INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2004; **46**:127–162 (DOI: 10.1002/fld.748)

Simulation of dam- and dyke-break hydrodynamics on dynamically adaptive quadtree grids

Q. Liang[1], A. G. L. Borthwick[1,*,†] and G. Stelling[2]

[1]*Department of Engineering Science, University of Oxford, Parks Road, Oxford OX1 3PJ, U.K.*
[2]*Department of Civil Engineering, Delft University of Technology, Delft, The Netherlands*

SUMMARY

Flooding due to the failure of a dam or dyke has potentially disastrous consequences. This paper presents a Godunov-type finite volume solver of the shallow water equations based on dynamically adaptive quadtree grids. The Harten, Lax and van Leer approximate Riemann solver with the Contact wave restored (HLLC) scheme is used to evaluate interface fluxes in both wet- and dry-bed applications. The numerical model is validated against results from alternative numerical models for idealized circular and rectangular dam breaks. Close agreement is achieved with experimental measurements from the CADAM dam break test and data from a laboratory dyke break undertaken at Delft University of Technology. Copyright © 2004 John Wiley & Sons, Ltd.

KEY WORDS: dam/dyke break; shallow water equations; Godunov-type scheme; HLLC approximate Riemann solver; MUSCL-Hancock method; adaptive quadtree grid

1. INTRODUCTION

World-wide, floods are likely to become increasingly severe and more frequent due to climate change, population growth, land-use, irrigation, deforestation, and urban development on flood plains. River flood defences usually consist of dams (which control river flows) and dykes (which act as lateral barriers). Numerical models for prediction of dam- and dyke-break hydrodynamics are essential analysis tools in the design of flood defence systems and for flood warning, control and management. However, accurate predictions of dam- and dyke-break flows pose a major challenge because of the steep free surface and velocity gradients that occur in such flows.

*Correspondence to: A. G. L. Borthwick, Department of Engineering Science, University of Oxford, Parks Road, Oxford, OX1 3PJ, U.K.
†E-mail: alistair.borthwick@eng.ox.ac.uk

Contract/grant sponsor: U.K. Engineering and Physical Sciences Research Council; contract/grant number: EPSRC grant GR/R13623

Published online 8 July 2004

Received 5 May 2003
Revised 5 April 2004

The first numerical dam-break models involved application of the method of characteristics to the 1D St Venant equations or the 2D shallow water equations (see e.g. References [1–3]). In non-conservative form, the characteristic equations correctly propagate information according to the physics of the flow field, provided the variables are smoothly variable (i.e. differentiable). However, for the discontinuous (shock-type) flows that occur in dam breaks, the governing equations must be in conservation form and shock-fitting approaches are required [4].

Stable non-oscillating high-resolution schemes for dam break simulation have been based on combinations of the *total-variation-diminishing* (TVD) concept with classical finite difference methods. García-Navarro *et al.* [5] and Aureli *et al.* [6] combined the MacCormack explicit scheme with the TVD technique. García-Navarro *et al.* [7] found that this scheme was superior to first-order Lax–Friedrich and second-order MacCormack methods when simulating flows with steep gradients. Rahman and Chaudhry [8] presented a 1D MacCormack finite difference scheme to simulate dam-break flow with grid adaptation. Although widely applied to dam-break simulations, the finite difference technique has the major drawback that it does not observe strict conservation of mass and momentum.

Weighted residual methods, such as the finite element method (FEM) and finite volume method (FVM), are increasingly used for simulating dam-break hydrodynamics in arbitrary-shaped flow domains. The methods utilize the integral form of the conservation laws, and thus properly conserve mass and momentum. However, the FEM (e.g. Reference [9]) can experience difficulty when both subcritical and supercritical flows are encountered [10], a feature of all dam-break flows. Moreover, FEM is computationally more expensive, so is less commonly used than FVM for simulating dam-break flows.

In applying the FVM method to solve dam-break problems, it has become commonplace to use Godunov-type shock-capturing schemes. Alcrudo and García-Navarro [11] developed a finite volume Godunov-type numerical model of wet-bed dam breaks. The inviscid 2D shallow water equations were solved on a fixed rectangular grid and a body-fitted co-ordinate system, with interface fluxes evaluated using the Roe-Pike [12] scheme. Zhao *et al.* [13] solved the inviscid 2D shallow water equations with Osher's [14] approximate Riemann solver on an unstructured grid. Anastasiou and Chan [15] presented a model for 2D shallow water equations based on a Godunov-type second-order finite volume method using Roe's [16] approximate solver on an unstructured mesh. Sleigh *et al.* [17] developed a finite volume numerical model for 2D river and estuary shallow flows with wet–dry boundary interfaces on dynamically adaptive triangular unstructured grids. Toro [4] gives a comprehensive review of recent developments in Godunov-type finite volume solvers for dam-break hydrodynamics.

In floods where there is a wet–dry interface it is important to model the contact wave properly [4], otherwise a spurious shock front may be predicted. To this end, Toro *et al.* [18] proposed a Harten, Lax and van Leer approximate Riemann solver with the contact wave restored, called the HLLC scheme. However, prediction of contact discontinuities and oblique shocks was not satisfactory due to the first-order accuracy of the scheme, and so methods were developed to extend the Godunov-type HLLC scheme to second-order accuracy. Toro [19] derived the weighted averaged flux (WAF) method for constructing second-order Godunov-type schemes, which has been successfully applied to dam-break simulation (see e.g. References [20, 21]). Second-order accuracy is also provided by monotone upwind schemes for conservation laws (MUSCL) [22, 23]. Hu *et al.* [24] and Causon *et al.* [25] have used the MUSCL scheme to predict dam-break flows.

This paper presents a Godunov-type finite volume solver that is based on dynamically adaptive quadtree grids and is applicable to both wet- and dry-bed dam/dyke-breaks. The model is extensively validated for benchmark idealized circular and rectangular dam breaks. It is then applied to a dam break into an open channel containing an elbow junction (CADAM test) and to a dyke break in a laboratory basin (Delft University of Technology study). Accurate results are obtained at high resolution due to the hierarchical nature of the quadtree grids, which are refined locally in regions of high flow gradient.

2. QUADTREE GRID

Quadtree grid generation is fast, automatic and robust. The procedure has been described in detail by Greaves and Borthwick [26] and Borthwick *et al.* [27], and so is summarized as follows:

(1) Scale the physical flow domain so that it fits within a unit square.
(2) Divide the unit square into four equal quadrant cells.
(3) Check each cell in turn, and subdivide if necessary according to specific subdivision criteria.
(4) Carry out further cell subdivision to ensure that no cell has a side length more than twice the size of its neighbours.

Two sets of subdivision criteria are used. The first relates to the discretized boundary geometry of the shallow flow domain, which is represented by a set of seeding points. Here, a cell is divided when it contains two or more seeding points and its subdivision level is less than the maximum specified. The second set of criteria relates to internal flow features, such as the root mean square values of the free surface gradients or the depth-averaged velocity component gradients. In dynamically adapting the grid, cells are added through further subdivision (enrichment) or removed (coarsened) by comparing the root mean square values to a look up table of prescribed maximum and minimum r.m.s. gradients.

Figure 1 shows an example regularized grid obtained about four seeding points located at (0.30, 0.16), (0.30, 0.16), (0.57, 0.91) and (0.91, 0.57), with the maximum subdivision level set to 4. During the grid generation process, a hierarchical data structure is simultaneously created as illustrated in Figure 2. The data structure comprises an integer tree of cell information where each cell has a parent pointer and four child pointers if they exist. A unique identification number consisting of a concatenation of local reference numbers is also given to each cell, which is determined during recursive division of the unit square. At any arbitrary level, the local reference numbers are defined so that 11, 21, 12, 22 denote the northwest, northeast, southwest and southeast quadrants. In order to give all cells identification numbers of the same length, trailing zeros are added to the end of identification numbers for cells that are at a lower level than the prescribed maximum subdivided level. The subdivision level is equal to the number of pairs of non-zero digits in the identification number. In Figure 1, cell 28's identification number is 12211200, and so its subdivision level is 3. Table I lists the cell information corresponding to the quadtree grid presented in Figure 1. In the neighbour finding algorithm, the identification number of the cell under consideration is manipulated to give the identification numbers of five outer hypothetical neighbours (including adjacent and

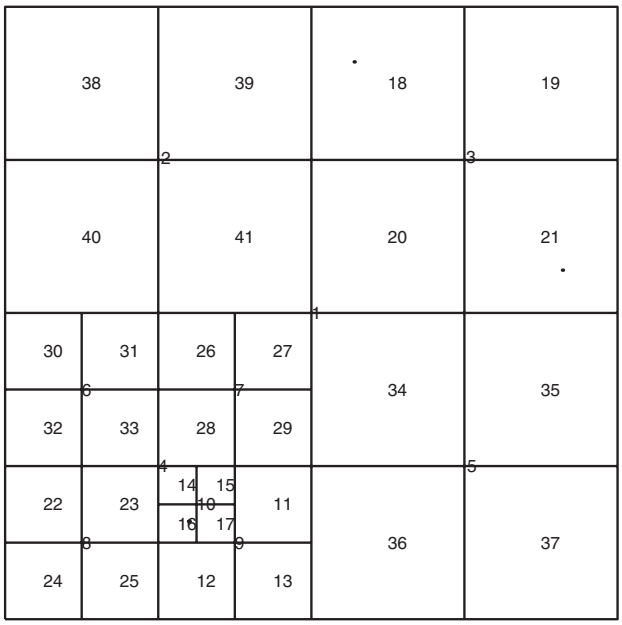


Figure 1. Regularized quadtree grid.

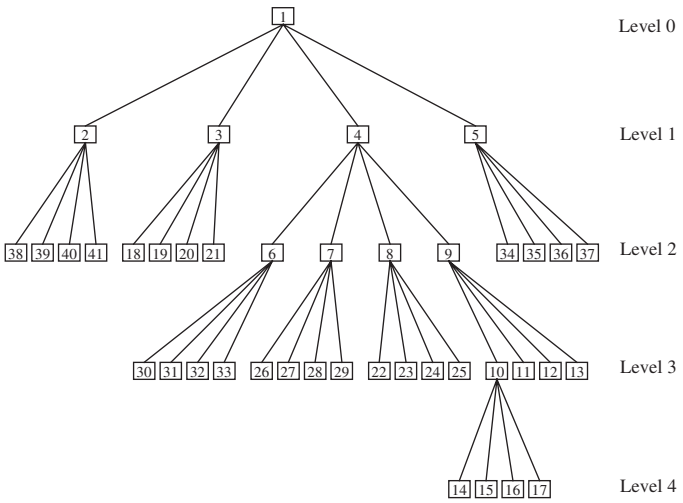


Figure 2. Hierarchical data structure.

corner neighbours) at the same subdivision level. The present method uses the fact that the three sister cells of a given leaf cell must exist and be neighbours. This is more efficient than the method of Greaves and Borthwick [26] which treated the sister cells as hypothetical neighbours. The nearest common ancestor (NCA) of the cell under consideration and each

Table I. Cell information for the quadtree grid in Figure 3.

Cell number	Reference number	Parent pointer	Child pointers			
1	00000000	0	2	3	4	5
2	11000000	1	38	39	40	41
3	21000000	1	18	19	20	21
4	12000000	1	6	7	8	9
5	22000000	1	34	35	36	37
6	12110000	4	30	31	32	33
7	12210000	4	26	27	28	29
8	12120000	4	22	23	24	25
9	12220000	4	10	11	12	13
10	12221100	9	14	15	16	17
11	12222100	9	—	—	—	—
12	12221200	9	—	—	—	—
13	12222200	9	—	—	—	—
14	12221111	10	—	—	—	—
15	12221121	10	—	—	—	—
16	12221112	10	—	—	—	—
17	12221122	10	—	—	—	—
18	21110000	3	—	—	—	—
19	21210000	3	—	—	—	—
20	21120000	3	—	—	—	—
21	21220000	3	—	—	—	—
22	12121100	8	—	—	—	—
23	12122100	8	—	—	—	—
24	12121200	8	—	—	—	—
25	12122200	8	—	—	—	—
26	12211100	7	—	—	—	—
27	12212100	7	—	—	—	—
28	12211200	7	—	—	—	—
29	12212200	7	—	—	—	—
30	12111100	6	—	—	—	—
31	12112100	6	—	—	—	—
32	12111200	6	—	—	—	—
33	12112200	6	—	—	—	—
34	22110000	5	—	—	—	—
35	22210000	5	—	—	—	—
36	22120000	5	—	—	—	—
37	22220000	5	—	—	—	—
38	11110000	2	—	—	—	—
39	11210000	2	—	—	—	—
40	11120000	2	—	—	—	—
41	11220000	2	—	—	—	—

hypothetical neighbour is determined, and the actual neighbour is then found by tracing the data tree downwards from the NCA until an undivided leaf cell is located. When regularizing the grid so that no cell has more than twice the linear dimension of its neighbour, the procedure is made most efficient by working outwards from those leaf cells of highest subdivision level so that the number of neighbouring cells to be checked is minimized.

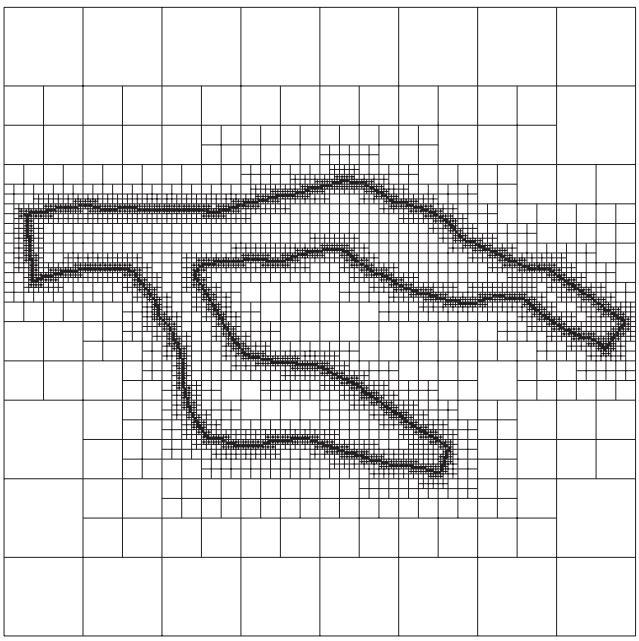


Figure 3. Quadtree grid for river bifurcation geometry.

