Augmented approximate Riemann solvers for the shallow water equations with varying bathymetry

Xin Chen
Jacob Ortega-Gingrich

Abstract

We describe an augmented Riemann solver for the one-dimensional shallow water equations with variable topography suggested by David George which addresses a number of the needs of applications involving flooding and small perturbations of delicate steady states. Fluid flows over varying topography, for example, add a source term which the numerical method must balance with jumps in momentum and depth in order to preserve delicate steady states, such as an ocean at rest. Furthermore, applications involving flooding, such as the modeling of tsunami inundation, require a Riemann solver that can handle dry cells and preserve depth non-negativity. The augmented solver herein discussed satisfies these properties by amalgamating various aspects of existing solvers such as the HLLE solvers and f-wave approaches and the addition of a stationary steady state wave to account for the source term. Additionally, we discuss some specific techniques which may be use to handle various challenging situations which may arise in a model such as the handling of steep shorelines. Finally, we discuss an implementation of this augmented Riemann solver for the one-dimensional shallow water equations and present a few numerical demonstrations.

1 Introduction

The shallow water equations are a system of nonlinear partial differential equations derived from integrating the Navier-Stokes equations over depth. They are useful for applications involving flows with little vertical variation from hydrostatic equilibrium, such as waves whose length scales greatly exceed the depths of the fluid (for example, in the modeling of tsunami waves). In one spatial dimension with varying bathymetry, the equations take the form

$$h_t + (hu)_x = 0$$
$$(hu)_t + \left(hu^2 + \frac{1}{2}gh^2\right)_x = -ghb_x$$

where $h(x,t)$ is the depth of the water (from the free surface to the sea floor), $u(x,t)$ is the velocity of the water and $b(x)$ is the fixed underwater topography. As these equations take the form of a hyperbolic conservation law (with a source term for the transfer of momentum between the fluid and the sea floor), a finite volume-based approach is natural for many applications. Central to such an approach is the solution of a large number of Riemann problems at every grid cell interface at each time step. Although analytic solutions to arbitrary Riemann problems with the shallow water equations are possible to obtain, the use of exact solutions at each cell interface is impractical and some sort of approximate solver must be used.

Of course, such a solver must be constructed in such a way as to possess certain prescribed properties demanded by the application. In this paper, we focus primarily on the demands which arise from the modeling of tsunami waves, both in terms of their propagation over an open ocean and their inundation. In order to allow for effective inundation modeling, we clearly require a Riemann solver that can handle dry states. For one thing, the Riemann solver must be positive semidefinite so as to disallow cells with negative depths; however, the inclusion of dry cells is not quite so simple as simply using a method which guarantees non-negative depths. Special precautions must also be taken at grid interfaces along shorelines, especially particularly steep ones, to prevent the appearance of spurious oscillations. Additionally, since tsunami waves are small perturbations of an ocean at rest over variable bathymetry, it is critical that the Riemann solver be able to preserve steady states involving a flat surface even with large variations in the underlying topography. We will discuss a few individual approaches which may be used to ensure that the resulting numerical method satisfies these properties.

1.1 The wave propagation algorithm

At the core of the finite volume methods with which the Riemann solvers herein discussed can be used is a wave propagation algorithm of some sort. For a first order (upwind) method, this updates cell values according to the formula

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

Note that we may improve the formal accuracy of the method to second order by adding in flux correction terms, as discussed in [3]; however, in the interest of conserving valuable space in this paper, we will mainly focus on the challenges of implementing Riemann solvers with the first order method, noting some of the adjustments which may need to be made to maintain stability when including the correction terms. There are a few standard ways in which the first order fluctuations $A^{\pm}\Delta Q_{i-\frac{1}{2}}$ may be determined. In a wave propagation-based approach, they are computed by decomposing the jump at the cell interface into a sum of M_w waves

$$Q_i^n - Q_{i-1}^n = \sum_{p=1}^{M_w} \mathcal{W}_{i-\frac{1}{2}}^p$$

which propagate from the discontinuity at speeds $s_{i-\frac{1}{2}}^p$. The fluctuations then used to update the cell values are

$$\mathcal{A}^-\Delta Q_{i-\frac{1}{2}} = \sum_{\{p:s_{i-\frac{1}{2}}^p<0\}} s_{i-\frac{1}{2}}^p \mathcal{W}_{i-\frac{1}{2}}^p, \qquad \mathcal{A}^+\Delta Q_{i-\frac{1}{2}} = \sum_{\{p:s_{i-\frac{1}{2}}^p>0\}} s_{i-\frac{1}{2}}^p \mathcal{W}_{i-\frac{1}{2}}^p.$$

