CHAPTER 1

HIGH-RESOLUTION FINITE VOLUME METHODS FOR THE SHALLOW WATER EQUATIONS WITH BATHYMETRY AND DRY STATES

Randall J. LeVeque and David L. George

Department of Applied Mathematics, University of Washington, Box 352420,
Seattle, WA 98195-2420
E-mail: rjl@amath.washington.edu, dgeorge@amath.washington.edu

We give a brief review of the wave-propagation algorithm, a high-resolution finite volume method for solving hyperbolic systems of conservation laws. These methods require a Riemann solver to resolve the jump in variables at each cell interface into waves. We present a Riemann solver for the shallow water equations that works robustly with bathymetry and dry states. This method is implemented in CLAWPACK and applied to benchmark problems from the Third International Workshop on Long-Wave Runup Models, including a two-dimensional simulation of runup during the 1993 tsunami event on Okushiri Island. Comparison is made with wave tank experimental data provided for the workshop. Some preliminary results using adaptive mesh refinement on the 26 December 2004 Sumatra event are also presented.

1. Introduction

We will present a brief introduction to a class of high-resolution finite volume methods for hyperbolic problems and discuss the application of these methods to long-wave run-up problems using the shallow water equations. To solve the benchmark problems for this workshop we have used such methods in one and two space dimensions that work robustly with bathymetry (bottom topography) and dry states, and that automatically handle the moving interface between water and land. Some results on the benchmark problems are presented in Sections 6 and 8, and more results, along with some animations, may be found at the website [16].

We use a mathematical framework known as the wave-propagation algorithm that has been implemented in the software package CLAWPACK (Conservation Laws Package) in 1, 2, and 3 space dimensions, and which also includes adaptive mesh refinement capabilities. This algorithm gives a general formulation of a class of finite volume methods known as "high-resolution shock-capturing Godunov-type methods" that are second order accurate in space and time on smooth solutions while automatically capturing discontinuities in the solution (including shocks or hydraulic jumps) with minimal numerical smearing and no spurious oscillations. More details on these algorithms and their application to hyperbolic problems may be found in LeVeque[13,15] and the CLAWPACK software documentation, available at `http://www.amath.washington.edu/~claw/`.

The fact that these are finite volume methods means that, rather than pointwise approximations to the solution, the numerical solution consists of approximations to the *cell averages* of the solution over grid cells. Here the grid cells are assumed to be intervals $[x_{i-1/2}, x_{i+1/2}]$ of uniform length Δx in one dimension or rectangles $[x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}]$ in two dimensions, but more general nonuniform grids can also be used.

Finite volume methods are particularly appropriate when solving systems of conservation laws, in which case the integral of the solution over each grid cell is modified only due to fluxes through the edges of the grid cell. Dividing this statement by the cell area leads to an update formula for the cell averages based on numerical approximations to the flux through each edge, as written out below. Such a finite volume method is based directly on the integral form of the conservation law and can be applied to problems with discontinuous solutions more reliably than finite difference approximations to the differential equation form of the conservation law, which does not hold at a discontinuity.

In one space dimension the integral form of a conservation law is

$$\frac{d}{dt}\int_{x_1}^{x_2} q(x,t)\,dx = f(q(x_1,t)) - f(q(x_2,t)) \qquad \forall x_1, x_2, \tag{1}$$

where $q(x,t) \in \mathbb{R}^m$ is the vector of conserved quantities and $f(q)$ is the flux function. This states that the total mass of q in any interval $[x_1, x_2]$ changes only due to fluxes through the edges.

The shallow water equations on a flat bottom have this form with $m = 2$ and

$$q = \begin{bmatrix} h \\ hu \end{bmatrix}, \qquad f(q) = \begin{bmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \end{bmatrix}, \tag{2}$$

where h is the fluid depth, u is the horizontal velocity, and g is the gravitational constant. These equations express the conservation of mass and momentum.

If the solution is sufficiently smooth, then the integral conservation law (1) can be manipulated to yield

$$\int_{x_1}^{x_2} q_t(x,t) + f(q(x,t))_x = 0 \qquad \forall x_1, x_2,$$

and hence

$$q_t + f(q)_x = 0, \tag{3}$$

which is the PDE form of the conservation law (with subscripts denoting partial derivatives). This equation is called *hyperbolic* if the Jacobian matrix $f'(q_0) \in \mathbb{R}^{m\times m}$ is diagonalizable and has real eigenvalues for any physically relevant state q_0. Hyperbolic problems typically model wave propagation and the eigenvalues correspond to the propagation velocities if we linearize about the state q_0. For the shallow water equations given by (2),

$$f'(q) = \begin{bmatrix} 0 & 1 \\ -u^2 + gh & 2u \end{bmatrix}, \tag{4}$$

with eigenvalues

$$\lambda^1 = u - \sqrt{gh}, \qquad \lambda^2 = u + \sqrt{gh}, \tag{5}$$

and corresponding eigenvectors

$$r^1 = \begin{bmatrix} 1 \\ u - \sqrt{gh} \end{bmatrix} = \begin{bmatrix} 1 \\ \lambda^1 \end{bmatrix}, \qquad r^2 = \begin{bmatrix} 1 \\ u + \sqrt{gh} \end{bmatrix} = \begin{bmatrix} 1 \\ \lambda^2 \end{bmatrix}. \tag{6}$$

A finite volume method in *conservation form* updates the cell average Q_i^n of the solution over the grid cell using an expression

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x}[F_{i+1/2}^n - F_{i-1/2}^n] \tag{7}$$

where

$$\begin{aligned} Q_i^n &\approx \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_{i+1/2}} q(x,t_n)\,dx \\ F_{i-1/2}^n &\approx \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} f(q(x_{i-1/2},t))\,dt \end{aligned} \tag{8}$$

are the numerical approximations to the cell average and interface flux, respectively. The update (7) comes directly from integrating (1) in time from t_n to t_{n+1} and dividing by Δx. Equation (7) can also be viewed as a direct discretization of the PDE (3), but viewing the value $F_{i-1/2}^n$ as an approximation to the interface flux is key in developing high-resolution methods for nonlinear problems.

We will generally drop the superscript n on quantities at time t_n to simplify the notation below, particularly since other superscripts will be needed.

1.1. *Godunov's method*

The methods we use are extensions of Godunov's method, a first-order method for gas dynamics developed in the 1950s in which the interface flux $F_{i-1/2}$ is computed by solving a *Riemann problem* between the states Q_{i-1} and Q_i. This is simply the conservation law with piecewise constant data (as in a shock tube or dam break problem). Such a problem naturally arises at the cell interface if the solution at time t_n is approximated by a piecewise constant function with values Q_j at all points in the jth grid cell.

Godunov's method is applicable to any hyperbolic conservation law. For a linear problem in which $f(q) = Aq$ for some matrix A (so $f'(q) = A$), the solution to the Riemann problem for any states Q_{i-1} and Q_i consists of m discontinuities (waves), each proportional to an eigenvector r^p of A, and propagating at speed equal to the corresponding eigenvalue λ^p (for $p = 1,\ 2,\ \ldots,\ m$). Since A must be diagonalizable, we can write

$$A = R\Lambda R^{-1}$$

where $\Lambda = \text{diag}(\lambda^1,\ \ldots,\ \lambda^m)$ and $R = [r^1 \cdots r^m]$ is the invertible matrix of eigenvectors.

Solving the Riemann problem between states Q_{i-1} and Q_i is then easily accomplished by decomposing $\Delta Q_{i-1/2} = Q_i - Q_{i-1}$ into eigenvectors of A, *i.e.*, writing $\Delta Q_{i-1/2}$ as a linear combination of the vectors r^p,

$$Q_i - Q_{i-1} = \sum_{p=1}^{m} \alpha_{i-1/2}^p r^p \equiv \sum_{p=1}^{m} \mathcal{W}_{i-1/2}^p. \tag{9}$$

We use $\mathcal{W}^p$ to denote the pth wave in this Riemann solution. The vector of coefficients $\alpha_{i-1/2}^p$ is given by $\alpha_{i-1/2} = R^{-1}\Delta Q_{i-1/2}$. Godunov's method in the linear case is then defined by setting

$$F_{i-1/2} = f(Q_{i-1/2}^{\Downarrow}) = AQ_{i-1/2}^{\Downarrow}$$

where $Q_{i-1/2}^{\Downarrow}$ denotes the value at the interface $x_{i-1/2}$ in the Riemann solution,

$$Q_{i-1/2}^{\Downarrow} = Q_{i-1} + \sum_{p:\lambda^p<0} \mathcal{W}_{i-1/2}^p.$$

Multiplying by A gives

$$\begin{aligned} F_{i-1/2} &= AQ_{i-1} + \sum_{p=1}^{m} \alpha^p_{i-1/2} (\lambda^p)^- r^p \\ &= AQ_{i-1} + A^- \Delta Q_{i-1/2}, \end{aligned} \tag{10}$$

where

$$A^- = R \operatorname{diag}((\lambda^p)^-) \, R^{-1}, \qquad \text{with} \;\; \lambda^- = \min(\lambda, 0). \tag{11}$$

Alternatively, we can write

$$Q^{\Downarrow}_{i-1/2} = Q_i - \sum_{p:\lambda^p > 0} \mathcal{W}^p_{i-1/2}$$

and obtain

$$\begin{aligned} F_{i-1/2} &= AQ_i + \sum_{p=1}^{m} \alpha^p_{i-1/2} (\lambda^p)^+ r^p \\ &= AQ_i + A^+ \Delta Q_{i-1/2}, \end{aligned} \tag{12}$$

where

$$A^+ = R \operatorname{diag}((\lambda^p)^+) \, R^{-1}, \qquad \text{with} \;\; \lambda^+ = \max(\lambda, 0). \tag{13}$$

For a linear problem the resulting method is simply the first-order upwind method, extended from the scalar advection equation to a general system by diagonalizing the system and applying the upwind method to each characteristic component in the appropriate direction based on the propagation velocity.