Boundary identification is performed on the initial quadtree grid before regularization. Each cell is checked by drawing two lines starting from the cell centre in the positive and negative y-directions until reaching the edge of the unit square. For each of the two lines, the number of intersections between it and the boundary is recorded. If both numbers are odd, then the cell lies within the flow domain; otherwise it is outside the boundary. The same approach may also be applied in the x-direction. The boundary perimeter, which is described by seeding points, is approximated by linear interpolation. Each cell is given a flag to indicate whether it lies entirely outside the computational domain. Grid regularization is only performed on those cells that lie inside the boundary or lie on the boundary.

In order to validate the performance of the grid generator, it has been used to create a quadtree grid for the benchmark case of a river bifurcation geometry [27, 28]. A total of 3143 seeding points are used to define the discretized river geometry, which is normalized to fit within the unit square root cell. The maximum level of resolution is set to 9 and regularization is carried out over the entire unit square. The resulting quadtree grid depicted in Figure 3 has a total of 12 293 cells, of which 9220 are undivided leaf cells. The grid is identical to those obtained by Yiu *et al.* [28] and Borthwick *et al.* [27]. On a Pentium 800 MHz computer, the CPU time for the entire grid generation process is 1.8 s, and is significantly faster than the aforementioned methods. This is due to fewer hypothetical neighbours being considered for neighbour finding and the more efficient single sweep regularization algorithm, which commences from the highest level leaf cell. The example also demonstrates the ability of the quadtree grid to provide locally high resolution where the geometry is most complicated.

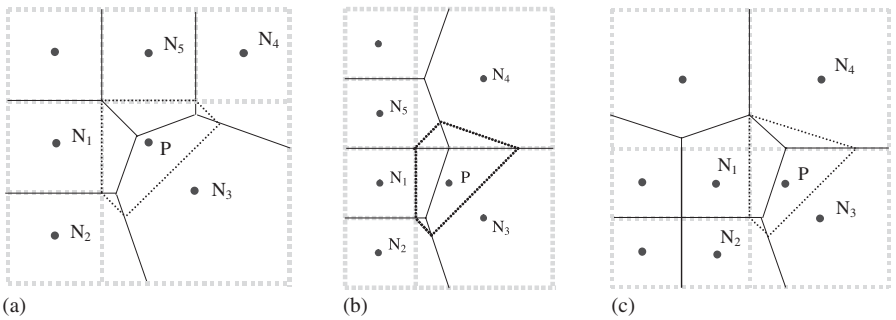


Figure 4. Natural neighbour interpolation for flux calculation.

Natural neighbour interpolation [29] is used to estimate fluxes at interfaces with different size adjacent grid cells, and in calculating new cell-centre values during grid adaptation. Natural neighbour interpolation is a weighted-average method that uses the immediate neighbouring data points to calculate the interpolant. The method involves using a Voronoi diagram to divide a planar area about a given set of data points. When an arbitrary point is inserted into the existing domain, a new cell and hence new Voronoi diagram are produced. The natural neighbours of the new cell are composed of all cells whose area has contributed to the formation of the new cell. The data value z for any inserted new point is expressed as

$$z = \sum_{i=1}^{m} \lambda_i z_i \tag{1}$$

in which m is the number of the natural neighbours, $\lambda_i = a_i/a$, where a_i and a denote the contributed area from the ith natural neighbour and the area of the new cell, respectively (λ_i is the so-called Sibson co-ordinate with respect to the ith natural neighbour), and z_i is the data value of the ith natural neighbour. The sum of the Sibson co-ordinates is obviously equal to 1. On a regularized quadtree grid (where there is at most one hanging node located at the middle of a cell interface) the number of grid configurations encountered during interpolation is limited to three primary patterns; all others are obtained by rotation. It is therefore more efficient to derive interpolation formulae for these cases than to use a general natural neighbour interpolation scheme. The interpolation scheme must ensure that fluxes are conserved. Figure 4 shows the three primary configurations for a regularized quadtree grid. N_1 is located at the centre of the cell of interest and N_3 at the centre of the larger adjacent cell. An interpolated value is required at point P (centred in the northwest quadrant of the undivided cell N_3). The original Voronoi areas are indicated by solid lines and the new tile (formed after point P has been inserted into the tessellation) is indicated by dashed lines. The contribution of each natural neighbour consists of the overlap area with that of the new tile. Calculation of the contribution area simply involves dividing a polygon into triangles and summing appropriately. The first case is illustrated in Figure 4(a), where the inserted point P has five natural neighbours represented by N_1, N_2, etc. The resulting Sibson co-ordinates with respect to the natural neighbours are

$$\lambda_1 = 23/96; \quad \lambda_2 = 1/48; \quad \lambda_3 = 23/48; \quad \lambda_4 = 1/48; \quad \lambda_5 = 23/96 \tag{2}$$

For the second case, in Figure 4(b)

$$\lambda_1 = 32/129; \quad \lambda_2 = 2/129; \quad \lambda_3 = 68/129; \quad \lambda_4 = 6/43; \quad \lambda_5 = 3/43 \tag{3}$$

For the third case, in Figure 4(c)

$$\lambda_1 = 22/69; \quad \lambda_2 = 1/69; \quad \lambda_3 = 34/69; \quad \lambda_4 = 4/23 \tag{4}$$

A similar approach is implemented when considering interpolation on enriched and coarsened grids during adaptation.

The natural neighbour interpolation is chosen herein due to its superior accuracy over alternative interpolation techniques [30]. Furthermore, the scheme is easy to implement on a quadtree grid because the hierarchical tree structure facilitates the identification of possible natural neighbours, while limiting the number of cells that need to be considered.

3. SHALLOW WATER EQUATIONS

In differential hyperbolic conservation form, the shallow water equations with the hydrostatic pressure assumption and omitting the viscous terms can be written as [31]

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{f}}{\partial x} + \frac{\partial \mathbf{g}}{\partial y} = \mathbf{s} \tag{5}$$

where $\mathbf{u}$ is the vector of conservative variables, $\mathbf{f}$ and $\mathbf{g}$ are flux vectors, and $\mathbf{s}$ is the vector of source terms. The vectors are given by

$$\mathbf{u} = \begin{bmatrix} \zeta \\ uh \\ vh \end{bmatrix}, \quad \mathbf{f} = \begin{bmatrix} uh \\ u^2 h + \frac{1}{2} g(\zeta^2 + 2\zeta h_s) \\ uvh \end{bmatrix}$$

$$\mathbf{g} = \begin{bmatrix} vh \\ uvh \\ v^2 h + \frac{1}{2} g(\zeta^2 + 2\zeta h_s) \end{bmatrix} \quad \text{and} \quad \mathbf{s} = \begin{bmatrix} 0 \\ (\tau_{wx} - \tau_{bx})/\rho - g\zeta S_{ox} + hfv \\ (\tau_{wy} - \tau_{by})/\rho - g\zeta S_{oy} - hfu \end{bmatrix}$$

where ζ is the water elevation above still water level (h_s); h ($=h_s+\zeta$) is the total water depth; u and v are depth-averaged velocity components in the x- and y-direction, respectively; g is acceleration due to gravity; f is Coriolis parameter; ρ is water density; τ_{bx} and τ_{by} are bed friction stresses; τ_{wx} and τ_{wy} are surface stresses; and S_{ox} ($=-\partial h_s/\partial x$) and S_{oy} ($=-\partial h_s/\partial y$) are bed slopes in the two Cartesian directions. Equation (5) ensures that the flux gradient and source terms are balanced mathematically, thus removing any need for additional numerical treatment when implementing an approximate solver in a Riemann scheme (see Reference [32] for details).

The bed stress terms represent the effect of bed roughness on the flow and may be estimated by using the following empirical formulae:

$$\tau_{bx} = \rho C_f u\sqrt{u^2 + v^2} \quad \text{and} \quad \tau_{by} = \rho C_f v\sqrt{u^2 + v^2} \tag{6}$$

The bed roughness coefficient C_f can be evaluated from the Chézy friction law, $C_f = g/C^2$, where C is the Chézy coefficient. Alternatively, C_f can be evaluated using $C_f = gn^2/h^{1/3}$, where n is the Manning coefficient.

With this set of balanced shallow water equations, the evaluation of source terms becomes trivial and easy. All the source terms can be evaluated within the grid, more precisely at the grid point, where the calculation is undertaken using the local values of the flow variables. In particular the bed slopes that are expressed in terms of the still water depth can be normally obtained analytically since the still water depth is related to the geometry of the problem domain that is known in advance. It is noted that the surface stresses and Coriolis effect in the source terms are neglected in all cases considered in this paper due to the relatively small computational domain for dam- and dyke-break problems.

The shallow water equations (5) form a system of conservation laws. Consider the flux Jacobian **A**, which is given by

$$\mathbf{A} = \begin{bmatrix} 0 & n_x & n_y \\ (c^2 - u^2)n_x - uvn_y & 2un_x + vn_y & un_y \\ -uvn_x + (c^2 - v^2)n_y & vn_x & un_x + 2vn_y \end{bmatrix} \tag{7}$$

where c $(= \sqrt{gh})$ is wave celerity, n_x and n_y are the Cartesian components of the unit vector $\mathbf{n}$ in the x- and y-directions, respectively. $\mathbf{n}$ is defined as $\mathbf{n} = \lfloor n_x \; n_y \rfloor^{\mathrm{T}}$ and $|\mathbf{n}| = \sqrt{n_x^2 + n_y^2} = 1$. The eigenvalues of the matrix **A** are

$$\lambda_1 = un_x + vn_y - c, \quad \lambda_2 = un_x + vn_y, \quad \lambda_3 = un_x + vn_y + c \tag{8}$$

which are all real and, provided $c > 0$, i.e. $h > 0$, they are also all distinct. This confirms the hyperbolicity of the foregoing 2D shallow water equations. The associated right and left eigenvector matrices are given by

$$\mathbf{R} = \begin{bmatrix} 1 & 0 & 1 \\ u - cn_x & n_y & u + cn_x \\ v - cn_y & -n_x & v + cn_y \end{bmatrix} \quad \text{and} \quad \mathbf{L} = \begin{bmatrix} \dfrac{un_x + vn_y}{2c} + \dfrac{1}{2} & -\dfrac{n_x}{2c} & -\dfrac{n_y}{2c} \\ -(un_y - vn_x) & n_y & -n_x \\ -\dfrac{un_x + vn_y}{2c} + \dfrac{1}{2} & \dfrac{n_x}{2c} & \dfrac{n_y}{2c} \end{bmatrix} \tag{9}$$

4. GODUNOV-TYPE NUMERICAL MODEL

4.1. HLLC approximate Riemann solver

Herein, the shallow water equations (5) are solved on quadtree grids, using a second-order Godunov-type scheme based on the HLLC approximate Riemann solver and MUSCL-Hancock method. The explicit updating conservative finite volume formula corresponding to Equation (5) is expressed as [4]

$$\mathbf{u}_{i,j}^{n+1} = \mathbf{u}_{i,j}^{n} + \frac{\Delta t}{\Delta x}(\mathbf{f}_{i-1/2,j} - \mathbf{f}_{i+1/2,j}) + \frac{\Delta t}{\Delta y}(\mathbf{g}_{i,j-1/2} - \mathbf{g}_{i,j+1/2}) + \Delta t \mathbf{s}_{i,j} \tag{10}$$

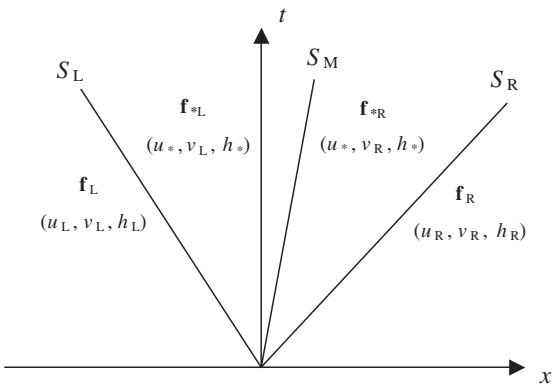


Figure 5. HLLC solution structure of the Riemann problem.

where subscripts i and j are the cell indices in x- and y-direction, respectively; superscript n denotes time level; Δt is the time step; Δx and Δy are the grid sizes along x- and y-direction, respectively; and $\mathbf{f}_{i-1/2,j}$ and $\mathbf{f}_{i+1/2,j}$ represent the fluxes through the left- and right-hand side cell interfaces, $\mathbf{g}_{i,j-1/2}$ and $\mathbf{g}_{i,j+1/2}$ are the fluxes through south and north cell interfaces, respectively. In the HLLC approximate Riemann solver (whose solution structure is depicted in Figure 5), the interface fluxes $\mathbf{f}_{i+1/2,j}$ are evaluated from

$$\mathbf{f}_{i+1/2,j} = \begin{cases} \mathbf{f}_L & \text{if } 0 \leqslant S_L \\ \mathbf{f}_{*L} & \text{if } S_L \leqslant 0 \leqslant S_M \\ \mathbf{f}_{*R} & \text{if } S_M \leqslant 0 \leqslant S_R \\ \mathbf{f}_R & \text{if } 0 \geqslant S_R \end{cases} \tag{11}$$

where $\mathbf{f}_L = \mathbf{f}(\mathbf{u}_L)$, $\mathbf{f}_R = \mathbf{f}(\mathbf{u}_R)$ with $\mathbf{u}_L$ and $\mathbf{u}_R$ representing the left and right Riemann states of a local Riemann problem with the subscripts L and R denoting left and right respectively, $\mathbf{f}_{*L}$ and $\mathbf{f}_{*R}$ are the numerical fluxes in the left and right parts of the middle region of the Riemann solution which is divided by a shear (contact) wave, and S_L, S_M and S_R are the speeds of the left, shear (contact) and right waves. In Equation (11), using subscripts 1 and 2 to denote the first and second components, respectively, of the flux vector $\mathbf{f}$, $\mathbf{f}_{*L}$ and $\mathbf{f}_{*R}$ are expressed as

$$\mathbf{f}_{*L} = \begin{bmatrix} f_{1*} \\ f_{2*} \\ f_{1*} v_L \end{bmatrix}, \quad \mathbf{f}_{*R} = \begin{bmatrix} f_{1*} \\ f_{2*} \\ f_{1*} v_R \end{bmatrix} \tag{12}$$

where v_L and v_R are the left and right initial values of a local Riemann problem for the tangential velocity component (Figure 5), and f_{1*} and f_{2*} are calculated using the HLL formula given by Harten *et al.* [33],

$$\mathbf{f}_* = \frac{S_R \mathbf{f}_L - S_L \mathbf{f}_R + S_L S_R (\mathbf{u}_R - \mathbf{u}_L)}{S_R - S_L} \tag{13}$$

In order to apply the HLLC scheme properly to the shallow water equations, it is important to identify the correct wave speeds, S_L, S_M and S_R. Fraccarollo and Toro [20] recommend the use of a two-rarefaction approximate Riemann solver for S_L and S_R. By simultaneously considering the dry-bed problem, S_L and S_R are given by

$$S_L = \begin{cases} u_R - 2\sqrt{gh_R} & \text{if } h_L = 0 \\ \min(u_L - \sqrt{gh_L}, u_* - \sqrt{gh_*}) & \text{if } h_L > 0 \end{cases}$$

and

$$S_R = \begin{cases} u_L + 2\sqrt{gh_L} & \text{if } h_R = 0 \\ \max(u_R + \sqrt{gh_R}, u_* + \sqrt{gh_*}) & \text{if } h_R > 0 \end{cases} \tag{14}$$

where u_L, u_R, h_L and h_R are left and right initial values for a local Riemann problem, and u_* and h_* are calculated from

$$u_* = \tfrac{1}{2}(u_L + u_R) + \sqrt{gh_L} - \sqrt{gh_R} \tag{15}$$

and

$$h_* = \frac{1}{g}\left[\frac{1}{2}(\sqrt{gh_L} + \sqrt{gh_R}) + \frac{1}{4}(u_L - u_R)\right]^2 \tag{16}$$

As to the middle wave speed S_M, Toro [4] suggests that the following choice is most suitable for dry-bed problems:

$$S_M = \frac{S_L h_R(u_R - S_R) - S_R h_L(u_L - S_L)}{h_R(u_R - S_R) - h_L(u_L - S_L)} \tag{17}$$

The above scheme is used for computing the interface flux in the x-direction. The method is extended to the y-direction in a similar way.