An alternate formulation of the fluctuations can be obtained by decomposing the jump in fluxes across the cell interface rather than the jump in Q. Doing so, we have

$$f(Q_i) - f(Q_{i-1}) = \sum_{p=1}^{M_w} \mathcal{Z}_{i-\frac{1}{2}}^p$$

and we may define our updating fluctuations as

$$\mathcal{A}^-\Delta Q_{i-1/2} = \sum_{\{p:s_{i-1/2}^p<0\}} \mathcal{Z}_{i-1/2}^p, \qquad \mathcal{A}^+\Delta Q_{i-1/2} = \sum_{\{p:s_{i-1/2}^p>0\}} \mathcal{Z}_{i-1/2}^p \tag{1}$$

This approach, which is known as the f-wave approach, shall be the one with which we incorporate the Riemann solver to be discussed below.

1.2 The Roe solver

In order to implement the wave propagation algorithms above described, we must decide on a basis of vectors $\mathcal{Z}^p_{i-\frac{1}{2}}$ at each grid interface into which to decompose the flux jump. Ideally, we wish for the waves propagating from the cell interface to reflect the sorts of waves which we would expect to observe in an actual Riemann problem. Although it would be possible to compute an exact solution to the Riemann problem at each cell interface, it is much more time efficient to use a linearized approximate solver. The Roe solver is one such solver based on a linearization about a specific mean state $(\bar{h}, \hat{u})$ where $\bar{h}$ is the arithmetic mean of the depths on either side of the interface and

$$\hat{u} = \frac{\sqrt{h_{i-1}}u_{i-1} + \sqrt{h_i}u_i}{\sqrt{h_{i-1}} + \sqrt{h_i}}. \tag{2}$$

The eigenvalues (known as the Roe wave speeds) of the linearized Jacobian $\hat{A}$ are $\hat{\lambda}^1 = \hat{u} - \hat{c}$, $\hat{\lambda}^2 = \hat{u} + \hat{c}$ where $\hat{c} = \sqrt{g\bar{h}}$. The corresponding eigenvectors are

$$\hat{r}^1 = \begin{pmatrix} 1 \\ \hat{u} - \hat{c} \end{pmatrix}, \hat{r}^2 = \begin{pmatrix} 1 \\ \hat{u} + \hat{c} \end{pmatrix}. \tag{3}$$

In a pure implementation of the Roe solver, we would choose $\hat{r}^p$ as the basis of vectors into which to decompose the flux jump, selecting corresponding wave speeds $s^p = \hat{\lambda}^p$. The result is a very efficient method which satisfies the so-called Roe condition, thus ensuring that if two neighboring states are such that they would be connected by a single shock discontinuity, the Roe solver will model that shock with a single (linear) discontinuity propagating at the correct speed. Even in cases involving small rarefaction waves, the Roe solver is effective, modeling each fan with a shock discontinuity. Problems arise, however, in cases involving transonic rarefaction fans which spread into both neighboring cells.

2 The HLLE solver and depth non-negativity

While efficient and very effective in cases of flat bathymetry, provided that there is no strong transonic rarefaction, the pure Roe Solver in itself does not quite meet our needs. As a first step, we wish to ensure depth non-negativity by using a modified version of the Roe method known as the HLLE method. Like the Roe solver, the HLLE solver uses two waves to approximate the solution to the Riemann problem. The waves themselves are very similar to the waves obtained from an eigenvalue decomposition of the Roe matrix, but modified so as to prevent outflow so fast as to allow the depth in any cell to become negative. The wave speeds, which are known as the Einfeldt speeds, are given by