For nonlinear problems, such as the shallow water equations, the exact solution to the Riemann problem is harder to calculate but can still be worked out (see, *e.g.*, LeVeque[15], Toro[20]) and the resulting interface flux used for $F_{i-1/2}$. In practice, however, it is usually more efficient to use some *approximate Riemann solver* to obtain $F_{i-1/2}$. One popular choice is to use a "Roe solver" following the work of Roe[18] for gas dynamics, in which the data Q_{i-1}, Q_i is used to define a "Roe-averaged" Jacobian matrix $A_{i-1/2}$ by a suitable combination of $f'(Q_{i-1})$ and $f'(Q_i)$. The numerical flux is then determined by solving the Riemann problem for the linear problem $q_t + A_{i-1/2} q_x = 0$. The Roe average is chosen to have the property that

$$f(Q_i) - f(Q_{i-1}) = A_{i-1/2}(Q_i - Q_{i-1}). \tag{14}$$

This leads to nice properties in the approximate solution. The Roe matrix for the shallow water equations is easily computed (see, *e.g.*, LeVeque[15]), and is simply the Jacobian matrix (4) evaluated at the Roe-averaged state

$$\hat{h}_{i-1/2} = \frac{h_i + h_{i-1}}{2}, \qquad \hat{u}_{i-1/2} = \frac{u_{i-1}\sqrt{gh_{i-1}} + u_i\sqrt{gh_i}}{\sqrt{gh_{i-1}} + \sqrt{gh_i}}. \tag{15}$$

The eigenvalues, or "Roe speeds", are therefore

$$\hat{s}^1_{i-1/2} = \hat{u}_{i-1/2} - \sqrt{g\hat{h}_{i-1/2}}, \qquad \hat{s}^2_{i-1/2} = \hat{u}_{i-1/2} + \sqrt{g\hat{h}_{i-1/2}}, \tag{16}$$

and the Roe eigenvectors are

$$\hat{r}^1_{i-1/2} = \begin{bmatrix} 1 \\ \hat{s}^1_{i-1/2} \end{bmatrix}, \qquad \hat{r}^2_{i-1/2} = \begin{bmatrix} 1 \\ \hat{s}^2_{i-1/2} \end{bmatrix}. \tag{17}$$

In one dimension the CLAWPACK software requires a Riemann solver that, for any states Q_{i-1} and Q_i, returns a set of M_w waves $\mathcal{W}^1_{i-1/2}, \ \ldots, \ \mathcal{W}^{M_w}_{i-1/2}$, and propagation speeds for the waves, $s^1_{i-1/2}, \ \ldots, \ s^{M_w}_{i-1/2}$. The number of waves M_w may be equal to m, the dimension of the system, but could be different. The Riemann solver must also return the *fluctuations* $\mathcal{A}^-\Delta Q_{i-1/2}$ and $\mathcal{A}^+\Delta Q_{i-1/2}$, two vectors that are used to update the solution according to

$$Q^{n+1}_i = Q_i - \frac{\Delta t}{\Delta x}(\mathcal{A}^+\Delta Q_{i-1/2} + \mathcal{A}^-\Delta Q_{i+1/2}). \tag{18}$$

These fluctuations should have the property that

$$\mathcal{A}^-\Delta Q_{i-1/2} + \mathcal{A}^+\Delta Q_{i-1/2} = f(Q_i) - f(Q_{i-1}), \tag{19}$$

so that they represent a "flux difference splitting". The fluctuations may be defined in terms of the interface fluxes as

$$\begin{aligned} \mathcal{A}^+\Delta Q_{i-1/2} &= f(Q_i) - F_{i-1/2} \\ \mathcal{A}^-\Delta Q_{i+1/2} &= F_{i+1/2} - f(Q_i) \end{aligned} \tag{20}$$

Then (19) is satisfied (note the shift in index) and (18) reduces to the flux-differencing update formula (7). The form (18) is used in CLAWPACK and the general formulation of the wave-propagation algorithms because it is more flexible and allows the extension of these methods to hyperbolic problems that are not in conservation form, in which case there is no "flux function" (see LeVeque[15]).

The notation $\mathcal{A}^{\pm}\Delta Q_{i-1/2}$ used for the fluctuation vectors is suggested by the fact that, for a linear problem, the natural choice is

$$\begin{aligned}\mathcal{A}^{-}\Delta Q_{i-1/2} &= A^{-}(Q_i - Q_{i-1}),\\ \mathcal{A}^{+}\Delta Q_{i-1/2} &= A^{+}(Q_i - Q_{i-1}),\end{aligned} \tag{21}$$

where the matrices $A^{\pm}$ are defined by (13) and (11). In general they are often defined by

$$\begin{aligned}\mathcal{A}^{-}\Delta Q_{i-1/2} &= \sum_{p=1}^{M_w}(s^p_{i-1/2})^{-}\mathcal{W}^p_{i-1/2},\\ \mathcal{A}^{+}\Delta Q_{i-1/2} &= \sum_{p=1}^{M_w}(s^p_{i-1/2})^{+}\mathcal{W}^p_{i-1/2}.\end{aligned} \tag{22}$$

For a linear problem or a nonlinear problem when the Roe solver is used, this agrees with the definition (20).

The first-order method (18) only uses the fluctuations returned from the Riemann solver, and does not make explicit use of the waves $\mathcal{W}^p$ or speeds s^p. These quantities are used in the high-resolution correction terms discussed in the next section.

1.2. *High-resolution corrections*

The method (18) is only first order accurate. The second-order Lax-Wendroff method for the linear problem can be written as a modification to (18), as

$$Q_i^{n+1} = Q_i - \frac{\Delta t}{\Delta x}(\mathcal{A}^{+}\Delta Q_{i-1/2} + \mathcal{A}^{-}\Delta Q_{i+1/2}) - \frac{\Delta t}{\Delta x}(\tilde{F}_{i+1/2} - \tilde{F}_{i-1/2}), \tag{23}$$

where the fluctuations are still given by (21) for the linear problem and the correction fluxes are

$$\tilde{F}_{i-1/2} = \frac{1}{2}\left(|A| - \frac{\Delta t}{\Delta x}A^2\right)\Delta Q_{i-1/2} = \frac{1}{2}\left(I - \frac{\Delta t}{\Delta x}|A|\right)|A|\Delta Q_{i-1/2} \tag{24}$$

where $|A| = A^{+} - A^{-}$. Inserting this in (23) and manipulating yields a more familiar form of the Lax-Wendroff method,

$$Q_i^{n+1} = Q_i - \frac{\Delta t}{2\Delta x}A(Q_{i+1} - Q_{i-1}) + \frac{\Delta t^2}{2\Delta x^2}A^2(Q_{i+1} - 2Q_i + Q_{i-1}). \tag{25}$$

This method is second order accurate on smooth solutions but is highly dispersive and nonphysical oscillations arise near steep gradients or discontinuities in the solution, which can completely destroy the accuracy. These oscillations can be avoided by using the form (23) and applying appropriate limiters to the correction terms. To this end we rewrite (24) as

$$\tilde{F}_{i-1/2} = \frac{1}{2}\sum_{p=1}^{m}\left(I - \frac{\Delta t}{\Delta x}|\lambda^p|\right)|\lambda^p|\widetilde{\mathcal{W}}^p_{i-1/2}, \tag{26}$$

where $\mathcal{W}^p_{i-1/2} = \alpha^p_{i-1/2} r^p$ are the waves obtained from the Riemann solution, as in (9), and $\widetilde{\mathcal{W}}^p_{i-1/2}$ represents a "limited" version of the wave. Each wave is limited by comparing it to the wave in the same family arising from the Riemann problem at the neighboring interface in the upwind direction, *i.e.*, we set

$$\widetilde{\mathcal{W}}^p_{i-1/2} = \text{limiter}(\mathcal{W}^p_{i-1/2}, \mathcal{W}^p_{I-1/2}) \tag{27}$$

where

$$I = \begin{cases} i-1 & \text{if } \lambda^p > 0, \\ i+1 & \text{if } \lambda^p < 0. \end{cases}$$

If $\mathcal{W}^p_{I-1/2}$ and $\mathcal{W}^p_{i-1/2}$ are "comparable" in some sense then this component of the solution is presumed to be smoothly varying. In this case the corresponding term in (26) can be expected to give a useful correction that will help improve accuracy and the limiter should return $\widetilde{\mathcal{W}}^p_{i-1/2} \approx \mathcal{W}^p_{i-1/2}$. However, if $\mathcal{W}^p_{I-1/2}$ and $\mathcal{W}^p_{i-1/2}$ quite different then this component is not smoothly varying and attempting to add an additional term from the Taylor series may make things worse rather than better. In this case the limiter should modify the wave, typically be reducing its magnitude. There is an extensive theory of limiters that will not be further discussed here.