4.2. *Unsplit MUSCL-Hancock method*

When directly implementing the HLLC approximate Riemann solver to evaluate interface fluxes in the context of a Godonov-type method, the scheme is merely first-order accurate. Herein, an unsplit MUSCL-Hancock method is used to construct a second-order accurate Godunov-type scheme by updating the conservative variables using a predictor step and a corrector step. The predictor step computes a temporary cell-centre value over the half time interval $\Delta t/2$

$$\bar{\mathbf{u}}_{i,j}^{n+1/2} = \mathbf{u}_{i,j}^{n} + \frac{\Delta t}{2\Delta x}(\mathbf{f}(\mathbf{u}_{i-1/2,j}) - \mathbf{f}(\mathbf{u}_{i+1/2,j})) + \frac{\Delta t}{2\Delta y}(\mathbf{g}(\mathbf{u}_{i,j-1/2}) - \mathbf{g}(\mathbf{u}_{i,j+1/2})) + \frac{\Delta t}{2}\mathbf{s}_{i,j} \tag{18}$$

where the flux vectors $\mathbf{f}(\mathbf{u}_{i-1/2,j})$, $\mathbf{f}(\mathbf{u}_{i+1/2,j})$, $\mathbf{g}(\mathbf{u}_{i,j-1/2})$ and $\mathbf{g}(\mathbf{u}_{i,j+1/2})$ are evaluated at mid-points of cell faces by linear interpolation of cell-centre values, via

$$\mathbf{u}(x, y) = \mathbf{u}_{i,j} + \mathbf{r} \cdot \nabla \mathbf{u}_{i,j} \tag{19}$$

in which (x, y) represents a point at the midpoint of a cell face of an arbitrary cell; $\mathbf{r}$ is the vector from the cell centre to point (x, y) with the right being defined as the positive direction,

and $\nabla \mathbf{u}_{i,j}$ is the gradient vector. The predictor step does not require Riemann solutions and the interface fluxes are evaluated within each cell using interpolated values at the extremities of the inner interfaces. In practice, the flow states are usually different at either side of an interface and so the fluxes are distinct at the inner and outer interfaces. Therefore in the predictor step, the use of inner fluxes to represent the overall fluxes through the cell face is non-conservative. However, this step does not really affect the conservative character of the overall method as the step is only intermediate, and the ultimate interface fluxes are calculated using the HLLC approximate Riemann solver in the corrector step over a full time step, which is fully conservative [34].

The corrector step involves employing Equation (10) to update the conservative variables over a time step. The left and right Riemann states at cell interface $i+1/2$ are evaluated from

$$\mathbf{u}^{\mathrm{L}}_{i+1/2} = \bar{\mathbf{u}}^{\mathrm{n}+1/2}_{i,j} + \mathbf{r} \cdot \nabla \mathbf{u}^{\mathrm{n}}_{i,j} \quad \text{and} \quad \mathbf{u}^{\mathrm{R}}_{i+1/2} = \bar{\mathbf{u}}^{\mathrm{n}+1/2}_{i+1,j} + \mathbf{r} \cdot \nabla \mathbf{u}^{\mathrm{n}}_{i+1,j} \tag{20}$$

where $\bar{\mathbf{u}}^{\mathrm{n}+1/2}_{i,j}$ and $\bar{\mathbf{u}}^{\mathrm{n}+1/2}_{i+1,j}$ are the predicted values at cells (i,j) and $(i+1,j)$ calculated by Equation (18); $\mathbf{r}$ is the normal distance vector from the cell centroids to the midpoint of cell face $i+1/2$ with right being the positive direction; $\nabla \mathbf{u}^{\mathrm{n}}_{i,j}$ and $\nabla \mathbf{u}^{\mathrm{n}}_{i+1,j}$ are gradient vectors. Hu *et al.* [35] suggest that the use of $\nabla \mathbf{u}^{\mathrm{n}}_{i,j}$ and $\nabla \mathbf{u}^{\mathrm{n}}_{i+1,j}$ to construct the Riemann states in the corrector step of the MUSCL-Hancock scheme gives better results than the use of $\nabla \bar{\mathbf{u}}^{\mathrm{n}+1/2}_{i,j}$ and $\nabla \bar{\mathbf{u}}^{\mathrm{n}+1/2}_{i+1,j}$. The Riemann states at the other three cell faces are calculated in a similar way.

4.3. Slope limiter

A slope limiter is used to damp numerical oscillations that would otherwise result from the second-order Godunov-type solver when applied to discontinuous flows. When evaluating the values of the conservation variables at the cell faces, Equations (19) and (20) are replaced by

$$\mathbf{u}_{i+1/2,j} = \mathbf{u}_{i,j} + \tfrac{1}{2}\,\Psi(r) \cdot \Delta \mathbf{u}_{\mathrm{loc}} \quad \text{and} \quad \mathbf{u}_{i-1/2,j} = \mathbf{u}_{i,j} - \tfrac{1}{2}\,\Psi(r) \cdot \Delta \mathbf{u}_{\mathrm{loc}} \tag{21}$$

in which $\Delta \mathbf{u}_{\mathrm{loc}} = \mathbf{u}_{i,j} - \mathbf{u}_{i-1,j}$, $\Psi(r)$ is a slope limiter which is a function of r, and r is the ratio of successive gradients given by

$$r = \Delta \mathbf{u}_{upw} / \Delta \mathbf{u}_{\mathrm{loc}} \tag{22}$$

where $\Delta \mathbf{u}_{upw} = \mathbf{u}_{i+1,j} - \mathbf{u}_{i,j}$.

There are several choices for the slope limiter. In the present work, a minmod limiter is used because numerical experiments show that the use of minmod limiter can produce more stable results. The minmod limiter is defined as

$$\Psi(r) = \max[0, \min(r, 1)] \tag{23}$$

4.4. Stability criterion

The foregoing numerical scheme is explicit, and its stability is governed by the Courant–Friedrichs–Lewy (CFL) criterion. For a 2D Cartesian grid, the CFL criterion for choosing an

appropriate time step, Δt, may be expressed

$$\Delta t = C \cdot \min\left[\frac{\Delta x_i}{|u_i| + \sqrt{gh_i}}, \frac{\Delta y_i}{|v_i| + \sqrt{gh_i}}\right] \tag{24}$$

where C is the Courant number specified in the range $0 < C \leqslant 1$, Δx_i and Δy_i are the dimensions of cell i in the x- and y-directions, u_i and v_i are the cell-centre depth-averaged velocity components, and h_i is the cell-centre water depth. In the present scheme, the time step is predicted in the previous iteration using Equation (24) and the Courant number C is set to 0.8 for all test cases to ensure stability.

4.5. Boundary conditions

Two types of simple boundary conditions are used herein, i.e. transmissive open and reflective slip closed boundary conditions [4]. In the context of a quadtree grid, the boundary conditions are imposed as follows. At open (transmissive) boundaries

$$\begin{aligned} h_0 = h_1, \quad u_0 = u_1, \quad v_0 = v_1 \\ h_{m+1} = h_m, \quad u_{m+1} = u_m, \quad v_{m+1} = v_m \end{aligned} \tag{25}$$

At solid (slip) boundaries

$$\begin{aligned} h_0 = h_1, \quad u_0 = -u_1, \quad v_0 = v_1 \\ h_{m+1} = h_m, \quad u_{m+1} = -u_m, \quad v_{m+1} = v_m \end{aligned} \tag{26}$$

Here, u is the velocity normal to the wall; v denotes velocity component tangential to the wall; 0 and $m + 1$ are fictitious cells outside the computational domain.

5. RESULTS

In all cases, the grid adaptation criteria are based on the root mean square values of the free surface gradient [31],

$$\Theta = \sqrt{\left(\frac{\partial \zeta}{\partial x}\right)^2 + \left(\frac{\partial \zeta}{\partial y}\right)^2} \tag{27}$$

The value of Θ is prescribed and adjusted numerically by trial and error in each case.

5.1. Test of equilibrium state of a flow with non-regular bed-slope

In order to validate the performance of the present flux-gradient and source term balanced shallow water equations and its ability to deal with source terms, test cases on a circular shallow basin with non-regular bed profiles are considered herein. The circular basin considered herein has an axially symmetric bathymetry with the still water depth being a function of radial distance from the basin's centre. The bathymetry has an x-direction cross-section as shown in

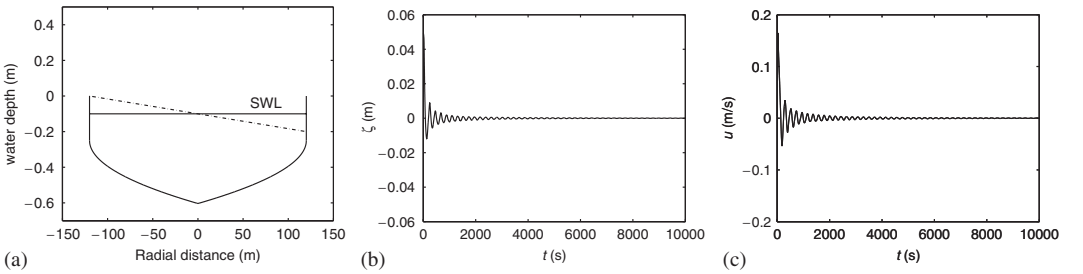


Figure 6. Equilibrium state test: (a) cross-section of the circular basin; (b) temporal history of u; and (c) temporal history of ζ.

Figure 6(a) with still water depth h_s being determined by the following formula:

$$h_s = H\left(\frac{1}{2} + \left(\frac{1}{2} - \frac{1}{2}\frac{r}{R_0}\right)^{1/2}\right) - 0.1 \tag{28}$$

where $H = 0.5\,\mathrm{m}$ is a weighted mean water depth, r is the distance from the basin centre, and $R_0 = 120\,\mathrm{m}$ is the radius of the circular basin. The still water surface is indicated in Figure 6(a) by the horizontal solid line. The bed roughness coefficient for calculating bed stress terms is set to $C_f = 0.08$ in order to allow the flow converge back to equilibrium state quickly after perturbation. Numerical computations are performed on a fixed quadtree grid with highest and lowest subdivision level of 8 and 6, respectively. The grid has greatest subdivision on the boundary with totally 5093 cells of which 2796 are leaf cells.

Firstly, the computations are carried out with the water surface right at the equilibrium state. The model is run for 10 000 s and, as expected, the flows are at rest throughout the whole numerical processes. Then perturbation is added into the system by inclining the water surface in x-direction as indicated by the dash-lines in Figure 6(a). Oscillating flows are then produced and damped out after some time due to the bed frictions. The temporal changing histories of free surface elevation and x-direction velocity component u are presented in Figures 6(b) and 6(c), and no velocity is induced in y-direction.

5.2. *Circular dam break*

Toro [4] proposed a circular dam-break test case involving a water cylinder of radius $R = 2.5\,\mathrm{m}$ in the middle of a $40\,\mathrm{m} \times 40\,\mathrm{m}$ square computational domain, which is horizontal and assumed to be frictionless. The water cylinder is 2.5 m in height and is enclosed by an infinitely thin circular wall which is removed instantaneously at $t = 0\,\mathrm{s}$. The water depth outside the dam is set to 0.5 m or dry. Initially, the water throughout the domain is at rest. Transmissive lateral wall boundary conditions are imposed. The initial quadtree grid has maximum and minimum division levels of 9 and 5 with the finest subdivision in the area where the initial water cylinder is located. The initial grid consists of a total of 6421 cells of which 4816 are leaf cells. The minimum cell width is 0.078125 m.

