Acta Numerica (2011), pp. 211–289
doi:10.1017/S0962492911000043

Tsunami modelling with adaptively refined finite volume methods*

Randall J. LeVeque
Department of Applied Mathematics,
University of Washington, Seattle, WA 98195-2420, USA
E-mail: rjl@uw.edu

David L. George
US Geological Survey, Cascades Volcano Observatory,
Vancouver, WA 98683, USA
E-mail: dgeorge@usgs.gov

Marsha J. Berger
Courant Institute of Mathematical Sciences,
New York University, NY 10012, USA
E-mail: berger@cims.nyu.edu

Numerical modelling of transoceanic tsunami propagation, together with the detailed modelling of inundation of small-scale coastal regions, poses a number of algorithmic challenges. The depth-averaged shallow water equations can be used to reduce this to a time-dependent problem in two space dimensions, but even so it is crucial to use adaptive mesh refinement in order to efficiently handle the vast differences in spatial scales. This must be done in a 'well-balanced' manner that accurately captures very small perturbations to the steady state of the ocean at rest. Inundation can be modelled by allowing cells to dynamically change from dry to wet, but this must also be done carefully near refinement boundaries. We discuss these issues in the context of Riemann-solver-based finite volume methods for tsunami modelling. Several examples are presented using the GeoClaw software, and sample codes are available to accompany the paper. The techniques discussed also apply to a variety of other geophysical flows.

* Colour online for monochrome figures available at journals.cambridge.org/anu.

CONTENTS

1. Introduction

Many fluid flow or wave propagation problems in geophysics can be modelled with two-dimensional depth-averaged equations, of which the *shallow water equations* are the simplest example. In this paper we focus primarily on the problem of modelling tsunamis propagating across an ocean and inundating coastal regions, but a number of related applications have also been tackled with depth-averaged approaches, such as storm surges arising from hurricanes or typhoons; sediment transport and coastal morphology; river flows and flooding; failures of dams, levees or weirs; tidal motions and internal waves; glaciers and ice flows; pyroclastic or lava flows; landslides, debris flows, and avalanches.

These problems often share the following features.

- The governing equations are a nonlinear hyperbolic system of conservation laws, usually with source terms (sometimes called *balance laws*).
- The flow takes place over complex topography or bathymetry (the term used for topography below sea level).
- The flow is of bounded extent: the depth goes to zero at the margins or shoreline and the 'wet–dry interface' is a moving boundary that must be captured as part of the flow.
- There exist non-trivial steady states (such as a body of water at rest) that should be maintained exactly. Often the wave propagation or flow to be modelled is a small perturbation of this steady state.
- There are multiple scales in space and/or time, requiring adaptively refined grids in order to efficiently simulate the full problem, even when

two-dimensional depth-averaged equations are used. For geophysical problems it may be necessary to refine each spatial dimension by five orders of magnitude or more in some regions compared to the grid used on the full domain.

Transoceanic tsunami modelling provides an excellent case study to explore the computational difficulties inherent in these problems. The goal of this paper is to discuss these challenges and present a set of computational techniques to deal with them. Specifically, we will describe the methods that are implemented in GeoClaw, open source software for solving this class of problems that is distributed as part of Clawpack (www3). The main focus is not on this specific software, however, but on general algorithmic ideas that may also be useful in the context of other finite volume methods, and also to problems outside the domain of geophysical flows that exhibit similar computational difficulties. For the interested reader, the software itself is described in more detail in Berger, George, LeVeque and Mandli (2010) and in the GeoClaw documentation (www7).

We will also survey some uses of tsunami modelling and a few of the challenges that remain in developing this field, and geophysical flow modelling more generally. This is a rich source of computational and modelling problems with applicability to better understanding a variety of hazards throughout the world.

The two-dimensional shallow water equations generally provide a good model for tsunamis (as discussed further below), but even so it is essential to use *adaptive mesh refinement* (AMR) in order to efficiently compute accurate solutions. At specific locations along the coast it may be necessary to model small-scale features of the bathymetry as well as levees, sea walls, or buildings on the scale of metres. Modelling the entire ocean with this resolution is clearly both impossible and unnecessary for a tsunami that may have originated thousands of kilometres away. In fact, the wavelength of a tsunami in the ocean may be 100 km or more, so that even in the region around the wave a resolution on the scale of several km is appropriate. In undisturbed regions of the ocean even larger grid cells are optimal. In Section 12.1 we show an example where the coarsest cells are 2° of latitude and longitude on each side. Five levels of mesh refinement are used, with the finest grids used only near Hilo, Hawaii, where the total refinement factor of $2^{14} = 16\,384$ in each spatial dimension, so that the finest grid has roughly 10 m resolution. With adaptive refinement we can simulate the propagation of a tsunami originating near Chile (see Figure 1.1 and Section 12) and the inundation of Hilo (see Section 12.1) in a few hours on a single processor.

The shallow water equations are a nonlinear hyperbolic system of partial differential equations and solutions may contain shock waves (hydraulic jumps). In the open ocean a tsunami has an extremely small amplitude

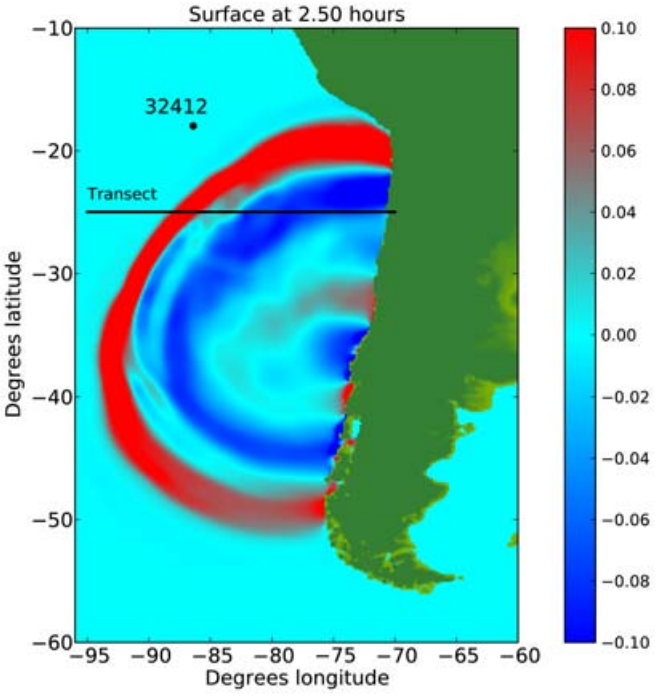


Figure 1.1. The 27 February 2010 tsunami as computed using GeoClaw. In this computation a uniform 216×300 grid with $\Delta x = \Delta y = 1/6$ degree (10 arcminutes) is used. Compare to Figure 12.1, where adaptive mesh refinement is used. The surface elevation and bathymetry along the indicated transect is shown in Figure 1.2. The colour scale for the surface is in metres relative to mean sea level. The location of DART buoy 32412 discussed in the text is also indicated.

(relative to the depth of the ocean) and long wavelength. Hence the propagation is essentially linear, with variable coefficients due to varying bathymetry. As a tsunami approaches shore, however, the amplitude typically increases while the depth of the water decreases and nonlinear effects become important. It is thus desirable to use a method that handles the nonlinearity well (*e.g.*, a high-resolution shock-capturing method), while also being efficient in the linear regime.

In general we would like the method to conserve mass to the extent possible (the momentum equations contain source terms due to the varying bathymetry and possibly Coriolis and frictional drag terms). In this paper we focus on shock-capturing finite volume designed for nonlinear problems that are extensions of *Godunov's method*. These methods are based on solving *Riemann problems* at the interfaces between grid cells, which consist of

the given equations together with piecewise constant initial data determined by the cell averages on either side. Second-order correction terms are defined using *limiters* to avoid non-physical oscillations that might otherwise appear in regions of steep gradients (*e.g.*, breaking waves or turbulent bores that arise as a tsunami approaches and inundates the shore). The methods exactly conserve mass on a fixed grid, but as we will see in Section 9.2 mass conservation is not generally possible or desirable near the shore when AMR is used. Even away from the shore, conserving mass when the grid is refined or de-refined requires some care when the bathymetry varies, as discussed in Section 9.1.

Studying the effect of a tsunami requires accurate modelling of the motion of the shoreline; a major tsunami can inundate several kilometres inland in low-lying regions. This is a free boundary problem and the location of the wet–dry interface must be computed as part of the numerical solution; in fact this is one of the most important aspects of the computed solution for practical purposes. Most tsunami codes do not attempt to explicitly track the moving boundary, which would be very difficult for most realistic problems since the shoreline topology is constantly changing as islands and isolated lakes appear and disappear. Some tsunami models use a fixed shoreline location with solid wall boundary conditions and measure the depth of the solution at this boundary, perhaps converting this via empirical expression to estimates of the inundation distances and run-up (the elevation above sea level at the point of maximum inundation). Most recent codes, however, use some 'wetting and drying' algorithm. The computational grid covers dry land as well as the ocean, and each grid cell is allowed to be wet ($h > 0$) or dry ($h = 0$) in the shallow water equations. The state of each cell can change dynamically in each time step as the wave advances or retreats. Of course accurate modelling of the inundation also requires detailed models of the local topography and bathymetry on a scale of tens of metres or less, while the water depth must be resolved to a fraction of a metre. Again this generally requires the use of mesh refinement to achieve a suitable resolution at the coast.

In the context of a Godunov-type method, it is necessary to develop a robust Riemann solver that can deal with Riemann problems in which one cell is initially dry, as well as the case where a cell dries out as the water recedes. This must be done in a manner that does not result in undershoots that might lead to negative fluid depth.

For tsunami modelling it is essential to accurately capture small perturbations to undisturbed water at rest; the ocean is 4 km deep on average while even a major tsunami has an amplitude less than 1 m in the open ocean. Moreover, the wavelength may be 100 km or more, so that over 1 km, for example, the ocean surface elevation in a tsunami wave varies by less than 1 cm while the bathymetry (and hence the water depth) may vary

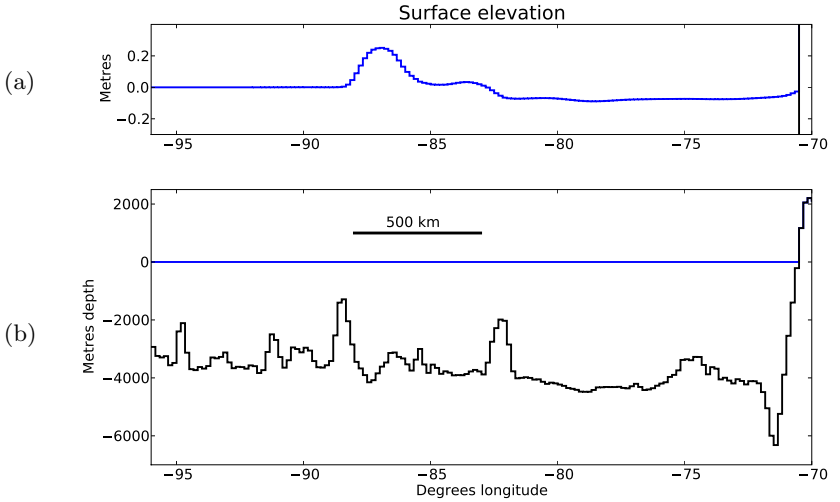


Figure 1.2. Cross-section of the Pacific Ocean on a transect at constant latitude 25°S, as shown in Figure 1.1. (b) Full depth of the ocean. (a) Zoom of the surface elevation from −20 cm to 20 cm showing the small amplitude and long wavelength of the tsunami, 2.5 hours after the earthquake. Note the difference in vertical scales and that in both figures the vertical scale is greatly exaggerated relative to the horizontal scale. The bathymetry and surface elevation are shown as piecewise constant functions over the finite volume cells used, in order to illustrate the large jump in bathymetry between neighbouring grid cells.

by hundreds of metres. This is illustrated in Figure 1.2, which shows a cross-section of the Pacific Ocean along the transect indicated in Figure 1.1, along with a zoomed view of the top surface exhibiting the long wavelength of the tsunami. This extreme difference in scales makes it particularly important that a numerical method be employed that can maintain the steady state of the ocean at rest, and that accurately captures small perturbations to this steady state. Such methods are often called 'well-balanced', because the balance between the flux gradient and the source terms must be maintained numerically. This must also be done in a way that remains well-balanced in the context of AMR, with no spurious waves generated at mesh refinement boundaries. We discuss this difficulty and our approach to well-balancing further in Section 3.1.

Two-dimensional finite volume methods can be applied either on regular (logically rectangular) quadrilateral grids or on unstructured grids such as triangulations. Unstructured grids have the advantage of being able to fit complicated geometries more easily, and for complex coastlines this may seem the obvious approach. For a fixed coastline this might be true, but

when inundation is modelled using a wetting and drying approach the advantage is no longer clear. Logically rectangular grids (indexed by (i, j)) in fact have several advantages: high accuracy is often easier to obtain (at least for smoothly varying grids), and refinement on rectangular patches is natural and relatively easy to perform. The GeoClaw software uses patch-based logically rectangular grids following the approach of Berger, Colella and Oliger (Berger and Colella 1989, Berger and Oliger 1984, Berger and LeVeque 1998). This approach to AMR has been extensively used over the past three decades in many applications and software packages, including Clawpack as well as Chombo (www2), AMROC (www1), SAMRAI (www11), and FLASH (www6). We review this approach in Section 8 and discuss several difficulties that arise in applications to tsunamis.

For many geophysical flow problems it is natural to use either purely Cartesian coordinates (over relatively small domains) or latitude–longitude coordinates on the sphere. The latter is generally used for tsunami propagation problems, for which the region of interest is usually far from the poles. For problems on the full sphere, other grids may be more appropriate, as discussed briefly in Section 6.2. For problems such as flooding of a serpentine river, it may be most appropriate to use a coarse grid that broadly follows the river valley, together with AMR to focus computational cells in the region where the river actually lies. In Section 6 we discuss a class of two-dimensional wave propagation algorithms that maintain stability and accuracy on general quadrilateral grids.

When developing methods to simulate complex geophysical flows it is very important to perform validation and verification studies, as discussed in Section 10. This requires both tests on synthetic problems where the accuracy of the solvers can be judged as well as comparison to observations from real events. Sections 11 and 12 present computational results of each type in order to illustrate the application of these methods.

2. Tsunamis and tsunami modelling

The term *tsunami* (which means 'harbour wave' in Japanese) generally refers to any impulse-generated gravity wave. Tsunamis can arise from many different sources. Most large tsunamis are generated by vertical displacement of the ocean floor during megathrust earthquakes on subduction zones. At a subduction zone, one plate (typically an oceanic plate) descends beneath another (typically continental) plate. The rate of this plate motion is on the order of centimetres per year. In the shallow part of the subduction zone, at depths less than 40 km, the plates are usually stuck together and the leading edge of the upper plate is dragged downwards. Slip during an earthquake releases this part of the plate, generally causing both upward and downward deformation of the ocean floor, and hence

the entire water column above it. The vertical displacement can be several metres, and it can extend across areas of tens of thousands of square kilometres. Displacing this quantity of water by several metres injects an enormous amount of potential energy into the ocean (as much as 10^{23} ergs for a large tsunami, equivalent to roughly 10 megatonnes of TNT). The potential energy is given by

$$\begin{aligned} \text{Potential energy} &= \iint \int_0^{\eta(x,y)} \rho g z \, \mathrm{d}z \, \mathrm{d}x \, \mathrm{d}y \\ &= \iint \frac{1}{2} \rho g \eta^2(x,y) \, \mathrm{d}x \, \mathrm{d}y, \end{aligned} \tag{2.1}$$

where ρ is the density of water, g is the gravitational constant, and $\eta(x, y)$ is the displacement of the surface from sea level. Here x and y are horizontal Cartesian coordinates and z is the vertical direction. This energy is carried away by propagating waves that tend to wreak their greatest havoc nearby, but if the tsunami is large enough can also cause severe flooding and damage thousands of kilometres away. Long-range tsunamis are often termed *teletsunamis* or *far-field tsunamis*, to distinguish them from local tsunamis that affect only regions near the source.

For example, the Aceh–Andaman earthquake on 26 December 2004 generated a tsunami along the zone where the Indian plate is subducting beneath the Burma platelet. The rupture extended for a length of roughly 1500 km and displaced water over a region approximately 150 km wide, with a vertical displacement of several metres. More recently, the Chilean earthquake of 27 February 2010 set off a tsunami along part of the South American subduction zone, where the Nazca plate descends beneath the South American plate. The fault-rupture length was shorter, perhaps 450 km, and fault displacement was also less, yielding a tsunami considerably smaller than the Indian Ocean tsunami of 2004.

The fact that a megathrust earthquake displaces the entire water column over a large surface area is advantageous to modellers, since it means that use of the two-dimensional shallow water equations is well justified. These equations, introduced in Section 3, model gravity waves with long wavelength (relative to the depth of the fluid) in which the entire water column is moving. These conditions are well satisfied as the tsunami propagates across an ocean.

A secondary source of tsunamis is *submarine landslides*, also called *subaqueous landslides*; see for example Bardet, Synolakis, Davies, Imamura and Okal (2003), Masson, Harbitz, Wynn, Pedersen and Løvholt (2006), Ostapenko (1999) and Watts, Grilli, Kirby, Fryer and Tappin (2003). These often occur on the continental slope, which can be several kilometres high and quite steep. The displacement of a large mass on the seafloor causes a corresponding perturbation of the water column above this region, which

again results in gravity waves that can appear as tsunamis. The local displacements may be much larger than in a megathrust earthquake, but usually over much smaller areas and so the resulting tsunamis have far less energy and rapidly dissipate as they radiate outwards. However, they can still do severe damage to nearby coastal regions. For example, an earthquake in 1998 resulted in a tsunami that destroyed several villages and killed more than 2000 people along a 30 km stretch of the north shore of Papua New Guinea. In this case it is thought that the tsunami was caused by a coseismic submarine landslide rather than by the earthquake itself (Synolakis *et al.* 2002). In the case of large earthquakes, it is possible that in addition to the seismic event itself, thousands of coseismic landslides may also occur, leading to additional tsunamis. As an example, Plafker, Kachadoorian, Eckel and Mayo (1969) documented numerous tsunamis in Alaskan fjords in connection with the 1964 earthquake.

There have also been submarine slumps of epic proportions that have caused large-scale destruction. An example is the Storegga slide roughly 8200 years ago on the Norwegian shelf, in which as much as 3000 km^3 of mass was set in motion, creating a tsunami that inundated areas as far away as Scotland (Dawson, Long and Smith 1988, Haflidason, Sejrup, Nygård, Mienert and Bryn 2004).

Subaerial landslides occurring along the coast can also cause localized tsunamis when the landslide debris enters the water. For example, a large-scale landslide on Lituya Bay in Alaska in 1958 caused a landslide within the bay that washed trees away to an elevation of 500 m on the far side of the bay, as documented by Miller (1960) and studied for example in Mader and Gittings (2002), Weiss, Fritz and Wünnemann (2009). Tsunamis and seiches can also arise in lakes as a result of earthquakes or landslides. As an example see McCulloch (1966).