The method (23) with correction fluxes (26) is easily extended to nonlinear problems. Recall that the (approximate) Riemann solver returns fluctuations $\mathcal{A}^\pm\Delta Q_{i-1/2}$, waves $\mathcal{W}^p_{i-1/2}$, and speeds $s^p_{i-1/2}$. The only change in the formulas required in order to apply (23) to a general nonlinear problem is to replace λ^p in (26) by the local speed $s^p_{i-1/2}$, obtaining

$$\tilde{F}_{i-1/2} = \frac{1}{2}\sum_{p=1}^{M_w}\left(I - \frac{\Delta t}{\Delta x}|s^p_{i-1/2}|\right)|s^p_{i-1/2}|\widetilde{\mathcal{W}}^p_{i-1/2}. \tag{28}$$

2. The f-wave approach

The method described above can be reformulated in a way that will prove particularly useful when source terms are added to the equations, as needed to handle variable bathymetry. Recall that the waves $\mathcal{W}^p_{i-1/2}$ correspond to a splitting of the jump in Q at the interface $x_{i-1/2}$,

$$Q_i - Q_{i-1} = \sum_{p=1}^{M_w} \mathcal{W}^p_{i-1/2}.$$

In a linear problem, or a nonlinear problem that has been locally linearized using a Roe matrix $A_{i-1/2}$, these waves are obtained by expressing $\Delta Q_{i-1/2}$ as a linear combination of the eigenvectors $r^p_{i-1/2}$ of the matrix, *i.e.*, $\mathcal{W}^p_{i-1/2} = \alpha^p_{i-1/2} r^p_{i-1/2}$ for some scalars $\alpha^p_{i-1/2}$, as is done in (9). Alternatively, we could split the jump in $f(Q)$ into eigencomponents as

$$f(Q_i) - f(Q_{i-1}) = \sum_{p=1}^{M_w} \beta^p_{i-1/2} r^p_{i-1/2} \equiv \sum_{p=1}^{M_w} \mathcal{Z}^p_{i-1/2}.$$

If the matrix $A_{i-1/2}$ satisfies Roe's condition (14), then we simply have $\mathcal{Z}^p_{i-1/2} = s^p_{i-1/2}\mathcal{W}^p_{i-1/2}$. For other approximate Riemann solvers it is necessary to determine an appropriate splitting of $f(Q)$ based on the splitting of Q

The vectors $\mathcal{Z}^p$ are called f-waves because they carry jumps in f rather than jumps in q. Since $\mathcal{Z}^p_{i-1/2} = \text{sgn}(s^p_{i-1/2})|s^p_{i-1/2}|\mathcal{W}^p_{i-1/2}$ for linearized Riemann solvers, the natural generalization of the correction flux (28) for the f-wave formulation is

$$\tilde{F}_{i-1/2} = \frac{1}{2}\sum_{p=1}^{M_w}\left(I - \frac{\Delta t}{\Delta x}\text{sgn}(s^p_{i-1/2})\right)\widetilde{\mathcal{Z}}^p_{i-1/2}, \tag{29}$$

where $\widetilde{\mathcal{Z}}^p_{i-1/2}$ is a limited version of $\mathcal{Z}^p_{i-1/2}$ calculated using the same limiter as previously applied to $\mathcal{W}^p$.

One advantage of the f-wave approach is that any linearly independent set of vectors r^p_{i-1} can be used to define the splitting of Δf into waves $\mathcal{Z}^p_{i-1/2}$ and the method remains conservative. Of course a reasonable choice is required in order to maintain consistency with the differential equation, but for example the eigenvectors of any reasonable approximate Jacobian matrix based on Q_{i-1} and Q_i could be used without needing to impose the Roe condition (14). For the shallow water equations this suggests using vectors

$$r^1 = \begin{bmatrix} 1 \\ s^1 \end{bmatrix}, \qquad r^2 = \begin{bmatrix} 1 \\ s^2 \end{bmatrix} \tag{30}$$

where s^1 and s^2 are some approximations to the wave speeds of the two waves in the Riemann solution. Taking s^1 and s^2 to be the Roe speeds

(16) recovers the Roe solver, but in some cases this is not a good choice. In particular the Roe solver can fail when dry states are expected in the solution. In Section 4 we present a different choice of speeds that can be used much more robustly.

3. Bathymetry and source terms

We now consider the shallow water equations over non-constant bathymetry with elevation $y = B(x)$. In this case $h(x,t)$ represents the depth of the water above the bathymetry. A sloping bottom can accelerate the fluid and gives rise to a source term in the momentum equation proportional to the slope $B_x(x)$. The shallow water equations now take the form

$$q_t + f(q)_x = \psi(q,x), \tag{31}$$

where q and f are as in (2) and the source term is

$$\psi(q,x) = \begin{bmatrix} 0 \\ -ghB_x \end{bmatrix}. \tag{32}$$

One way to tackle (31) numerically is is to use a *fractional step* procedure. In each time step one first solves the homogeneous conservation law (3) to advance Q^n by Δt to obtain an intermediate state Q^*, and then solves

$$q_t = \psi(q,x) \tag{33}$$

over time Δt to advance Q^* to Q^{n+1}. This often works well, but is subject to numerical difficulties, particularly in situations where there is a steady state with $f(q) = \psi(q,x)$ and the desired dynamic solution is a relatively small perturbation of this steady state. In this case solving (3) and (33) may each lead to significant changes in the solution that should nearly cancel out. Numerically, the use of distinct numerical methods in separate steps can lead to errors that swamp the desired solution when the waves of interest are small perturbations of the steady state.

Instead of using a fractional step method, we modify the f-wave formulation of the hyperbolic solver and incorporate the source term into the flux difference before decomposing this into waves, *i.e.*, we decompose

$$f(Q_i) - f(Q_{i-1}) - \Delta x \Psi_{i-1/2} = \sum_{p=1}^{M_w} \mathcal{Z}^p_{i-1/2}, \tag{34}$$

where $\Psi_{i-1/2}$ is a discretization of the source term. For the shallow water equations with bathymetry the source term $\Delta x ghB_x$ at $x_{i-1/2}$

is approximated by $\frac{1}{2}g(h_{i-1}+h_i)(B_i-B_{i-1})$, resulting in the vector $f(Q_i)-f(Q_{i-1})-\Delta x\Psi_{i-1/2}$ taking the form

$$\begin{bmatrix} h_iu_i - h_{i-1}u_{i-1} \\ \left(h_iu_i^2+\frac{1}{2}gh_i^2\right)-\left(h_{i-1}u_{i-1}^2+\frac{1}{2}gh_{i-1}^2\right)+\frac{1}{2}g(h_{i-1}+h_i)(B_i-B_{i-1}) \end{bmatrix}. \tag{35}$$

This vector is decomposed into f-waves, for example by writing it as a linear combination of the eigenvectors $r^p_{i-1/2}, (p=1,2)$ of the Roe matrix or of (30).

Of particular importance is the special case of a motionless body of water over variable bathymetry, in which case $u\equiv 0$ and $h(x,t)+B(x)$ is initially constant and should remain so. In this case the vector (35) becomes

$$\begin{bmatrix} 0 \\ \frac{1}{2}g\left(h_i^2-h_{i-1}^2\right)+\frac{1}{2}g(h_{i-1}+h_i)(B_i-B_{i-1}) \end{bmatrix}. \tag{36}$$

If h_j+B_j is constant in j then $B_i-B_{i-1}=h_{i-1}-h_i$ and (36) is the zero vector, resulting in $\mathcal{Z}^1_{i-1/2}=\mathcal{Z}^2_{i-1/2}=0$ and $\mathcal{A}^{\pm}\Delta Q_{i-1/2}=0$ in (34). Hence $Q_i^{n+1}=Q_i$ and the steady state is exactly maintained numerically.

Moreover, if we are modeling waves in which perturbations in $h+B$ are small compared to variations in B, then this vector captures the perturbations after canceling out the steady state portion of the variation. The wave limiters and high resolution correction terms are applied to these perturbations rather than to waves that include the large variation in h due to the bathymetry. As a result, this method is much more accurate in modeling the propagation of small amplitude perturbations than a fractional step method. This may be quite important in calculating the long-range propagation of small-amplitude tsunamis through the ocean.

The f-wave approach and its use for source terms is discussed further in Bale et. al.[2] and LeVeque[15]. A variety of other approaches have also been developed recently for this problem, for example Gosse[8], Greenberg and LeRoux[9], Jenny and Müller[11], Kurganov and Levy[12], LeVeque[14],

4. Dry states

It is well known that if the Roe solver is used to solve a Riemann problem in which $u_{i-1}<u_i$ with sufficient difference in velocities, then the approximate Riemann solution will have a negative depth in the intermediate state (see Figure 15.3 in LeVeque[15] for an illustration of this). This nonphysical behavior often causes the computation to crash. In reality a dry state should form if the velocity difference is sufficiently large, although the Roe solver

can fail even for smaller velocity differences (see Einfeldt, et. al.[7] for some discussion of related issues for the Euler equations). Similar problems arise when solving a Riemann problem with one state initially dry, $h_{i-1} = 0$ or $h_i = 0$, as happens at some cell interfaces for any problem involving wave motion at the shore.

This difficulty can be avoided by using the f-wave approach discussed in Sections 2 and 3 using eigenvectors r^1 and r^2 from (30) with a better choice of speeds than the Roe speeds. In most cases we use the "Einfeldt speeds"

$$s^1_{\scriptscriptstyle E} = \min(u_{i-1} - \sqrt{gh_{i-1}}, \hat{s}^1), \qquad s^2_{\scriptscriptstyle E} = \max(u_i + \sqrt{gh_i}, \hat{s}^2), \tag{37}$$

obtained by comparing the Roe speeds with the characteristic speeds λ^1_{i-1} and λ^1_i (the eigenvalues of the Jacobian matrices in states Q_{i-1} and Q_i). This choice is adapted from the suggestion of Einfeldt[6] that speeds corresponding to these be used in the HLL method for gas dynamics. The HLL approximate Riemann solver (after Harten, Lax, and van Leer[10]) simply uses two waves to approximate the Riemann solution (regardless of the dimension m of the system) with speeds s^1 and s^2 approximating the minimum and maximum wave speeds arising in the system. The HLLE method, using the Einfeldt speeds, chooses these speeds by comparing the Roe speed, a reasonable choice if the wave is a shock, and the extreme characteristic speed, which may be faster if the wave is instead a spreading rarefaction wave. Since $m = 2$ in the one-dimensional shallow water equations, the method we use is closely related to the HLLE method, though not the same and will be modified further to handle source terms and dry states below. (See LeVeque and Pelanti[17] for some more discussion of the relation between the HLL and Roe solvers and the f-wave approach.)

Using the f-wave approach with the choice of speeds (37) and corresponding eigenvectors (30) nearly always maintains non-negative depth. In fact, it can be shown that the total mass in the intermediate Riemann solution is always positive given these speeds and eigenvectors. However, as explained below, it is sometimes necessary to further modify the wave speeds to maintain non-negativity, at least in the subcritical case when $s^1_{\scriptscriptstyle E} < 0 < s^2_{\scriptscriptstyle E}$. In the supercritical case when both speeds have the same sign, no modification is needed.