$$\check{s}^1_{i-\frac{1}{2}} = \min(\lambda^-(Q^n_{i-1}), \hat{\lambda}^-_{i-\frac{1}{2}}), \quad \check{s}^2_{i-\frac{1}{2}} = \max(\lambda^+(Q^n_i), \hat{\lambda}^+_{i-\frac{1}{2}}) \tag{4}$$

where $\lambda^\pm(Q)$ are the eigenvalues of the Jacobian matrix of the flux function evaluated at Q. The corresponding wave basis vectors (which are no longer necessarily eigenvectors of any matrix in particular) are given by $w^p_{i-\frac{1}{2}} = (1, \check{s}_{i-\frac{1}{2}^p})^T$. The result is an approximate Riemann solver which preserves depth non-negativity while retaining the efficiency advantages of the Roe solver.

Unfortunately, however, this method continues to fail to capture large transonic rarefactions, as each of the two waves radiating from the discontinuity are still represented each by a single discontinuity, thus requiring some sort of entropy fix. One natural way to implement this is to represent a large rarefaction fan with two discontinuities, which may propagate into different cells in the case of transonic rarefaction. To do this, however, we must decompose our flux difference into more than two waves. Thus, we arrive at the augmented Riemann solver proposed by David George.

3 The augmented solver

Instead of using the standard wave decomposition with only two waves, the most basic augmented Riemann solver adds a third "corrector" wave to the decomposition which may, among other purposes, be used as a natural entropy fix to better represent large potentially transonic rarefactions. In order to achieve a three-wave decomposition, we augment the vector $\Delta Q^n_{i-\frac{1}{2}}$ containing the jumps in depth and momentum with a third component containing $\Delta\varphi$, the jump in momentum flux $\varphi = hu^2 + \frac{1}{2}gh^2$ across the interface. This choice is partly motivated by our observation that ΔHU is already the jump in mass flux, thus suggesting that we may, by including the momentum flux, develop a method which hybridizes the standard wave decomposition and the f-wave approach. Initially keeping with the traditional wave decomposition, we seek a decomposition of the form

$$\begin{bmatrix} H_i - H_{i-1} \\ HU_i - HU_{i-1} \\ \varphi(Q_i) - \varphi(Q_{i-1}) \end{bmatrix} = \sum_{p=1}^{3} \alpha^p_{i-1/2} w^p_{i-1/2}.$$

Although this method approach so far appears akin to a traditional wave decomposition, we implement the f-wave propagation algorithm to update the cell values, defining the flux waves as

$$\mathcal{Z}^p_{i-1/2} = [\mathbf{0}_{2\times 1} \quad \mathbf{I}_{2\times 2}]\, \alpha^p_{i-1/2} w^p_{i-1/2}$$

and updating using the fluctuations given by (1).

Next we need to decide upon basis vectors $w^p_{i-1/2}$ for our decomposition. Desiring to remain as close to the HLLE method as possible (so as to inherit its depth non-negativity property), we choose to include the Einfeldt speeds $\check{s}_{i-\frac{1}{2}^{\pm}}$ and their corresponding in our wave decomposition

$$\left\{w^1_{i-1/2}, s^1_{i-1/2}\right\} = \left\{\left(1, \check{s}^-_{i-1/2}, \left(\check{s}^-_{i-1/2}\right)^2\right)^T, \check{s}^-_{i-1/2}\right\}$$

$$\left\{w^3_{i-1/2}, s^3_{i-1/2}\right\} = \left\{\left(1, \check{s}^+_{i-1/2}, \left(\check{s}^+_{i-1/2}\right)^2\right)^T, \check{s}^+_{i-1/2}\right\}.$$

For the intermediate "corrector" wave, our choice is much less clear cut. For now, we shall make the simplest choice possible of

$$\left\{w^2_{i-1/2}, s^2_{i-1/2}\right\} = \left\{(0,0,1)^T, \frac{1}{2}\left(\check{s}^-_{i-1/2} + \check{s}^+_{i-1/2}\right)\right\},$$

noting that it may be optimal to modify this to better track the edges of large rarefaction fans. We note that with this choice of waves, the depth of each cell is updated only by the outer (HLLE) waves (since the mass flux/momentum component of w^2 is zero), thus in terms of updates to H, this augmented method is no different from the HLLE method. As such, we conclude that this method must retain the depth non-negativity property of the HLLE solver.