The example we use in this paper is the tsunami generated by the Chilean earthquake of 27 February 2010. The computational advantages of the AMR techniques discussed in this paper are particularly dramatic in modelling far-field effects of transoceanic tsunamis, but are also important in modelling localized tsunamis or the near-field region (which is hardest hit by any tsunami). Typically much higher resolution is needed along a small portion of the coast of primary interest than elsewhere, and over much of the computational domain there is dry land or quiescent water where a very coarse grid can be used.

2.1. Available data sets

Modelling a tsunami requires not only a set of mathematical equations and computational techniques, it also requires data sets, often very large ones. We must specify the bathymetry of the ocean and coastal regions, the topo-

graphy onshore in regions that may be inundated, and the motion of the seafloor that initiates the tsunami. For validation studies we also need observed data from past events, which might include DART buoy (Deep-ocean Assessment and Reporting of Tsunamis) or tide gauge data as well as post-tsunami field surveys of run-up and inundation.

Fortunately there are now ample sources of real data available online that are relatively easy to work with. One of the goals of our own work has been to provide tools to facilitate this, and to provide templates that may be useful in setting up and solving a new tsunami problem. This is still work in progress, but some pointers and documentation are provided in the GeoClaw documentation (www7).

Large-scale bathymetry at the resolution of 1 minute (1/60 degree) for the entire Earth is available from the National Geophysical Data Center (NGDC). The National Geophysical Data Center (NGDC) GEODAS Grid Translator (www9) allows one to specify a rectangular latitude–longitude domain and download bathymetry at a choice of resolutions. Note that one degree of latitude is about 111 km and one degree longitude varies from 111 km at the equator to half that at 60° North, for example. For modelling transoceanic propagation we have found that 10-minute data, with a resolution of roughly 18 km, are often sufficient. In coastal regions greater resolution is required. In particular, in order to model inundation of a target region it may be necessary to have data sets with a resolution of tens of metres or less. The availability of such data varies greatly. In some countries coastal bathymetry is virtually impossible to obtain. In other locations it is easily available online. In particular, many coastal regions of the US are covered by data sets available from NOAA DEMs (www10).

In addition to bathymetry, it is necessary to have matching onshore topography for regions where inundation is to be studied. Unfortunately bathymetry and topography are generally measured by different techniques and sometimes the data sets do not match up properly at the coastline, which of course is exactly the region of primary interest in modelling inundation. Often a great deal of work has already gone into creating the public data sets in order to reconcile these differences, but an awareness of potential difficulties is valuable.

When studying landslide-induced tsunamis, an additional difficulty is that detailed bathymetry of the region around the slide is typically obtained only after the slide has occurred. Without pre-slide bathymetry at the same resolution it can be difficult to determine the correct initial bathymetry or the mass of the slide, which of course is crucial to know in order to generate the correct tsunami numerically.

For subduction zone events it is also necessary to know the seafloor displacement in order to generate the tsunami. In this case the modeller is aided by the fact that the mechanics of some earthquakes have been well

studied. For large events there are generally ample seismic data available from around the world that can be used to attempt to reconstruct the *focal mechanism* of the quake: the direction of slip and orientation of the fault, along with the depth at which the rupture occurred, the length of the rupture, the magnitude of the displacement, *etc.* An event can sometimes be modelled by a simplified representation consisting of a few such parameters, for example the USGS model of the Chile 2010 earthquake (www12) that we use in some of our examples later in this paper. To convert these parameters into seafloor deformation in each grid cell would require solving three-dimensional elasticity equations with a dislocation within the earth, and would require detailed knowledge of the elastic parameters and the geological substructure of the earth in the region of the quake. Instead, a simplified model is generally used to quickly convert parameters into approximate seafloor deformation, such as the well-known model introduced by Mansinha and Smylie (1971) and later modified by Okada (1985, 1992). We use a Python implementation of the Okada model that we based on the models in the COMCOT (www4) (Liu, Woo and Cho 1998).

Larger events are often subdivided into a finite collection of such parametrizations, by breaking the fault into pieces with different sets of parameters. For each piece, the focal mechanism parameters can then be converted into the resulting motion of the seafloor, and these can be summed to obtain the approximate seafloor deformation resulting from the earthquake. It may also be necessary to use time-dependent deformations for large events, such as the 2004 Aceh–Andaman event, which lasted more than 10 minutes as the rupture propagated northwards.

Although large earthquakes are well studied, determining the correct mechanism is non-trivial and there are often several different mechanisms proposed that may be substantially different, particularly in regard to the tsunamis that they generate. One use of tsunami modelling is to aid in the study of earthquakes, providing additional constraints on the mechanism beyond the seismic evidence; see for example Hirata, Geist, Satake, Tanioka and Yamaki (2003). However, the existence of competing descriptions of the earthquake can also make it more difficult to validate a numerical method for the tsunami itself.

In addition to seismic data, real-time data during a tsunami are also measured by tide gauges at many coastal locations, from which the amplitude and waveform of the tsunami can be estimated. The tides and any coseismic deformation must be filtered out from these data in order to see the tsunami, particularly for large-scale tsunamis that can extend through several tidal periods. The observed waves (particularly in shallow water) are also highly dependent on the local bathymetry, and can vary greatly between nearby points. Tide gauges in bays or harbours often register much more wave action than would be seen farther from shore, due to reflections and

resonant sloshing. To have any hope of properly capturing this numerically it is generally necessary to provide the model with fine-scale local bathymetry.

The wave amplitude in the deep ocean cannot be measured by traditional tide gauges, but in recent years a network of gauges have been installed on the ocean floor that measure the water pressure with sufficient sensitivity to estimate the depth. In Section 12 we use data from a DART buoy (Meinig, Stalin, Nakamura, González and Milburn 2006), which transmits data from a pressure sensor at a depth of more than 4000 m. The DART system was developed by NOAA and originally deployed only along the western coast of the United States. Many other nations have also developed similar buoy systems, and after the 2004 Indian Ocean tsunami the world-wide network was greatly expanded. Real-time and historical data sets are available online via DART Data (www5).

Also useful in tsunami modelling is the wealth of data collected by tsunami survey teams that respond after any tsunami event. Attempts are made to map the run-up and inundation along stretches of the affected coast, by examining water marks on buildings, wrack lines, debris lodged in trees, and other markers. This evidence often disappears relatively quickly after the event and the rapid response of scientists and volunteers is critical. The findings are generally published and are valuable sources of data for validation studies. Again it is often necessary to have high-resolution local bathymetry and topography in order to model the great variation in run-up and inundation that are often seen between nearby coastal locations. Survey teams sometimes collect these data as well. For some sample survey results, see for example Gelfenbaum and Jaffe (2003), Liu, Lynett, Fernando, Jaffe and Fritz (2005) and Yeh *et al.* (2006).

Information about past tsunamis can also be gleaned from the study of tsunami deposits (Bourgeois 2009). As a tsunami approaches shore it generally becomes quite turbulent, even forming a bore, and picks up sediment such as sand and marine microorganisms that may be deposited inland as the tsunami decelerates. These deposits can often be identified, either near the surface from a recent tsunami or in the subsurface from prehistoric events, as illustrated in Figure 2.1. In some coastal regions, excavations and core samples reveal more than ten distinct layers of deposits from tsunamis in the past few thousand years. Much of what is known about the frequency of megathrust earthquakes along subduction zones has been learned from studying tsunami deposits, as these deposits are commonly the only remaining evidence of past earthquakes. For example, Figure 2.2 shows the record of 17 sand layers interpreted as tsunami deposits, from the coast of Oregon state, indicating that megathrust events along the Cascadia Subduction Zone (CSZ) occur roughly every 500 years. The CSZ runs from northern California to British Columbia, and the last great earthquake and triggered

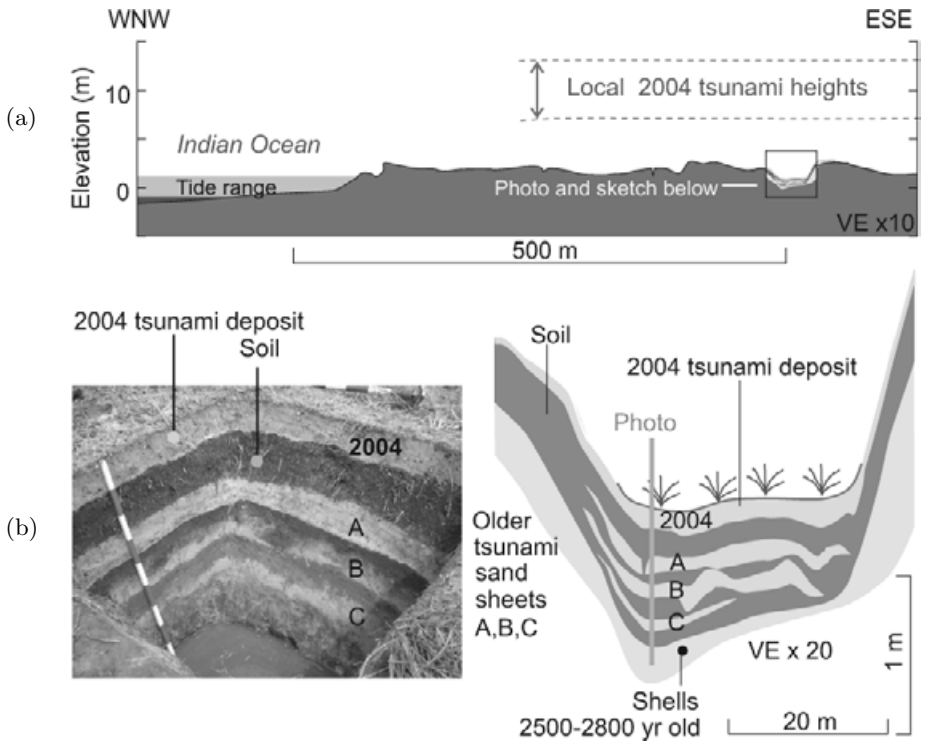


Figure 2.1. 2004 and older tsunami deposits in western Thailand (Jankaew *et al.* 2008). (a) Coastal profile of a part of western Thailand hit by the 2004 Indian Ocean tsunami (simplified from Figure 2 in Jankaew *et al.* (2008)). (b) Photo and sketch of a trench along this profile, showing the 2004 tsunami deposit and three older tsunami deposits, all younger than about 2500 years ago.

tsunami were on 26 January 1700, as determined from matching Japanese historical records of a tsunami with dated tsunami deposits in the Pacific Northwest of the US (Satake, Shimazaki, Tsuji and Ueda 1996, Satake, Wang and Atwater 2003). An interesting account of this scientific discovery can be found in Atwater *et al.* (2005). The next such event will have disastrous consequences for many communities in the Pacific Northwest, and the tsunami is expected to cause damage around the Pacific.

2.2. Uses of tsunami modelling

There are many reasons to study tsunamis computationally, and ample motivation for developing faster and more accurate numerical methods. Applications include the development of more accurate real-time warning systems, the assessment of potential future hazards to assist in emergency planning, and the investigation of past tsunamis and their sources. In this section we give a brief introduction to some of the issues involved.

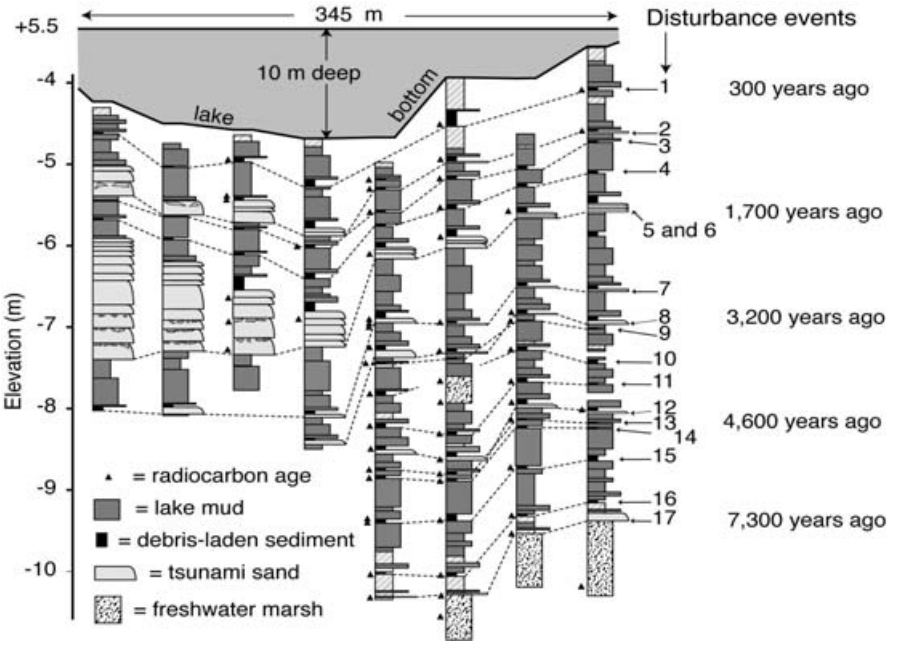


Figure 2.2. An example of long-term records of tsunami deposits interpreted to be from the Cascadia subduction zone: from Bradley Lake on the coast of southern Oregon. Seventeen different sediment deposits were identified and correlated at eight different locations. The far right column shows the approximate age of each set of deposits. From Bourgeois (2009), based on a figure of Kelsey, Nelson, Hemphill-Haley and Witter (2005).

Real-time warning systems rely on numerical models to predict whether an earthquake has produced a dangerous tsunami, and to identify which communities may need to be warned or evacuated. Mistakes in either direction are costly: failing to evacuate can lead to loss of life, but evacuating unnecessarily is not only very expensive but also leads to poor response to future warnings. Real-time prediction is difficult for many reasons: a code is required that will run faster than real time and still provide detailed results, usually for many different locations. Moreover, the source is usually poorly known initially since solving the inverse problem of determining the focal mechanism from seismic signals takes considerable time and consolidation of data from multiple sites. The DART buoys were developed in part to address this problem. By measuring the actual wave at one or more locations near the source, a better estimate of the tsunami can be quickly generated and used to select initial data for real-time prediction, as discussed by Percival *et al.* (2010).

Most codes used for studying tsunamis are not designed for real-time warning; this is a specialized and demanding application (Titov *et al.* 2005). However, there are many other applications where research codes can play a role. For example, hazard assessment and mitigation requires the use of tsunami models to investigate the potential damage from a future tsunami, to locate safe havens and plan evacuation routes, and to assist government agencies in planning for emergency response. For this, information about past tsunamis in a region is valuable both in validating the code and in designing hypothetical tsunami sources for assessing the vulnerability to future tsunamis.

A topic of growing interest is the development of probabilistic models that take into account the uncertainty of future earthquakes. Seismologists can often provide information about the likelihood of ruptures of various magnitudes along several fault planes, and tsunami modellers then seek to produce from this a probabilistic assessment of the risk of inundation to varying degrees. Although these simulations do not need to be set up and run in real time, the need to do large numbers of simulations for a probabilistic study is additional motivation for developing fast and accurate techniques that can handle the entire simulation from tsunami generation to detailed modelling of specific distant communities. For more on this topic, see for example Geist and Parsons (2006), González, Geist, Jaffe, Kanoglu *et al.* (2009) and Geist, Parsons, ten Brink and Lee (2009).

Another use of tsunami modelling is to better understand past tsunamis, and to identify the earthquakes that generated them. Much of what is known about earthquakes that happened before the age of seismic monitoring or historical records has been determined through the study of tsunami deposits, as illustrated in Figures 2.1 and 2.2 and discussed above. Tsunami modelling is often required to assist in solving the inverse problem of determining the most likely earthquake source and magnitude from a given set of deposits. For this it would be desirable to couple the tsunami model to sedimentation equations capable of modelling the suspension of sediments and their transport and deposition, ideally also taking into account the resulting changes in bathymetry and topography that may affect the fluid dynamics. Moreover, tsunami deposits often exhibit layers in which the grain size either increases or decreases with depth, and this grading contains information about how the flow was behaving at this location while the sediment was deposited; *e.g.*, Higman, Gelfenbaum, Lynett, Moore and Jaffe (2007) and Martin *et al.* (2008). Ideally the model would include multiple grain sizes and accurately simulate the entrainment and sedimentation of each. The development of sufficiently accurate sedimentation models and computational tools adequate to do this type of analysis is an active area of research; see for example Huntington *et al.* (2007).

3. The shallow water equations

The shallow water equations are the standard governing model used for transoceanic tsunami propagation as well as for local inundation: *e.g.*, Yeh, Liu, Briggs and Synolakis (1994) and Titov and Synolakis (1995, 1998). Because we use shock-capturing methods that can converge to discontinuous weak solutions, we solve the most general form of the equations: a nonlinear system of hyperbolic conservation laws for depth and momentum. In one space dimension these take the form

$$h_t + (hu)_x = 0, \tag{3.1a}$$

$$(hu)_t + (hu^2 + \tfrac{1}{2}gh^2)_x = -ghB_x, \tag{3.1b}$$

where g is the gravitational constant, $h(x,t)$ is the fluid depth, $u(x,t)$ is the vertically averaged horizontal fluid velocity. A drag term $-D(h,u)u$ can be added to the momentum equation and is often important in very shallow water near the shoreline. This is discussed in Section 7.

The function $B(x)$ is the bottom surface elevation relative to mean sea level. Where $B < 0$ this corresponds to submarine bathymetry and where $B > 0$ to topography. Although in tsunami studies the term bathymetry is commonly used, in much of this paper we will use the term *topography* to refer to both bathymetry and onshore topography, both for conciseness and because in many other geophysical flows (debris flows, lava flows, *etc.*) there is only topography.

We will also use $\eta(x,t)$ to denote the water surface elevation,

$$\eta(x,t) = h(x,t) + B(x,t).$$

We allow the topography to be time-dependent since most tsunamis are generated by motion of the ocean floor resulting from an earthquake or landslide. Figure 3.1 shows a simple sketch of the variables. Note that (3.1) is in fact a 'balance law', since variable bottom topography and drag introduce source terms in the momentum equation. The physically relevant form (3.1) introduces some difficulties for numerical solution, particularly with regard to steady state preservation. As mentioned above, this has led to the development of well-balanced schemes for such systems (see *e.g.* Bale, LeVeque, Mitran and Rossmanith (2002), Bouchut (2004), George (2008), Greenberg and LeRoux (1996), Botta, Klein, Langenberg and Lützenkirchen (2004), Gallardo, Parés and Castro (2007), Gosse (2000), LeVeque (2010) and Noelle, Pankrantz, Puppo and Natvig (2006)). This is sometimes circumvented by using alternative non-conservative forms of the shallow water equations for $\eta(x,t)$ and $u(x,t)$, but these forms are problematic if discontinuities appear in the inundation regime (bore formation), and conservation of mass is not easily guaranteed.

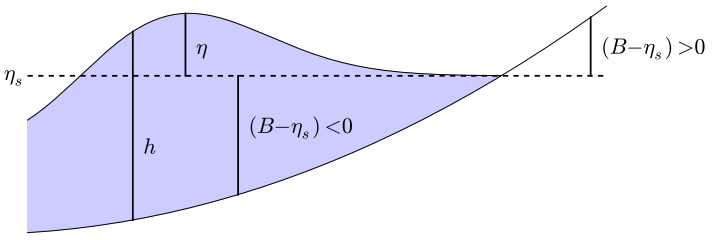


Figure 3.1. Sketch of the variables of the shallow water equations. The shaded region is the water of depth $h(x,t)$, and the water surface is $\eta(x,t) = B(x,t) + h(x,t)$. The dashed line shows the mean sea level η_s.