Firstly, the wet-bed case is considered. During the simulation grid enrichment occurs up to level 9 when $\Theta > 0.5$, and grid coarsening is implemented when all four child cells of a given cell have $\Theta < 0.4$ and their subdivision level is greater than 5. Figure 7 shows the free surface

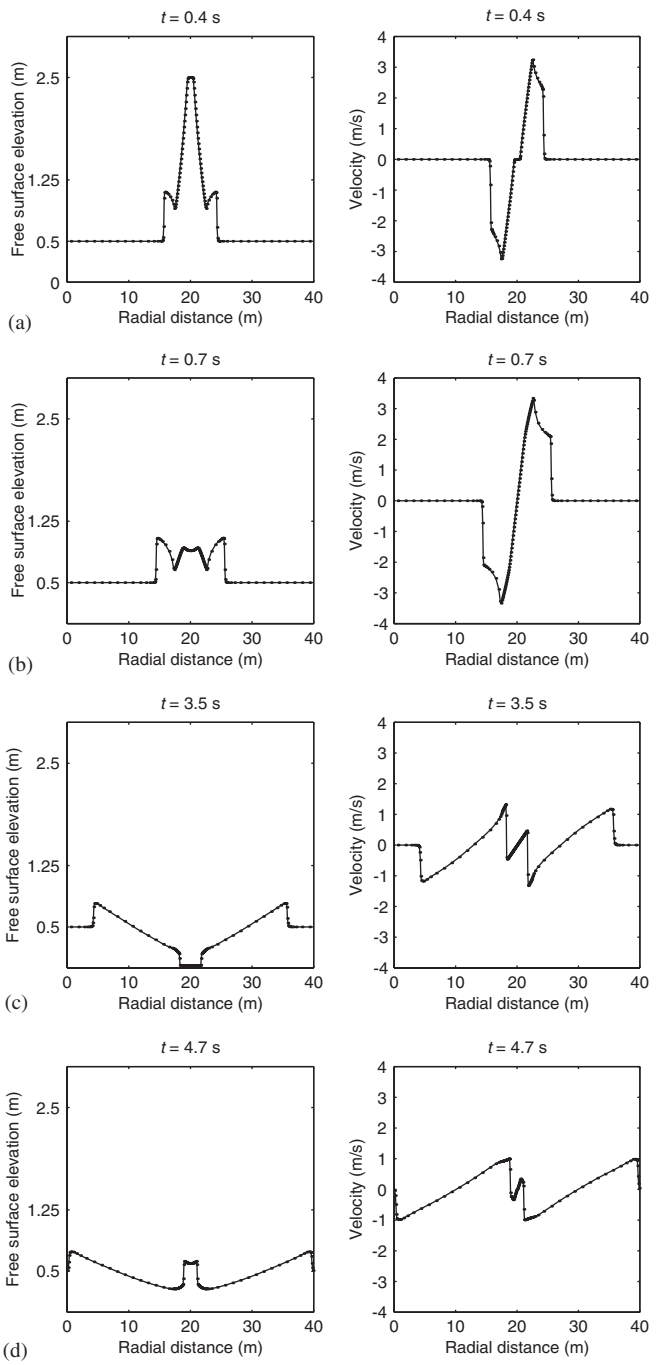


Figure 7. Wet-bed circular dam break. Left: radial profile of free surface; and right: radial profile of velocity.

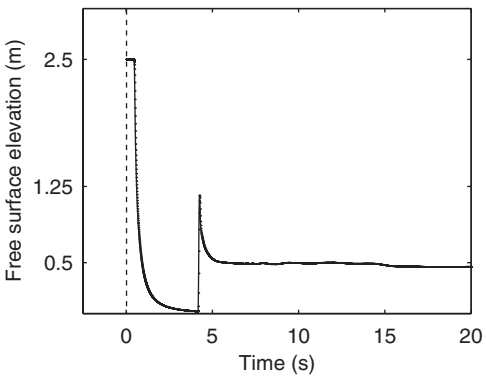


Figure 8. Wet-bed circular dam break: temporal variation in surface elevation at the centre of the domain.

elevation and particle velocity along a radial line crossing the centre of the initial dam-break cylinder at different output times. At $t = 0.4$ s, it is observed that a primary shock wave is propagating radially away from the centre. There is also a rarefaction moving radially towards the centre, and about to reach the centre point of the original water cylinder. At $t = 0.7$ s, the rarefaction wave has reflected at the centre, and has begun to travel outward causing a dip that can be seen in the free surface. The reflected rarefaction wave then continues to move radially away from the centre, the water level drops almost to the bed and a secondary inward moving circular shock wave forms by time $t = 3.5$ s. The secondary shock is due to the reflected rarefaction wave over-expanding the flow [4]. At $t = 4.7$ s, the primary shock wave has almost reached the boundary, the secondary shock wave has reflected at the centre and begun also to travel radially outward, and a slight dip at the centre has appeared. From the figures, we can see that the quadtree grid (represented by the symbols at cell-centres) evolves with the dam-break. Figure 8 shows the temporal variation in surface elevation at the centre of the domain, and is in close agreement with the result obtained by Toro [4], who used an inhomogeneous 1D version of the shallow water equations. Toro used an extremely fine mesh, and so his prediction may be treated as almost corresponding to an analytical solution.

Next, consider the dry-bed case. In order to better capture the wet–dry front, the trigger value for Θ is differently prescribed so that grid enrichment occurs up to level 9 when $\Theta > 0.02$, and grid coarsening is implemented when $\Theta < 0.008$. The plots for free surface elevation and particle velocity along a radial line crossing the centre of the initial dam-break cylinder are presented in Figure 9 for different output times. Figure 10 shows the temporal variation in surface elevation at the centre of the domain which is significantly different from that for wet-bed case. It is evident from the results that no shock wave forms at the front. And the front velocity is about four times bigger than that of wet-bed case. These findings confirm Toro's [4] conclusion that dry and wet-bed problems have solutions that are distinctly different.

In order to examine the performance of the dynamically adaptive quadtree grid based scheme, the wet-bed case has also been simulated on fixed uniform quadtree grids of subdivision level 7 and 8, respectively. As can be seen in Figure 11, the dip in the free surface at $t = 0.7$ s is properly simulated on the adaptive and level 8 uniform quadtree grids. This is not

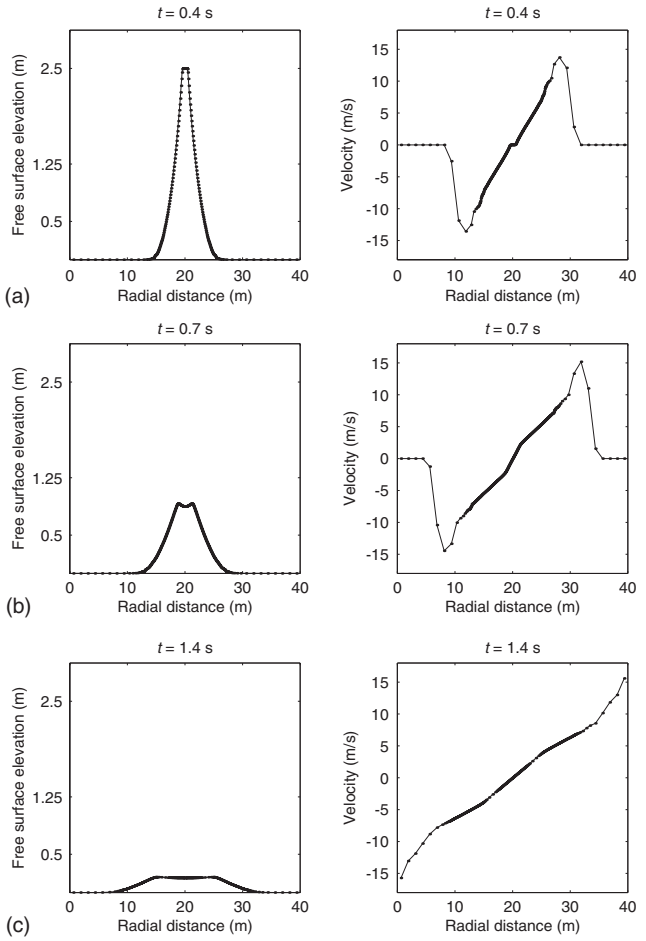


Figure 9. Dry-bed circular dam break. Left: radial profile of free surface; and right: radial profile of velocity.

the case for level 7 uniform quadtree grid, which gives inaccurate results at the dip. The CPU times are 104, 715 and 583 s for the uniform level 7, uniform level 8 and adaptive quadtree grids, respectively, on a Pentium 800 MHz with 256 MB RAM personal computer. Therefore by using a dynamically adaptive quadtree grid, the numerical model efficiently achieves high resolution in cases where there are locally high gradients.

5.3. Frictionless rectangular dam break

The rectangular dam-break takes place in a $200\,\mathrm{m} \times 200\,\mathrm{m}$ domain with a flat horizontal frictionless bed. Initially, the square basin is divided into two halves by an infinitesimally thin dam. The still water depth in the left (upstream) half is 10 m while that in the right (downstream) half is set to be either 5 m or dry. At $t = 0$, a section of the thin dam breaks abruptly, as indicated in Figure 12. For both the wet and dry bed dam breaks, all the frictionless solid

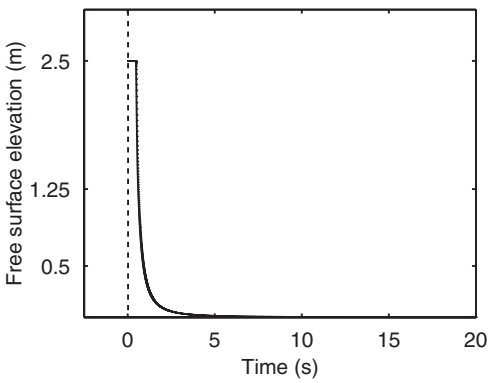


Figure 10. Wet-bed circular dam break: temporal variation in surface elevation at the centre of the domain.

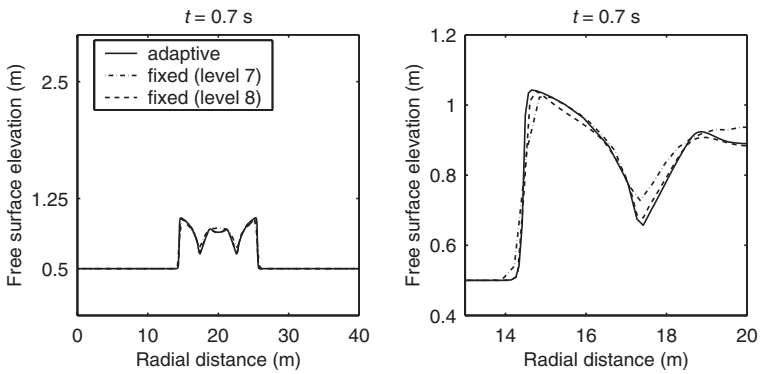


Figure 11. Predicted surface elevation for wet-bed circular dam break using different grids.

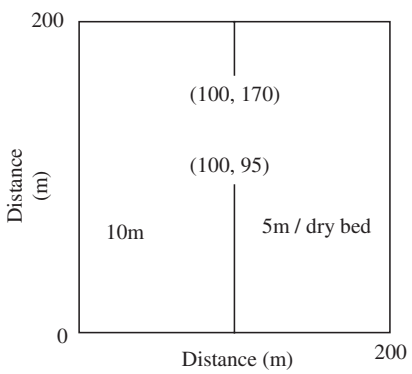


Figure 12. Frictionless rectangular dam break. Definition sketch.

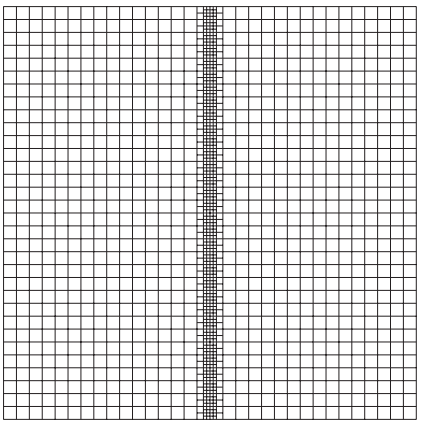
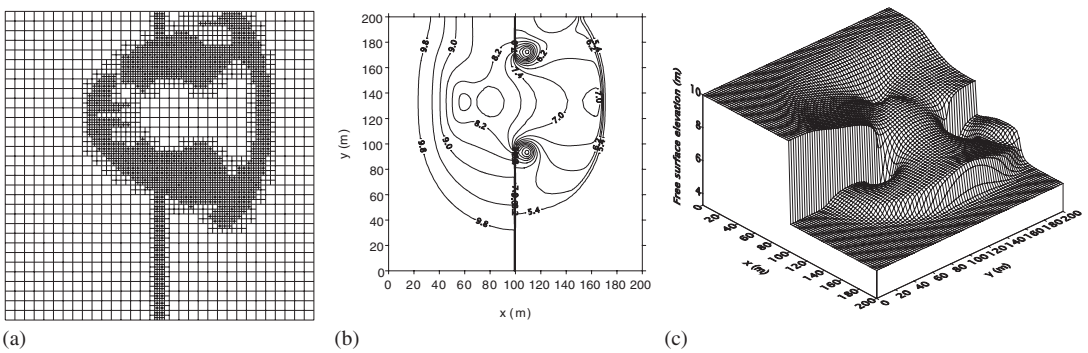


Figure 13. Frictionless rectangular dam break. Initial quadtree grid.



(a) (b) (c)

Figure 14. Frictionless rectangular dam break. Wet-bed results at $t = 7.2$ s: (a) adapted grid; (b) water depth contours; and (c) 3D representation of free surface.

walls are treated as reflective boundaries. Figure 13 shows the initial quadtree grid with finest grid level set to 7 and coarsest to 5. The grid has a total of 2133 cells of which 1600 are leaf cells.

In the wet-bed simulation, grid enrichment is undertaken when $\Theta > 0.1$, and coarsening when $\Theta < 0.02$. Figure 14 presents the dynamically adapted quadtree grid, water depth contours and a 3D projection of the water surface at time $t = 7.2$ s after the dam has breached. A nearly semi-circular shock wave can be seen propagating into the downstream half of the basin, as water pours through the opening of the dam. Meanwhile, a rarefaction wave propagates into the upstream half of the basin. Vortices occur immediately downstream of the sharp-edges of the breach. The results agree closely with those obtained by many other researchers, including Alcrudo and Garcia-Navarro [11], Anastasiou and Chan [15], Mingham and Causon [36], Fujihara and Borthwick [37], and Rogers *et al.* [31]. By using a dynamically adaptive quadtree grid, high resolution is automatically obtained in those regions where the free surface gradient is steep, such as the shock front and the two vortices.