3.1 Adding a source term

All of the above discussed methods pertain only to the homogeneous shallow water equations, corresponding to flows over a flat-bottomed basin. In order to use these previously described methods with a nonzero source term $-ghb_x$, we would have to implement some sort of splitting. As this would involve some sort of discretization of the bathymetric gradient b_x, it would be very difficult, if not impossible, to ensure that the momentum flux and source terms balance perfectly in a steady state. However, our formulation of an augmented solver provides us with a natural way to incorporate the source term directly into the Riemann solver itself, preventing any spurious waves which may appear in a stationary fluid over variable topography. The solution is to add yet another component to the decomposition

$$\begin{bmatrix} H_i - H_{i-1} \\ HU_i - HU_{i-1} \\ \varphi(Q_i) - \varphi(Q_{i-1}) \\ B_i - B_{i-1} \end{bmatrix} = \sum_{p=0}^{3} \alpha^p_{i-1/2} w^p_{i-1/2},$$

adding, in the process, a fourth wave w_0 which we hope will capture jumps in H and HU which are present only due to the underlying terrain and not due to any sort of fluid dynamics. Just as in the homogeneous case, we define the flux waves to be used with the wave propagation algorithm as

$$\mathcal{Z}^p_{i-1/2} = \left[\mathbf{0}_{2\times 1} \quad \mathbf{I}_{2\times 2} \quad \mathbf{0}_{2\times 1}\right] \alpha^p_{i-1/2} w^p_{i-1/2}$$

We are now left with the task of choosing a form for the fourth wave w_0 which suits our purposes. For motivation, we consider an augmented (and homogeneous) reformulation of the (originally inhomogeneous) shallow water system $\tilde{q}_t + W(\tilde{q})\,\tilde{q}_x = 0$ with

$$\begin{aligned} \tilde{q} &= (h, hu, \varphi, b)^T \\ W(\tilde{q}) &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ -u^2 + gh & 2u & 0 & gh \\ 0 & -u^2 + gh & 2u & 2ugh \\ 0 & 0 & 0 & 0 \end{bmatrix}. \end{aligned}$$

The eigenpairs of the Jacobian matrix in this augmented system are

$$\begin{aligned} \{r^0(\tilde{q}), \lambda^0(\tilde{q})\} &= \left\{ \left(\frac{gh}{\lambda^1(\tilde{q})\lambda^3(\tilde{q})}, 0, -gh, 1 \right)^T, 0 \right\} \\ \{r^1(\tilde{q}), \lambda^1(\tilde{q})\} &= \left\{ (1, \lambda^1(\tilde{q}), (\lambda^1(\tilde{q}))^2, 0)^T, u - \sqrt{gh} \right\} \\ \{r^2(\tilde{q}), \lambda^2(\tilde{q})\} &= \left\{ (0, 0, 1, 0)^T, 2u \right\} \\ \{r^3(\tilde{q}), \lambda^3(\tilde{q})\} &= \left\{ (1, \lambda^3(\tilde{q}), (\lambda^3(\tilde{q}))^2, 0)^T, u + \sqrt{gh} \right\}. \end{aligned}$$

Noting that the last three eigenpairs bear close resemblance to the waves considered in the augmented solver for the homogeneous equations, we define the waves

$$\left\{w^1_{i-1/2}, s^1_{i-1/2}\right\} = \left\{\left(1, \check{s}^-_{i-1/2}, \left(\check{s}^-_{i-1/2}\right)^2, 0\right)^T, \check{s}^-_{i-1/2}\right\} \tag{5}$$

$$\left\{w^2_{i-1/2}, s^2_{i-1/2}\right\} = \left\{(0,0,1,0)^T, \frac{1}{2}\left(\check{s}^-_{i-1/2} + \check{s}^+_{i-1/2}\right)\right\} \tag{6}$$

$$\left\{w^3_{i-1/2}, s^3_{i-1/2}\right\} = \left\{\left(1, \check{s}^+_{i-1/2}, \left(\check{s}^+_{i-1/2}\right)^2, 0\right)^T, \check{s}^+_{i-1/2}\right\}. \tag{7}$$