For tsunami modelling we solve the two-dimensional shallow water equations

$$h_t + (hu)_x + (hv)_y = 0, \tag{3.2a}$$

$$(hu)_t + (hu^2 + \tfrac{1}{2}gh^2)_x + (huv)_y = -ghB_x, \tag{3.2b}$$

$$(hv)_t + (huv)_x + (hv^2 + \tfrac{1}{2}gh^2)_y = -ghB_y, \tag{3.2c}$$

where $u(x,y,t)$ and $v(x,y,t)$ are the depth-averaged velocities in the two horizontal directions, $B(x,y,t)$ is the topography. Again a drag term might be added to the momentum equations.

For simplicity, we will discuss many issues in the context of the one-dimensional shallow water equations (3.1) whenever possible. We also first consider the equations in Cartesian coordinates, with x and y measured in metres, as might be appropriate when modelling local effects of waves on a small portion of the coast or in a wave tank. For transoceanic tsunami propagation it is necessary to propagate on the surface of the earth, as discussed further in Section 6.2. For this it is common to use latitude and longitude coordinates, assuming the earth is a perfect sphere. A more accurate geoid representation of the earth could be used instead. Latitude–longitude coordinates present difficulties for many problems posed on the sphere due to the fact that grid lines coalesce at the poles and cells are much smaller in the polar regions than elsewhere, which can lead to time step restrictions. For tsunamis on the earth we are generally only interested in the mid-latitudes and this is not a problem, but in Section 6.2 we mention an alternative grid that may be useful in other contexts.

On a rotating sphere the equations should also include Coriolis terms in the momentum equations. For tsunami modelling these are generally neglected. During propagation across an ocean, the fluid velocities are small and are concentrated within the wave region and Coriolis effects have been shown to be very small (*e.g.*, Kowalik, Knight, Logan and Whitmore

(2005)). Our own tests have also indicated that Coriolis terms can be safely ignored. On the other hand, they are simple to include numerically along with the drag terms via a fractional step approach, as discussed in Section 7.

3.1. Hyperbolicity and Riemann problems

The shallow water equations (3.1) belong to the more general class of hyperbolic systems

$$q_t + f(q)_x = \psi(q, x), \tag{3.3}$$

where $q(x, t)$ is the vector of unknowns, $f(q)$ is the vector of corresponding fluxes, and $\psi(q, x)$ is a vector of source terms:

$$q = \begin{bmatrix} h \\ hu \end{bmatrix}, \quad f(q) = \begin{bmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \end{bmatrix}, \quad \psi = \begin{bmatrix} 0 \\ -ghB_x \end{bmatrix}. \tag{3.4}$$

We will also introduce the notation $\mu = hu$ for the momentum and $\phi = hu^2 + \frac{1}{2}gh^2$ for the momentum flux, so that

$$q = \begin{bmatrix} h \\ \mu \end{bmatrix}, \quad f(q) = \begin{bmatrix} \mu \\ \phi \end{bmatrix}. \tag{3.5}$$

The Jacobian matrix $f'(q)$ then has the form

$$f'(q) = \begin{bmatrix} \partial\mu/\partial h & \partial\mu/\partial\mu \\ \partial\phi/\partial h & \partial\phi/\partial\mu \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ gh - u^2 & 2u \end{bmatrix}. \tag{3.6}$$

Hyperbolicity requires that the Jacobian matrix be diagonalizable with real eigenvalues and linearly independent eigenvectors. For the shallow water equations the matrix in (3.6) has eigenvalues

$$\lambda^1 = u - \sqrt{gh}, \quad \lambda^2 = u + \sqrt{gh} \tag{3.7}$$

and corresponding eigenvectors

$$r^1 = \begin{bmatrix} 1 \\ u - \sqrt{gh} \end{bmatrix}, \quad r^2 = \begin{bmatrix} 1 \\ u + \sqrt{gh} \end{bmatrix}. \tag{3.8}$$

We will use superscripts to index these eigenvalues and eigenvectors since subscripts corresponding to grid cells will be added later.

Note that the eigenvalues are always real for physically relevant depths $h \geq 0$. For $h > 0$ they are distinct and the eigenvectors are linearly independent. Hence the equations are hyperbolic for $h > 0$, and the solution consists of propagating waves. The eigenvalues correspond to velocities of propagation and the eigenvectors give information about the relation between h and hu in a wave propagating at this speed.

Note that waves propagate at velocities $\pm\sqrt{gh}$ relative to the background fluid velocity u. The velocity $c = \sqrt{gh}$ is the *gravity wave speed* and is

analogous to the sound speed for small-amplitude acoustic waves. For two-dimensional shallow water equations the theory is somewhat more complicated, since waves can propagate in any direction, but the speed of propagation in any direction is again $\sqrt{gh}$ relative to the fluid velocity.

Note also that in general the eigenvalues satisfy $\lambda^1 < \lambda^2$, but they could both be negative (if $u < -\sqrt{gh}$) or both positive (if $u > \sqrt{gh}$). Such flows are called *supercritical* and correspond to supersonic flow in gas dynamics. For tsunami modelling, the flow is nearly always subcritical, with $\lambda^1 < 0 < \lambda^2$, except in very shallow water near the shore. The ratio $|u|/\sqrt{gh}$ is called the *Froude number* and is analogous to the Mach number of gas dynamics.

For a tsunami propagating in the ocean, the fluid velocity is very small relative to $\sqrt{gh}$ and so the velocity of propagation depends primarily on the depth. For a typical ocean depth of 4000 m the propagation speed is nearly 200 m s^{-1}, roughly the speed of a commercial jet. In shallower water the wave speed decreases. On a continental shelf with a typical depth of 100 m, the speed is about 30 m s^{-1}, about 6 times smaller. This is worth bearing in mind when using explicit numerical methods, since the time step allowed by stability considerations is directly proportional to the wave speed. We will return to this in Section 8.1.

3.2. Eliminating the source term

There is a technique that is often used to eliminate the source term in a hyperbolic system with the structure of the one we are considering, which we introduce now since we will use it in developing Riemann solvers below. Rewrite the original system of nonlinear equations (3.1) as a system of three equations, by viewing the topography $B(x,t)$ as a function of x and t that does not vary with time:

$$\begin{aligned} h_t + \mu_x &= 0, \\ \mu_t + \phi_x + ghB_x &= 0, \\ B_t &= 0. \end{aligned} \tag{3.9}$$

This gives a *homogeneous* hyperbolic system, though at the expense of turning the system into a nonlinear system that is not in conservation form, due to the 'non-conservative product' hB_x. This has potential difficulties associated with it (see for example Castro, LeFloch, Munoz and Parés (2008)), but this form is useful in deriving Riemann solvers. The system (3.9) is hyperbolic since the eigenvalues of the Jacobian matrix

$$\begin{bmatrix} 0 & 1 & 0 \\ -u^2 + gh & 2u & gh \\ 0 & 0 & 0 \end{bmatrix} \tag{3.10}$$

are easily seen to be $\lambda^{1,2} = u \pm \sqrt{gh}$, as in the original system, along with

$\lambda^0 = 0$. The new wave we have introduced with speed 0 comes from the stationary discontinuity in B. Note that the eigenvector associated with this wave is

$$r^0 = \begin{bmatrix} gh/(u^2 - gh) \\ 0 \\ 1 \end{bmatrix}. \tag{3.11}$$

This indicates that the stationary wave with a small jump in bathymetry ΔB also has a jump in h, and if $u = 0$ then the first component of r^0 is -1, so that $\Delta h = -\Delta B$ and hence $\Delta\eta = 0$, corresponding to the ocean at rest. More generally, if the Froude number $|u|/\sqrt{gh}$ is small then $\Delta\eta \approx -(u^2/gh)\Delta B$.

The momentum μ is always constant across this wave. This makes sense physically since μ is also the mass flux, and a stationary jump in mass flux would lead to the creation of a delta function singularity in mass at this point.

3.3. Linearized equations

The easiest case to analyse is the *linearized equation* governing small-amplitude waves relative to the fluid depth. Consider flat topography for the moment (so the source term disappears) and suppose we consider very small-amplitude waves against a background steady state with constant depth $\hat{h}$ and velocity $\hat{u}$. For tsunami modelling it is natural to take $\hat{u} = 0$, but one could also study small waves on a steady flow with some non-zero velocity. Then, if we write $q(x,t) = \hat{q} + \tilde{q}(x,t)$ and insert this into the shallow water equations, we find that the small perturbation $\tilde{q}$ satisfies

$$\tilde{q}_t + \hat{A}\tilde{q}_x = \mathcal{O}(\|\tilde{q}\|^2), \tag{3.12}$$

where $\hat{A} = f'(\hat{q})$ is the constant Jacobian matrix evaluated at the background state $\hat{q} = (\hat{h},\ \hat{h}\hat{u})^T$. If we drop the higher-order terms and also drop the tildes in (3.13), we obtain the linearized equations

$$q_t + \hat{A}q_x = 0. \tag{3.13}$$

This is a linear hyperbolic partial differential equation (PDE) with constant eigenvalues

$$\hat{\lambda}^1 = \hat{u} - \hat{c}, \quad \hat{\lambda}^2 = \hat{u} + \hat{c}, \quad \text{where } \hat{c} = \sqrt{g\hat{h}}. \tag{3.14}$$

The eigenvectors $\hat{r}^1$ and $\hat{r}^2$ from (3.8) are also constant. If we form a matrix $\hat{R} = [\hat{r}^1,\ \hat{r}^2]$ with these columns, then this eigenvector matrix diagonalizes $\hat{A}$:

$$\hat{A} = \hat{R}\hat{\Lambda}\hat{R}^{-1}, \quad \text{or} \quad \hat{\Lambda} = \hat{R}^{-1}\hat{A}\hat{R}. \tag{3.15}$$

Because this matrix is independent of x and t, we can multiply (3.13) by R^{-1}, replace A by $A\hat{R}\hat{R}^{-1}$, and hence obtain the diagonal system

$$w_t + \hat{\Lambda} w_x = 0, \tag{3.16}$$

where $w = \hat{R}^{-1}q$. This decouples into two scalar advection equations for the *characteristic variables* w^1 and w^2, with solutions that simply translate at speeds $\hat{\lambda}^1$ and $\hat{\lambda}^2$ respectively. The linear PDE with arbitrary initial conditions can thus be solved by computing initial characteristic data $w(x,0) = \hat{R}^{-1}q(x,0)$, solving the scalar advection equations for each component of $w(x,t)$, and finally computing $q(x,t) = \hat{R}w(x,t)$. Note that $q(x,t)$ is always a linear combination of the two eigenvectors, and $w^1(x,t)$ and $w^2(x,t)$ are simply the weights.

3.4. The linear Riemann problem

Since the ocean does not have constant depth, and is not one-dimensional, we cannot use the above exact solution procedure directly. However, understanding the eigenstructure displayed above is critical to the development of Godunov-type numerical methods that we concentrate on here. These methods, and also much of the theory of both linear and nonlinear hyperbolic PDEs, are based on solutions to the so-called *Riemann problem*. This consists of the original PDE under study together with very special initial data at some time $t = \bar{t}$ consisting of piecewise constant data with a single jump discontinuity at some point $\bar{x}$,

$$q(x,\bar{t}) = \begin{cases} Q_\ell & \text{if } x < \bar{x}, \\ Q_r & \text{if } x > \bar{x}. \end{cases} \tag{3.17}$$

For the linear hyperbolic problem (3.13), it is easy to see (using the construction of the exact solution described above), that the solution consists of two discontinuities propagating away from the point $\bar{x}$ at velocities $\hat{\lambda}^1$ and $\hat{\lambda}^2$. Moreover the jump in q across each of these waves must be proportional to the corresponding eigenvector, and so the solution has the form

$$q(x,t) = \begin{cases} Q_\ell & \text{if } x < \bar{x} + \hat{\lambda}^1(t-\bar{t}), \\ Q_m & \text{if } \bar{x} + \hat{\lambda}^1(t-\bar{t}) < x < \bar{x} + \hat{\lambda}^2(t-\bar{t}), \\ Q_r & \text{if } x > \bar{x} + \hat{\lambda}^2(t-\bar{t}), \end{cases} \tag{3.18}$$

where the middle state Q_m satisfies

$$Q_m = Q_\ell + \alpha^1 \hat{r}^1 = Q_r - \alpha^2 \hat{r}^2 \tag{3.19}$$

for some scalars α^1 and α^2. We will denote the waves by

$$\mathcal{W}^1 = Q_m - Q_\ell = \alpha^1 \hat{r}^1, \quad \mathcal{W}^2 = Q_r - Q_m = \alpha^2 \hat{r}^2. \tag{3.20}$$

The weights α^1 and α^2 can be found as the two components of the vector α by solving the linear system

$$\hat{R}\alpha = Q_r - Q_\ell. \tag{3.21}$$

The solution is easily determined to be

$$\alpha^1 = \frac{\hat{\lambda}^2 \Delta h - \Delta\mu}{2\hat{c}}, \quad \alpha^2 = \frac{-\hat{\lambda}^1 \Delta h - \Delta\mu}{2\hat{c}}. \tag{3.22}$$

where $\Delta h = h_r - h_\ell$ and $\Delta\mu = \mu_r - \mu_\ell = h_r u_r - h_\ell u_\ell$. Note in particular that if $u_\ell = u_r = \hat{u}$ then $\alpha^1 = \alpha^2 = (h_r - h_\ell)/2$, and the initial jump in h resolves into equal-amplitude waves propagating upstream and downstream.

For the constant coefficient linear problem the characteristic structure determines the Riemann solution. For variable coefficient or nonlinear problems, the exact solution for general initial data can no longer be computed by characteristics in general, but the Riemann problem can still be solved and is a key tool in analysis and numerics.

3.5. Varying topography

To linearize the shallow water equations in the case of variable topography, it is easiest to work in terms of the surface elevation $\eta(x,t) = B(x) + h(x,t)$. We will linearize about a flat surface $\hat{\eta}$ and zero velocity $\hat{u} = 0$. We will define $\hat{h}(x) = \hat{\eta} - B(x)$, which is no longer constant and may have large variations if the topography $B(x)$ varies. The momentum equation can be rewritten as

$$\mu_t + (hu^2)_x + gh(h + B)_x = 0, \tag{3.23}$$

and linearizing this gives the equation

$$\tilde{\mu}_t + g\hat{h}(x)\tilde{\eta}_x = 0 \tag{3.24}$$

for the perturbation $(\tilde{\eta},\ \tilde{\mu})$ about $(\hat{\eta},\ 0)$. Combining this with the already linear continuity equation $\tilde{\eta}_t + \tilde{\mu}_x = 0$ and dropping tildes gives the variable coefficient linear hyperbolic system

$$\begin{bmatrix} \eta \\ \mu \end{bmatrix}_t + \begin{bmatrix} 0 & 1 \\ g\hat{h}(x) & 0 \end{bmatrix} \begin{bmatrix} \eta \\ \mu \end{bmatrix}_x = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{3.25}$$

If we try to diagonalize these equations, we find that because the eigenvector matrix R now varies with x, the advection equations for the characteristic variables w^1 and w^2 are coupled together by source terms that only vanish where the bathymetry is flat. Over varying bathymetry a wave in one characteristic family is constantly losing energy into the other family, corresponding to wave reflection from the bathymetry.

Nonetheless, we can define a *Riemann problem for this variable coefficient system* by allowing a jump in $\hat{h}$ from $\hat{h}_\ell$ to $\hat{h}_r$ at $\bar{x}$, along with a jump in the

data from $(\eta_\ell,\ \mu_\ell)$ to $(\eta_r,\ \mu_r)$. The solution to this Riemann problem consists of a left-going wave with speed $\hat{c}_\ell = -(g\hat{h}_\ell)^{1/2}$ and a right-going wave with speed $\hat{c}_r = (g\hat{h}_r)^{1/2}$. Each wave propagates across a region of constant topography (B_ℓ or B_r respectively) at the appropriate speed, and hence the jump in $(\eta,\ \mu)$ across each wave must be an eigenvector corresponding to the coefficient matrix on that side of $\bar{x}$:

$$\mathcal{W}^1 = \alpha^1 \hat{r}^1_\ell = \alpha^1 \begin{bmatrix} 1 \\ -\hat{c}_\ell \end{bmatrix}, \quad \mathcal{W}^2 = \alpha^2 \hat{r}^2_r = \alpha^2 \begin{bmatrix} 1 \\ \hat{c}_r \end{bmatrix}, \tag{3.26}$$

The weights α^1 and α^2 can be determined by solving the linear system

$$\begin{bmatrix} 1 & 1 \\ -\hat{c}_\ell & \hat{c}_r \end{bmatrix} \begin{bmatrix} \alpha^1 \\ \alpha^2 \end{bmatrix} = \begin{bmatrix} \eta_r - \eta_\ell \\ \mu_r - \mu_\ell \end{bmatrix} \equiv \begin{bmatrix} \Delta\eta \\ \Delta\mu \end{bmatrix}, \tag{3.27}$$

yielding

$$\alpha^1 = \frac{\hat{c}_r \Delta\eta - \Delta\mu}{c_\ell + c_r}, \quad \alpha^2 = \frac{\hat{c}_l \Delta\eta + \Delta\mu}{c_\ell + c_r}. \tag{3.28}$$

Note that in the case when there is no jump in topography, $\hat{h}_\ell = \hat{h}_r = \hat{h}$, we find that $-c_\ell = c_r = (g\hat{h})^{1/2}$, and $\Delta\eta = \Delta h$, so that (3.28) agrees with (3.22).

Another way to derive this linearized solution is to linearize the system (3.9) that we obtained by introducing $B(x, y)$ as a new component. Linearizing about $\hat{h}$ and $\hat{u} = 0$ gives the variable coefficient matrix

$$\hat{A}(x) = \begin{bmatrix} 0 & 1 & 0 \\ g\hat{h}(x) & 0 & g\hat{h}(x) \\ 0 & 0 & 0 \end{bmatrix}, \quad \hat{h}(x) = \begin{cases} \hat{h}_\ell & \text{if } x < \bar{x}, \\ \hat{h}_r & \text{if } x > \bar{x}. \end{cases} \tag{3.29}$$

The Riemann solution consists of three waves, found by decomposing

$$\Delta q = \begin{bmatrix} \Delta h \\ \Delta\mu \\ \Delta B \end{bmatrix} = \alpha^1 \begin{bmatrix} 1 \\ -\hat{c}_\ell \\ 0 \end{bmatrix} + \alpha^2 \begin{bmatrix} 1 \\ \hat{c}_r \\ 0 \end{bmatrix} + \alpha^0 \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}. \tag{3.30}$$

From the third equation we find $\alpha^0 = \Delta B$, and then α^1 and α^2 can be found by solving

$$\begin{bmatrix} \Delta h + \Delta B \\ \Delta\mu \\ 0 \end{bmatrix} = \alpha^1 \begin{bmatrix} 1 \\ -\hat{c} \\ 0 \end{bmatrix} + \alpha^2 \begin{bmatrix} 1 \\ \hat{c} \\ 0 \end{bmatrix}. \tag{3.31}$$

Since $\Delta h + \Delta B = \Delta\eta$, this gives the same system as (3.27), and the same propagating waves as before.