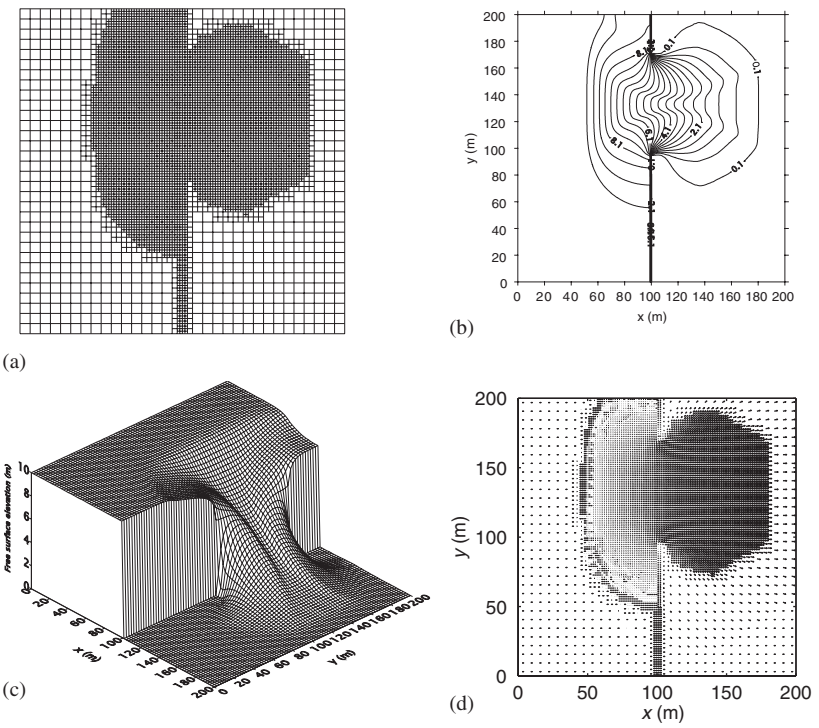


Figure 15. Frictionless rectangular dam break. Dry-bed results at $t = 5$ s: (a) adapted grid; (b) water depth contours; (c) 3D representation of free surface; and (d) velocity field.

In the dry-bed simulation, different criteria are applied for the grid adaptation in order to capture the wet–dry front accurately. Grid enrichment is undertaken when $\Theta > 0.02$, and coarsening when $\Theta < 0.005$. Figure 15 gives the results at $t = 5$ s. Figure 16 shows the equivalent results, for a wet downstream bed where the initial water depth was set to 0.01 m. No shock wave front occurs in the dry bed simulation, unlike the wet-bed case. As the flood develops, the wet- and dry-bed velocity fields are significantly different, as can be seen in Figures 15(d) and 16(d). This confirms Toro's [4] view that it is incorrect in principle to use finite water depths to approximate the dry-bed problem. However, the results from wet-bed cases are expected to be asymptotic to those of dry-bed problems when the initial downstream water depth is small enough.

5.4. Dam-break flow into an L-shaped channel

In 1997, the CADAM group carried out laboratory dam break experiments using a rectangular upstream reservoir connected to an L-shaped open channel, as sketched in Figure 17. The channel bed elevation is 0.33 m above that of the reservoir bed so that there is a vertical step at the entrance of the channel. Initially the total still water depth in the reservoir is 0.53 m, and the water depth in the channel is either set to 0.01 m (wet-bed) or zero (dry-bed). The reservoir and the channel are separated by a gate, which is opened suddenly to produce dam-break flow.

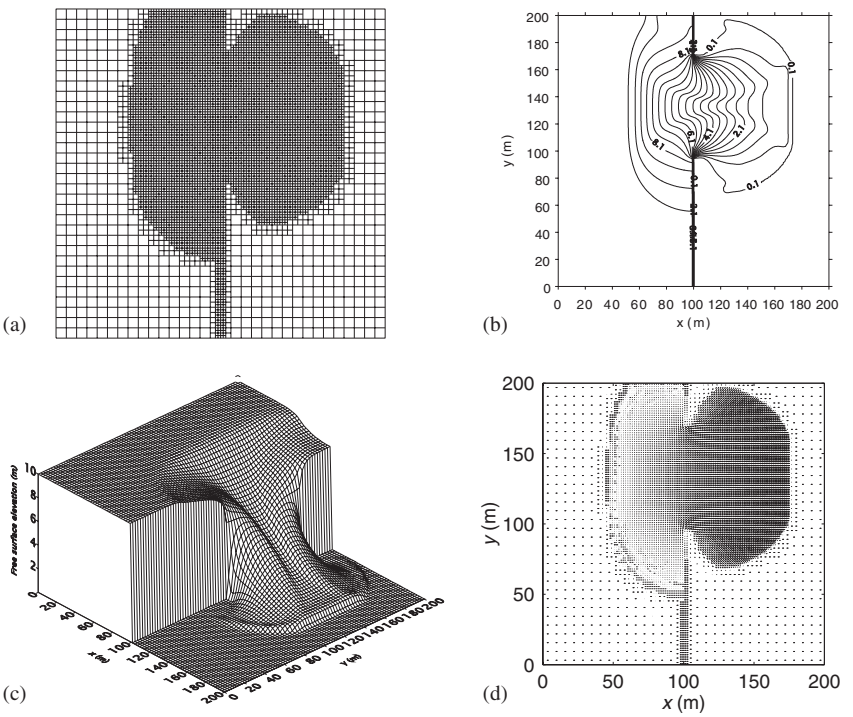


Figure 16. Frictionless rectangular dam break. Slightly wet-bed results at $t = 5$ s: (a) adapted grid; (b) water depth contours; (c) 3D representation of free surface; and (d) velocity field.

At the channel entrance, the local head loss h_ξ due to a sudden contraction (at cell interface $i + 1/2$ where the contraction occurs) is

$$h_\xi = \xi \frac{u_{i+1}^2}{2g} \tag{29}$$

where ξ is an empirical coefficient and u_{i+1} is the flow velocity in cell $i + 1$. The local head loss effect is incorporated in the shallow water equations as a resistance stress [38], with components

$$\tau_{\xi x} = \rho \xi u_{i+1} \sqrt{u_{i+1}^2 + v_{i+1}^2}, \quad \tau_{\xi y} = \rho \xi v_{i+1} \sqrt{u_{i+1}^2 + v_{i+1}^2} \tag{30}$$

which are simply added to the source terms in the momentum equations. Herein, the value of ξ is set to 0.8. Numerical experiments show that the results are not sensitive to the value of ξ, perhaps because the vertical step from the reservoir to the channel is approximated by a very steep slope $S_{ox} \approx 10.75$, which is equal to the ratio between the step height and the minimum grid size.

Figure 18 depicts the initial quadtree grid. The maximum and minimum grid levels of resolution are 8 and 6, respectively. There are a total of 2523 cells of which 1916 are leaf.

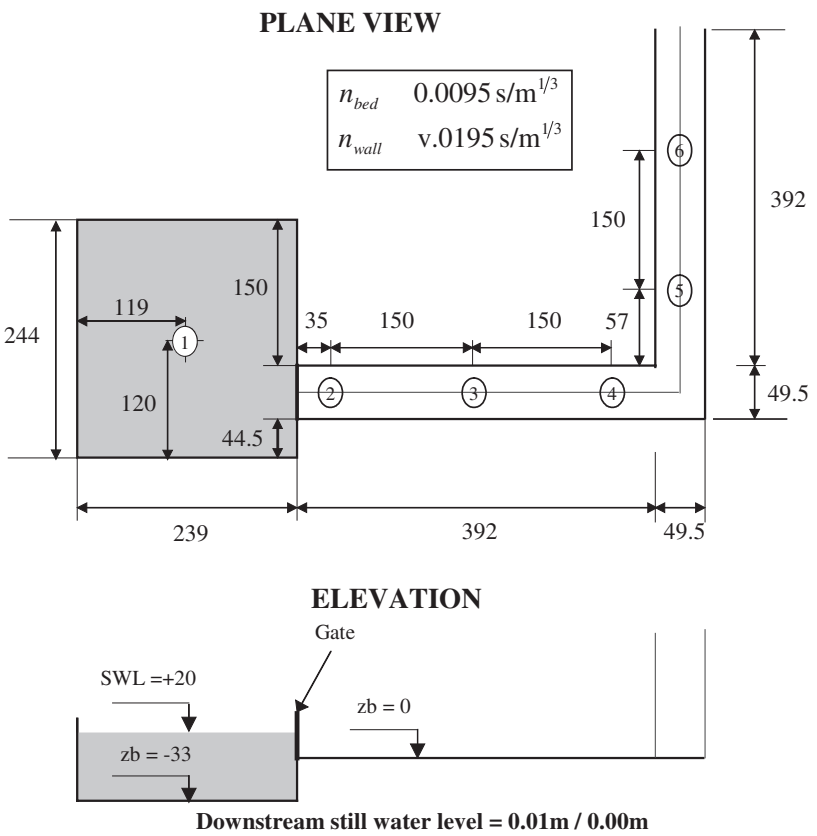


Figure 17. CADAM dam break. Laboratory test layout (all dimensions are in cm).

Grid enrichment takes place when $\Theta > 0.1$ and coarsening when $\Theta < 0.05$. According to Soares Frazão *et al.* [39], the bottom and wall friction Manning coefficients are set to 0.0095 $sm^{-1/3}$ and 0.0195 $sm^{-1/3}$, respectively. Figures 19 and 20 present predicted and measured water surface levels with time at different gauging points for the wet- and dry-bed cases, respectively. In both wet- and dry-bed cases, the results at Gauge 1 depict the initially steep fall in free surface elevation in the header tank from 0 to 15 s after the initial dam break, followed by a more gradual decrease in free surface elevation from 20 to 40 s, and confirm that the numerical tank is emptying in a physically correct manner. The predicted and measured primary step-like changes in water level as the bore and its reflection from the elbow arrive at Gauges 2–4 are in close agreement, indicating that the propagation speeds have been properly modelled. The theoretical and experimental results are reasonably similar at Gauges 5 and 6, located beyond the elbow, for both the wet- and dry-bed cases. At Gauge 2, the predicted water elevations are higher than the measured values after the bore has arrived, and slightly lower afterwards. These discrepancies are also evident in results presented by Soares Frazão *et al.* [39] who used a Boltzmann scheme to predict the dry-bed dam break hydrodynamics.

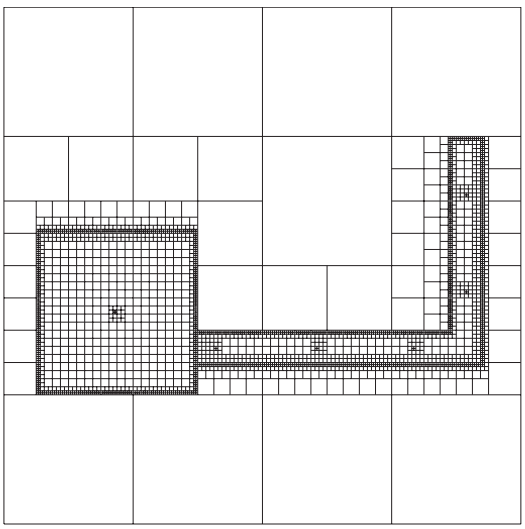


Figure 18. CADAM dam break. Initial quadtree grid.

5.5. *Dyke break in a laboratory-scale rectangular basin*

A dyke break flood onto a flat, horizontal basin has been undertaken in the Fluid Mechanics Laboratory of Delft University of Technology, the Netherlands. Figure 21 illustrates the experimental layout, which consists of a closed reservoir containing water of initial depth equal to 0.6 m separated by a solid dyke wall from a flat basin. The basin was initially dry or contained still water to a low depth (0.05 m for the case considered here). A gate opening of 40 cm represented the breach, and was located in the middle of the dyke. The dyke-break was modelled by lifting the gate at a constant speed of about 16 cm/s. According to Duinmeijer [40], the basin has a smooth concrete bed corresponding to a Manning roughness coefficient of $0.012\,\mathrm{ms}^{-1/3}$. Reflective boundary conditions are imposed at all solid walls.

Figure 22 depicts the initial quadtree grid representing the problem domain. The grid has maximum and minimum grid levels of 9 and 6, respectively, and consists of 6197 grid cells in total, of which 3011 are leaf cells. Grid enrichment is undertaken when $\Theta > 0.01$, and coarsening when $\Theta < 0.005$.

It should be noted that because the gate is lifted up slowly during the experiment, the initial stages of the computational procedure are modified at the gate flow cells before the gate is fully opened. When the vertical gate opening distance h_w is less than the water level either side of the gate, the gate is treated as a solid boundary. In this case, the water depth at the gate is set to h_w and the discharge per unit width Q calculated using the formula for submerged culvert flow [41]

$$Q = C_h h_w \sqrt{2g(H - C_h h_w)} \tag{31}$$

where H is headpond water level and is set to 0.6 m here, and C_h is the contraction coefficient representing the combined effect of vertical and horizontal contractions and equal to 0.6 for

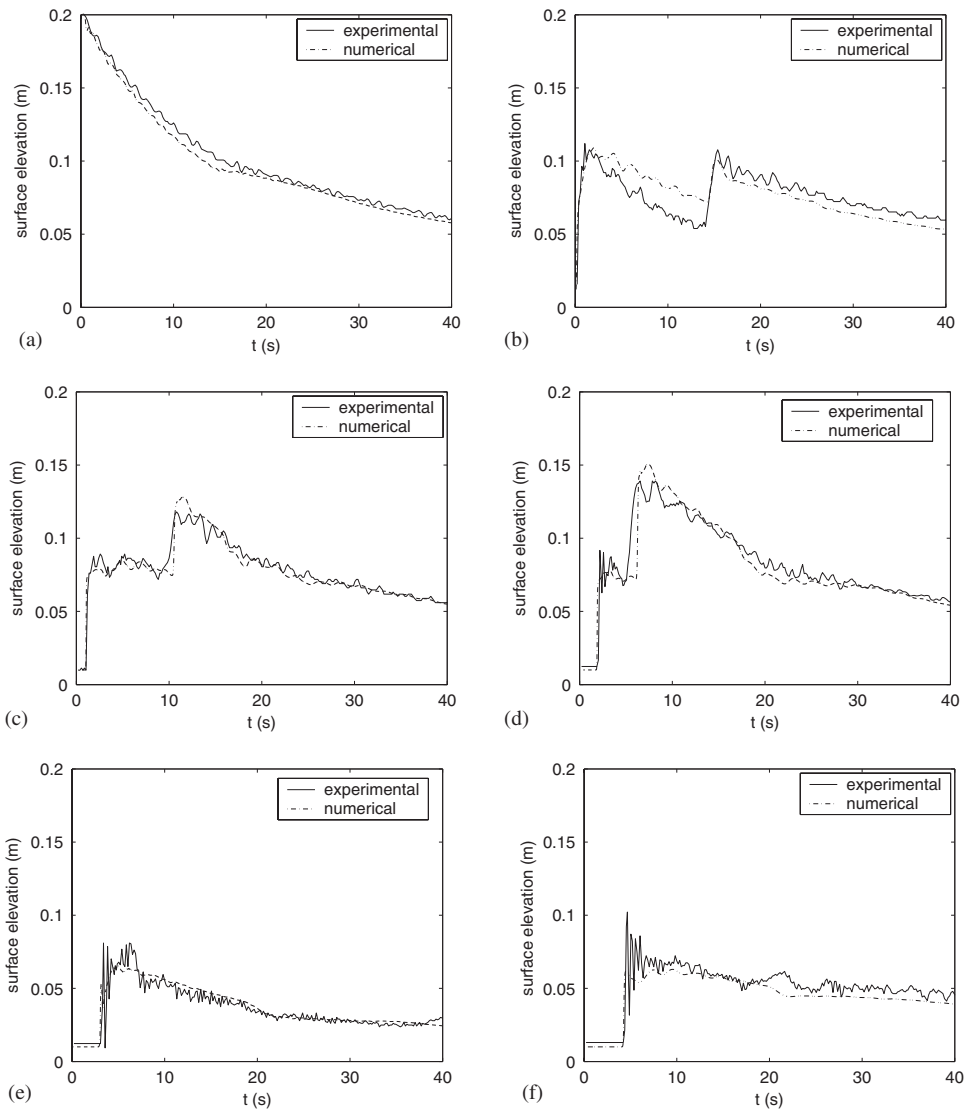


Figure 19. CADAM dam break. Wet-bed test predicted and measured water surface time histories: (a) Gauge 1; (b) Gauge 2; (c) Gauge 3; (d) Gauge 4; (e) Gauge 5; and (f) Gauge 6.

culvert with square edges. In this case, the culvert is very short, and so C_h has been modified to 0.9 and 0.6 for the wet- and dry-bed cases, respectively (from numerical experiments). After the gate is fully lifted, further local head losses should be taken into account because the width of the opening is small compared with the length of the dyke wall. Here, the local head loss is again modelled using Equation (30) with ξ set to 0.8, at cells immediately either side of the gate opening.