We first consider the case of preserving non-negativity in a cell that has positive depth initially, $h_{i-1} > 0$ or $h_i > 0$, and later consider the case of preserving non-negativity in a cell that is already initially dry. Even in the case where both cells have a positive depth initially, $h_{i-1} > 0$ and $h_i > 0$, a negative depth can be generated if $s^1_E < 0 < s^2_E$ when the choice (37) is used. Although the total mass in the intermediate Riemann solution is positive, it may happen that the approximate Riemann solution leads to the mass going negative on one side of the interface. In this case it can be shown that the mass must be increasing on the other side by at least the same amount, and so negativity can be avoided by a transfer of mass that can be accomplished by increasing the speed on the side losing mass. Working out the formula to increase this speed just to the point where the negative state reaches $h = 0$, we find that in general the following speeds can always be used:

$$s^1 = \min\left(s^1_E,\ s^2_E/2 - \sqrt{(s^2_E/2)^2 + \max(0, s^2_E\Delta^1 - \Delta^2)/h_{i-1}}\right), \quad (38)$$

$$s^2 = \max\left(s^2_E,\ s^1_E/2 + \sqrt{(|s^1_E|/2)^2 + \max(0, \Delta^2 - s^1_E\Delta^1)/h_i}\right). \quad (39)$$

That is, given $h_{i-1} > 0$ initially, (38) will maintain that $h_{i-1} \geq 0$, and given $h_i > 0$ initially (39) will maintain that $h_i \geq 0$. Here Δ^1 and Δ^2 are the components of the modified flux difference

$$f(Q_i) - f(Q_{i-1}) - \Delta x \Psi_{i-\frac{1}{2}}, \quad (40)$$

which takes into account the bathymetry. Each speed (38) and (39) always corresponds to the Einfeldt speed unless a negative state would arise on that side, so for a given Riemann problem, at most one of s^1 and s^2 is different from the Einfeldt speed, and only when necessary to maintain non-negativity.

Maintaining the non-negativity in a cell with a depth that is initially 0 is handled somewhat differently. If $B_{i-1} = B_i$, and only one of h_{i-1} or h_i is positive, then the true solution to this Riemann problem consists only of a rarefaction wave with the leading edge propagating at speed $u_i - 2\sqrt{gh_i}$ if $h_{i-1} = 0$ or speed $u_{i-1} + 2\sqrt{gh_{i-1}}$ if $h_i = 0$. We use these speeds as s^1 or s^2 if $h_{i-1} = 0$ or $h_i = 0$ respectively. As stated above, using the speed (38) will preserve $h_{i-1} \geq 0$ if it is initially positive, and (39) will preserve $h_i \geq 0$ if it is initially positive. Using the speed of the leading edge of a rarefaction wave for the complimentary speed however, will not necessarily prevent the dry state from becoming negative. It can be shown however that negativity is only possible if the true rarefaction wave is transonic ($s^1 < 0 < u_{i-1} + 2\sqrt{gh_{i-1}}$ if $h_i = 0$ or $u_i - 2\sqrt{gh_i} < 0 < s^2$ if $h_{i-1} = 0$). Transonic rarefactions are a more general problem, and are discussed in the following section.

5. Entropy conditions

Integral conservation laws can be satisfied by discontinuous weak solutions subject to the Rankine-Hugoniot jump conditions. This results in possible non-uniqueness of weak solutions—an initial value problem might be satisfied by both a smooth solution and a discontinuous one. Determining the physically relevant solution requires additional admissibility conditions, often taking the form of an "entropy" function that is conserved except across a discontinuity. The name arises from the Euler equations of gas dynamics, where the entropy must increase when gas passes through a shock. For the shallow water equations, the "entropy" function is actually mechanical energy, which must decrease when passing through a discontinuity.

Since the integral conservation laws alone do not guarantee a unique solution to an initial value problem, a numerical method based on the conservation laws alone might converge to an entropy violating weak solution. An "entropy fix" is therefore often needed. Determining such a fix for Godunov-type methods requires a careful look at the particular Riemann solver being used. If an approximate solver is used, true solutions to Riemann problems which consist of m waves—any combination of rarefactions and shock waves, are replaced by m jump discontinuities locally at each grid cell. These discontinuities might approximate a physically relevant shock wave, or perhaps a smooth rarefaction. In the latter case, the jump discontinuity still approximates the conservation law, however it more closely resembles the entropy violating shock than the physically relevant rarefaction.

With the wave propagation methods, the waves arising from a particular Riemann problem at a grid cell interface are averaged onto the adjacent cells. Therefore, the local discrepancy between an entropy violating discontinuity and a smooth rarefaction may have no effect on the numerical solution, if both produce the same average within a grid cell. This will be the case if the wave structure of the rarefaction remains entirely within a grid cell. The exceptional case is a transonic rarefaction—a rarefaction where one of the eigenvalues passes through zero. This type of rarefaction has a wave structure that should overlap a cell interface, yet it is approximated by a jump discontinuity moving either to the left or the right. This does affect the numerical solution, and can cause a method to converge globally to an entropy violating weak solution. The fix is to determine when the correct entropy solution to a Riemann problem corresponds to a transonic rarefaction, and then split the entropy violating single wave, apportioning it to the adjacent grid cells appropriately.

An alternative and numerically more useful formulation of the entropy condition for the shallow water equations is that a physically correct shock in the p^{th} $(p = 1, 2)$ characteristic family must have the p-characteristics impinging on it. That is,

$$\lambda^p(q_l) > s > \lambda^p(q_r), \tag{41}$$

where s is the shock speed and λ^p is the p^{th} eigenvalue evaluated at q_l and q_r—the states directly to the left and right of the shock respectively. Therefore, if the entropy solution to a Riemann problem contains a shock connecting the left state Q_{i-1} to the middle state, denoted Q_m, then

$$\lambda^1(Q_{i-1}) > \lambda^1(Q_m). \tag{42}$$

Similarly if a shock connects the right state Q_i to the middle state Q_m, then

$$\lambda^2(Q_m) > \lambda^2(Q_i). \tag{43}$$

If (42) or (43) is violated, then in fact a rarefaction connects the corresponding states in the entropy solution. As explained above, an entropy violating Riemann solution will not affect the numerical solution, except in the case of a transonic rarefaction. Therefore the only cases in which an entropy fix is required are when

$$\lambda^1(Q_{i-1}) < 0 < \lambda^1(Q_m) \tag{44}$$

or

$$\lambda^2(Q_m) < 0 < \lambda^2(Q_i), \tag{45}$$

which indicate a transonic rarefaction in the first or second families respectively.

It is easy to check $\lambda^1(Q_{i-1})$ and $\lambda^2(Q_i)$. However, with the f-wave approach, Q_m is never explicitly computed so we cannot simply evaluate $\lambda^1(Q_m)$ and $\lambda^2(Q_m)$ directly. In fact the f-wave approach is not equivalent to using a single value for Q_m, but two middle values, one to the left and one to the right of the cell interface. The approach we've taken is to instead compare the Roe speeds, $\hat{s}^1_{i-\frac{1}{2}}$ and $\hat{s}^2_{i-\frac{1}{2}}$, with the right and left speeds, $\lambda^1(Q_{i-1})$ and $\lambda^2(Q_i)$. The motivation for this approach is that the Roe speeds can serve as an estimate for the shock speeds, allowing us to estimate whether (41) is satisfied. For instance, to detect the presence of a transonic rarefaction in the second family, $u_i + \sqrt{gh_i}$ is compared to $\hat{s}^2_{i-\frac{1}{2}}$. If $u_i + \sqrt{gh_i} > \hat{s}^2_{i-\frac{1}{2}}$ then most likely the true Riemann solution has a rarefaction in the second family. Furthermore $\hat{s}^2_{i-\frac{1}{2}}$ serves as an estimate for the characteristic speed at the center of the rarefaction fan, $\hat{s}^2_{i-1/2} \approx \frac{1}{2}(\lambda^2(Q_m) + \lambda^2(Q_i))$ and hence $\lambda^2(Q_m) \approx 2s^2_{i-1/2} - \lambda^2(Q_i)$. Therefore if

$$2(\hat{s}^2_{i-\frac{1}{2}}) - (u_i + \sqrt{gh_i}) < 0 < u_i + \sqrt{gh_i}, \tag{46}$$

then the true Riemann solution is likely to have a transonic rarefaction in the second characteristic family. The second f-wave $\mathcal{Z}^2$ should then be split into two waves, one moving to the right the other to the left. A similar test is done for the first characteristic family.

The entropy fix when one state in the Riemann problem is initially dry is somewhat different, and also acts to ensure that the depth in the corresponding cell remains non-negative. As mentioned in the previous section, the true Riemann solution in such a case consists of a single rarefaction. The speeds of the edges of the rarefaction wave are given by quantities in the wet cell ($s^1 = u_{i-1} - \sqrt{gh_{i-1}}$ and $s^2 = u_{i-1} + 2\sqrt{gh_{i-1}}$ if $h_{i-1} > 0$ or $s^1 = u_i - 2\sqrt{gh_i}$ and $s^2 = u_i + \sqrt{gh_i}$ if $h_i > 0$). In the event of a single transonic rarefaction, $s^1 < 0 < s^2$, an entropy fix such that the f-waves simply carry an appropriate proportion of the true single wave is necessary. We simply apportion to each f-wave an amount based on the proportion of the rarefaction in each cell.

This approximate Riemann solver works quite robustly in the context of the first-order accurate Godunov's method, in the f-wave formulation with bathymetry source terms and dry states. Addition of the second-order correction terms (28) adds new potential problems when dry states are present, as these waves can lead to a negative depth even when the first order method would not. Standard wave limiters devised to avoid nonphysical oscillations may not completely avoid this problem, and so we have added an additional limiting procedure to maintain nonnegative depth.