We will make use of this Riemann solution for the linearized shallow water equations in developing an approach for the full nonlinear equations in Section 5.

3.6. Interaction with the continental shelf

Often there is a broad and shallow continental shelf that is separated from the deep ocean by a very steep and narrow continental slope (narrow relative to the wavelength of the tsunami, that is). Figure 12.4 shows the continental shelf near Lima, Peru and the refraction of the 27 February 2010 tsunami wave hitting this shelf. In this section we consider an idealized model to help understand the amplification of a tsunami that takes place as it approaches the coast.

Consider piecewise constant bathymetry with a jump from an undisturbed depth h_ℓ to a shallower depth of h_r. Figure 3.2 shows an example of a small-amplitude wave interacting with such bathymetry, in this case a step discontinuity 30 km offshore at the location indicated by the dashed line. The undisturbed depths are $h_\ell = 4000$ and $h_r = 200$ m. At time $t = 0$ a hump of stationary water is introduced with amplitude 0.4 m. This hump splits into left-going and right-going waves of equal amplitude, sufficiently small that propagation is essentially linear on both sides of the discontinuity. A purely positive perturbation of the depth is used here to make the figures clearer, but any small-amplitude waveform would behave in the same manner.

We observe in Figure 3.2 that the right-going wave is split into transmitted and reflected waves when it encounters the discontinuity in bathymetry. The transmitted wave has large amplitude, but shorter wavelength, while the reflected wave has smaller amplitude. At later times the right-going wave on the shelf reflects off the right boundary and becomes a left-going wave. In this model problem the shore is simply a solid vertical wall, but a similar reflection would be observed from a beach. This left-going wave reflected from shore later hits the discontinuity in bathymetry and is itself split into a transmitted wave (left-going in the ocean) and a reflected wave (right-going on the shelf). The reflected right-going wave is now a wave of depression, which later reflects off the shore, then off the discontinuity, *etc.*

It is important to note that much of the wave energy is trapped on the continental shelf and reflects multiple times between the discontinuity in bathymetry and the shore. This has practical implications and is partly responsible for the fact that multiple destructive tsunami waves are often observed on the coast. Moreover, the trapped wave continues to radiate energy back into the ocean each time the wave reflects off the discontinuity. This leads to a more complex wave pattern elsewhere in the ocean than would be observed from the initial tsunami alone, or from including only the single reflection that would be seen from a shore with no shelf. This suggests that to accurately simulate tsunamis it may be important to adequately resolve continental shelves, even in regions away from the coastline of primary interest in the simulation. As an example of this, the simulation shown in Figures 12.1–12.4 shows that large-amplitude waves remain trapped on the shelf off Peru long after the main tsunami has passed by.

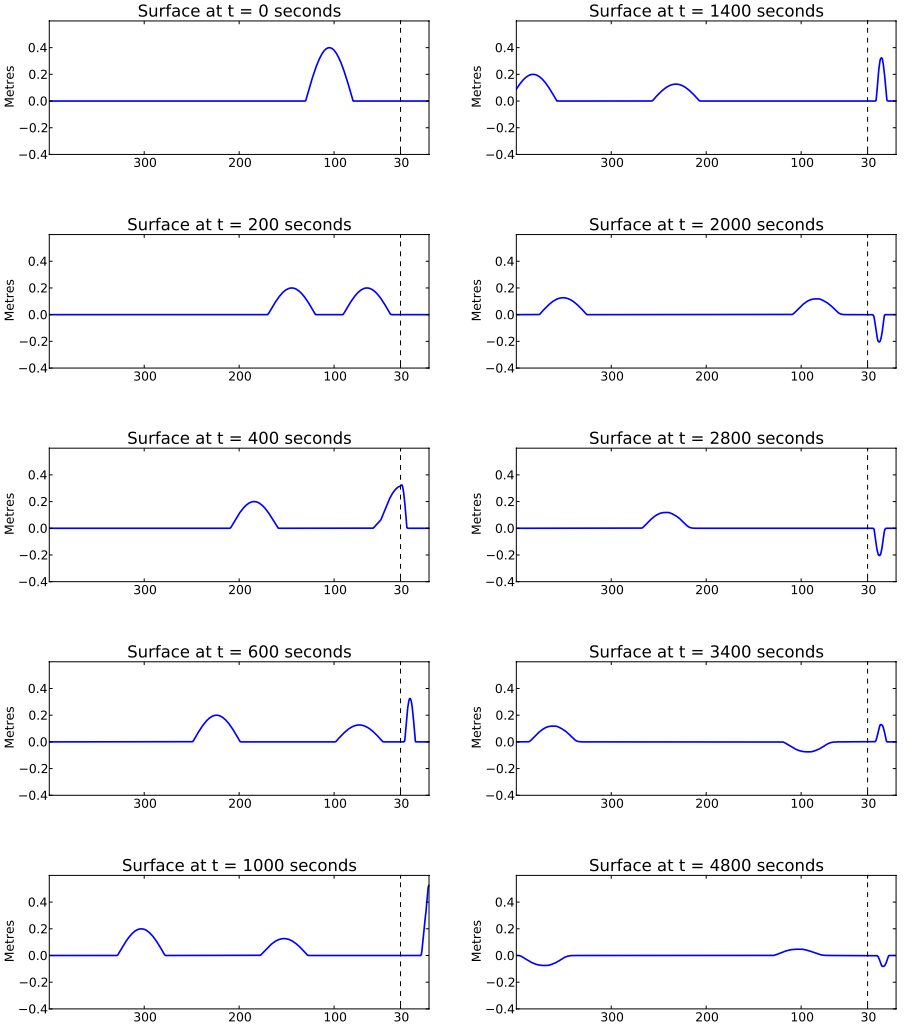


Figure 3.2. An idealized tsunami interacting with a step discontinuity representing a continental shelf. The dashed line indicates the location of the discontinuity, 30 km offshore. See Figure 3.3 for the same solution as a contour plot in the x–t plane.

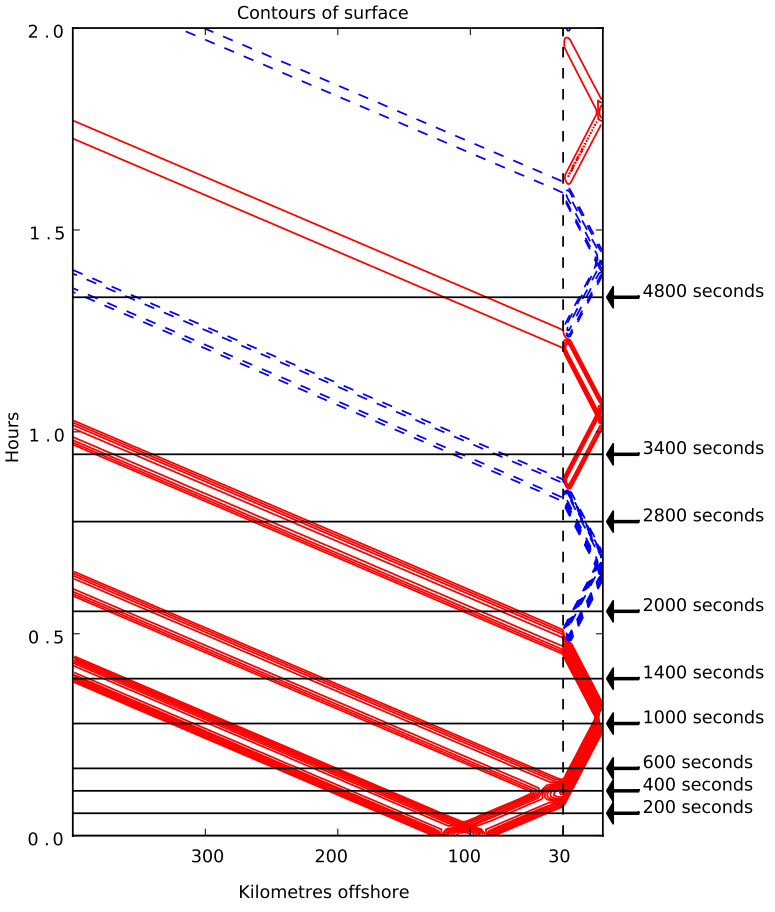


Figure 3.3. Contour plot in the x–t plane of an idealized tsunami interacting with a step discontinuity representing a continental shelf. Solid contour lines are at $0.025, 0.05, \ldots, 0.35$ m. Dashed contour lines are at -0.025, -0.05, -0.1, -0.15 m. This is a different view of the results shown in Figure 3.2, and the times shown there are indicated as horizontal lines.

Consider the first interaction of the wave shown in Figure 3.2 with the discontinuity. Note that the lower wave speed on the shelf results in a shorter-amplitude wave. To understand this, suppose the initial wave has wavelength W_ℓ. The tail of the wave reaches the step at time $\Delta t = W_\ell/\sqrt{gh_\ell}$ later than the front of the wave. At this time the front of the transmitted wave on the shallow side has moved a distance $\Delta t\sqrt{gh_r}$ and so the wavelength observed on the shallow side is $W_r = \sqrt{h_r/h_\ell}W_\ell < W_\ell$. The wavelength decreases by the same factor as the decrease in wave speed.

On the other hand, the amplitude of the transmitted wave is larger than the amplitude of the original wave by a factor $C_T > 1$, the *transmission coefficient*, while the reflected wave is smaller by a factor $C_R < 1$, the *reflection coefficient*. For the idealized step discontinuity, these coefficients are given by

$$C_T = \frac{2c_\ell}{c_\ell + c_r}, \quad C_R = \frac{c_\ell - c_r}{c_\ell + c_r}, \tag{3.32}$$

analogous to the transmission and reflection coefficients of linear acoustics, for example, at an interface between materials with different impedance. For the example shown in Figures 3.2 and 3.3, the coefficients are $C_T \approx 1.63$ and $C_R = C_T - 1 \approx 0.63$.

There are several ways to derive these coefficients. An approach that fits well here is to use the structure of the Riemann solution derived above, as is done for acoustics in LeVeque (2002). Consider a pure right-going wave consisting of a jump discontinuity of magnitude $\Delta\eta$ in depth, that hits the discontinuity in bathymetry at some time $\bar{t}$. From this time forward we have a Riemann problem in which $\Delta\mu = c_\ell\Delta\eta$ by the jump conditions across a right-going wave in the deep water. The Riemann solution consists of a left-going wave (the reflected wave) and a right-going wave (the transmitted wave) of the form (3.26), and the formulas (3.28) when applied to this particular Riemann data yield directly the coefficients (3.32). A more general waveform can be viewed as a sequence of small step discontinuities approaching the shelf, each of which must have the same relation between $\Delta\eta$ and $\Delta\mu$, and so each is split in the same manner into transmitted and reflected waves.

Note that if $c_\ell = c_r$ there is no discontinuity, and in this case $C_T = 1$ while $C_R = 0$. On the other hand, in the limiting case of very shallow water on the right, $C_T \to 2$ while $C_R \to 1$. This limiting case corresponds to a solid wall boundary condition, and this factor of 2 amplification is apparent at time $t = 1000$ s in Figure 3.2, when the wave is reflecting off the shore.

In general the amplification factor for a wave transmitted into shallower water is between 1 and 2, while the reflection coefficient is between 0 and 1 if $c_\ell > c_r$. When a wave is transmitted from shallow water into deeper water (*e.g.*, if $c_\ell < c_r$) then the reflection coefficient in (3.32) is negative,

explaining the negation of amplitude seen in Figures 3.2 and 3.3 when the trapped wave reflects off the discontinuity, for example between times 1400 and 2000 seconds in those plots.

We can also calculate the fraction of energy that is transmitted and reflected at the shelf. In a pure right-going wave (or a pure left-going wave) the energy is equally distributed between potential and kinetic energy by the equipartition principle. If $\eta(x)$ is the displacement of the surface from sea level $\eta_s = 0$ and $u(x)$ is the velocity of the fluid, then these are given by

$$\begin{aligned} \text{Potential energy} &= \int \frac{1}{2}\rho g \eta^2(x)\,\mathrm{d}x, \\ \text{Kinetic energy} &= \int \frac{1}{2}\rho u^2(x)\,\mathrm{d}x, \end{aligned} \tag{3.33}$$

where ρ is the density of the water. It is easy to check that these are equal for a wave in a single characteristic family (for the linearized equations about a constant depth h and zero velocity) by noting that the form of the eigenvectors (3.8) shows that $hu(x) = \pm\sqrt{gh}\,\eta(x)$ for each x. Let E_ℓ be the energy in the wave approaching the step. The reflected wave has the same shape but the amplitude of $\eta(x)$ is reduced by C_R everywhere, and hence the energy in the reflected wave is $C_R^2 E_\ell$. By conservation of energy, the amount of energy transmitted is $(1 - C_R^2)E_\ell$. This result can also be found by calculating the potential energy of the transmitted wave directly from the integral in (3.33), taking into account both the amplitude of the wave by the factor C_T and the reduction in wavelength by $\sqrt{h_r/h_\ell}$. For the example shown in Figures 3.2 and 3.3, approximately 60% of the energy is transmitted onto the shelf at the first reflection time. At the kth reflection of the wave trapped on the shelf, the energy radiated can be calculated to be $(1-C_R)^2 C_R^{(k-1)} E_\ell$. The total of the initially reflected energy plus all the radiated energy is given by an infinite series that sums to E_ℓ.

4. Finite volume methods

Before continuing our discussion of Riemann problems for the shallow water equations, we pause to introduce the basic ideas of finite volume methods, both as motivation and in order to see what information will be required from Riemann solutions.

Nonlinear hyperbolic systems (3.3) present some well-known difficulties for numerical solution, and a considerable amount of research has been dedicated to the development of suitable numerical methods for them; see LeVeque (2002) for an overview. A class of numerical methods that has been very successful for these problems are the shock-capturing *Godunov-type methods*: finite volume methods making use of Riemann problems to determine the numerical update.

In a one-dimensional finite volume method, the numerical solution Q_i^n is an approximation to the average value of the solution in the ith grid cell $\mathcal{C}_i = [x_{i-1/2}, x_{i+1/2}]$:

$$Q_i^n \approx \frac{1}{V_i} \int_{\mathcal{C}_i} q(x, t_n)\, \mathrm{d}x, \tag{4.1}$$

where V_i is the volume of the grid cell (simply the length in one dimension, $V_i = x_{i+1/2} - x_{i-1/2}$). The wave propagation algorithm updates the numerical solution from Q_i^n to Q_i^{n+1} by solving Riemann problems at $x_{i-1/2}$ and $x_{i+1/2}$, the boundaries of $\mathcal{C}_i$, and using the resulting wave structure of the Riemann problem to determine the numerical update. For a homogeneous system of conservation laws $q_t + f(q)_x = 0$, such methods are often written in *conservation form*,

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x}(F_{i+1/2}^n - F_{i-1/2}^n) \tag{4.2}$$

where $F_{i-1/2}^n$ is a *numerical flux* approximating the time average of the true flux across the left edge of cell $\mathcal{C}_i$ over the time interval:

$$F_{i-1/2}^n \approx \frac{1}{\Delta t} \int_{t_n}^{t_{n+1}} f(q(x_{i-1/2}, t))\, \mathrm{d}t. \tag{4.3}$$

If the method is in conservation form, then no matter how the numerical fluxes are chosen the method will be conservative: summing Q_i^{n+1} over all grid cells gives a cancellation of fluxes except for fluxes at the boundaries. The classical *Godunov's method* is obtained by solving the Riemann problem at each cell edge (using $\bar{x} = x_{i-1/2}$ and $\bar{t} = t_n$ in our general description of the Riemann problem, for example) and then evaluating the resulting Riemann solution at $x_{i-1/2}$ to define the numerical flux, setting

$$F_{i-1/2}^n = f(Q(x_{i-1/2})).$$

This gives a first-order accurate method that can be viewed as a generalization of the upwind method for scalar advection.

For equations (3.3) with a source term, one common approach is to use a *fractional step method* in which each time step is subdivided into a step on the homogeneous conservation law $q_t + f(q)_x = 0$, followed by a step on the source terms alone, solving $q_t = \psi(q, x)$. This approach generally works well for the friction or Coriolis terms in the shallow water equations, as discussed further in Section 7, but is not suitable for handling the bathymetry terms. For the steady state solution of the ocean at rest, the bathymetry source term must exactly cancel out the gradient of hydrostatic pressure that appears in the momentum flux. A fractional step method will not achieve this and will generate large spurious waves. Instead these source terms must be incorporated into the Riemann solution directly, as discussed further below.

To incorporate source terms, it is no longer possible to use the conservation form (4.2). Instead we will write the method in *fluctuation form*

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x}(\mathcal{A}^+ \Delta Q_{i-1/2}^n + \mathcal{A}^- \Delta Q_{i+1/2}^n), \tag{4.4}$$

where the vector $\mathcal{A}^+ \Delta Q_{i-1/2}^n$ represents the net effect of all waves propagating into the cell from the left boundary, while $\mathcal{A}^- \Delta Q_{i+1/2}^n$ is the net effect of all waves propagating into the cell from the right boundary. For a homogeneous conservation law, this will be conservative if we choose these fluctuations as a *flux-difference splitting* at each interface, so that for example

$$\mathcal{A}^- \Delta Q_{i-1/2}^n + \mathcal{A}^+ \Delta Q_{i-1/2}^n = f(Q_i^n) - f(Q_{i-1}^n). \tag{4.5}$$

When source terms are incorporated, the right-hand side of (4.5) must be suitably modified as discussed below.

The notation $\mathcal{A}^\pm \Delta Q$ is motivated by the linear case. If $f(q) = Aq$, then Godunov's method is the simple generalization of the scalar upwind method obtained by taking

$$\mathcal{A}^\pm \Delta Q_{i-1/2}^n = A^\pm (Q_i^n - Q_{i-1}^n), \tag{4.6}$$

where the matrices $A^\pm$ are defined by

$$A^\pm = R\Lambda^\pm R^{-1}, \quad \Lambda^\pm = \begin{bmatrix} (\lambda^1)^\pm & 0 \\ 0 & (\lambda^2)^\pm \end{bmatrix}, \tag{4.7}$$

where $\lambda^+ = \max(\lambda,\ 0)$ and $\lambda^- = \min(\lambda,\ 0)$. For the linearized shallow water equations, note that in the subcritical case these fluctuations are simply

$$\mathcal{A}^- \Delta Q_{i-1/2} = \hat{\lambda}^1 \mathcal{W}_{i-1/2}^1, \quad \mathcal{A}^+ \Delta Q_{i-1/2} = \hat{\lambda}^2 \mathcal{W}_{i-1/2}^2. \tag{4.8}$$

In the supercritical case, one of the fluctuations would be the zero vector while the other is the sum of $\hat{\lambda}^p \mathcal{W}_{i-1/2}^p$ over $p = 1, 2$, which gives the full jump in the flux difference $A(Q_i^n - Q_{i-1}^n)$.