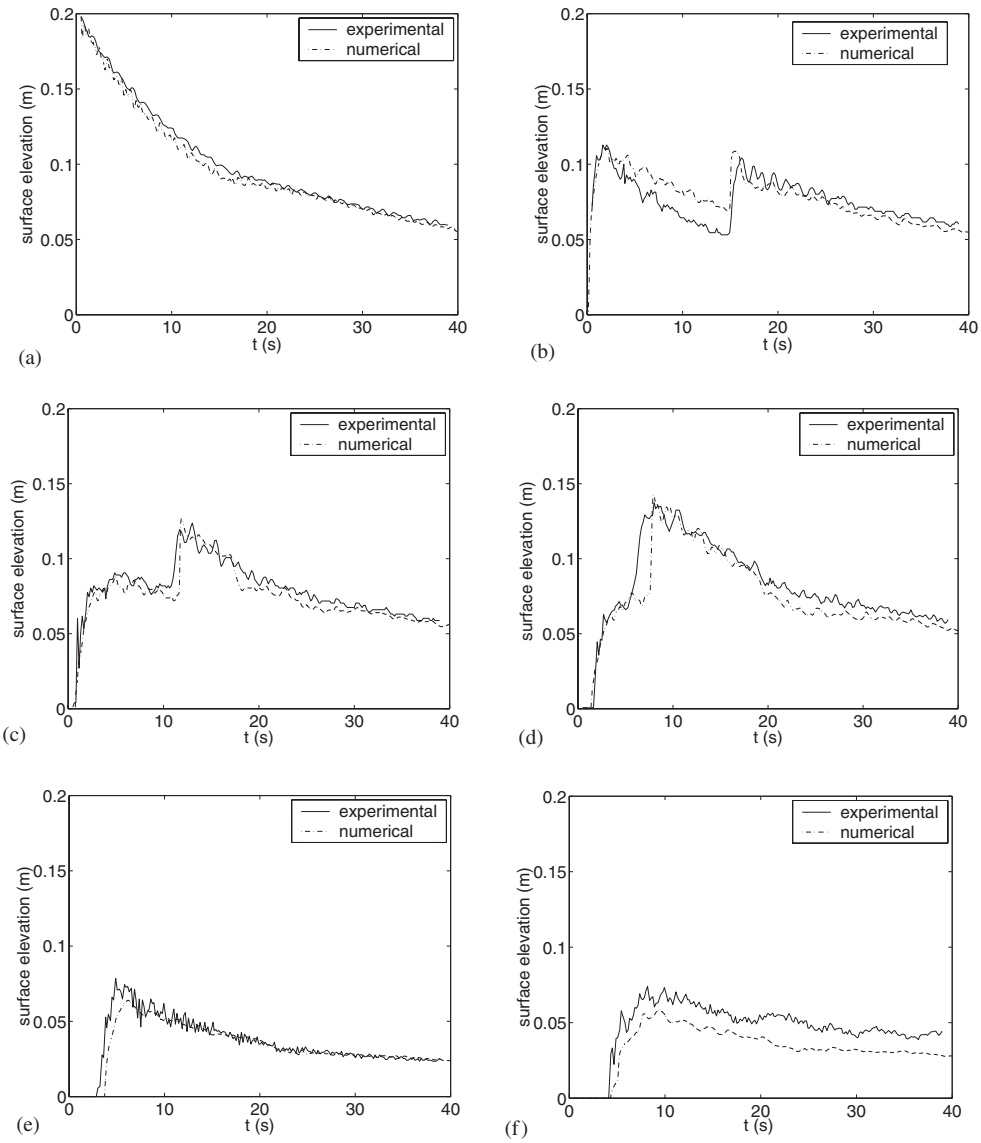


Figure 20. CADAM dam break. Dry-bed test predicted and measured water surface time histories: (a) Gauge 1; (b) Gauge 2; (c) Gauge 3; (d) Gauge 4; (e) Gauge 5; and (f) Gauge 6.

Figure 23 presents the free surface elevations and adapted quadtree grids at $t = 2$, 4, 10 and $t = 18$ s, respectively, for a case where the flat basin contains water of initial depth 0.05 m. During the early stages of the dyke break, an almost semi-circular shock front (i.e. a bore) propagates radially away from the middle of the gate opening. The bore is followed by a rarefaction wave (a moving hydraulic drop) that is pushed radially downstream by the

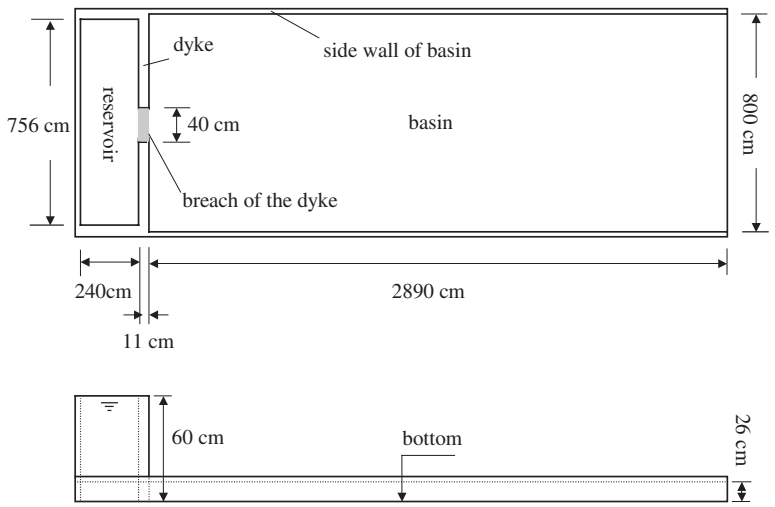


Figure 21. Delft University of Technology dyke break: top view and side view of the experiment layout.

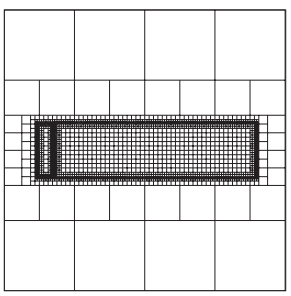


Figure 22. Delft University of Technology dyke break: initial quadtree grid.

strength of the jet through the gate opening. By $t = 4$ s, the bore front is nearly circular, except where it reflects at the lateral walls. There is a plateau behind the bore, and then a hydraulic drop (the rarefaction wave) that continues to be pushed downstream by the supercritical radial jet-like flow through the gate opening. Divergent hydraulic jumps are located either side of the jet. At later times, the bore front becomes nearly straight as it propagates downstream. Interacting curved bore-reflection waves are evident in the elongating flow region upstream of the bore until the rarefaction wave is reached. Further upstream, the bore reflection waves coalesce with the hydraulic jump on either side of the jet. The rarefaction wave remains centred about 5 m from the gate opening from $t = 10$ to 18 s, but with increasing sharply defined curvature about the centreline of the basin. The rarefaction wave extends across the basin in a laterally complicated manner, being affected by wall reflections. Figure 24 presents a plot of the predicted bore front locations at times $t = 1$, 2, 3 and 4 s. Figure 25 shows predicted and measured time-dependent water depth histories at gauge points located—1, 6

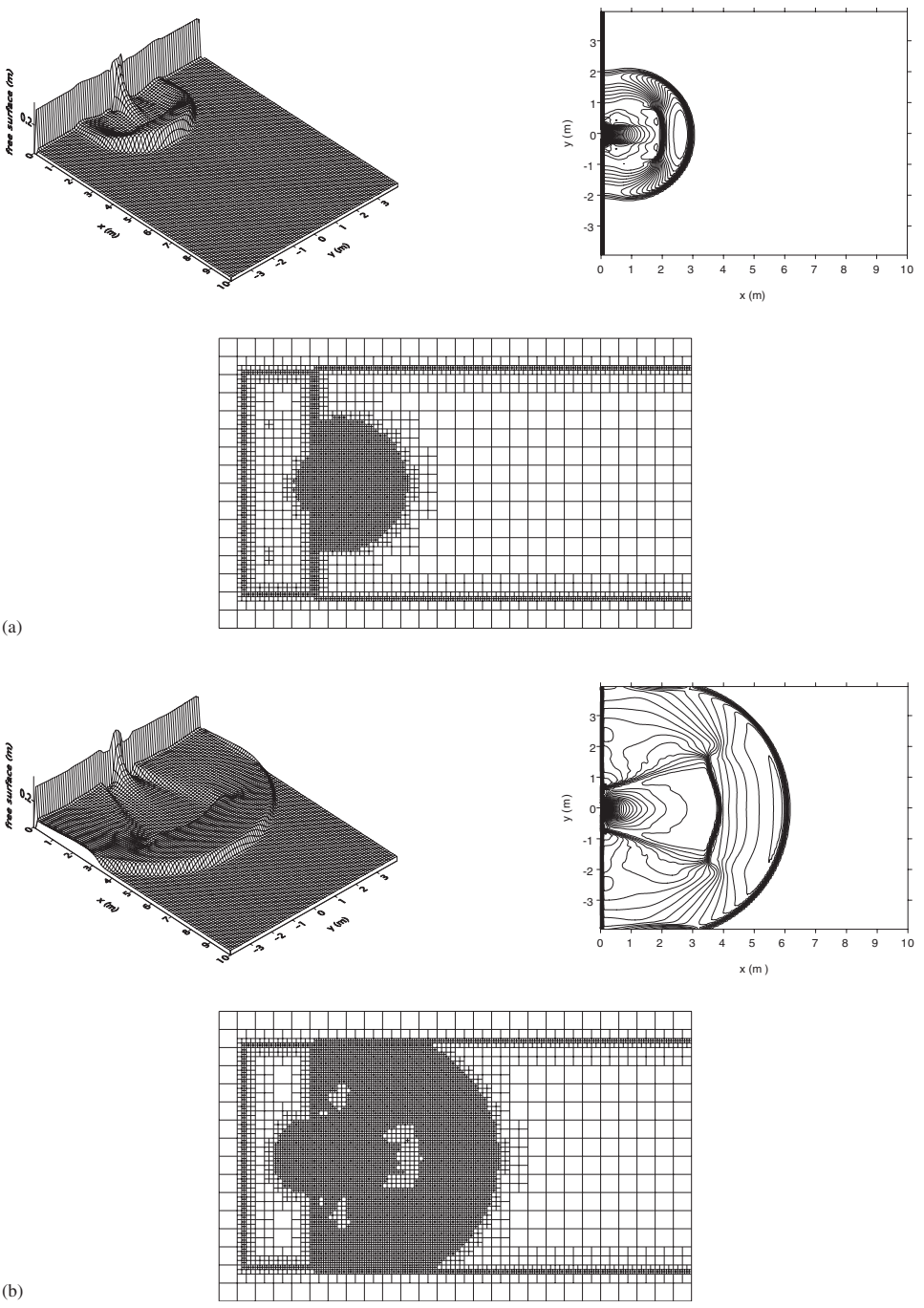
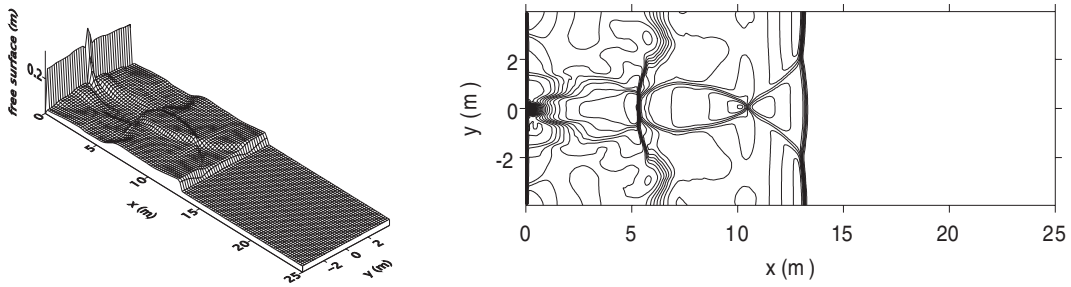
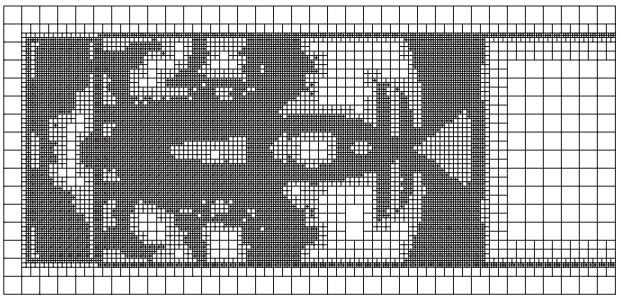
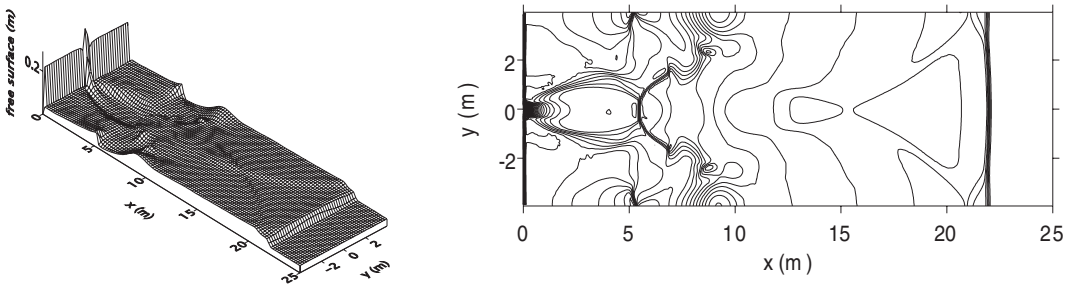


Figure 23. Delft University of Technology wet-bed dyke break: free surface elevation, contour plots and adapted quadtree grids: (a) $t = 2$ s; (b) $t = 4$ s; (c) $t = 10$ s; and (d) $t = 18$ s.



