resolution of 2D free-surface equations, and a finite volume technique of integration over body-fitted grids is proposed.

The construction of an approximate Jacobian of the normal flux function allows conservative upwind discretization of the equations for arbitrary cell shapes. This treatment enables efficient calculation of subcritical as well as supercritical two-dimensional flows and makes possible dealing with surface discontinuities. The main advantage of the model stems from the adaptability of the grid to the geometry of the problem and the subsequent ability to produce correct results near the boundaries.

Verification of the technique is made by comparison with analytical solutions and very good agreement is found. Three cases of rapidly varying two-dimensional flows are presented to show

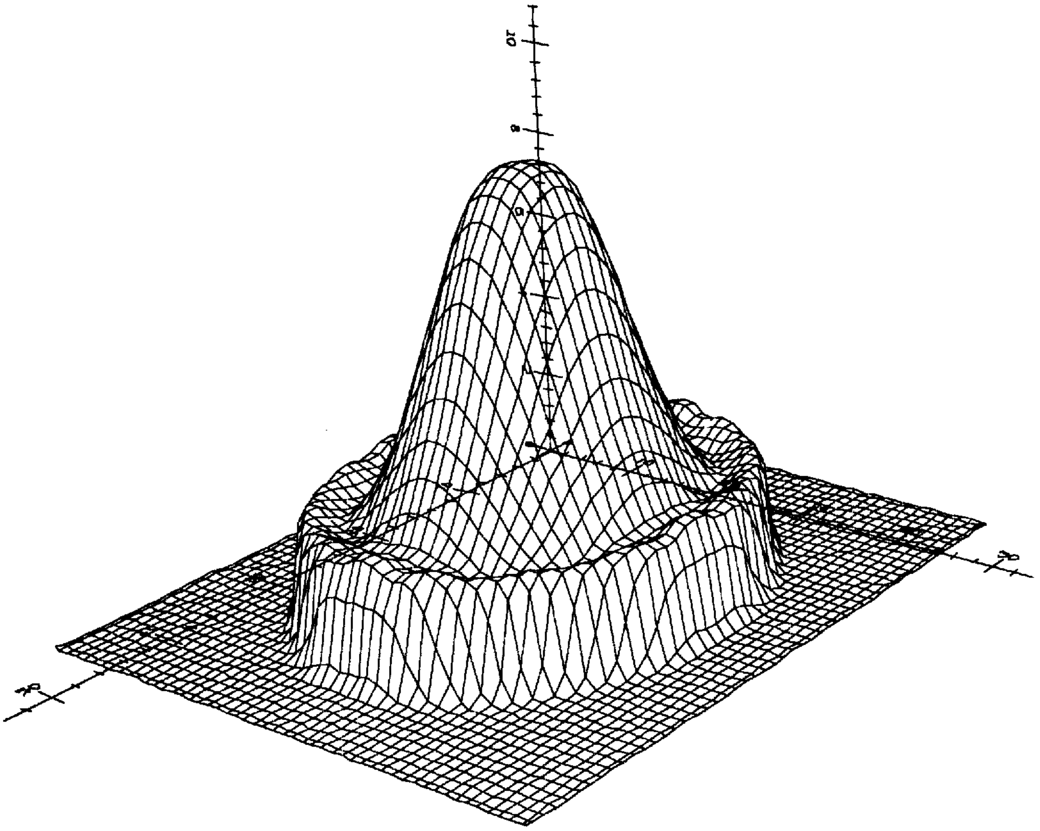


Figure 12. Water surface elevation calculated on the rectangular mesh.

the efficiency and stability of this method which contains no terms depending on adjustable parameters.

REFERENCES

1. Ch. Hirsch, *Numerical Computation of Internal and External Flows. Vol. 2: Computational Methods for Inviscid and Viscous Flows*, Wiley, Chichester, 1990.
2. B. van Leer, 'Towards the ultimate conservative difference scheme, III', *J. Comput. Phys.*, **23** (1977).
3. F. Alcrudo and P. García-Navarro, 'Flux difference splitting for 1D open channel flow equations', *Int. j. numer. methods fluids*, **14**, 1009–1018 (1992).
4. J. J. Barley, 'A survey of operator splitting applied to upwind differencing', *Numerical Analysis Report*, University of Reading, 1988.
5. R. J. Fennema and M. H. Chaudhry, 'Explicit methods for 2-D transient free-surface flows', *J. Hydraul. Eng. ASCE*, **116**, 1013–1034 (1990).
6. C. V. Bellòs, J. V. Soulis and J. G. Sakkas, 'Computation of two-dimensional dam-break-induced flows', *Adv. Water Resources*, **14**, 31–41 (1991).
7. N. Katopodes and T. Strelkoff, 'Computing two-dimensional dam-break flood waves', *J. Hydraul. Eng. ASCE*, **104**, 1269–1288 (1978).
8. A. Bento Franco and A. Betamio, 'Simulaçao uni e bidimensional de cheias provocadas por roturas de barragens em planicies de inundaçao', *V Simposio Luso-Brasileiro de Hidraulica e Recursos Hidricos SILUSB*, 1991.
9. R. Garcia and R. A. Kahawita, 'Numerical solution of the St. Venant equations with the MacCormack finite difference scheme', *Int. j. numer. methods fluids*, **6**, 259–274 (1986)
10. P. Lax and B. Wendroff, 'Systems of conservation laws', *Comm. on Pure and Applied Maths.*, **XIII**, 217–237 (1960).