4.1. Second-order corrections and limiters

Godunov's method is only first-order accurate and introduces a great deal of numerical diffusion into the solution. In particular, steep gradients are badly smeared out. To obtain a *high-resolution method*, we add additional terms to (4.4) that model the second derivative terms in a Taylor series expansion of $q(x, t + \Delta t)$ about $q(x, t)$, and then apply *limiters* to avoid the nonphysical oscillations that often arise near discontinuities when a dispersive second-order method is used. To maintain conservation, these corrections

can be expressed in a flux-differencing form, and so we replace (4.4) by

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x}(\mathcal{A}^+\Delta Q_{i-1/2}^n + \mathcal{A}^-\Delta Q_{i+1/2}^n) - \frac{\Delta t}{\Delta x}(\tilde{F}_{i+1/2}^n - \tilde{F}_{i-1/2}^n). \quad (4.9)$$

For a constant coefficient linear system, second-order accuracy is achieved by taking

$$\tilde{F}_{i-1/2}^n = \frac{1}{2}\left(I - \frac{\Delta t}{\Delta x}|A|\right)|A|(Q_i^n - Q_{i-1}^n), \quad (4.10)$$

where $|A| = R(\Lambda^+ - \Lambda^-)R^{-1}$. Inserting (4.10) and (4.6) into (4.9) and simplifying reveals that this is simply the Lax–Wendroff method,

$$Q_i^{n+1} = Q_i^n - \frac{1}{2}\frac{\Delta t}{\Delta x}A(Q_{i+1}^n - Q_{i-1}^n) + \frac{1}{2}\left(\frac{\Delta t}{\Delta x}\right)^2 A^2(Q_{i+1}^n - Q_i^n + Q_{i-1}^n). \quad (4.11)$$

Although this is second-order accurate on smooth solutions, the dominant term in the error is dispersive, and so non-physical oscillations appear near steep gradients. This can be disastrous, particularly if they lead to negative values of the depth. By viewing the Lax–Wendroff method in the form (4.9), as a modification to the upwind Godunov method, we can apply limiters to produce 'high-resolution' results. To do so, note that the correction flux (4.10) can be rewritten in terms of the waves $\mathcal{W}^1$ and $\mathcal{W}^2$ as

$$\tilde{F}_{i-1/2} = \frac{1}{2}\sum_{i=1}^{2}\left(1 - \frac{\Delta t}{\Delta x}|\lambda^p|\right)|\lambda^p|\mathcal{W}_{i-1/2}^p, \quad (4.12)$$

where we have dropped the time step index n and the superscript p refers to the wave family. We introduce limiters by replacing $\mathcal{W}_{i-1/2}^p$ by a limited version $\widetilde{\mathcal{W}}_{i-1/2}^p = \Phi(\theta_{i-1/2}^p)\mathcal{W}_{i-1/2}^p$, where $\theta_{i-1/2}^p$ is a scalar measure of the strength of the wave $\mathcal{W}_{i-1/2}^p$ relative to the wave in the same family arising from a neighbouring Riemann problem, while $\Phi(\theta)$ is a scalar-valued *limiter function* that takes values near 1 where the solution appears to be smooth and is typically closer to 0 near perceived discontinuities. See LeVeque (2002) for more details. There is a vast literature on limiter functions and methods with a similar flavour. Often the limiter is applied to the numerical flux function (giving flux-limiter methods) or to slopes in a reconstruction of a piecewise polynomial approximate solution from the cell averages (*e.g.*, slope limiter methods). The above formulation in terms of 'wave limiters' has the advantage that it extends very naturally to arbitrary hyperbolic systems of equations, even those that are not in conservation form. This *wave propagation* approach is the basic method used throughout the Clawpack software. The generalization to two space dimensions is briefly discussed in Section 6.

4.2. The f-wave formulation

Another formulation of the wave propagation algorithms known as the *f-wave form* has been found to be very useful in many contexts, including the incorporation of source terms as discussed below. An approximate Riemann solver generally produces a set of *wave basis vectors* $r^p_{i-1/2}$ (often as the eigenvectors of some matrix) and then determines the waves by decomposing the vector $Q_i - Q_{i-1}$ as a linear combination of these basis vectors,

$$Q_i - Q_{i-1} = \sum_p \alpha^p_{i-1/2} r^p_{i-1/2} \equiv \sum_p \mathcal{W}^p_{i-1/2}. \tag{4.13}$$

The f-wave approach instead splits the flux difference as a linear combination of these vectors,

$$f(Q_i) - f(Q_{i-1}) = \sum_p \beta^p_{i-1/2} r^p_{i-1/2} \equiv \sum_p \mathcal{Z}^p_{i-1/2}. \tag{4.14}$$

From this splitting we can easily define fluctuations $\mathcal{A}^{\pm}\Delta Q_{i-1/2}$ satisfying (4.5) by assigning the f-waves $\mathcal{Z}^p_{i-1/2}$ for which the corresponding eigenvalue or approximate wave speed is negative to $\mathcal{A}^-\Delta Q_{i-1/2}$, and the remaining f-waves to $\mathcal{A}^+\Delta Q_{i-1/2}$. For the linearized shallow water equations in the subcritical case, this reduces to

$$\begin{aligned} \mathcal{A}^-\Delta Q_{i-1/2} = \mathcal{Z}^1_{i-1/2}, \quad \mathcal{A}^+\Delta Q_{i-1/2} = \mathcal{Z}^2_{i-1/2}, \\ \tilde{F}_{i-1/2} = \frac{1}{2}\sum_{p=1}^{2}\left(1 - \frac{\Delta t}{\Delta x}|\hat{\lambda}^p|\right)\operatorname{sgn}(\lambda^p)\widetilde{\mathcal{Z}}^p_{i-1/2}, \end{aligned} \tag{4.15}$$

where $\widetilde{\mathcal{Z}}^p_{i-1/2}$ is a limited version of $\mathcal{Z}^p_{i-1/2}$. The f-waves are limited in exactly the same manner as waves $\mathcal{W}^p_{i-1/2}$ would be.

One advantage of this formulation is that the requirement (4.5) is satisfied no matter how the eigenvectors r^1 and r^2 are chosen for the nonlinear case. Another advantage is that source terms are easily included into the Riemann solver in a well-balanced manner.

5. The nonlinear Riemann problem

Although linearized equations may be suitable in deep water, as a tsunami approaches shore the nonlinearities cannot be ignored. In the nonlinear equations the characteristic speeds (eigenvalues of the Jacobian matrix) vary with the solution itself. Over flat bathymetry the fluid depth is greater at the peak of a wave than in the trough, so the peak travels faster and can even overtake the trough in water that is shallow relative to the wavelength. This wave breaking is clearly visible for ordinary wind-generated waves on

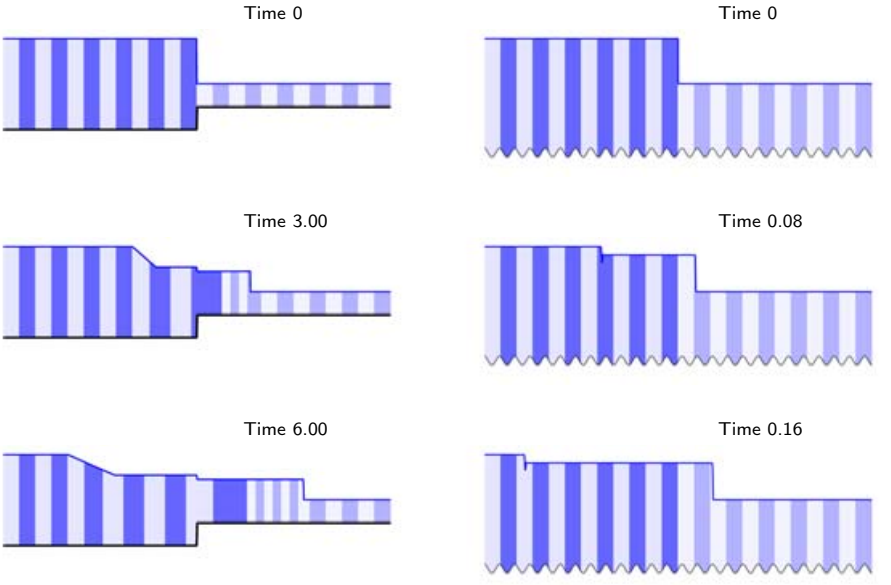


Figure 5.1. Solution to the 'dam-break' Riemann problem for the shallow water equations with initial velocity 0. The shading shows a passively advected tracer to help visualize the fluid velocities, compression, and rarefaction. The bathymetry is (a) $B_\ell = -1$ and $B_r = -0.5$, (b) $B_\ell = -4000$ and $B_r = -200$. In both cases, $\eta_\ell = 1$ and $\eta_r = 0$.

the ocean as they move into sufficiently shallow water in the surf zone. In the shallow water equations the depth must remain single-valued and so overturning waves cannot be modelled directly. Instead a *shock wave* forms, also called a *hydraulic jump* in shallow water theory. This models a *bore*, a near-discontinuity in the surface elevation that is often seen at the leading edge of tsunamis as they approach shore or propagate up a river.

The nonlinear Riemann problem over flat bathymetry can be solved and consists of two waves moving at constant velocities, though now each wave is generally either a shock wave (if characteristics are converging) or a spreading rarefaction wave (if characteristics are diverging, *i.e.*, the eigenvalue is strictly increasing from left to right across the wave). For details on solving the nonlinear Riemann problem exactly, see for example LeVeque (2002) or Toro (2001).

On varying topography we can consider a generalized Riemann problem in which the bathymetry is allowed to be discontinuous at the point $\bar{x}$ along with the state variables. The solution to this nonlinear Riemann problem

generally consists of three waves. In addition to the two propagating waves, which each propagate over flat bathymetry to one side or the other of $\bar{x}$ as in the linear case discussed above, there will also be a stationary wave (propagating with speed zero) at $\bar{x}$, where the jump in bathymetry leads to a jump in depth h, and also in the surface η if water is flowing across the step. This is illustrated in Figure 5.1. In the linearized model this stationary jump in η does not appear because the jump in the surface at a stationary discontinuity is of order u^2/gh for small perturbations. Figure 5.1(b) shows the solution to the nonlinear Riemann problem with the same jump in the surface η as in Figure 5.1(a), but over much deeper water. The spread of characteristics across the rarefaction wave is so small that it appears as a discontinuity and the fluid velocity is so small that the jump in surface at the stationary discontinuity can not be seen.

5.1. Approximate Riemann solvers

For the linearized shallow water equations on flat topography, the exact eigenstructure is known and easily used to compute the exact Riemann solution for any states Q_ℓ and Q_r, as has been done in Section 3.4. For the nonlinear problem, the exact solution is more difficult to compute and generally not worth the effort, since the waves and speeds are used in a finite volume method that introduces errors when computing cell averages in each time step. Since a Riemann problem is solved at every cell interface in each time step, the cost of the Riemann solver often dominates the computational cost of the method and it is important to develop efficient approximate solvers. Moreover, rarefaction waves such as those shown in Figure 5.1(a) are not directly handled by the wave propagation algorithms, which assume each wave is a jump discontinuity.

Instead of using the exact Riemann solution, most Godunov-type methods use approximate Riemann solvers. For GeoClaw we use approximate solvers that always return a set of waves (or f-waves) that are simple discontinuities propagating at constant speeds. These must be chosen in a manner that:

- gives a good approximation to the nonlinear Riemann solution,
- preserves steady states, in particular the ocean at rest,
- handles dry states $h_\ell = 0$ or $h_r = 0$,
- works well in conjunction with AMR.

The Riemann solver used in GeoClaw is rather complicated and will not be described in detail. We will just give a flavour of how it is constructed. Full details can be found in George (2006, 2008), and the dry state problem is discussed further in George (2010).

The f-wave approach developed in Section 3.5 is expanded to an *augmented Riemann solver* in which the vector

$$\begin{bmatrix} \Delta h \\ \Delta \mu \\ \Delta \phi \\ \Delta B \end{bmatrix} \tag{5.1}$$

is decomposed into 4 waves. Note that the first two components of this vector correspond to the jump in $q = (h,\ \mu)$ in the Riemann problem data, while the second and third components together correspond to the jump in flux $f(q) = (\mu,\ \phi)$. The jump in h is explicitly included in order to apply techniques that ensure that no negative depths are generated in the Riemann solutions near the shoreline.

The equations defining the Riemann problem consist of the equations (3.9) for $h,\ \mu$, and B, together with an equation for the momentum flux ϕ derived by differentiating

$$\phi = \mu/h + \frac{1}{2}gh^2 \tag{5.2}$$

with respect to t and using the equations for the time derivatives of h and μ to obtain

$$\phi_t + 2(u^2 - gh)\mu_x + 2u\phi_x + 2ghuB_x = 0. \tag{5.3}$$

This results in the non-conservative system

$$\begin{bmatrix} h \\ \mu \\ \phi \\ B \end{bmatrix}_t + \begin{bmatrix} 0 & 1 & 0 & 0 \\ gh - u^2 & 2u & 0 & gh \\ 0 & gh - u^2 & 2u & 2ghu \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} h \\ \mu \\ \phi \\ B \end{bmatrix}_x = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{5.4}$$

The eigenvalues of this matrix are

$$\lambda^1 = u - \sqrt{gh}, \quad \lambda^2 = u + \sqrt{gh}, \quad \lambda^3 = 2u, \quad \lambda^0 = 0, \tag{5.5}$$

and the corresponding eigenvectors are

$$r^1 = \begin{bmatrix} 1 \\ \lambda^1 \\ (\lambda^1)^2 \\ 0 \end{bmatrix}, \quad r^2 = \begin{bmatrix} 1 \\ \lambda^2 \\ (\lambda^2)^2 \\ 0 \end{bmatrix}, \quad r^3 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \quad r^0 = \begin{bmatrix} gh/\lambda^1\lambda^2 \\ 0 \\ -gh \\ 1 \end{bmatrix}. \tag{5.6}$$

Again the eigenvector r^0 corresponds to the stationary wave induced by the jump in topography. Note that the first component of r^0 can be written as $-1/(1 - u^2/gh)$ and for zero velocity reduces to -1, corresponding to the jump $\Delta h = -\Delta B$ that gives the ocean at rest, $\Delta\eta = 0$. It is shown in George (2006, 2008) that a well-balanced method for both the ocean at rest and also a flowing steady state is obtained by defining a discrete

approximation to the steady wave as

$$\mathcal{W}^0 = \Delta B \begin{bmatrix} -\rho_1 \\ 0 \\ -g\bar{h}\rho_2 \\ 1 \end{bmatrix}, \tag{5.7}$$

where $\bar{h} = (h_\ell + h_r)/2$ and the ratios ρ_1 and ρ_2 are nearly 1 for small velocities:

$$\rho_1 = \frac{g\bar{h}}{g\bar{h} - \bar{u}^2}, \quad \rho_2 = \frac{\max(u_\ell u_r, 0) - g\bar{h}}{\bar{u}^2 - g\bar{h}}, \tag{5.8}$$

where $\bar{u} = \frac{1}{2}(u_\ell + u_r)$. Subtracting this wave from the vector (5.1) reduces the problem to a system of three equations for the remaining waves.

The eigenvalues λ^1 and λ^2 are replaced by wave speeds s^1 and s^2 estimated from the Riemann data, and these values are also used in the discrete eigenvectors r^1 and r^2. The wave speeds are approximated using a variant of the approach suggested by Einfeldt (1988) in connection with the HLL solver of Harten, Lax and van Leer (1983) to avoid difficulties with the vacuum state in gas dynamics, which is analogous to the dry state problem in shallow water. This HLLE solver is further discussed in Einfeldt, Munz, Roe and Sjogreen (1991) and elsewhere. These *HLLE speeds* are given by

$$s^1 = \min(\hat{s}^1,\ u_\ell - c_\ell), \quad s^2 = \max(\hat{s}^2,\ u_r + c_r) \tag{5.9}$$

where $\hat{s}^1$ and $\hat{s}^2$ are the speeds used in the *Roe solver* for the shallow water equations,

$$\hat{s}^1 = \bar{u} - \hat{c}, \quad \hat{s}^2 = \bar{u} + \hat{c}, \tag{5.10}$$

where

$$\hat{c} = \sqrt{g\bar{h}}, \quad \hat{u} = \frac{u_\ell\sqrt{h_\ell} + u_r\sqrt{h_r}}{\sqrt{h_\ell} + \sqrt{h_r}}. \tag{5.11}$$

The wave decomposition is then done by solving the linear system to determine the weights β^1, β^2, and β^3 in

$$\begin{bmatrix} \Delta h \\ \Delta\mu \\ \Delta\phi \end{bmatrix} = \beta^1 \begin{bmatrix} 1 \\ s^1 \\ (s^1)^2 \end{bmatrix} + \beta^2 \begin{bmatrix} 1 \\ s^2 \\ (s^2)^2 \end{bmatrix} + \beta^3 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \tag{5.12}$$

Further improvements can be made by replacing the third eigenvector by a different choice in certain situations, as discussed further in George (2008). Finally, the second and third components of these waves are used as f-waves in the algorithm described in Section 4.2 along with the wave speeds s^1, s^2, and $s^3 = 2\hat{u}$. This results in a method that conserves mass (and momentum when $\Delta B = 0$), avoids dry states, and is well-balanced. A number of related Riemann solvers and Godunov-type methods have been

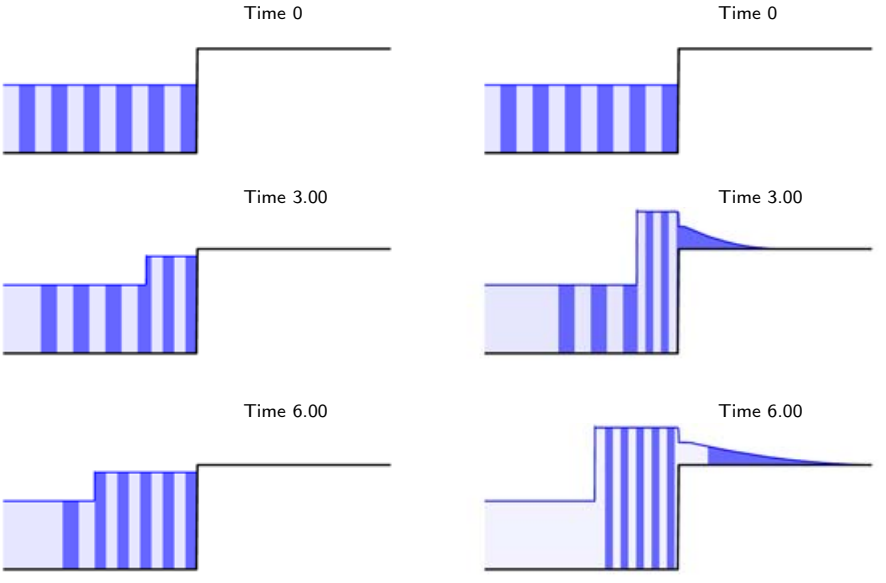


Figure 5.2. Solution to the Riemann problem for the shallow water equations with a dry state on the right and positive velocity in the left state. The velocity is larger in the case shown in column (b). The shading shows a passively advected tracer to help visualize the fluid velocities, compression, and rarefaction.

proposed in the literature that can also achieve these goals. The approach outlined above that splits the jump in q and in $f(q)$ is also related to the relaxation approaches discussed in Bouchut (2004) and LeVeque and Pelanti (2001). See also Bale *et al.* (2002), Gosse (2001, 2000) and In (1999).