(c)





(d)

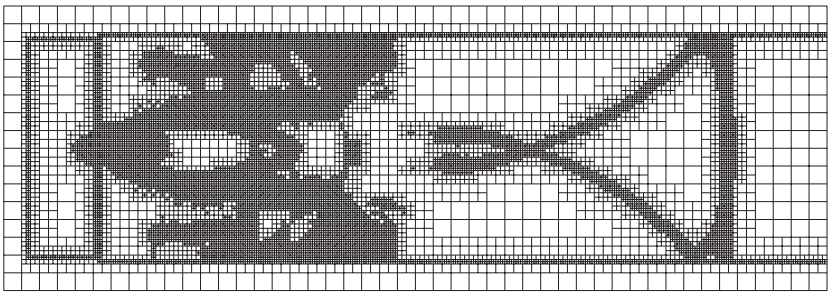


Figure 23. *Continued.*

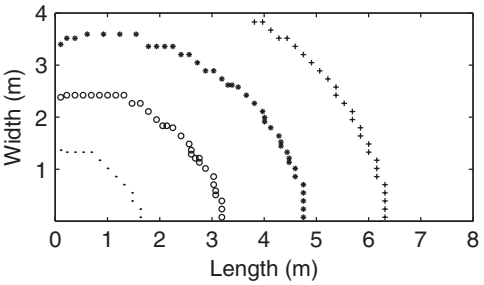


Figure 24. Delft University of Technology wet-bed dyke break: numerical front bore positions at $t = 1$ s; $t = 2$ s; $t = 3$ s; $t = 4$ s.

and 13 m from the centre of the gate opening in the downstream direction. There is a steady decrease in water level upstream of the gate opening. The other plots show the development of the bore front as it progresses downstream from the gate opening. The present results are in close agreement with the experimental measurements and alternative numerical predictions obtained at Delft University of Technology [40, 42]. The predicted and measured front speeds downstream from the gate opening are each approximately 1.6 m/s.

Figures 26–28 show the equivalent results when the basin is initially dry everywhere. The time-histories of water surface elevation are very similar to those obtained in the laboratory by Duinmeijer [40]. The bore front is greatly reduced, and perhaps only exists in the simulation as a consequence of the assumption of culvert flow as the gate opens. At the early stages, the flow spreads away from the gate opening with an almost circular flood of fairly uniform depth. The wet–dry interface is more that of a contact wave than a front. After $t = 4$ s, the wet–dry interface reaches the side walls, and reflection occurs. The wet–dry front straightens, and behind it a pattern of reflected curved waves can be seen by $t = 18$ s. Overall, the contact-type wet–dry front moves significantly faster than the bore in the previous case, as can be seen from Figure 27, where the downstream speed varies from 2.4 to 1.8 m/s, in reasonable agreement with the experimental data of Duinmeijer [40].

6. CONCLUSIONS

This paper has described a Godunov-type finite volume numerical solver of the hyperbolic shallow water equations based on dynamically adaptive quadtree grids. The model uses the MUSCL-Hancock-type HLLC approximate Riemann solver to evaluate cell interface fluxes. Detailed predictions of standard dam break tests are presented, and the results found to be in close agreement with alternative schemes in the literature. The performance of the dynamically adaptive quadtree-based scheme has been compared against uniform fixed quadtree grids for a circular wet-bed dam break. It is found that the dynamically adaptive approach offers high resolution while maintaining computational efficiency in cases where the flow gradients are locally steep. The results presented for the dry-bed circular dam break are at a resolution that should be useful for benchmarking 2D shallow water solvers. The predictions of the dam break into an L-shaped channel are remarkably similar to the experimental measurements obtained

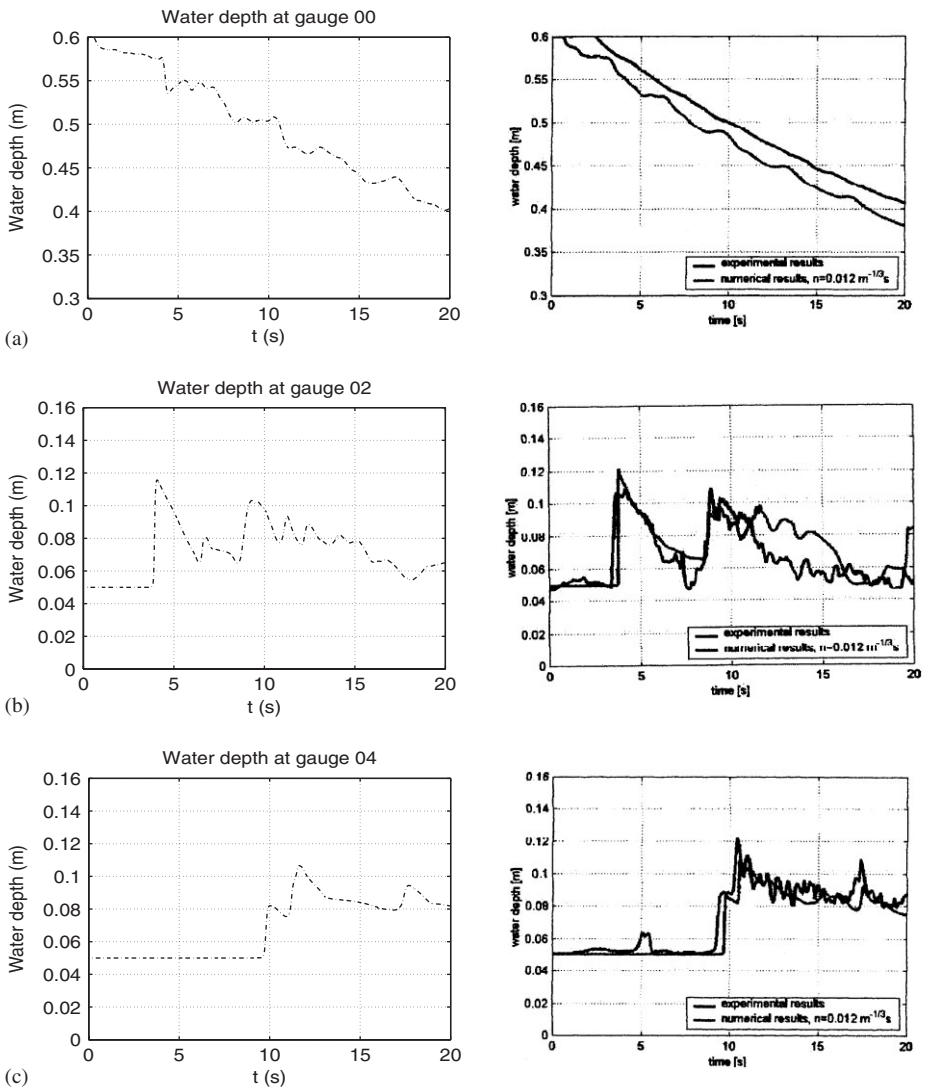


Figure 25. Delft University of Technology wet-bed dyke break: time histories of water depth at different gauge points (left: numerical predictions from the present model; right: alternative numerical results and experimental measurements obtained by Duinmeijer [40]).

by the CADAM group. The present scheme reproduces correctly the bore hydrodynamics including reflection effects as well as the emptying of the tank. Numerical predictions of a dyke-break at laboratory-scale are very similar to experimental data from Delft University of Technology, and provide a detailed picture of the complicated evolution of the flood hydrodynamics over a flat bed where lateral walls are present.

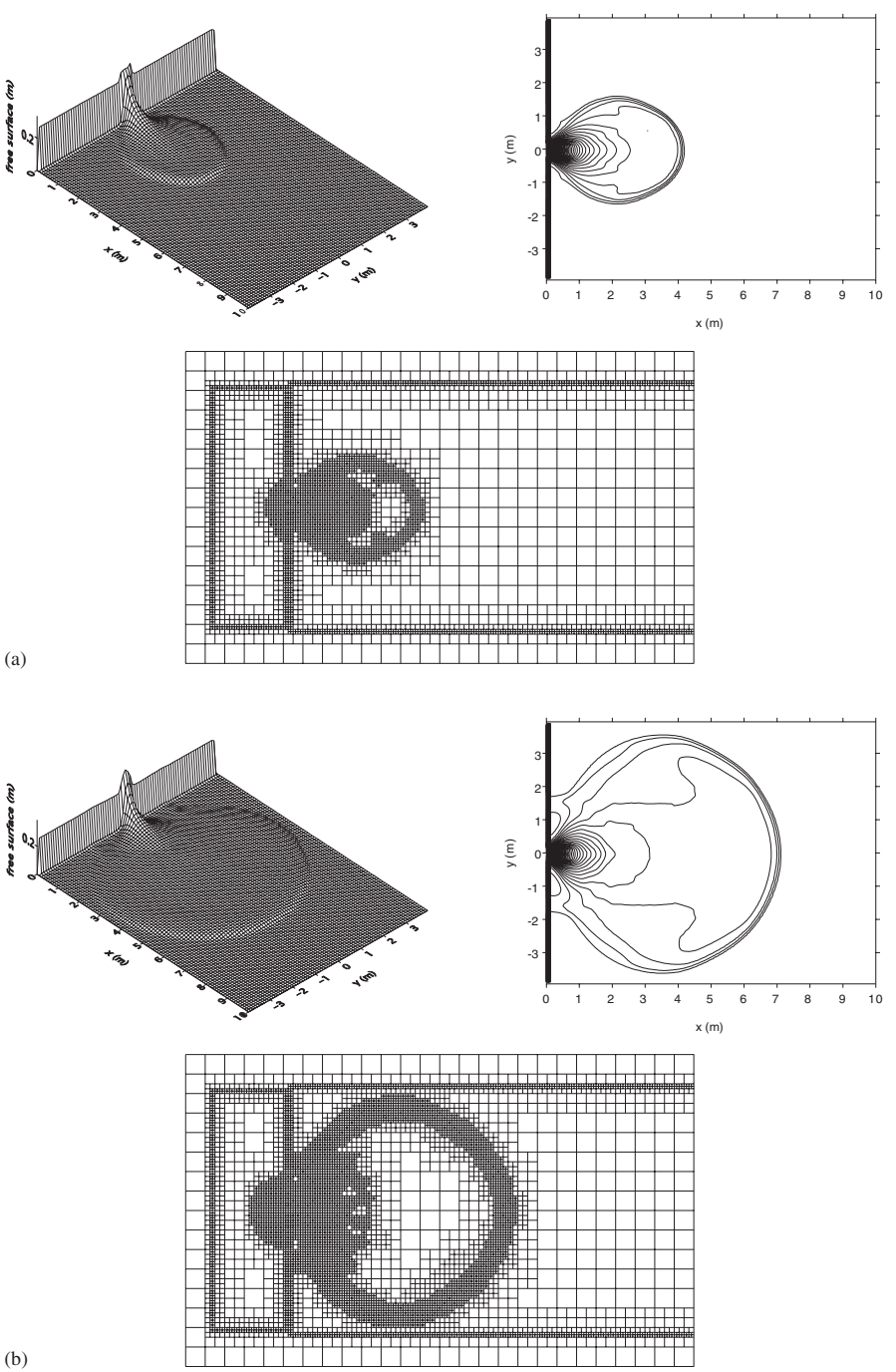
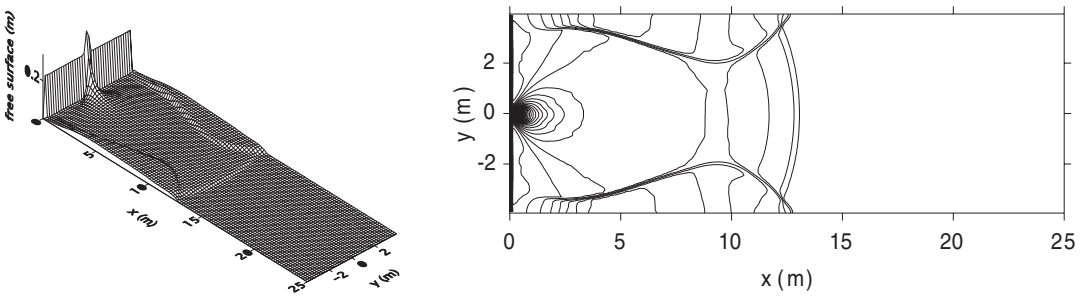
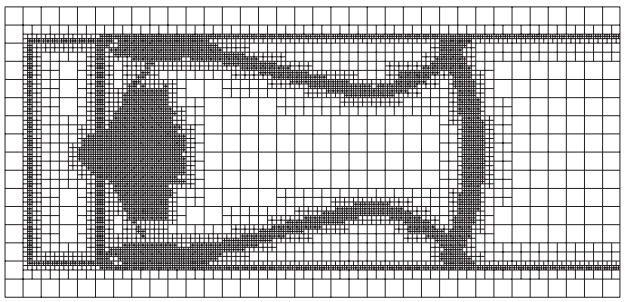
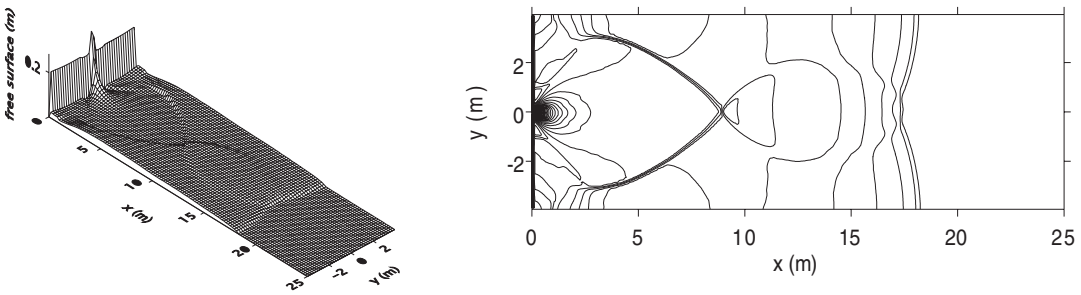


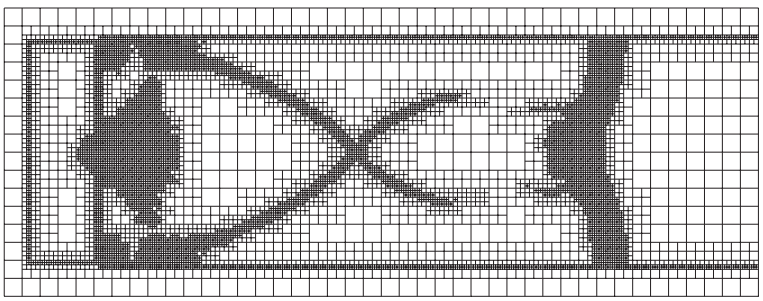
Figure 26. Delft University of Technology dry-bed dyke break: free surface elevation, contour plots and adapted quadtree grids: (a) $t = 2$ s; (b) $t = 4$ s; (c) $t = 10$ s; and (d) $t = 18$ s.