Now it remains only to choose the final wave and its speed $\{w^0_{i-1/2}, s^0_{i-1/2}\}$. Since we do not want the new wave, which is supposed to capture variations in depth and momentum consistent with a steady state, to update either cell, it is clear from the beginning that this "steady state" wave will have to be stationary. As such, we see that we must ultimately choose $s^0_{i-\frac{1}{2}} = 0$. But, as we have seen above, we already know that the augmented Jacobian matrix has a zero eigenvalue. Noting that steady state solutions $\tilde{q}$ must satisfy

$$W(\tilde{q})\tilde{q}_x = 0, \tag{8}$$

we see that $\tilde{q}_x$ must be an eigenvector corresponding to this zero eigenvalue. Motivated by this observation, we have a theorem, which is proven by David George and whose proof we shall not reproduce here in the interest of space.

Theorem 1. *Suppose that a smooth steady state solution to the shallow water equations exists between two points x_l and x_r, with $b(x_l) \neq b(x_r)$. If the vector $\tilde{q}(x,t)$ is differenced between x_l and x_r, then the difference must satisfy*

$$\tilde{q}(x_r,t) - \tilde{q}(x_l,t) = (b(x_r) - b(x_l))\begin{bmatrix} \frac{g\bar{H}(q(x_l,t),q(x_r,t))}{\overline{\lambda^+\lambda^-}(q(x_l,t),q(x_r,t))} \\ 0 \\ -g\tilde{H}(q(x_l,t),q(x_r,t)) \\ 1 \end{bmatrix}$$

where

$$\begin{aligned} \overline{\lambda^+\lambda^-}(q(x_l,t),q(x_r,t)) &= \left(\frac{u_l+u_r}{2}\right)^2 - g\left(\frac{h_l+h_r}{2}\right) \\ \bar{H}(q(x_l,t),q(x_r,t)) &= \frac{h_l+h_r}{2} \\ \widetilde{\lambda^+\lambda^-}(q(x_l,t),q(x_r,t)) &= \max(0, u_r u_l) - g\left(\frac{h_l+h_r}{2}\right) \\ \tilde{H}(q(x_l,t),q(x_r,t)) &= \bar{H}(q(x_l,t),q(x_r,t))\frac{\widetilde{\lambda^+\lambda^-}(q(x_l,t),q(x_r,t))}{\overline{\lambda^+\lambda^-}(q(x_l,t),q(x_r,t))}. \end{aligned}$$

This theorem gives us an important condition on the jump between two adjacent cell averages in a steady state solution which we may utilize to select a wave $w^0_{i-\frac{1}{2}}$ to capture precisely these steady state jumps. Doing this, we define

$$w^0_{i-1/2} = \begin{bmatrix} \frac{g\bar{H}(Q_{i-1},Q_i)}{\overline{\lambda^+\lambda^-}(Q_{i-1},Q_i)} \\ 0 \\ -g\tilde{H}(Q_{i-1},Q_i) \\ 1 \end{bmatrix}, \tag{9}$$

which we note takes the form of the relevant eigenvector of some averaged version of the Jacobian matrix W. Thus, we see that including this vector in the decomposition of the augmented jump vector will "absorb" all variation in H which can be attributed to varying topography, thus giving us a well balanced method.

3.2 Practical considerations for problems involving dry cells and shorelines

At this stage, we have a numerical method which is positive semi-definite and well balanced. Nonetheless, in order to achieve our goals of having a Riemann solver which can be used in inundation models, there remain a few difficulties that must be dealt with. As it turns out, using a positive semi-definite method is not all that is necessary for effective handling of dry or near dry states. Because of the potential of small numerical errors in cells where H_i is very small to result in spurious oscillatory instabilities, it is oftentimes best not to attempt to execute the Riemann solver at interfaces between nearly-dry cells. As such, for practical purposes in the implementation of the method here described, we expect that we will have to specify some dryness tolerance, a minimum depth below which the method simply designates a cell as dry and reduces its depth and momentum to zero. David George does not explicitly discuss the use of such a tolerance in his article, but examination of the Riemann solver used by the GeoClaw software reveals that he also used this approach to try to prevent instabilities in cells with depths not much larger than the expected numerical errors. Although we have no theoretical guidelines governing the choice of this tolerance parameter, we note that a tolerance of approximately 10^{-3} meters seems to work fairly well and is, in fact, a value which is oftentimes used with GeoClaw for tsunami modeling.