Riemann problems with an initial dry state on one side raise additional issues that we will not discuss in detail here. Figure 5.2 shows two examples to illustrate one aspect of this problem. In each case there is a step discontinuity in bathymetry with the left cell wet and the right cell dry, data of the sort that naturally arise along the shoreline. In the case illustrated in Figure 5.2(a), the velocity in the left state is positive but sufficiently small that the step discontinuity acts as a solid wall and the Riemann solution consists of a left-moving 1-shock, with stationary water to the right of the shock. The case illustrated in Figure 5.2(b) has a larger positive fluid velocity, in which case the flow overtops the step and there is a right-going 1-rarefaction invading the dry cell along with a left-going 1-shock. For more details about the handling of dry states in the Riemann solver used in GeoClaw, see George (2008, 2010).

6. Algorithms in two space dimensions

In two space dimensions, hyperbolic systems such as (3.2) more generally take the form

$$q_t + f(q)_x + g(q)_y = \psi(q, x, y). \tag{6.1}$$

Godunov-type finite volume algorithms can be naturally extended to two dimensions by solving 1D Riemann problems normal to each edge of a finite volume cell, and using the Riemann solution to define an edge flux or a set of waves propagating into the neighbouring cells. High-resolution correction terms can then be added to achieve greater accuracy without spurious oscillations. The methods used in GeoClaw are the standard wave propagation algorithms of Clawpack, which are described in detail in LeVeque (2002). For a logically rectangular quadrilateral grid, the cells can be indexed by (i, j) and each cell has four neighbours. In this case the numerical solution Q_{ij}^n is an approximation to the average value of the solution over the grid cell $\mathcal{C}_{ij}$,

$$Q_{ij}^n \approx \frac{1}{V_{ij}} \int_{\mathcal{C}_{ij}} q(x, y, t_n)\, \mathrm{d}x\, \mathrm{d}y, \tag{6.2}$$

where V_{ij} is the area of the cell. For a regular Cartesian grid, the cell areas are simply $V_{ij} = \Delta x \Delta y$, but the methods can also be applied on any quadrilateral grid defined by a mapping of the uniform computational grid.

The basic idea of the wave propagation algorithms in two dimensions is illustrated in Figure 6.1, where six quadrilateral grid cells are shown. Figure 6.1(a) shows the left-going and right-going waves that might be generated by solving the Riemann problem normal to the cell edge in the middle of this patch. The shallow water equations are rotationally invariant, and the Riemann problem normal to any edge can easily be solved by rotating the momentum components of the cell averages Q to normal and tangential components. The normal components are used in solving a 1D Riemann problem along with the depth h on either side. The jump in tangential velocity is simply advected by a third wave propagating at the intermediate velocity found from the 1D Riemann solution.

Using these waves to update the cell averages in the two cells neighbouring this edge gives the natural generalization of Godunov's method, which is first-order accurate and stable only for Courant numbers up to 0.5 (because of the waves that also enter the cell from above and below when solving Riemann problems in the orthogonal direction).

To increase the accuracy we need to add second-order correction terms that model the next terms in a Taylor series expansion of the solution at the end of the time step about the starting values, requiring an estimate of q_{tt}. In the Lax–Wendroff framework used in the wave propagation algorithms, this is replaced by spatial derivatives by differentiating the original

system of equations in time. The result involves q_{xx} and q_{yy} and these terms can be incorporated by a direct extension of the one-dimensional correction terms, with limiters used as in one dimension to give high resolution (sharp gradients without overshoots or undershoots). The time derivative q_{tt} also involves mixed derivatives q_{xy} and it is important to include these terms as well, both to achieve full second-order accuracy and also to improve the stability properties of the method. The cross-derivative terms are included by taking the waves propagating normal to the interface shown in Figure 6.1(a) and splitting each wave into up-going and down-going pieces that modify the cell averages above or below. This is accomplished by decomposing the fluctuations $\mathcal{A}^{\pm}\Delta Q$ into eigenvectors of the Jacobian matrix in the transverse direction (tangent to the cell interface we started with). The resulting eigen-decomposition is used to split each of the fluctuations into an down-going part (illustrated in Figure 6.1(b)) and a up-going part (illustrated in Figure 6.1(c)), and is done in the *transverse Riemann solver* of Clawpack. The triangular portions of these waves that lie in the adjacent row of grid cells can be used to define a flux from the cells in the middle row to the cells in the bottom or top row of cells respectively. The algorithms must of course be modified to take into account the areas swept out by the waves relative to the area of the grid cells in order to properly update cell averages. This approach is described in more detail in LeVeque (2002) and has been successfully used in solving a wide variety of hyperbolic systems in two space dimensions, and also in three dimensions after introducing an additional set of transverse terms (Langseth and LeVeque 2000). See also LeVeque (1996) for a simpler discussion in the context of advection equations.

With the addition of these transverse terms, the resulting method is stable up to a Courant number of 1. The methods can be used on an arbitrary logically rectangular grid: the mapping from computational to physical space need not be smooth, an advantage for some applications such as the quadrilateral grid on the sphere used for AMR calculations in Berger, Calhoun, Helzel and LeVeque (2009).

6.1. Ghost cells and boundary conditions

Boundary conditions are imposed by introducing an additional two rows of grid cells (called *ghost cells*) around the edge of the grid. In each time step values of Q are set in these cells in some manner, depending on the physical boundary condition, and then the finite volume method is applied over all cells in the original domain. Updating cells adjacent to the original boundaries will use ghost cell values in determining the update, and in this way the physical boundary conditions indirectly affect the solution.

For tsunami modelling we typically take the full domain to be sufficiently large that any waves leaving the domain can be safely ignored; we assume

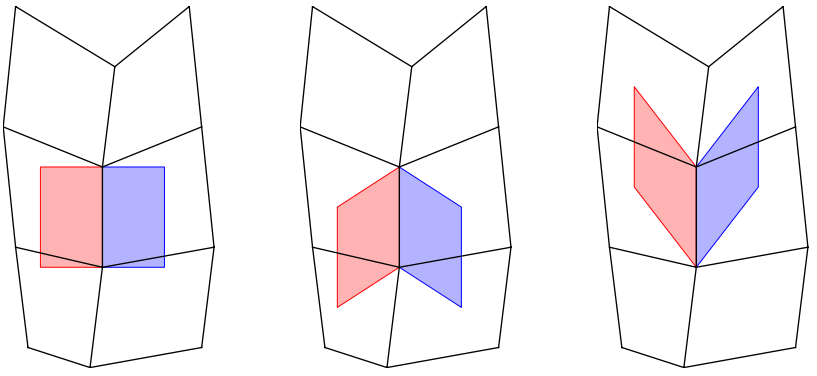


Figure 6.1. (a) Six quadrilateral grid cells and the waves moving normal to a cell interface after solving the normal Riemann problem. (b) Down-going portions of these waves resulting from transverse Riemann solve. (c) Up-going portions of these waves resulting from transverse Riemann solve.

they should not later reflect off a physical feature and re-enter the domain. So we require non-reflecting boundary conditions (also called absorbing boundary conditions) that allow outgoing waves to leave the domain without unphysical numerical reflections at the edge of the computational domain. For Godunov-type methods such as the wave propagation methods we employ, a very simple extrapolation method gives a reasonable non-reflecting boundary condition as discussed in LeVeque (2002): in each time step we simply copy the values of Q in the cells adjacent to each boundary into the adjacent ghost cells. Solving a Riemann problem between two identical states results in zero-strength waves and so the Riemann problems at the cell interfaces at the domain boundary give no spurious incoming waves. This is illustrated in Figure 12.1, for example, where the tsunami is seen to leave the computational grid with very little spurious reflection.

When adaptive mesh refinement is used, many grid patches will have edges that are within the full computational domain. In this case ghost cell values are filled either from an adjacent grid at the same level of refinement, if such a grid exists, or by interpolating from coarser levels. This is described further in Section 9. It is important to ensure that spurious waves are not generated from internal interfaces between grids at different levels. Again Godunov-type methods seem to handle this quite well, as is also apparent from the results shown in Figure 12.1, for example.

6.2. Solving on the sphere

To properly model the propagation of tsunamis across the ocean, it is necessary to solve the shallow water equations on the surface of the sphere rather than in Cartesian coordinates. This can be done using the approach discussed above and illustrated in Figure 6.1, where now the cell area is calculated as an area on the sphere. The coordinate lines bounding the quadrilaterals are assumed to lie along great circles on the sphere between the corner vertices and so these areas are easily computed.

The current implementation in GeoClaw assumes that latitude–longitude coordinates are used on the sphere. This gives some simplification of the Riemann solvers since the cell edges are then orthogonal to one another and the momenta that are stored in the Q vectors are the components of momentum in these two directions. Latitude–longitude grids are generally used for teletsunami modelling since interest is generally focused on the mid-latitudes. To obtain an accurate representation of flow on the sphere, it is necessary to compute the cell volumes V_{ij} using surface area on the sphere. The grid cells are viewed as patches of the sphere obtained by joining the four corners by great circle arcs between the specified latitude and longitude values. The length of the cell edges also come into the finite volume methods and must be calculated using great circle distance.

On the full sphere, latitude–longitude coordinates have the problem that grid lines coalesce at the poles. The cells are very small near the poles relative to those near the equator, requiring very small time steps in order to keep the global Courant number below 1. A variety of other grids have been proposed for solving problems on the full sphere, particularly in atmospheric sciences where flow at the poles is an important part of the solution. One approach that fits well with the AMR algorithms described in this paper is discussed in Berger *et al.* (2009).

7. Source terms for friction

Topographic source terms are best incorporated into the Riemann solver, as described in Section 3.1. Additional source terms arise from bottom friction in shallow water, and are particularly important in modelling inundation. Run-up and inundation distance are affected by the roughness of the terrain, and would be much larger on a bare sandy beach than through a mangrove swamp, for example.

To model friction, we replace the momentum equations of (3.2) by

$$(hu)_t + (hu^2 + \tfrac{1}{2}gh^2)_x + (huv)_y + ghB_x = -D(h,u,v)hu, \tag{7.1a}$$

$$(hv)_t + (huv)_x + (hv^2 + \tfrac{1}{2}gh^2)_y + ghB_y = -D(h,u,v)hv, \tag{7.1b}$$

with some frictional drag coefficient $D(h,u,v)$. Various models are available

in the literature. We generally use the form

$$D(h,u,v) = n^2 g h^{-7/3}\sqrt{u^2+v^2}. \tag{7.2}$$

The parameter n is the *Manning coefficient* and depends on the roughness. If detailed information about the surface is known then this could be a spatially varying parameter, but for generic tsunami modelling a constant value of $n = 0.025$ is often used.

Note that in deep water the friction term in (7.1) is generally negligible, being of magnitude $\mathcal{O}(|u|^2 h^{-4/3})$, and so we only apply these source terms in coastal regions, *e.g.*, in depths of 100 m or less. In these regions the source term is applied as an update to momentum at the end of each time step. We loop over all grid cells and in shallow regions update the momenta $(hu)_{ij}$ and $(hv)_{ij}$ by

$$\begin{aligned} D_{ij} &= n^2 g h_{ij}^{-7/3}\sqrt{u_{ij}^2+v_{ij}^2}, \\ (hu)_{ij} &= (hu)_{ij}/(1+\Delta t D_{ij}), \\ (hv)_{ij} &= (hv)_{ij}/(1+\Delta t D_{ij}), \end{aligned} \tag{7.3}$$

This corresponds to taking a step of a linearized backward Euler method on the ordinary differential equations for momentum obtained from the source alone. By using backward Euler, we ensure that the momentum is driven to zero when $\Delta t D_{ij}$ is large, rather than potentially changing sign as might happen with forward Euler, for example. A higher-order method could be used, but given the uncertainty in the Manning coefficient (and indeed in the friction model itself), this would be of questionable value. Including friction is particularly important at the shoreline where the depth h approaches zero, and the above procedure helps to stabilize the method and ensure that velocities remain bounded as the shoreline moves while a wave is advancing or retreating.

8. Adaptive mesh refinement

In this section we will first describe the general block-structured adaptive mesh refinement (AMR) algorithms that are widely used on structured logically rectangular grids. This approach is discussed in numerous papers including Berger and Oliger (1984) and Berger and Colella (1989). The implementation specific to Clawpack and hence to GeoClaw is described in more detail in Berger and LeVeque (1998). We will summarize the basic approach and then concentrate on some of the challenges that arise when combining AMR with geophysical flow algorithms, in particular in dealing with dry states and with the need for well-balanced algorithms that maintain steady states.

8.1. AMR overview

Block-structured AMR algorithms are designed to solve hyperbolic systems on a hierarchy of logically rectangular grids. A single coarse (level 1) grid comprises the entire domain, while grids at a given level $\ell+1$ are finer than the coarser level ℓ grids by fixed integer refinement ratios r_x^ℓ and r_y^ℓ in the two spatial directions,

$$\Delta x^{\ell+1} = \Delta x^\ell / r_x^\ell, \quad \Delta y^{\ell+1} = \Delta y^\ell / r_y^\ell. \tag{8.1}$$

In practice we normally take $r_x^\ell = r_y^\ell$ at each level since in this application there is seldom any reason to refine differently in the two spatial directions. The nesting requirements of subgrids are not restrictive, in that a single level $(\ell+1)$ grid may overlap several level ℓ grids, and may be adjacent to level $(\ell-1)$ grids.

Since subgrids at a given level can appear and disappear adaptively, the highest grid level present at a given point in the domain changes with time. The subgrid arrangement changes during the process of regridding, which occurs every few time steps. This allows subgrids to essentially 'move' with features in the solution. On the current set of grids, the solution on each grid is advanced using the same numerical method that would be used on a single rectangular grid, together with some special procedures at the boundaries of subgrids.

The time steps on level $\ell+1$ grids are typically smaller than the time step on the level ℓ grids by a factor r_t^ℓ. Since Godunov-type explicit methods like the wave propagation method are stable only if the Courant number is bounded by 1, it is common practice to choose the same refinement factor in time as in space, $r_t^\ell = r_x^\ell = r_y^\ell$, since this usually leads to the same Courant number on the finer grids as on the coarser grid. The Courant number can be thought of as a measure of the fraction of a grid cell that a wave can traverse in one time step, and is given by $|s_{\max}\Delta t/\Delta x|$, where $s_{\max}$ is the maximum wave speed over the grid.

However, for tsunami applications of the type considered in this paper, it is often desirable to choose r_t^ℓ to be smaller than the spatial refinement factor for the levels ℓ corresponding to the finest grids, which are often introduced only near the shoreline in regions where run-up and inundation are to be studied. This is because the Courant number is based on the wave speed $|u \pm \sqrt{gh}| \approx \sqrt{gh}$, which depends on the water depth. For grids that are confined to coastal regions, h is much smaller than on the coarser grids that cover the ocean. If the coarsest grid covers regions where the ocean is 4000 m deep while a fine level is restricted to regions where the depth is at most 40 m, for example, then refining by the same factor in space and time would lead to a Courant number of 0.1 or less on the fine grid and potentially require 10 times as many time steps on the fine grids than

are necessary for stability. Since the vast majority of grid cells are often associated with fine grids near the shore, this can have a huge impact on the efficiency of the method (and also its accuracy, since solving a hyperbolic equation with very small Courant number introduces additional numerical viscosity and is typically less accurate than if a larger time step is used).

We will first give a brief summary of the AMR integration algorithm and the regridding strategy. We then focus on the modifications that are required for tsunami modelling, which are also important in modelling other depth-averaged geophysical flows of the type mentioned in Section 1.

8.2. AMR procedure

The basic AMR integrating algorithm applies the following steps recursively, starting with the coarsest grids at level $\ell = 1$.

AMR Integration Strategy.

(1) Take a time step of length Δt^ℓ on all grids at level ℓ.

(2) Using the solution at the beginning and end of this time step, perform space–time interpolation to determine ghost cell values for all level $\ell+1$ grids at the initial time and all $r_t^\ell - 1$ intermediate times, for any ghost cells that do not lie in adjacent level $\ell + 1$ grids. (Where there is an adjacent grid at the same level, values are copied directly into the ghost cells at each intermediate time step.)

(3) Take r_t^ℓ time steps on all level $\ell + 1$ grids to bring these grids up to the same advanced time as the level ℓ grids.

(4) For any grid cell at level ℓ that is covered by a level $\ell + 1$ grid, replace the solution Q in this cell by an appropriate average (described in Section 9) of the values from the $r_x^\ell r_y^\ell$ grid cells on the finer grid that cover this cell.

(5) Adjust the coarse cell values adjacent to fine grids to maintain conservation of mass (and of momentum in regions where the source terms vanish). This step is described in more detail in Section 9.4, after discussing the interpolation issues.

After each of the level $\ell+1$ time steps in step (3) above, the same algorithm is applied recursively to advance even finer grids (levels $\ell + 2, \ldots$).

Every few time steps on each level a *regridding step* is applied (except on the finest allowed level). The frequency depends on how fast the waves are moving, and how wide a *buffer region* around the grid patches there is. The larger the buffer region, the less frequently regridding needs to be performed. On the other hand a wide buffer region results in more grid cells to integrate on the finer level. We typically use a buffer width of 2 or 3 cells and regrid every 2 or 3 time steps on each level.

AMR Regridding Algorithm.

(1) Flag cells at level ℓ that require refinement to level $\ell + 1$. Our flagging strategy for tsunami modelling is summarized below.

(2) Cluster the flagged cells into rectangular patches using the algorithm of Berger and Rigoutsos (1991). This heuristic tries to strike a balance between minimizing the number of grids (to reduce patch overhead), and minimizing the number of unnecessarily refined cells when clustering into rectangles.

(3) Initialize the solution on each level $\ell+1$ grid. For each cell, either copy the data from an existing level $\ell + 1$ grid or, if no such grid exists at this point, interpolate from level ℓ grids using procedures described in the next section.

8.3. AMR cell flagging criteria

Depending on the application, a variety of different criteria might be used for flagging cells. In many applications an error estimation procedure or a feature detection algorithm is applied to all grid points on levels $l < L_{\max}$, where $L_{\max}$ is the maximum number of levels allowed. Cells where a threshold is exceeded are flagged for inclusion in a finer grid patch. A common choice is to compute the spatial gradient of one or more components of the solution vector q. For the simulation of tsunamis, we generally use the elevation of the sea surface relative to sea level, $|h + B - \eta_s|$. This is non-zero only in the wave and is a much better flagging indicator than the gradient of h, for example, which can be very large even in regions where the ocean is at rest due to variations in topography.