The procedure is best illustrated by writing the second-order method as the sum of the first-order Godunov update

$$Q_i^G = Q_i^n - \frac{\Delta t}{\Delta x}(\mathcal{A}^+\Delta Q_{i-1/2} + \mathcal{A}^-\Delta Q_{i+1/2}), \tag{47}$$

and the second-order correction fluxes (29),

$$Q_i^{n+1} = Q_i^G - \frac{\Delta t}{\Delta x}\left[\tilde{F}_{i+\frac{1}{2}} - \tilde{F}_{i-\frac{1}{2}}\right]. \tag{48}$$

Note that the correction flux at a cell interface takes mass away from one cell and adds mass to the adjacent cell, with the direction depending on the sign of its first component. The gross mass-flux out of the i^{th} grid cell, due to these correction fluxes, is therefore

$$M_i = \left[\max(0, \tilde{F}^1_{i+\frac{1}{2}}) - \min(0, \tilde{F}^1_{i-\frac{1}{2}})\right]. \tag{49}$$

If $\Delta t M_i$ is larger than the mass present after the Godunov update, $\Delta x(Q_i^G)^1$, the correction fluxes could potentially create a negative depth in this cell. This is prevented by re-limiting the correction fluxes based on which cell they take mass away from

$$\tilde{F}_{i-\frac{1}{2}} \to \varphi_{i-\frac{1}{2}} \tilde{F}_{i-\frac{1}{2}} \tag{50}$$

where

$$\varphi_{i-\frac{1}{2}} = \begin{cases} \min(1, \Delta x(Q_i^G)^1/\Delta t M_i) & \text{if} \quad \tilde{F}^1_{i-\frac{1}{2}} < 0 \\ \min(1, \Delta x(Q_{i-1}^G)^1/\Delta t M_{i-1}) & \text{if} \quad \tilde{F}^1_{i-\frac{1}{2}} > 0 \end{cases}. \tag{51}$$

This procedure is consistent with the standard limiting strategy, in that second-order accuracy is achieved throughout most of the domain, and limiting only occurs where there are problematic features, such as shock waves or near dry-states.

6. Results for Benchmark Problem 1

The first benchmark problem consists of a linear sloping beach, $b(x,t) = -x$, and initially motionless water with a shoreline at $x = 0$. An incoming wave is induced by a non-zero initial surface elevation, $\eta(x, 0)$, specified by data provided from $x = 0$ to 50000 meters in increments of 50 meters. We compare the resulting motion of the shoreline, the surface elevation, and the velocity field of our numerical solution with a provided analytical solution. The surface elevation and velocity data for the analytical solution were provided at three separate times, $t = 160$, 175, and 220 seconds. The position and velocity of the shoreline were provided for $t \in [0, 300]$ seconds.

We computed this problem using CLAWPACK, on a series of grids of varying resolution (from 500 to 5000 grid cells), each over a domain $x \in [-500, 50000]$ meters. The computational grid cells were clustered densely near the shoreline, using a piecewise linear mapping. This was required to efficiently capture the fine-scale motion of the shoreline, which occurred over a small fraction of the entire domain. Convergence to the analytical solutions was observed as the grids were refined.

Some of our computational results are shown with the analytical solutions in figures 1 and 2. We have chosen to show the results on a grid of

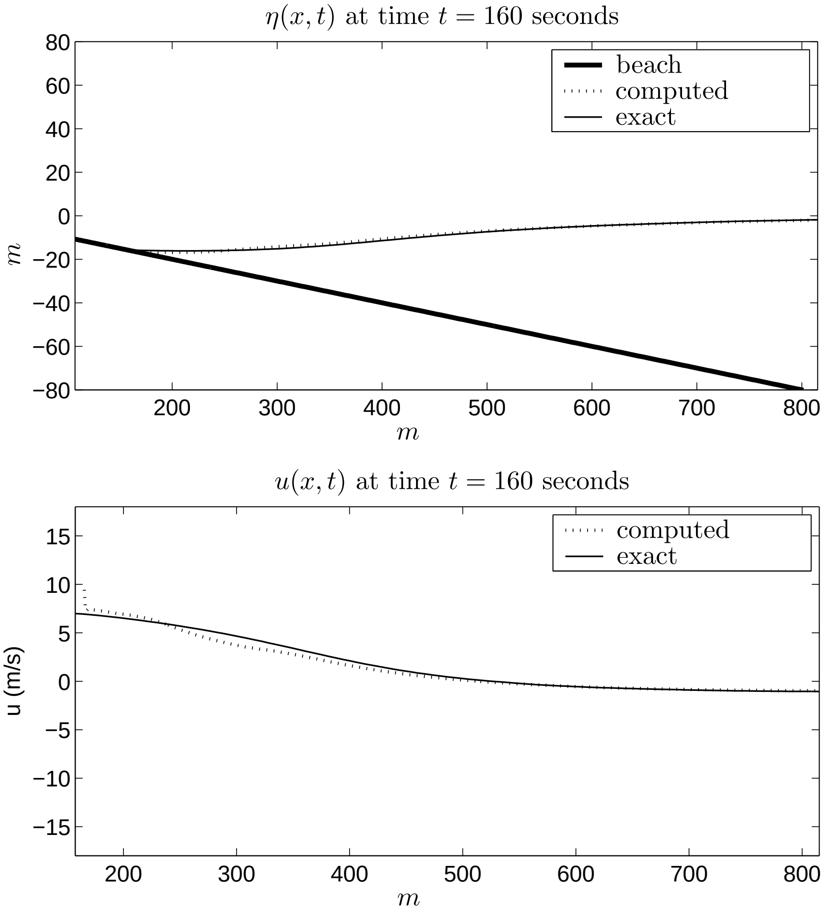


Fig. 1. Top: Water surface elevation at $t = 160s$, shown near the beach. Bottom: Velocity field at $t = 160s$ in the same region. Both computations were done on a 1000-point grid.

1000 points so that the numerical results are still distinguishable from the analytical solution. The surface elevation and the velocity field are shown at $t = 160s$ in figure 1. Note that the figures show only a small portion of the domain near the beach. Figure 2 shows the motion of the shoreline for the same computation. For additional results at other times and grid resolutions, see the website [16]

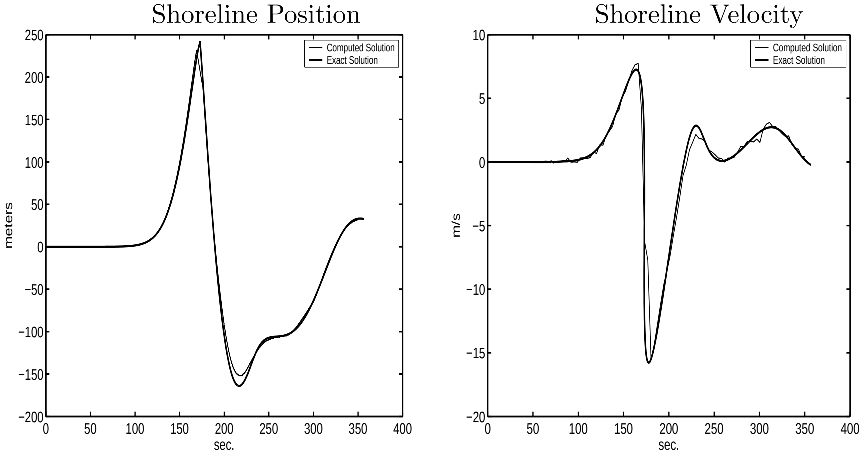


Fig. 2. Left: Position of the shoreline as a function of time, computed on a 1000-point grid. Right: Velocity of the shoreline for the same computation.

7. Extension to two space dimensions

In two space dimensions the shallow water equations with bathymetry take the form

$$q_t + f(q)_x + g(q)_y = \psi(q,x,y), \tag{52}$$

where

$$q = \begin{bmatrix} h \\ hu \\ hv \end{bmatrix}, \quad f(q) = \begin{bmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \\ huv \end{bmatrix}, \quad g(q) = \begin{bmatrix} hv \\ huv \\ hv^2 + \frac{1}{2}gh^2 \end{bmatrix}, \tag{53}$$

and the source term is $\psi = \psi_1 + \psi_2$ with

$$\psi_1(q,x,y) = \begin{bmatrix} 0 \\ -ghB_x(x,y) \\ 0 \end{bmatrix}, \quad \psi_2(q,x,y) = \begin{bmatrix} 0 \\ 0 \\ -ghB_y(x,y) \end{bmatrix}, \tag{54}$$

The general form of the wave-propagation algorithm is now

$$\begin{aligned} Q_{ij}^{n+1} = Q_{ij} &- \frac{\Delta t}{\Delta x}(\mathcal{A}^+\Delta Q_{i-1/2,j} + \mathcal{A}^-\Delta Q_{i+1/2,j}) \\ &- \frac{\Delta t}{\Delta y}(\mathcal{B}^+\Delta Q_{i,j-1/2} + \mathcal{B}^-\Delta Q_{i,j+1/2}) \\ &- \frac{\Delta t}{\Delta x}(\tilde{F}_{i+1/2,j} - \tilde{F}_{i-1/2,j}) - \frac{\Delta t}{\Delta y}(\tilde{G}_{i,j+1/2} - \tilde{G}_{i,j-1/2}). \end{aligned} \tag{55}$$

The fluctuations $\mathcal{A}^{\pm}\Delta Q$ and $\mathcal{B}^{\pm}\Delta Q$ are determined by solving one-dimensional Riemann problems normal to each cell interface, using the f-wave formulation described above to incorporate the appropriate portion of the source term. In the x-direction, for example, we solve the Riemann problem $q_t + f(q)_x = \psi_1$. The solution to this Riemann problem is exactly the same as the one-dimensional case with the addition of a contact discontinuity wave that passively advects the jump in the orthogonal velocity v. Using only these terms in (55) without the correction fluxes $\tilde{F}$ or $\tilde{G}$ gives a two-dimensional generalization of Godunov's method sometimes called donor-cell upwind. A better first-order method (corner transport upwind) is obtained by splitting the waves in each one-dimensional Riemann solution into waves moving in the transverse direction, so that wave motion oblique to the grid is more properly modeled. In CLAWPACK this requires the specification of a "transverse Riemann solver" that takes as input a fluctuation, say $\mathcal{A}^{+}\Delta Q$, and returns a splitting of this vector into up-going and down-going portions that affect the fluxes $\tilde{G}_{i,j+1/2}$ and $\tilde{G}_{i,j-1/2}$ respectively. This is typically based on the eigenvalues and eigenvectors of some approximate Jacobian matrix $g'(q)$ in the transverse direction, and is described in more detail for the shallow water equations in LeVeque[15]. Second-order correction terms can also be incorporated as in one space dimension, based on the waves obtained from the one-dimensional Riemann solution normal to each cell edge.