Although there is no need to run the Riemann solver between two dry cells (as there are no waves traveling across such an interface to update the values of either cell), it is very important for inundation modeling that we be able to handle Riemann problems between neighboring wet and dry cells. Although one might naïvely implement the above described method, the obtained results may prove to be disappointing. Over flat topography, the exact solution to a Riemann problem involving a wet and a dry state consists of a single rarefaction wave with one edge proceeding into the next cell. Considering, for concreteness, a case with a dry state on the right $H_i = 0$, the wet-dry interface (the right edge of the rarefaction fan) travels at speed

$$U_{i-1} + 2\sqrt{gH_{i-1}}.$$

Except in the case of a supercritical flow where $U_{i-1} > \sqrt{gH_{i-1}}$, in which case the rarefaction fan is contained completely in the previously dry right cell, this rarefaction wave is generally transonic and, as a result, should "update" both cells. In order to capture the effects of the rarefaction on both cells, we see that we require two waves at a minimum. An effective way to accomplish this goal would be to modify the Einfeldt speeds suitably so that the primary HLLE-based waves follow the edges of the rarefaction fan. For example, for a dry state state on the right ($H_i = 0$), we might define the modified Einfeldt speeds as

$$\check{s}^-_{i-\frac{1}{2}} = U_{i-1} - \sqrt{gH_{i-1}}, \quad \check{s}^+_{i-\frac{1}{2}} = U_{i-1} + 2\sqrt{gH_{i-1}}$$

and then use these (and their corresponding basis waves) in the decomposition described in the previous section.

In addition to difficulties due to transonic rarefaction at wet-dry interfaces, we find that it is also necessary to make some changes to our method in such situations to deal with sharply varying topography. In situations where the variation in topography across the cell boundary is relatively small compared to the depth of the wet cell, or if the wet cell is at a higher elevation, the Riemann solver with the above-described modified Einfeldt speeds could be used with few problems. More problematic, however, is the case involving a dry cell with much higher topography than surface level of the neighboring wet cell. If the topography is sufficiently tall that any incoming water would not overtop it, the interface is essentially a boundary with a solid wall and should be treated as such; whether the jump in topography is 100 meters or 100 kilometers should have no bearing on the numerical solution. Unfortunately, however, the magnitude of the steady state wave in our augmented decomposition is proportional to the topographical jump, so a naïve execution of the usual Riemann solver in such a case would introduce spurious waves, even with steady state initial data, thus ruining our otherwise well-balanced method. One easy way to address this difficulty is to run a quick test to see if the run up of water against a hypothetical wall at the interface would be high enough to surmount the discontinuity. A decision can then be made, based on this result, whether to treat the interface as a solid wall boundary or, if the run up is sufficiently high, to use the Riemann solver to update both cells. This method requires an estimate to the middle state of the solid wall Riemann problem, an estimate which can easily be obtained from the HLLE middle state

$$\check{H}^* = \frac{HU_L - HU_R + \check{s}^+_{i-1/2} H_R - \check{s}^-_{i-1/2} H_L}{\check{s}^+_{i-1/2} - \check{s}^-_{i-1/2}} \tag{10}$$

using ghost cells for a solid wall boundary. For example, if we are dealing with a dry state on the right with $\Delta b > 0$, we would set $H_R = H_L = H_{i-1}$ with $HU_R = -HU_L = -HU_{i-1}$. Once this estimate is obtained, the usual Riemann solver can then be used with either a solid wall condition (if $H^* < \Delta b$) or with the dry cell data ($H^* > \Delta b$). This technique should preserve the well-balanced nature of the method, as any shoreline with steady state initial data would be treated as a solid wall.