The sheer scale of tsunami modelling makes it necessary to allow much more refinement in some spatio-temporal regions than in others. In particular, the maximum refinement level and refinement ratios may be chosen to allow a very fine resolution of some regions of the coast that are of particular interest, for example a harbour or bay where a detailed inundation map is desired. Other regions of the coast may be of less interest and require less refinement. We may also wish to allow much less refinement away from the coast where the tsunami can be well represented on a much coarser grid. Conversely it is sometimes useful to *require* refinement up to a given level in certain regions. This is useful, for example, to force some refinement of a region before the wave arrives. These regions of required or allowed refinement may vary with time, since one part of the coast may be of interest at early times and another part of the coast (more distant from the source) of interest at later times. To address this, in the GeoClaw software the user

can specify a set of space–time regions of the form

$$L_1,\ L_2,\ x_1\ , x_2,\ y_1,\ y_2,\ t_1,\ t_2$$

to indicate that on the given space–time rectangle $[x_1, x_2] \times [y_1, y_2] \times [t_1, t_2]$, refinement to at least level L_1 is required, and to at most level L_2 is allowed.

9. Interpolation strategies for coarsening and refining

If the refinement level increases in a region during regridding, the solution in the cells of the finer grid may need to be interpolated from coarser levels in order to initialize the new grids. In the other direction, averaging from fine grids to coarser underlying grids is done in step (4) of the AMR algorithm of Section 8.2. This produces the best possible solution on the coarse grid at each time. When a fine grid disappears in some region during regridding, the remaining coarser grid already contains the averaged solution based on the finer grid, and so no additional work is required to deal with coarsening during the regridding stage.

We will first discuss refining and coarsening in the context of a one-dimensional problem where it is easier to visualize. The formulas we develop all extend in a natural way to the full two-dimensional case, discussed in Section 9.3.

When refining and coarsening it is important to maintain the steady states of an ocean at rest. This is particularly important since refinement often occurs just before the tsunami wave arrives in an undisturbed area of the ocean, and coarsening occurs as waves leave an area and the ocean returns to a steady state.

Since the interpolation procedures are intimately tied to the representation of the bathymetry and its interpolation between grids at different levels, we start the discussion there. We consider a cell $\mathcal{C}_k^\ell$ at some level ℓ, and say that a cell $\mathcal{C}_i^{\ell+1}$ at the finer level is a *subcell* of $\mathcal{C}_k^\ell$ if it covers a subset of the interval $\mathcal{C}_k^\ell$ (recall we are still working in one space dimension). The set of indices i for which $\mathcal{C}_i^{\ell+1}$ is a subcell of $\mathcal{C}_k^\ell$ will be denoted by Γ_k^ℓ. We will say that the topography is *consistent* between the different levels if the topography value B_k^ℓ in a cell at level ℓ is equal to the average of the values $B_i^{\ell+1}$ in all subcells of $\mathcal{C}_k^\ell$ at level $\ell + 1$:

$$B_k^\ell = \frac{1}{r_x^\ell} \sum_{i \in \Gamma_k^\ell} B_i^{\ell+1}. \tag{9.1}$$

If the cells have non-uniform sizes, for example on a latitude–longitude grid in two dimensions, then this formula generalizes to the requirement that

$$B_k^\ell = \frac{1}{V_k^\ell} \sum_{i \in \Gamma_k^\ell} V_i^{\ell+1} B_i^{\ell+1}. \tag{9.2}$$

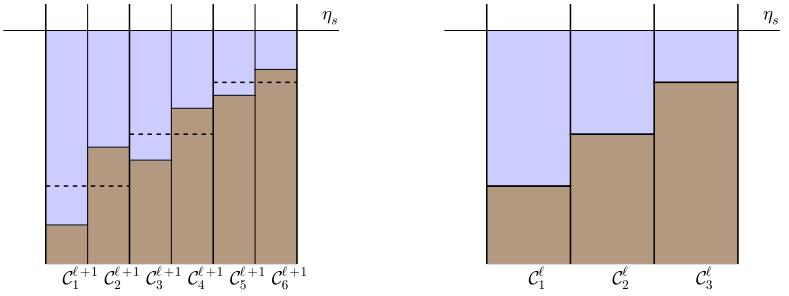


Figure 9.1. (a) Level $\ell+1$ topography and water depth with a constant sea surface elevation η_s. The dashed lines are the level ℓ topography. (b) Level ℓ topography and water depth on the coarse grid.

where the cell volumes (lengths in 1D, areas in 2D) satisfy

$$V_k^\ell = \sum_{i \in \Gamma_k^\ell} V_i^{\ell+1}. \tag{9.3}$$

Since a discussion of this consistency is most relevant in 2D, we will defer discussion of how we accomplish (9.2) to Section 9.3, and assume that it holds for now.

9.1. Coarsening and refining away from shore

We first consider a situation such as illustrated in Figure 9.1, where all the cells are wet on both levels. In this figure and the following figures, the darker region is the earth below the topography B_i and the lighter region is the water between B_i and $\eta_i = B_i + h_i$. The coarse-grid topography of Figure 9.1(b) (which is also shown as a dashed line in Figures 9.1(a) and (b)) is consistent with the fine grid topography: each coarse-grid value of B is the average of the two fine grid values. The water depths illustrated in Figures 9.1(a) and 9.1(b) are consistent with each other (the total mass of water is the same) and both correspond to an undisturbed ocean with $\eta \equiv \eta_s$.

Suppose we are given the solution $Q_i^{\ell+1}$ for $i = 1, 2, \ldots, 6$ on the fine (level $\ell+1$) grid shown in Figure 9.1(a) and we wish to coarsen it to obtain Figure 9.1(b). Assume the topography $B_i^{\ell+1}$ is consistent, so that

$$B_k^\ell = \frac{1}{2}(B_{2k-1}^{\ell+1} + B_{2k}^{\ell+1}), \quad k = 1,\ 2,\ 3. \tag{9.4}$$

To compute the water depth h_k^ℓ in the coarser cells we can simply set

$$h_k^\ell = \frac{1}{2}(h_{2k-1}^{\ell+1} + h_{2k}^{\ell+1}), \quad k = 1,\ 2,\ 3. \tag{9.5}$$

This preserves the steady state of water at rest $B_i^{\ell+1} + h_i^{\ell+1} = \eta_s$ for all $i \in \Gamma_k^\ell$, since then $B_k^\ell + h_k^\ell = \eta_s$ as well. More generally, with an arbitrary refinement factor r_x^ℓ and possibly varying cell volumes, we would set

$$h_k^\ell = \frac{1}{V_k^\ell} \sum_{i \in \Gamma_k^\ell} V_i^{\ell+1} h_i^{\ell+1}. \tag{9.6}$$

The momentum μ_k^ℓ can be averaged from level $\ell+1$ to level ℓ in the same manner, replacing h by μ in (9.6).

To go in the other direction, now suppose we are given the coarse-grid solution of Figure 9.1(b) and wish to interpolate to the fine grid, for example after a new grid is created. We would like to obtain Figure 9.1(a) on the fine grid in this case, with the flat water surface preserved. Unfortunately, the standard approach using linear interpolation of the conserved variables in the coarse cell and evaluating them at the centre of each fine grid cell described in Berger and LeVeque (1998) works very well for most conservation laws but fails miserably here.

For the data shown in Figure 9.1(b), the depth is decreasing linearly over the three coarse-grid cells. Using this linear function as the interpolant to compute the fluid depth h in the cells $\mathcal{C}_i^{\ell+1}$ on the finer grid would conserve mass but would not preserve the sea surface, because the fine grid bathymetry is not varying linearly. Variation in the sea surface would generate gradients of $h + B$ and hence spurious waves. In tsunami calculations on coarse ocean grids this interpolation strategy can easily generate discontinuities in the surface level on the order of tens or hundreds of metres, destroying all chances of modelling a tsunami. Instead, the interpolation must be based on surface elevation $\eta_k^\ell = B_k^\ell + h_k^\ell$, which for Figure 9.1(b) would all be equal to η_s. We construct a linear interpolant to these data over each grid cell and evaluate this at the fine cell centres to obtain values $\eta_i^{\ell+1}$, and then set $h_i^{\ell+1} = \eta_i^{\ell+1} - B_i^{\ell+1}$. The interpolant in coarse cell k is

$$\eta(x) = \eta_k^\ell + \sigma_k^\ell (x - x_k^\ell), \tag{9.7}$$

where x_k^ℓ is the centre of this cell. The slope σ_k^ℓ is chosen to be

$$\sigma_k^\ell = \text{minmod}(\eta_k^\ell - \eta_{k-1}^\ell,\ \eta_{k+1}^\ell - \eta_k^\ell)/\Delta x^\ell. \tag{9.8}$$

We generally use the standard minmod function (*e.g.*, LeVeque (2002)), which returns the argument of minimum modulus, or zero if the two arguments have opposite sign. This interpolation strategy prevents the introduction of new extrema in the water surface elevation, preserves a flat sea surface (provided that all depths are positive), and produces Figure 9.1(a) from the data in Figure 9.1(b).

In a tsunami wave the sea surface is not flat but is nearly so, and the surface is a smoothly varying function of x even when the topography varies

rapidly. The approach of interpolating the water surface elevation also works well in this case and produces a second-order accurate approximation to smooth waves. Note that although we switch to the variable η when doing the interpolation, the equations are still being solved in terms of the conserved quantities.

To interpolate the momentum, we begin with the standard approach; we determine a linear interpolant to μ_k^ℓ and then evaluate this at the fine cell centres. Again we use minmod slopes to prevent the introduction of new local extrema in momentum. However, because we might be interpolating to fine cells that are much shallower than the coarse cell, we ensure that the interpolation does not introduce new extrema in velocities as well. We check the velocities in the fine cells, defined by $\mu_i^{\ell+1}/h_i^{\ell+1}$, for all $i \in \Gamma_k^\ell$, to see if there are new local extrema that exceed the coarse velocities $\mathcal{C}_k^\ell$, $\mathcal{C}_{k-1}^\ell$ and $\mathcal{C}_{k+1}^\ell$. If so, we redefine the fine cell momenta, for all $i \in \Gamma_k^\ell$, by

$$\mu_i^{\ell+1} = h_i^{\ell+1}\big(\mu_k^\ell / h_k^\ell\big). \tag{9.9}$$

Note that this still conserves momentum, assuming that (9.6) is satisfied, since

$$\begin{aligned} \sum_{i\in\Gamma_k^\ell} V_i^{\ell+1}\mu_i^{\ell+1} &= \big(\mu_k^\ell/h_k^\ell\big)\sum_{i\in\Gamma_k^\ell} V_i^{\ell+1}h_i^{\ell+1} \quad \text{by (9.9).} \\ &= V_k^\ell \mu_k^\ell \quad \text{by (9.6)} \end{aligned} \tag{9.10}$$

While this additional limiting may at first seem unnecessary or overly restrictive, without it the velocities created in shallow regions where fine cells have vanishingly small depths can become unbounded. This makes the interpolation procedures near the shore, at the interface of wet and dry cells, especially difficult. This is the subject of the next section.

9.2. Coarsening and refining near the shore

The averaging and interpolation strategies just presented break down near the shoreline where one or more cells is dry. Figure 9.2 illustrates two possible situations. In both cases it is impossible to maintain conservation of mass and also preserve the flat sea surface. In this case we forgo conservation and maintain the flat surface, since otherwise the resulting gradient in sea surface will generate spurious waves near the coast that can easily have larger magnitude than the tsunami itself.

In Figure 9.2(b) the middle coarse cell is wet, $h_2^\ell > 0$, while on the refined grid only one of the two refined cells is wet, $h_3^{\ell+1} > 0$ but $h_4^{\ell+1} = 0$ in Figure 9.2(a). Figures 9.2(c) and 9.2(d) show a case where the middle coarse cell is dry, but on the fine grid one of the underlying fine cells must be wet in order to maintain a constant sea surface. In both cases, the total

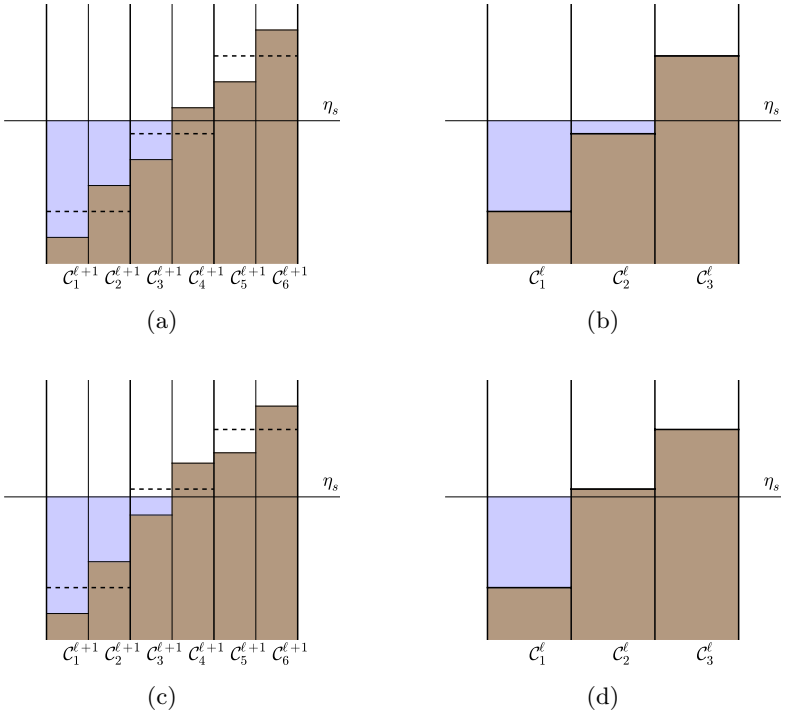


Figure 9.2. (a) Level $\ell + 1$ topography and water depth on a beach where the three rightmost cells are dry. (b) Corresponding level ℓ representation with one dry cell. (c) Second example of level $\ell + 1$ topography and water depth on a beach where the three rightmost cells are dry. (d) The corresponding level ℓ representation with two dry cells. Note that refining the middle dry cell leads to one wet cell and one dry cell.

mass of water is *not* preserved either when going from the coarse to fine or from the fine to coarse grid.

The lack of conservation of mass near shorelines is perhaps troubling, but there is no way to avoid this when different resolutions of the topography are used. For ocean-scale tsunami modelling it may easily happen that the entire region of interest along the coast lies within a single grid cell on the coarsest level, and this cell will be dry if the average topography value in this coarse cell is above sea level. Obviously, when this cell is refined as the wave approaches land, water must be introduced on the finer grids in order to properly represent the fine-scale topography and shoreline. Stated more generally, in order to prevent the generation of new sea-surface extrema and hence hydraulic gradients near the shoreline, the coarsening and refining formulas presented in the previous section require additional modifications, which do not conserve mass in general. To mitigate this we try to ensure

that the shoreline is appropriately refined, using fine topography and a flat sea, before the wave arrives. Then the change in mass does not affect the computed solution at all: exactly the same solution would be computed if the shoreline had been fully resolved from the start of the computation.

When coarsening, the simple averaging (9.6) cannot be used near the shoreline unless all fine cells are dry $h_i^{\ell+1} = 0$, in which case the coarse cell $h_k^\ell = 0$ is also dry and is the average. In general, to go from the fine grid values to a coarse-grid value we average over only those subcells that are wet, setting

$$\tilde{\eta}_k^\ell = \frac{\sum_{i\in\Gamma_k^\ell} V_i^{\ell+1}\mathrm{sgn}(h_i^{\ell+1})\eta_i^{\ell+1}}{\sum_{i\in\Gamma_k^\ell} V_i^{\ell+1}\mathrm{sgn}(h_i^{\ell+1})}, \tag{9.11}$$

and then set

$$h_k^\ell = \max(0,\ \tilde{\eta}_k^\ell - B_k^\ell). \tag{9.12}$$

Note that $\mathrm{sgn}(H)$ is always 0 (if the cell is dry) or 1 (if it is wet) and by assumption at least one subcell is wet, so the denominator of (9.11) is non-zero. If all cells are wet then we will have $\tilde{\eta}_k^\ell > B_k^\ell$ and mass is conserved. In fact the formula (9.12) reduces to (9.6) in this case, and in practice we always use (9.12) for coarsening.

Now consider refinement. To interpolate the depth from a coarse cell to the underlying fine cells in a situation such as those shown in Figure 9.2, we first construct a linear interpolant for the surface elevation η, a function of the form

$$\mathcal{N}_k^\ell(x) = \bar{\eta}_k^\ell + \bar{\sigma}_k^\ell(x - x_k^\ell), \tag{9.13}$$

Here $\bar{\eta}_k^\ell$ is not the usual surface variable η, but is modified to account for dry cells. This will be defined below. Once defined on all coarse cells, $\bar{\sigma}_k^\ell$ is computed again using minmod slopes based on the values of $\bar{\eta}^\ell$. We then compute the depth in the subcells using this linear function and the fine grid topography,

$$h_i^{\ell+1} = \max(0,\ \mathcal{N}_k^\ell(x_i^{\ell+1}) - B_i^{\ell+1}), \tag{9.14}$$

for each subcell $\mathcal{C}_i^{\ell+1}$ of the coarse cell $\mathcal{C}_k^\ell$.

If the cell $\mathcal{C}_k^\ell$ is wet then the surface value $\bar{\eta}_k^\ell$ in (9.13) is taken to be

$$\bar{\eta}_k^\ell = \eta_k^\ell = B_k^\ell + h_k^\ell \quad \text{if } h_k^\ell > 0. \tag{9.15}$$

If the coarse cell is dry, we need to determine an appropriate surface elevation $\bar{\eta}_k^\ell$ for use in the linear function (9.13). In this case $\eta_k^\ell = B_k^\ell$, and this topography value may be above sea level. Instead of using this value we set $\bar{\eta}_k^\ell = \eta_s$, the specified sea level, in this case.

For interpolating and coarsening momentum near the shore, because mass is not conserved in general, we must treat momentum carefully by adopting

additional procedures to change the momentum in a way that is consistent with the change in mass. Upon coarsening the momentum, the procedure we use conserves momentum whenever mass is conserved. If mass is lost upon coarsening, such as would occur when coarsening the solutions shown in Figures 9.2(a) and 9.2(c), the momentum associated with the mass that no longer exists is removed. That is, for cells with non-zero mass, we define the coarse momentum by

$$\mu_k^\ell = \frac{\min(V_k^\ell h_k^\ell, \sum_{i\in\Gamma_k^\ell} V_i^{\ell+1} h_i^{\ell+1})}{\sum_{i\in\Gamma_k^\ell} V_i^{\ell+1} h_i^{\ell+1}} \frac{1}{V_k^\ell} \sum_{i\in\Gamma_k^\ell} V_i^{\ell+1}\mu_i^{\ell+1}. \tag{9.16}$$

Note that (9.16) reduces to the standard coarsening formula when the mass is conserved, yet when mass is reduced upon coarsening the coarse momentum is multiplied by the ratio of the coarse mass to the mass in the fine subcells.