(c)





(d)

Figure 26. *Continued.*

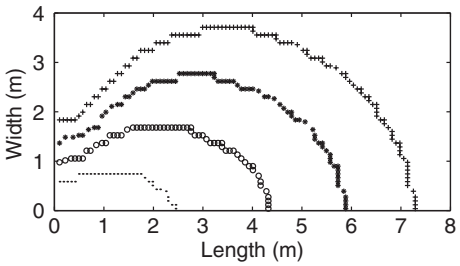


Figure 27. Delft University of Technology dry-bed dyke break: numerical wet–dry interface positions at $t = 1$ s; $t = 2$ s; $t = 3$ s; $t = 4$ s.

NOTATION

$\mathbf{A}$	inviscid flux Jacobian matrix
C	Chézy coefficient or Courant number
c	wave celerity
C_{f}	bed roughness coefficient
C_h	contraction coefficient
f	Coriolis parameter
$\mathbf{f}, \mathbf{g}$	flux vectors
ζ	free surface elevation above still water level
ξ	empirical coefficient for local head loss
g	acceleration due to gravity
H	headpond water level
h	total water depth
h_{s}	still water depth
h_{w}	height from the bed to the foot of a gate
h_ξ	local head loss due to a sudden contraction or vertical step
$\mathbf{L}$	left eigenvector matrix
n	Manning coefficient
$\mathbf{n}$	unit vector
n_x, n_y	Cartesian components of the unit vector $\mathbf{n}$
Q	discharge per unit width
$\mathbf{r}$	distance vector
r	gradient ratio
$\mathbf{R}$	right eigenvector matrix
$\mathbf{s}$	vector of source terms of the shallow water equations
S_{L}, S_{M}, S_{R}	speeds of the left, shear and right waves of the Riemann solution
S_{ox}, S_{oy}	bottom slope components in the x- and y-directions
t	time
$\mathbf{u}$	vector of conserved variables
u, v	depth-averaged velocity components in the x- and y-directions
ρ	water density
x, y	horizontal and vertical Cartesian co-ordinates in physical plane

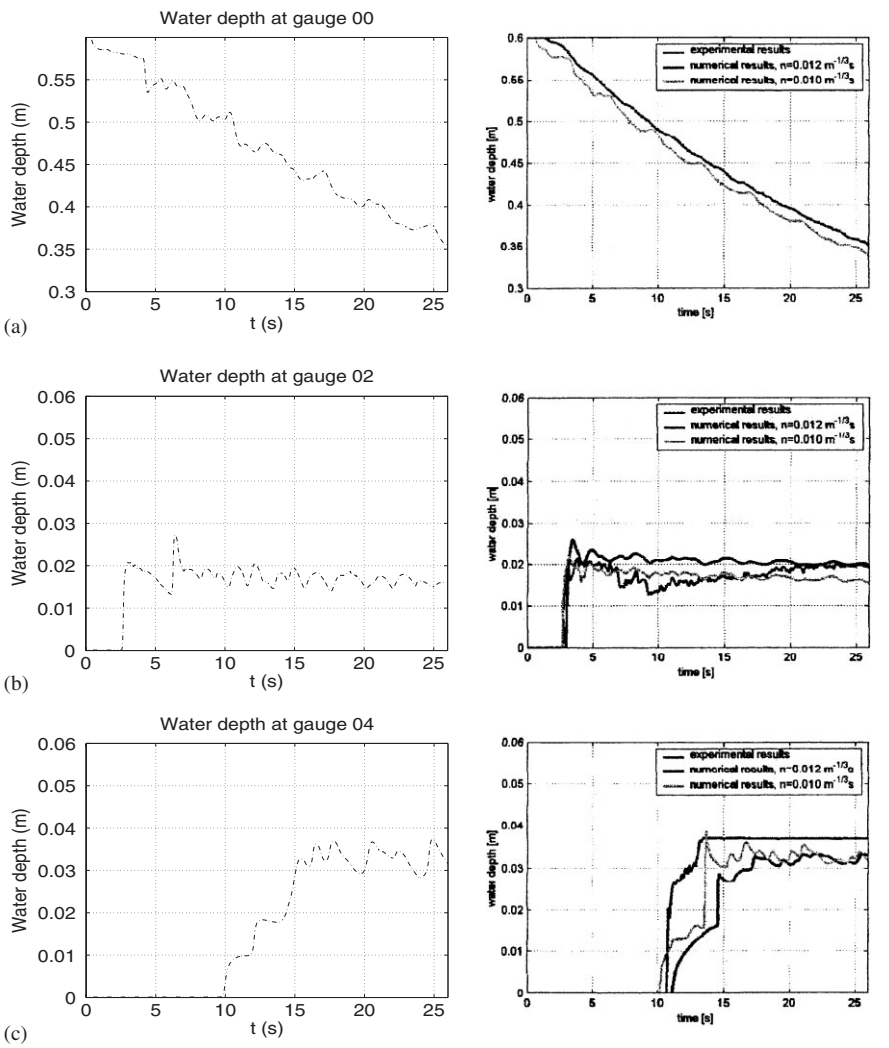


Figure 28. Delft University of Technology dry-bed dyke break: time histories of water depth at different gauge points (left: numerical predictions from the present model; right: alternative numerical results and experimental measurements obtained by Duinmeijer [40]).

τ_{bx}, τ_{by}	bed shear stress components in the x- and y-directions
τ_{ξ}	resistance stress due to local head loss
$\Delta x, \Delta y$	cell length in x- and y-directions, respectively
Θ	free surface gradient adaptivity parameter
Ψ	slope limiter
Δt	time step
$\lambda_1,\ \lambda_2,\ \lambda_3$	eigenvalues

τ_{wx}, τ_{wy}	free surface shear stress components in the x- and y-directions
$\frac{\partial}{\partial t}$, $\frac{\partial}{\partial x}$, $\frac{\partial}{\partial y}$	partial differential coefficients

Subscripts

i, j	grid cell indices
L, R	left and right directions
$*$	star region of the Riemann solution

Superscripts

L, R	left and right Riemann states of either side of a cell interface
n	time levels

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support provided by the U.K. Engineering and Physical Sciences Research Council through EPSRC Grant GR/R13623. The authors would also like to thank Dr Ben Rogers (Johns Hopkins University), Professor Jinren Ni (Peking University) and Dr Paul Taylor (Oxford University).

REFERENCES

1. Sakkas JG, Strelkoff T. Dam-break flood in a prismatic dry channel. *Journal of Hydraulic Division* (ASCE) 1973; **99**(HY12):2195–2216.
2. Chen C. Laboratory verification of a dam-break flood model. *Journal of Hydraulic Division* (ASCE) 1980; **106**:535–556.
3. Katopodes ND, Schamber DR. Applicability of dam-break flood wave models. *Journal of Hydraulic Engineering* (ASCE) 1983; **109**(5):702–721.
4. Toro EF. *Shock-Capturing Methods for Free-surface Shallow Flows*. Wiley: Chichester, 2001.
5. García-Navarro P, Alcrudo F, Savirón JM. 1-D Open-channel flow simulation using TVD-McCormack scheme. *Journal of Hydraulic Engineering* (ASCE) 1992; **118**(10):1359–1372.
6. Aureli F, Mignosa P, Tomirotti M. Numerical simulation and experimental verification of dam-break flows with shocks. *Journal of Hydraulic Research* 2000; **38**(3):197–206.
7. Garcia-Navarro P, Fras A, Villanueva I. Dam-break flow simulation: some results for one-dimensional models of real cases. *Journal of Hydrology* 1999; **216**:227–247.
8. Rahman M, Chaudhry MH. Simulation of dam-break flow with grid adaptation. *Advances in Water Resources* 1998; **21**(1):1–9.
9. Akanbi AA, Katopodes ND. Models for flood propagation on initially dry land. *Journal of Hydraulic Engineering* (ASCE) 1988; **114**(7):686–706.
10. Zoppou C, Roberts S. Catastrophic collapse of water supply reservoirs in urban areas. *Journal of Hydraulic Engineering* (ASCE) 1999; **125**(7):686–695.
11. Alcrudo F, García-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.
12. Roe PL, Pike J. Efficient construction and utilisation of approximate Riemann solutions. In *Computing Methods in Applied Science and Engineering*. North-Holland: Amsterdam, 1984.
13. Zhao DH, Shen HW, Tabious III GQ, Lai SJ, Tan WY. Finite-volume two-dimensional unsteady-flow model for river basins. *Journal of Hydraulic Engineering* (ASCE) 1994; **120**(7):864–883.
14. Osher S. Numerical solution of singular perturbation problems and hyperbolic systems of conservation laws. In *Mathematical Studies*, Axelsson O *et al.* (eds), vol. 47. North-Holland: Amsterdam, 1981.
15. 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.
16. Roe PL. Approximate Riemann solvers, parameter vectors and difference schemes. *Journal of Computational Physics* 1981; **43**:357–372.

17. Sleigh PA, Gaskell PH, Berzina M, Wright NG. An unstructured finite-volume algorithm for predicting flow in rivers and estuaries. *Computers and Fluids* 1998; **27**(4):479–508.
18. Toro EF, Spruce M, Speares W. Restoration of the contact surface in the HLL Riemann solver. *Shock Waves* 1994; **4**:25–34.
19. Toro EF. A weighted average flux method for hyperbolic conservation laws. *Proceedings of the Royal Society of London* (*A*) 1989; **423**:401–418.
20. Fraccarollo L, Toro EF. Experimental and numerical assessment of the shallow water model for two-dimension dam-break type problems. *Journal of Hydraulic Research* 1995; **33**(6):843–863.
21. Zoppou C, Roberts S. Numerical solution of the two-dimensional unsteady dam break. *Applied Mathematical Modelling* 2000; **24**:457–475.
22. Van Leer B. Towards the ultimate conservative difference scheme IV: a new approach to numerical convection. *Journal of Computational Physics* 1977; **23**:276.
23. Van Leer B. Towards the ultimate conservative difference scheme V: a second order sequel to Godunov's method. *Journal of Computational Physics* 1979; **32**:101.
24. 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.
25. Causon DM, Ingram DM, Mingham CG, Yang G, Pearson RV. Calculation of shallow water flows using a Cartesian cut cell approach. *Advances in Water Resources* 2000; **23**:545–562.
26. Greaves DM, Borthwick AGL. Hierarchical tree-based finite element mesh generation. *International Journal for Numerical Methods in Engineering* 1999; **45**:447–471.
27. Borthwick AGL, Marchant RD, Copeland GJM. Adaptive hierarchical grid model of water-borne pollutant transport. *Advances in Water Resources* 2000; **23**:849–865.
28. Yiu C, Greaves DM, Cruz S, Saalehi A, Borthwick AGL. Quadtree grid generation: information handling, boundary fitting and CFD applications. *Computers and Fluids* 1996; **25**(8):759–769.
29. Sibson R. A vector identity for the Dirichlet Tessellation. *Mathematical Proceedings of the Cambridge Philosophical Society* 1980; **87**:151–155.
30. Lloyd PM, Ball DJ, Stansby PK. Unsteady surface-velocity field measurement using particle tracking. *Journal of Hydraulic Research* 1995; **33**(4):519–534.
31. Rogers BD, Fujihara M, Borthwick AGL. Adaptive *Q*-tree Godunov-type scheme for shallow water equations. *International Journal for Numerical Methods in Fluids* 2001; **35**:247–280.
32. Rogers BD, Borthwick AGL, Taylor PH. Mathematical balancing of flux gradient and source terms prior to using Roe's approximate Riemann solver. *Journal of Computational Physics* 2003; **192**(2):422–451.
33. Harten A, Lax PD, van Leer B. On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM Review* 1983; **25**(1):35–61.
34. Toro EF. *Riemann Solvers and Numerical Methods for Fluid Dynamics—A Practical Introduction*. Springer: Berlin, 1997.
35. Hu K, Mingham GG, Causon DM. Numerical simulation of wave overtopping of coastal structures using the non-linear shallow water equations. *Coastal Engineering* 2000; **41**:433–465.
36. Mingham CG, Causon DM. A high resolution finite volume method for the shallow water equations. *Journal of Hydraulic Engineering* 1998; **124**(6):605–614.
37. Fujihara M, Borthwick AGL. Godunov-type solution of curvilinear shallow-water equations. *Journal of Hydraulic Engineering* (ASCE) 2000; **126**(11):827–836.
38. Zhou JG, Causon DM, Ingram DM, Mingham CG. Numerical solutions of the shallow water equations with discontinuous bed topography. *International Journal for Numerical Methods in Fluids* 2002; **38**:769–788.
39. Soares Frazão S, Sillen S, Zech Y. Dam-break flow through sharp bends: physical model and 2D Boltzmann model validation. *Proceedings of the CADAM Meeting*, Wallingford, UK, 1998.
40. Duinmeijer SPA. Verification of Delft FLS. *Technical Report*, Delft University of Technology, 2002.
41. Henderson FM. *Open Channel Flow*. Macmillan: New York, 1966.
42. Battjes JA, Gerritsen H. Coastal modelling for flood defence. *Philosophical Transactions of the Royal Society* 2002; **360**(1796):1461–1475.