The presence of dry states leads to new complications when the algorithm is extended to two space dimensions. A wave moving transversely into a dry or nearly dry cell can produce a negative depth unless special care is taken by incorporating a more fully multidimensional limiter. Currently we do not use transverse propagation or second-order correction terms in two dimensions and are further developing this approach. Our goal is to ultimately be able to use transverse propagation and high-resolution correction terms for dry state problems in two dimensions. However, even without these terms reasonable results are obtained, as demonstrated in the next section.

8. Results for Benchmark Problem 2

In the second benchmark problem, we compare our computational solutions to laboratory data collected from a wave tank experiment. The wave tank was built as a 1:400 scale model, approximating coastline bathymetry near Monai, Japan, a region that suffered inundation from the 1993 Okushiri

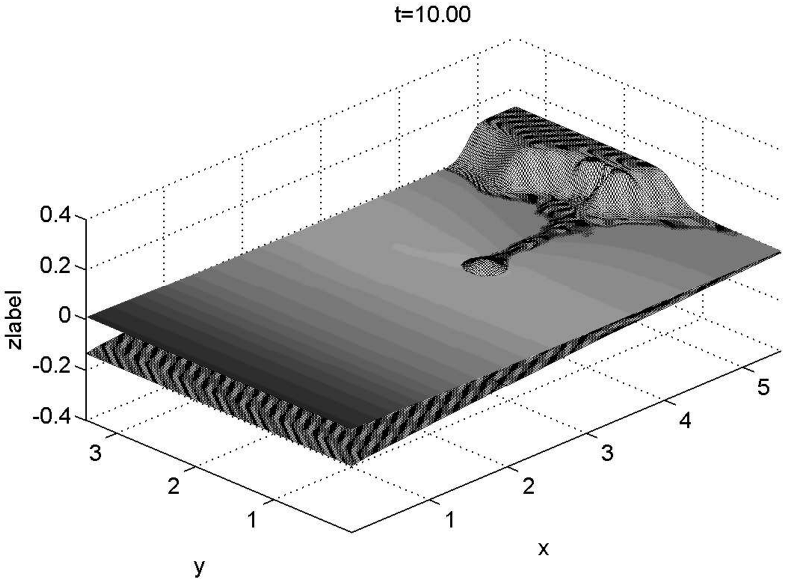


Fig. 3. Three-dimensional view of the numerical solution at one time.

tsunami. In the experiment, an incoming wave was induced by mechanical wave paddles along one side of the tank, and the resulting motion of the surface elevation was measured by gages at three separate locations. Additionally, a movie was made showing the tank from overhead during the experiment. This data and movie are available at the workshop website.

We computed this problem with CLAWPACK, originally using a uniform single grid of 275×175 cells. The computational domain modeled a subregion of the tank, $(x, y) \in [0, 5\text{m}] \times [0, 3.5\text{m}]$, which includes an inlet region that experienced large wave runup in the tsunami. We used the provided data that specifies the measured surface elevation of an incoming wave along one side of a subregion of the tank, $x = 0$, for the period $t = [0, 22.5\text{s}]$. (Note: the computational domain is rotated in the figures to match the orientation of the movie, so x runs upward and y from left to right.)

The computed surface elevation is shown at various times in Figures 3 through 6. Figure 3 shows a three-dimensional view of the computed solution at one time, Figure 4 shows the solution during the primary runup period, and Figure 5 compares the solution with several close-up snapshots from the wave tank movie, demonstrating similarities in the zone where the maximum runup occurred.

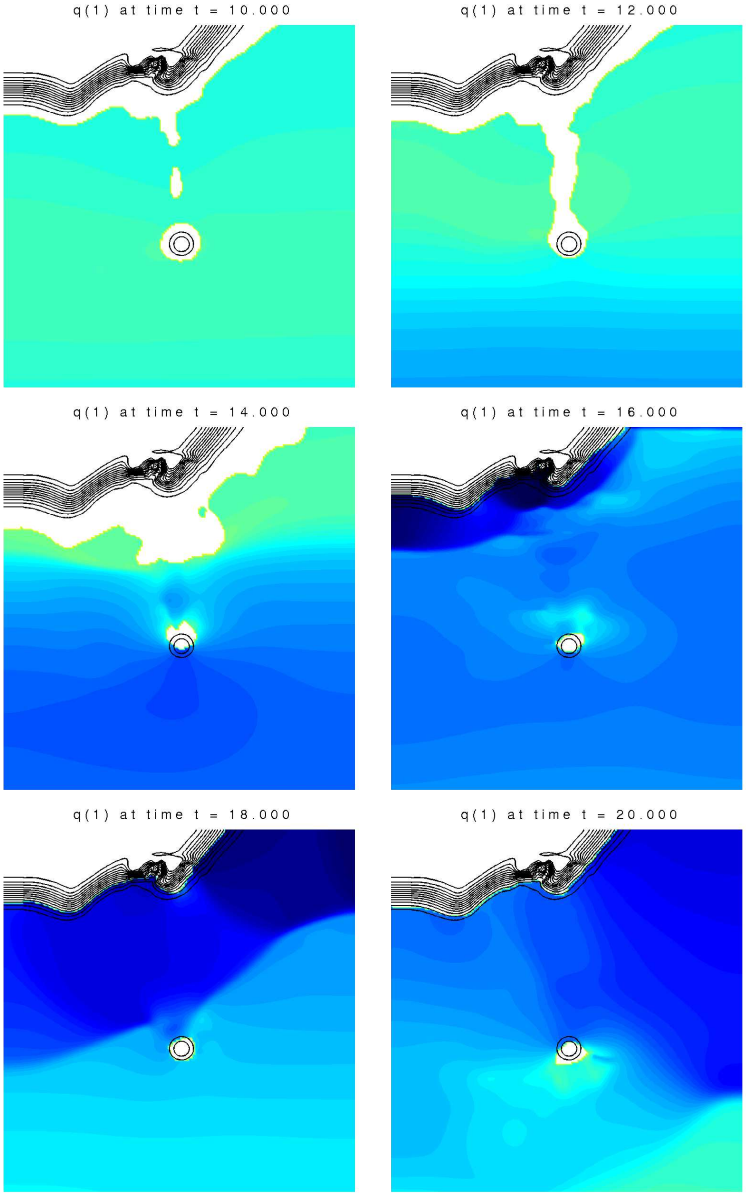


Fig. 4. The computed solution during the primary runup of the wave. The primary runup occurred in the first 30 seconds. Contour lines show the topography that was initially above the water surface, including an island. Gray scale indicates elevation above the original water surface.

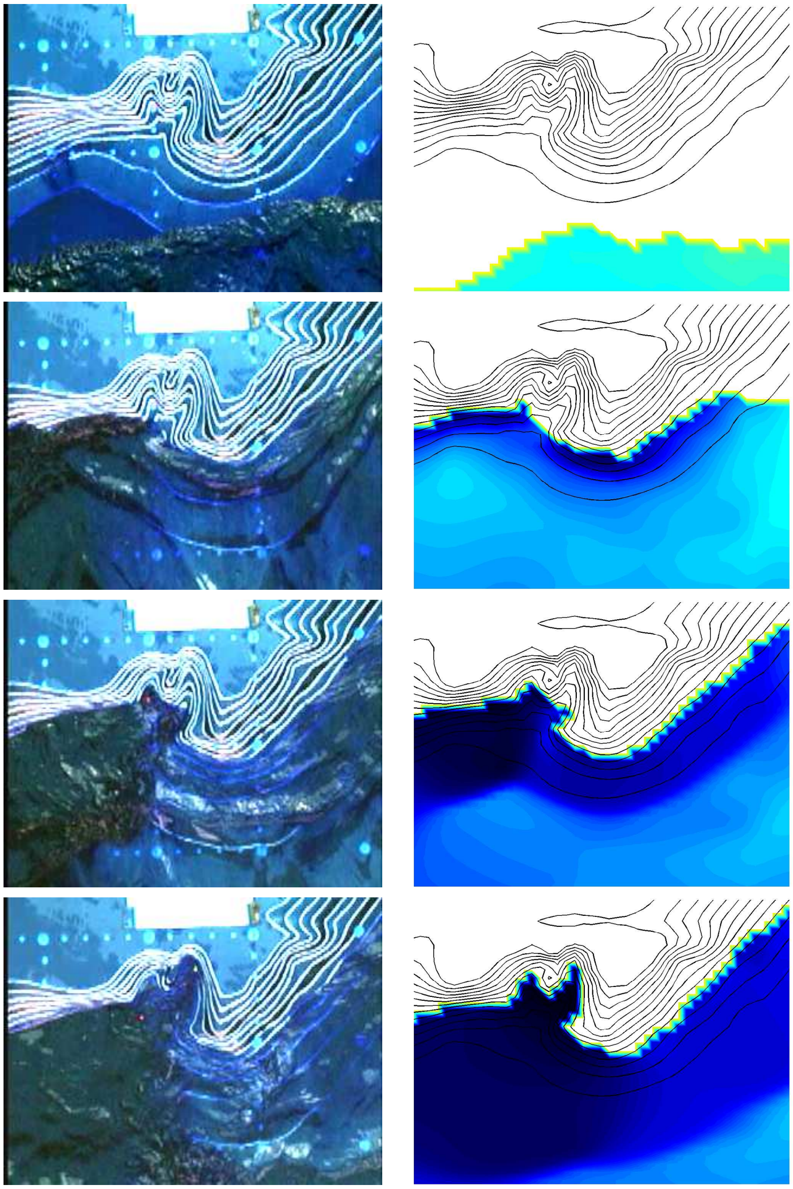


Fig. 5. Comparison of the numerical solution (right) with snapshots from the overhead movie of the laboratory experiment (left). Frames 11, 26, 41, 56 from the movie are shown and computed results are shown at corresponding 30 second intervals. (The movie shows 30 frames per second.)