3.3 Transonic rarefactions

In most situations, the augmented Riemann solver would be expected to behave very similarly to the HLLE solver, with the vast majority of the contributions to the updating fluctuations coming from the outer waves $\mathcal{Z}^1$ and $\mathcal{Z}^2$. In these cases, we expect that the intermediate corrector wave $\mathcal{Z}^2$ will be much smaller in magnitude, so our choice of the second basis vector w^1 is not particularly important. As such, we typically choose the simplest, namely $w^2 = (0, 0, 1, 0)^T$, as described in the preceding sections. In certain situations, however, it is preferable to select a different intermediate wave to help track large transonic rarefaction fans which would be expected to update both cells. Specifically, we might choose a corrector wave which follows the edge of a transonic rarefaction bordering the middle state. For a 1-rarefaction, for example, we would choose

$$\{w^2_{i-\frac{1}{2}}, s^2_{i-\frac{1}{2}}\} = \{(1, \lambda^{-*}_{i-\frac{1}{2}}(H^*_{i-\frac{1}{2}}), (\lambda^{-*}_{i-\frac{1}{2}}(H^*_{i-\frac{1}{2}}))^2, 0), \lambda^{+*}_{i-\frac{1}{2}}(H^*_{i-\frac{1}{2}})\}.$$

While this alternate form of the corrector wave does appear to address some of our difficulties with capturing transonic rarefactions, it comes at the cost of potentially interfering with the depth non-negativity of the method. Since the mass flux component of w^2 in this formulation is non-zero, the corrector wave will result in updates to the depth the the cell into which it travels. Thus, the depth is no longer strictly updated as in the HLLE method, thereby removing any guarantee of positive semi-definiteness. To combat this possibility, we must take care to use this corrector wave form only when we are working with cells of sufficient depth. Note that this is precisely why we use modified HLLE waves instead of the corrector wave to help with transonic rarefaction at a wet-dry interface, as described in the previous section.

3.4 Second order corrections

At this point, a few very brief statements should be made pertaining to the use of this Riemann solver with a high-resolution second-order limited method. Once the f-waves $\mathcal{Z}^p_{i-\frac{1}{2}}$ have been computed, second order correction fluxes may then be calculated simply as

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

where $\tilde{\mathcal{Z}}^p_{i-\frac{1}{2}}$ are some sort of limited for form of the f-waves. Overall, the inclusion of these second order correction terms with limiters does not greatly affect the Riemann solver or its properties. Since the steady state wave is stationary ($s^0_{i-\frac{1}{2}} = 0$), it does not contribute anything to the correction terms either, so the inclusion of these correction fluxes will not have any impact on the well-balancedness of the method. It is, however, possible that the correction terms may cause spurious negative depth in certain contrived circumstances. This occurrence can be prevented by the application of some sort of additional limiter which would ensure positive semi-definiteness; however, this is beyond the scope of the present paper.

4 Numerical Experiments

We shall now describe a simple one-dimensional implementation of the Riemann solver described above with an f-wave update algorithm both with and without second order corrections. The source code of this implementation, which takes the form of a MATLAB program, is included in the group GitHub repository with the final submission of this assignment. The code used herein is heavily based on David George's two dimensional Riemann solver, which is the basis of the GeoClaw software, as well as an unreleased one-dimensional version provided to us by Dr. George himself. It should be noted that while most of the functionality of the above-described numerical method is implemented in this program, due to the shortness of the available time, we have neglected to include a few features. Most notably, we have not included any additional limiting scheme to prevent second order terms from instigating negative depth, instead manually setting any spurious negative depths which may arise in an update cycle to zero. This does, unfortunately, cause some spurious oscillations, especially in cases where we have discontinuities in very shallow water and limits the situations in which we may effectively use the second order method.