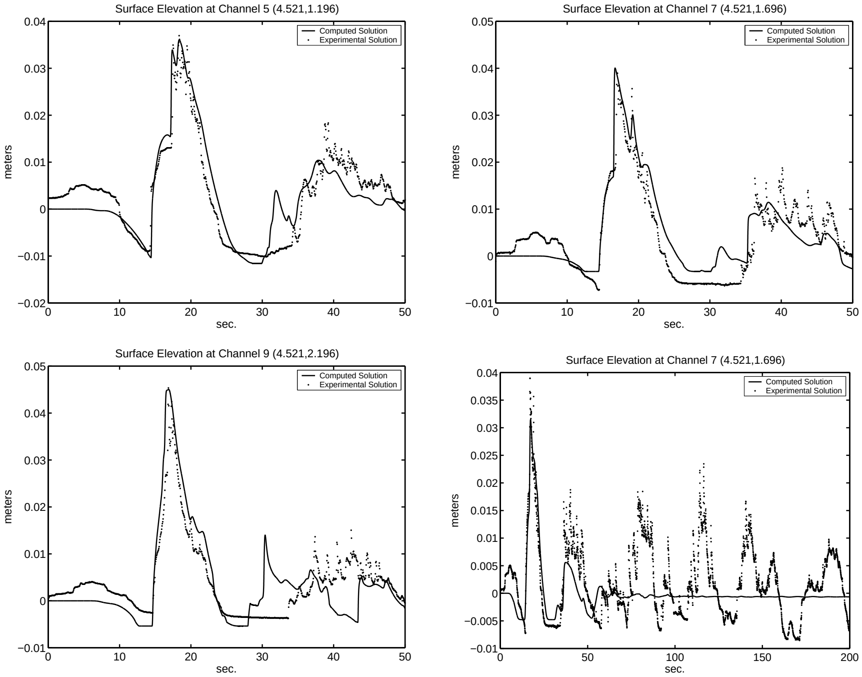


Fig. 6. Comparison of the surface elevation, with the laboratory measurements at the three wave gages (Channels 5, 7, 9). The numerical solution is comparable during the primary runup in the first 50 seconds. The bottom right figure shows the comparison for the duration of the laboratory measurements, at one of the gages (Channel 5). Sustained oscillations in the laboratory measurements are not seen in the numerical solution, as discussed in the text. The other gages showed similar patterns at later times.

Laboratory measurements at three gage locations were provided for the workshop, denoted Channel 5, 7, and 9. This data and the numerical solution (surface elevation as a function of time) at these three locations are shown in Figure 6. The numerical solution was comparable to the laboratory measurements for the first 50 seconds of the experiment, during which the primary runup of the wave and several reflected waves are seen. For the last 150 seconds of the experiment, the wave tank measurements exhibited large oscillations that were not evident in the numerical solution. This discrepancy is shown in the last graph of Figure 6. Based on discussions at the workshop, it is believed that this is due to the fact that the wave maker in the experiment did not generate a perfectly clean wave. The incoming wave data provided for the benchmark problem was only over 22.5 seconds, and was followed by other waves in the wave tank for which no data was provided.

9. Adaptive mesh refinement and large-scale tsunami propagation

In two-dimensional calculations over large spatial domains it may be crucial to use nonuniform grids to achieve the desired resolution in some regions without an excessive number of grid cells overall. For tsunami modeling there are two types of nonuniformity that may be desirable. First, we may want to have a finer grid near the shore where the details of the runup must be computed, or in other regions where the bathymetry has a large impact on the wave propagation behavior and eventual run up. In principle this can be achieved with "static refinement" of some spatial regions relative to others, though ideally computations on the finer grid would only be done at times when the wave is present. Second, we may want to use an adaptively refined grid in which the region of refinement moves along with the tsunami to provide good resolution of the wave at all times, with a minimal number of grid points in regions where nothing is happening.

Since the Catalina workshop took place, the tragic Sumatra tsunami of December 26, 2004 has prompted increased study of the all aspects of tsunamis. Our own efforts in recent months have focused on developing an adaptive mesh refinement code that works well on the scale of the Indian Ocean (or other oceans), with the ability to capture tsunami propagation across the ocean and couple it to the study of run up on much smaller scales along particular stretches of coastline. In the course of this work we have also improved the Riemann solver beyond what is described in this paper to make it even more robust. The new version no longer requires increasing the wave speeds above the Einfeldt speeds, and instead is based on introducing additional waves in the Riemann solution using ideas from LeVeque and Pelanti [17]. This work will be reported in more detail elsewhere. See the webpage [16] for pointers.

The CLAWPACK software includes AMRCLAW, an adaptive mesh refinement version of the code developed with Marsha Berger and described in more detail in Berger and LeVeque[3]. More recently the CLAWPACK formulation has also been incorporated into ChomboClaw by Donna Calhoun at the University of Washington, allowing the Chombo package [5] of C++ routines for adaptive refinement to be applied. This package uses similar algorithms to AMRCLAW but has additional features such as an MPI implementation with load balancing for adaptive refinement on parallel computers, and support for implicit algorithms that may be needed if dissipative or dispersive terms are added to the equations.

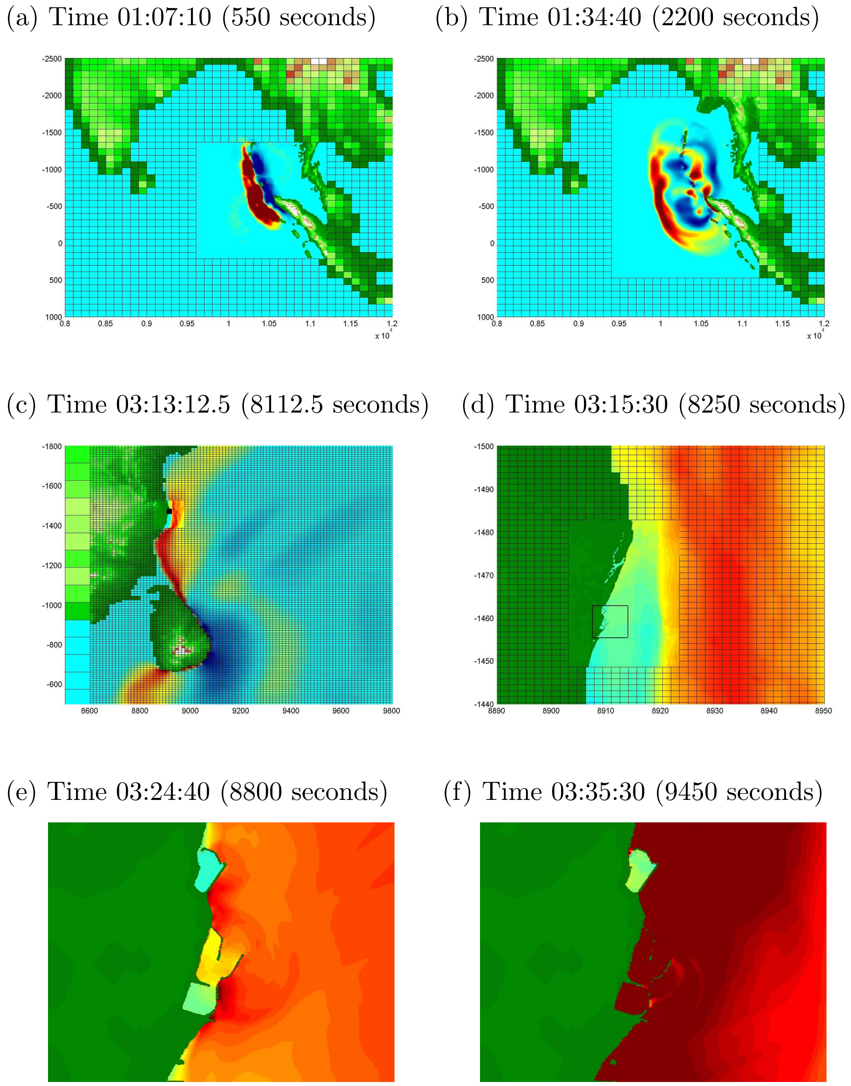
A rectangular grid is refined by covering a portion of the domain by one or more rectangular patches of grids that are finer by a factor of k, some integer. This process can be repeated recursively, with each grid level further refined by grid patches at a higher level or refinement. The maximum number of levels is specified along with the refinement factor at each level. The grid patches are chosen by flagging grid cells at each level that "need refining" (see below) and then clustering these cells into a set of rectangular patches that cover these cells and a limited number of other cells. This is done by solving an optimization problem (using the algorithm of Berger and Rigoutsos[4]) that takes into account the trade-off between refining too many cells unnecessarily and creating too many grid patches, since there is some overhead associated with applying the algorithm on each patch. The boundary data (ghost cell values) on a patch must be generated by space-time interpolation from data on the coarser grid. A time step of length Δt is first taken on the coarse grid, boundary data is then generated from the fine grid patches, and k' time steps of length $\Delta t/k'$ are then taken on the fine grids to reach the same time. For most AMR applications on hyperbolic problems we take $k' = k$, refining in time by the same factor as in space in order to maintain a comparable Courant number at all levels. However, for tsunami propagation and runup applications where we only have the finest grids near the shore, it may be desirable to refine in time by a smaller factor $k' < k$. Recall that the wave speeds are roughly $\sqrt{gh}$, which differs by more than an order of magnitude between the deep ocean and the coastal regions. The smaller wave speed near shore allows a larger time step.