Figures 1 and 2 contain plots of two test problems in which this method is used. The test illustrated in figure 1 involves an evenly-sloping beach with a initially Gaussian wave washing up onto it. Extrapolation (non-reflecting) boundary conditions are used at both sides, ensuring that once the waves have departed from the domain, the water stabilizes into a steady state. This is a useful test in that it illustrates both the well-balanced nature of the method (in that it is able not only to maintain but to approach a steady state after the wave has washed up onto the beach) and the effectiveness of the handling of wet-dry interface Riemann problems. We see the appearance of no spurious oscillations in the vicinity of the moving interface, indicating that the "artificial wall" scheme described in section 3.2 is effective in ensuring that the augmented solver is only used at the interface if the wave is traveling sufficiently rapidly as to surmount the jump in topography. Although this problem involves very shallow water, we have no issues implementing a second order method with the monotonized-centers (MC) limiter, likely because of the smoothness of the wave profile through the experiment. Figure 2 contains plots demonstrating a much more complicated situation involving a dam break problem with a wave washing up onto a hillside with

Figure 1: A wave running up onto a beach.

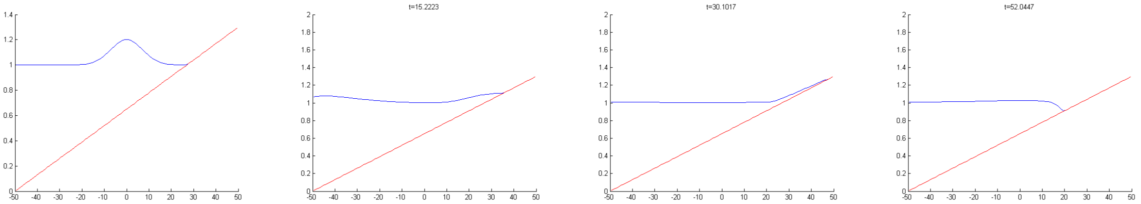
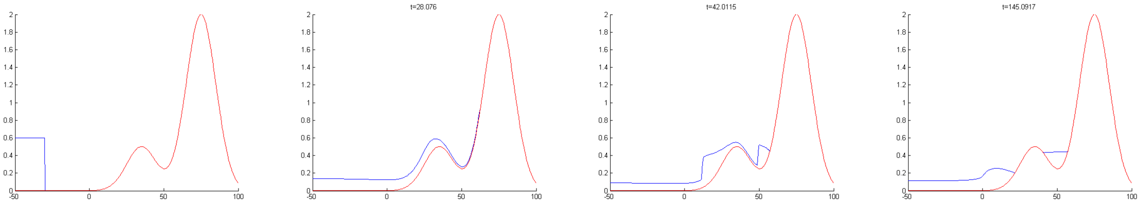


Figure 2: Dam break problem with run up onto a hill with a valley.



a small valley, trapping some water in the depression. For this problem, we encountered great deal of unwanted behavior when we attempted to implement the second order method, behavior which did not appear to be present when omitting the correction terms (as done to produce this figure). We suspect that these difficulties could probably be fixed, possibly with the implementation of the additional limiters described by David George, but did not have time to explore this possible avenue. Overall, however, even the first order method illustrates the effectiveness of the Riemann solver in handling the difficulties of flows over dry and varied terrain.

5 Conclusions

We have described and discussed an augmented Riemann solver developed by David George which addresses a great many of the difficulties in the modeling of fluid flows over varying terrain and the handling of dry states. The method is well-balanced in the sense that it can maintain very delicate steady states involving balances between a source term containing a topographical gradient and a flux gradient, something which would not generally be possible treating the source term as one normally might using a split method. Furthermore, this solver proves very effective at handling wet-dry interfaces, ensuring that spurious oscillations do not appear in cases cases steeply sloping terrain. Furthermore, the method provides a natural entropy fix useful for dealing with transonic rarefactions. Finally, so long as care is taken not to use this entropy fix in cases involving very shallow depths and an additional limiting scheme is used with second order methods, the method inherits the property of positive semi-definiteness from the underlying HLLE method.

References

[1] D.L. George, Finite Volume Methods and Adaptive Refinement for Tsunami Propagation and Inundation, Ph.D. Thesis, University of Washington, Seattle, WA, 2006

[2] D.L. George, Augmented Riemann solvers for the shallow water equations over variable topography with steady states and inundation, Journal of Computational Physics, 227 (2008) 3089-3113.

[3] R.J. LeVeque, Finite Volume Methods for Hyperbolic Problems. Texts in Applied Mathematics, Cambridge University Press, Cambridge, 2002.