After updating to the same time on the fine grids, the coarse grid solution on any grid cell covered by the finer grid is then replaced by the average of the fine grid values in this cell in order to transfer the more accurate solution to the coarse grid. Additional modifications to the values are needed near the patch edge to maintain global conservation as described in Berger and LeVeque[3]. This is done recursively at each level. While the inner workings are somewhat complicated, the general formulation in AMRCLAW allows extension of most CLAWPACK computations directly to adaptively refined grids. The computational time required for the overhead associated with multiple grids is often negligible compared to the savings achieved by concentrating fine grids only where needed.

We can flag the cells that "need refinement" however we wish. For the calculation presented below, we have allowed some refinement everywhere based on a measure of the variation in the solution, so that the propagating wave is well resolved. In addition, we allow additional levels of refinement near the shore in particular regions of interest, though only once the wave is approaching. Regridding is performed every few time steps to modify the region of refinement as the wave propagates.

In the course of this work, we ran into several difficulties at the interfaces between grids that have not been observed in other applications of the AMR codes. These arise from the representation of the bathymetry and shore on a Cartesian grid. One problem, for example, is that a coarse grid cell that is dry (if the bathymetry value in this cell is above sea level) may be refined into some cells that are above sea level and others below. Even though $h = 0$ on the coarse cell we cannot initialize h to zero on all the fine cells without generating nonphysical wave motion. The fine cells below sea level must be initialized with a positive h to maintain the constant sea level. This problem and related difficulties could only be solved by some substantial reworking of the AMRCLAW code. The result is a special-purpose program that incorporates these algorithmic modifications and can now be applied to many tsunami scenarios. It is currently being tested by comparing predictions with measurements made at various places around the Indian Ocean in the wake of the 26 December 2004 Sumatra earthquake. Some preliminary results are shown in Figure 7, and additional results and movies are linked from the webpage [16] and will be reported more fully in future publications.

The top two frames of Figure 7 show the Bay of Bengal at two early times. A coarse grid is used where nothing is yet happening and the grid cells are shown on this "Level 1 grid", which has a mesh width of one degree (approximately 111 km). The rectangular region where no grid lines are shown is a Level 2 grid with mesh width 8 times smaller, about 14 km. Red represents water elevation above sea level, dark blue is below the undisturbed surface. Figure (c) shows a zoomed view of the southern tip of India and Sri Lanka at a later time. The original Level 1 grid is still visible along the left-most edge, but the rest of the region shown has been refined by a Level 2 grid. Due north from Sri Lanka, along the coast of India, there is a region near Chennai where two additional levels of refinement have been allowed at this stage in the computation. The grid lines on Level 3 are not shown; the mesh width on this level is about 1.7 km, a factor of 8 finer than Level 2. A Level 4 grid is also visible in the center of this region and appears as a small black rectangle.


(a) Time 01:07:10 (550 seconds)

(b) Time 01:34:40 (2200 seconds)

(c) Time 03:13:12.5 (8112.5 seconds)

(d) Time 03:15:30 (8250 seconds)

(e) Time 03:24:40 (8800 seconds)

(f) Time 03:35:30 (9450 seconds)

Fig. 7. Propagation of the 26 December 2004 tsunami across the Indian Ocean, using adaptive mesh refinement with refinement by a factor of 4096 from the coarsest grid shown in Figures (a)–(b) to the finest grid shown in Figures (d)–(f). The latter figures show zoomed views of the region near the harbors of Chennai, India. Times are GMT (and seconds since initial rupture at 0:58 GMT). The earthquake source used in this computation was obtained from the Caltech Seismological Laboratory [1]. See the text for more details.

Figure (d) shows a further zoomed view of the coast near Chennai. In this figure the grid lines show the Level 3 grid. The Level 4 grid is refined by a factor of 64 relative to the Level 3 grid, so the mesh width is about 27 meters. Grid lines on this finest level are not shown. The rectangle in this figure shows a region that is expanded in Figures (e) and (f) to show the two harbors of Chennai. The fine-scale bathymetry used in this computation was obtained by digitizing navigational charts. In this simulation the commercial harbor to the south is inundated, with the tsunami overtopping the surrounding sea wall, while the fishing harbor to the north is largely untouched. This appears to agree with what was actually observed, although we are still investigating this. Moreover, we do not yet have sufficiently accurate data on the height of the sea walls enclosing each harbor. One data point known accurately from tide gage data is the arrival time for the initial wave in Chennai, which was at 9:05 local time (3:35 GMT). This is well matched by our simulation: this is essentially the time shown in Figure 7(f).

A more careful study of this region will be performed in the future and presented elsewhere, including comparisons with runup data collected by Harry Yeh as part of the Tsunami Survey Team that visited this area in February, 2005. See the webpage [16] for more recent results and movies of the simulations. In the future we plan to also compare with data collected by other teams at various other points around the Indian Ocean. We will also make our computer code available to the community for others to use, in a form that should allow the local bathymetry from other regions to be easily incorporated. See the webpage for more details.

Adaptive mesh refinement is crucial for this simulation. The calculation shown here was run on a single-processor 3.2 GHz PC under linux. Figure (c) was obtained after about 15 minutes of running time, Figure (d) about 2.5 hours later, indicating that most of the grid cells are concentrated on the Level 4 grid, which is introduced only when the wave approaches the shore in this one region. Were it possible to use the finest grid over the entire domain, the result of Figure (c) would require more than 4000 years of computing on the same machine.

10. Conclusions and future work

Finite volume methods using an approximate Riemann solver devised to deal with bathymetry and dry states have been successfully used for Benchmark Problems 1 and 2. Benchmark problems 3 and 4 at the workshop model landslide-induced tsunamis, and involve bathymetry that changes with time. In principle our code can handle this situation but we have not yet done extensive tests of this.

Adaptive mesh refinement is crucial for large scale problems and we are further developing and testing the AMR version of our method. We are also working on formulating this method on the full earth, by switching to latitude-longitude coordinates on the sphere and calculating on a computational rectangle in these coordinates using periodic boundary conditions in longitude. The AMR code is already capable of handling this situation and expect to soon be able to study the global propagation and compare with tide gage data (see, for example, Titov et. al. [19]) that is available from many points around the world following the 26 December 2004 event.

11. Acknowledgments

This work was supported in part by NSF grants DMS-0106511 and CMS-0245206, and by DOE grant DE-FC02-01ER25474. The authors would like to thank Marsha Berger and Donna Calhoun for assistance with the adaptive refinement aspect of this work, and researchers at the Caltech Seismological Laboratory for providing the source data for the Indian Ocean simulation.

References

1. C. J. Ammon, C. Ji, H.-K. Thio, D. Robinson, S. Ni, V. Hjorleifsdottir, H. Kanamori, T. Lay, S. Das, D. Helmberger, G. Ichinose, J. Polet, and D. Wald, *Rupture process of the 2004 sumatra-andaman earthquake*, Science, 308 (2005), pp. 1133–1139.
2. D. Bale, R. J. LeVeque, S. Mitran, and J. A. Rossmanith, *A wave-propagation method for conservation laws and balance laws with spatially varying flux functions*, SIAM J. Sci. Comput., 24 (2002), pp. 955–978.
3. M. J. Berger and R. J. LeVeque, *Adaptive mesh refinement using wave-propagation algorithms for hyperbolic systems*, SIAM J. Numer. Anal., 35 (1998), pp. 2298–2316.
4. M. J. Berger and I. Rigoutsos, *An algorithm for point clustering and grid generation*, IEEE Trans. Sys. Man & Cyber., 21 (1991), pp. 1278–1286.
5. P. Colella et al., CHOMBO *software*. `http://seesar.lbl.gov/anag/chombo/`, 2005.
6. B. Einfeldt, *On Godunov-type methods for gas dynamics*, SIAM J. Num. Anal., 25 (1988), pp. 294–318.
7. B. Einfeldt, C. D. Munz, P. L. Roe, and B. Sjogreen, *On Godunov type methods near low densities*, J. Comput. Phys., 92 (1991), pp. 273–295.

8. L. Gosse, *A well-balanced flux-vector splitting scheme designed for hyperbolic systems of conservation laws with source terms*, Comput. Math. Appl., 39 (2000), pp. 135–159.
9. J. M. Greenberg, A. Y. LeRoux, R. Baraille, and A. Noussair, *Analysis and approximation of conservation laws with source terms*, SIAM J. Numer. Anal., 34 (1997), pp. 1980–2007.
10. A. Harten, P. D. Lax, and B. van Leer, *On upstream differencing and Godunov-type schemes for hyperbolic conservation laws*, SIAM Review, 25 (1983), pp. 35–61.
11. P. Jenny and B. Müller, *Rankine-Hugoniot-Riemann solver considering source terms and multidimensional effects*, J. Comp. Phys., 145 (1998), pp. 575–610.
12. A. Kurganov and D. Levy, *Central-upwind schemes for the Saint-Venant system*, Math. Model. and Numer. Anal., 36 (2002), pp. 397–425.
13. R. J. LeVeque, *Wave propagation algorithms for multi-dimensional hyperbolic systems*, J. Comput. Phys., 131 (1997), pp. 327–353.
14. R. J. LeVeque, *Balancing source terms and flux gradients in high-resolution Godunov methods: The quasi-steady wave-propagation algorithm*, J. Comput. Phys., 146 (1998), pp. 346–365.
15. R. J. LeVeque, *Finite Volume Methods for Hyperbolic Problems*, Cambridge University Press, 2002.
16. R. J. LeVeque and D. L. George . `http://www.amath.washington.edu/~rjl/catalina04/`.
17. R. J. LeVeque and M. Pelanti, *A class of approximate Riemann solvers and their relation to relaxation schemes*, J. Comput. Phys., 172 (2001), pp. 572–591.
18. P. L. Roe, *Approximate Riemann solvers, parameter vectors, and difference schemes*, J. Comput. Phys., 43 (1981), pp. 357–372.
19. V. Titov, A. B. Rabinovich, H. O. Mofjeld, R. E. Thomson, and F. I. Gonzalez, *The global reach of the 26 December 2004 Sumatra tsunami*, Science, 309 (2005), pp. 2045–2048.
20. E. F. Toro, *Shock-Capturing Methods for Free-Surface Shallow Flows*, Wiley and Sons Ltd., UK, 2001.