Upon refinement, we begin with the standard procedure used away from the shore: a linear interpolation of momentum is performed, and then the momentum in each fine subcell is checked to see if new extrema in velocities are generated (in determining velocity bounds, we define the velocity to be zero in dry neighbouring coarse cells), in which case we resort to (9.9) for all subcells $i \in \gamma_k^\ell$. This certainly includes the case where a dry ($h_i^{\ell+1} = 0$) subcell has non-zero momentum and hence an infinite velocity. That is, when velocity bounds are violated, each fine-cell momentum becomes the product of the fine-cell depth and the coarse velocity (for coarse cells that are dry ($h_k^\ell = 0$), the velocity is defined to be zero). Note that this procedure alone does not conserve momentum if mass is not conserved: rather, if mass is altered on the fine grid, the momentum would be altered by the ratio of fine subcells' mass to the coarse cell's mass. To prevent the addition of momentum to the system purely through refinement, we modify (9.9) to

$$\mu_i^{\ell+1} = h_i^{\ell+1} \frac{\mu_k^\ell}{\max\bigl(h_k^\ell, \frac{1}{V_k^\ell}\sum_{i\in\Gamma_k^\ell} V_i^{\ell+1} h_i^{\ell+1}\bigr)}. \tag{9.17}$$

Note that (9.17) implies that momentum is conserved even when mass has been added, since in that case

$$\sum_{i\in\Gamma_k^\ell} V_i^{\ell+1}\mu_i^{\ell+1} = \frac{\mu_k^\ell}{\max\bigl(h_k^\ell, \frac{1}{V_k^\ell}\sum_{i\in\Gamma_k^\ell} V_i^{\ell+1} h_i^{\ell+1}\bigr)} \sum_{i\in\Gamma_k^\ell} V_i^{\ell+1} h_i^{\ell+1} = V_k^\ell \mu_k^\ell. \tag{9.18}$$

When mass is lost, (9.17) implies that the momentum is multiplied by the ratio of the remaining fine-cell mass and coarse mass, essentially removing momentum associated with the lost mass.

All of the formulas in this section reduce to those of the previous section when mass is conserved. Therefore we can implement the coarsening and refining strategy in a uniform manner for all cases. While the formulas may seem overly complicated, they ensure the following properties upon regridding.

- Mass is conserved except possibly near the shore.
- Mass conservation implies momentum conservation.
- If mass is gained, momentum is conserved.
- If mass with non-zero momentum is lost, the momentum associated with that mass is removed as well.
- New extrema in surface elevation and hence hydraulic gradients are not created.
- New extrema in water velocity are not created.

In general, coarsening and refinement near the shoreline should ideally happen just prior to the arrival of waves, while the shoreline is still at a steady state. In this case, all of the specialized procedures described in this section produce the same solution on the fine grids as would exist if the fine grids had been initialized with a constant sea level long before the tsunami arrival. Therefore, although mass and momentum are not necessarily conserved upon refinement, the adaptive solution is ideally close to the solution that would exist if fixed (non-AMR) grids were used, yet at a much reduced computational expense.

9.3. Extension to two dimensions

Interpolation and averaging

All of the interpolation and averaging strategies described above extend naturally to two dimensions. In fact, if we continue to let a single index represent grid cells, *e.g.*, $i \in \Gamma_k^\ell$ represents the index of a level $\ell+1$ rectangular subcell $\mathcal{C}_i^{\ell+1}$ within a level ℓ rectangular cell $\mathcal{C}_k^\ell$, then most of the formulas described above need only minor modification. The length ratio r_x^ℓ becomes an area ratio $r_x^\ell r_y^\ell$ in the case of a Cartesian grid. More generally we continue to use V_k^ℓ to represent the area of cell k on level ℓ.

For interpolation, we simply extend the linear interpolants (9.7) and (9.13) to two dimensions,

$$f(x,y) = f_k^\ell + (\sigma^x)_k^\ell (x - x_k^\ell) + (\sigma^y)_k^\ell (y - y_k^\ell). \tag{9.19}$$

Here, (x_k^ℓ, y_k^ℓ) is the centre of this cell, and the slopes $(\sigma^x)_k^\ell$ and $(\sigma^y)_k^\ell$ are the minmod limited slopes in the x and y directions respectively. Lastly, when considering neighbouring cells to determine if new velocity extrema are generated in $\mathcal{C}_i^{\ell+1}$, we consider all nine coarse cells including and surrounding $\mathcal{C}_k^\ell$.

Consistent computational topography

The topographic data used in a computation are often specified by several different rectangular gridded digital elevation models (DEMs) that are at different resolutions. For example, over the entire ocean 10-minute data may be sufficient, while in the region near the earthquake source or along a coastal region of interest one or more finer-scale DEMs must be provided. The DEMs also do not necessarily align with the finite volume computational grids, and so the consistency property (9.2) for topography on different grid levels requires careful consideration.

We accomplish (9.2) in the following manner. The topography data sets are ordered in terms of their spatial resolution (if two data sets have the same resolution they are arbitrarily ordered). We define the topographic surface $B(x, y)$ as the piecewise bilinear function that interpolates the topography data set of the highest resolution DEM at any given point (x, y) in the domain. Away from boundaries of DEMs, this function is continuous and defined within each rectangle of the DEM grid using bilinear interpolation between the four corner points. Where a fine DEM grid is overlaid on top of a coarser one, there are potentially discontinuities in $B(x, y)$ across the outer boundaries of the finer DEM. This procedure defines a unique piecewise bilinear function $B(x, y)$ based only on the DEM grids, independent of the computational grid(s). When a new computational grid is created, either at the start of a computation or when regridding, the computational topography in each finite volume cell is defined by integrating $B(x, y)$ over the cell:

$$B_k^\ell = \frac{1}{V_i^\ell} \int_{\mathcal{C}_k^\ell} B(x, y)\, \mathrm{d}x\, \mathrm{d}y. \tag{9.20}$$

Note that each integral may span several DEM cells if it overlaps a DEM boundary, but since it is a piecewise bilinear function the integral can be computed exactly. Since these topography values are based on exact integrals of the same surface, at all refinement levels, the consistency property (9.2) will always be satisfied.

9.4. Maintaining conservation at grid interfaces

The discussion above focused on maintaining constant sea level and conserving mass and momentum when grid cells are coarsened and refined. We now turn to step (5) in the AMR Integration Strategy of Section 8.2. We wish to ensure that the method conserves mass away from the shoreline at least, and also momentum in the case without source terms. The final solution will not be strictly conservative due to the source terms of topography and friction, and due to the shoreline algorithms which favour maintaining a constant sea level over maintaining conservation of mass, but the underlying method should be conservative to reduce these effects.

Recall that in step (4) of the algorithm in Section 8.2 the value of Q in any coarse-grid cell that is covered by fine grids is replaced by the appropriate weighted average of the more accurate fine-grid values. This potentially causes a conservation error since the fine grid cells at the boundary of the patch were updated using fluxes from ghost cells rather than from the neighbouring coarse cells directly. The standard fix for this, applied to finite volume methods written in flux-differencing form, is to adjust the adjacent coarse-grid values by the difference between the coarse grid flux at the patch boundary (originally used to compute the value in this cell) and the weighted average of fine grid fluxes that was instead used interior to the patch (Berger and Colella 1989). This restores global conservation and presumably also improves the value in these coarse-grid cells by using a more accurate approximation of the flux, as determined on the fine cells.

We use the f-wave propagation algorithm instead of flux-differencing since this allows the development of a well-balanced method in the non-conservative form described above. This requires a modification of the flux-based fix-up procedure that is described in detail in Berger and LeVeque (1998) and implemented more generally in the AMR algorithms of Clawpack. This modification works in general for wave propagation algorithms based on fluctuations rather than fluxes.

Note that since the fine grid typically takes many time steps between each coarse-grid step, performing this fix-up involves saving the fluctuations around the perimeter of each fine grid at each intermediate step, which is easily done. The harder computation is the modification of the coarse cell values based on these fluctuations, since the coarse cells affected are generally interior to a coarse grid and appear in an irregular manner. Each coarse grid keeps a linked list of cells needing this correction, and saves the fluctuations on the edge adjacent to a fine grid. For example, if the coarse cell is to the left of the fine grid, the left-going fluctuation is needed.

One additional correction step is needed for conservation when using the wave propagation approach. A Riemann problem between a coarse-grid cell and the fine grid ghost cell needs to be accounted for to maintain conservation. This leads to an additional Riemann problem at the boundary of each fine grid cell at each intermediate time, as discussed by Berger and LeVeque (1998). In the absence of source terms and away from the shoreline, these two steps ensure that conservation is maintained in spite of the fact that the method does not explicitly calculate fluxes.

10. Verification, validation, and reproducibilty

Verification and validation (V&V) is an important aspect of research in computational science, and often poses a large-scale challenge of its own for complex applications: see Roache (1998) for a general discussion. Our

goal in this paper is not to provide detailed V&V studies of the GeoClaw software, but only to give a flavour of some of the issues that arise and approaches one might take in relation to tsunami models. For some other discussions of this topic and possible test problems, see for example Synolakis and Bernard (2006) or Synolakis, Bernard, Titov, Kânoğlu and González (2008).

Verification in the present context consists of verifying that the computational algorithms and software can give a sufficiently accurate solution to the shallow water equations they purportedly solve, with the specified topography and initial conditions. In particular, this requires checking that the adaptive refinement algorithms provide accurate results in the regions of interest even when much coarser grids are used elsewhere, without generating spurious reflections at grid interfaces, for example. Exact solutions to the shallow water equations over topography are difficult to come by, but a few solutions are known that are useful, in particular as tests of the shoreline algorithms. The paper of Carrier, Wu and Yeh (2003) provides the exact solution for a one-dimensional wave on a beach that is suggested as a verification problem in Synolakis *et al.* (2008) and was one of the benchmark problems discussed by many authors in Liu, Yeh and Synolakis (2008). The paper of Thacker (1981) presents some other exact solutions, including water sloshing in a parabolic bowl, which has often been used as a test problem for numerical methods, for example by Gallardo *et al.* (2007) and in the GeoClaw test suite. In Section 11 we illustrate another technique for testing whether a tsunami code accurately solves the shallow water equations in the necessary regimes for modelling both transoceanic propagation and local inundation. No finite set of tests will prove that the program always gives correct solutions, and of course no numerical method will: the accuracy depends on the grid resolution used and other factors. However, by exercising the code on problems where an exact or highly accurate reference solution is available, it is possible to gain a useful appreciation for the accuracy and limitations of a code.

Validation of a code is generally more difficult, since this concerns the question of whether the computational results provide a useful approximation to reality under a certain range of conditions. The depth-averaged shallow water equations provide only an approximation to the full three-dimensional Navier–Stokes equations, so even the exact solution to these equations will only be an approximation to the real flow. Several assumptions are made in deriving the shallow water equations, in particular that the wavelength of the waves of interest is long relative to the fluid depth. This is often true for tsunamis generated by megathrust events, at least for the transoceanic propagation phase. It is less clear that this assumption holds as tsunamis move into shallower water and interact with small-scale local features. A great deal of effort has gone into validation studies for

the shallow water equations and into the development and testing of other model equations that may give a better representation of reality.

Validation studies of tsunami models take many forms. Comparison with actual tsunami events similar to ones the code is designed to model is the best form of validation in many ways. Due to the recent frequency of large tsunamis there is a wealth of data now available, far beyond what was available 10 years ago. These data have been used in numerous validation studies of tsunami models, for example Grilli *et al.* (2007) and Wang and Liu (2007). These studies are seldom clear-cut, however, due to the wide range of unknowns concerning the earthquake source structure, the resulting seafloor deformation, the proper drag coefficient to use in friction terms, and various other factors.

To perform more controlled experiments, large-scale wave tanks are used to simulate tsunami inundation, with scaled-down versions of coastal features and precisely controlled sources from wave generators. The resulting flow and inundation can then be accurately measured with tools such as depth gauges, flow meters, and high-speed cameras. By running a tsunami code on the wave tank topography with the same source, careful comparisons between numerical results and the actual flow can be performed. Some standard test problems are described in Liu *et al.* (2008) and Synolakis *et al.* (2008). A problem with this, of course, is that this can only validate the code relative to the wave tank, which is itself a scaled-down model of real topography. There is still the question of how well this flow corresponds to reality.

Reproducibility of computational experiments is an issue of growing concern in computational science; see for example Fomel and Claerbout (2009), Merali (2010) and Quirk (2003). By this we mean the performance of computational experiments in a controlled and documented manner that can potentially be reproduced by other scientists. While this is a standard part of the scientific method in laboratory sciences, in computational science the culture has put little emphasis on this. Many publications contain numerical experiments where neither the method used or the test problem itself are described in sufficient detail for others to verify the results or to perform meaningful comparisons against competing methods.

Part of our goal with GeoClaw, and with the Clawpack project more generally, is to facilitate the specification, sharing, and archiving of computational experiments (LeVeque 2009). The codes for all the experiments in the next sections can be found on the webpage for this paper (www13), along with pointers to additional codes such as those used for the experiments presented in Berger *et al.* (2010).

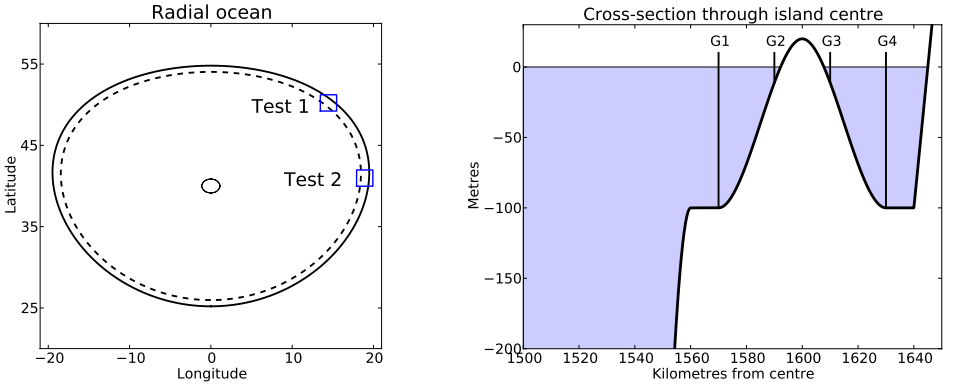
11. The radial ocean

As a verification test that the shallow water equations are solved correctly on the surface of the sphere, and that the wetting and drying algorithms give similar results regardless of the orientation of the shoreline relative to the grid, it is useful to test the code on synthetic problems where comparisons are easy to perform. We illustrate this with an example taken from Berger *et al.* (2010), which should be consulted for more details on the bathymetry and a related test problem. The domain consists of a radially symmetric ocean with a radius of 1645 km on the surface of a sphere of radius comparable to the earth's radius, $R = 6367.5$ km, centred at 40°N. The extent of the ocean in latitude–longitude space is shown in Figure 11.1. The bathymetry is flat at -4000 m up to a 1500 km, and then is followed by a smooth continental slope and continental shelf with a depth of 100 m, and finally a linear beach. The initial conditions for the tsunami consist of a Gaussian hump of water at the centre, given by

$$\eta(r) = A_0 \exp\bigl(-2r^2/10^9\bigr), \tag{11.1}$$

where r is the great-circle distance from the centre, measured in metres. The amplitude A_0 is varied to illustrate the effect of different size tsunamis.

At some location along the shelf we place a circular island with a radius of roughly 10 km, centred 45 km offshore. In theory the flow around the island should be identical regardless of where it is placed, though numerically this will not be true. We compare the results for two different locations as a test of consistency.



(a) (b)

Figure 11.1. (a) Geometry of the radially symmetric ocean, as described in the text. (b) A zoom view of the topography of the continental shelf along the ray going through the centre of the island. The location of the four gauges is also shown.

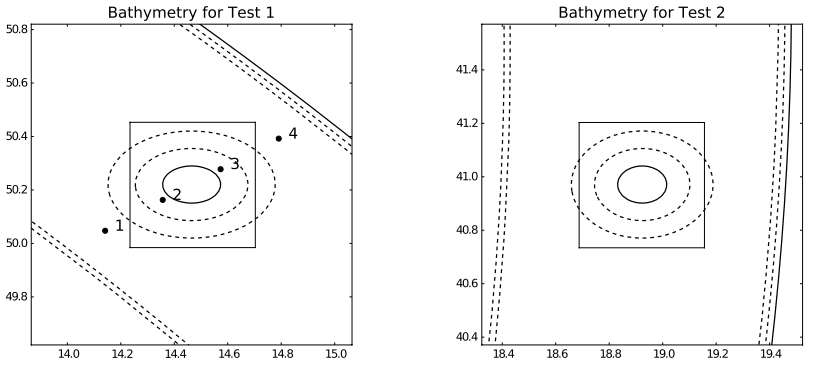


Figure 11.2. Bathymetry in regions near an island placed at different locations along the coast, indicated by the rectangles in Figure 11.1(a). The location of four gauges is also shown. The time history of the surface at these gauges is shown in Figure 11.4.

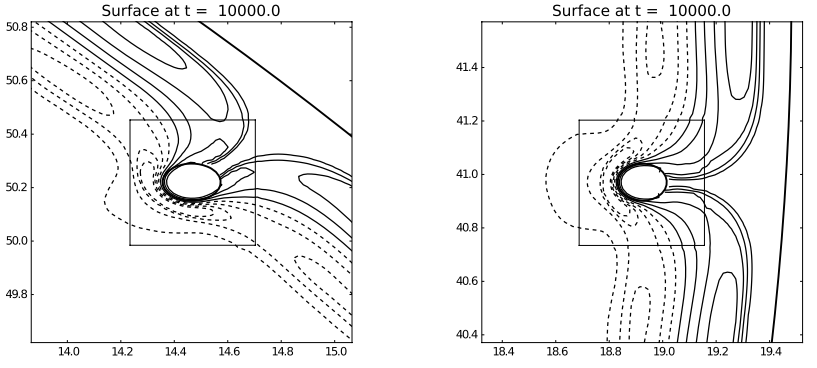


Figure 11.3. Contours of tsunami height for flow around the island from Test 1 and Test 2, in the case $A_0 = 150$ in (11.1). In each case solid contours are at $\eta = 1, 3, 5, 7$ m, and dashed contours are at $\eta = -1, -3, -5, -7$.

Figure 11.1(a) shows the ocean (which does not look circular in latitude–longitude coordinates). The outer solid curve is the position of the shoreline, with constant distance from the centre when measured on the surface of a sphere. The dashed line shows the extent of the continental shelf. The boxes labelled Test 1 and Test 2 are regions where the island is located in the tests presented in the following figures. The small circle near the centre shows roughly the extent of the hump of water used as initial data. Figure 11.1(b) shows a cross-section of the bathymetry through the island.

Figure 11.2 shows zoomed views of the two boxes labelled Test 1 and Test 2 in Figure 11.1(a), with contours of the bathymetry. The solid contour lines are shoreline ($B = 0$) and the dashed contour lines are at elevations

$B = -40, -80, -120, -160$ m. Note that the continental shelf has a uniform depth of -100 m. In this figure we also indicate the locations of four gauges where the surface elevation as a function of time is recorded as the computation progresses. Specifying gauges is a standard feature in GeoClaw and at each time step the finest grid available near each gauge location is used to interpolate the surface elevation to the gauge location. Figure 11.4 shows results at these gauges for several different tests, described below.

We solve using five levels of AMR with the same parameters as in Berger *et al.* (2010): a 40×40 level 1 grid over the full domain (1° on a side), and factor of 4 refinement in each subsequent level (a cumulative factor 256 refinement on the finest level). Levels 4 and 5 are used only near the island.

We show results for three different values of the amplitude in (11.1): $A_0 = 0.005,\ 5.0$ and 150.0. Figure 11.4 shows the surface elevation measured at the four gauges shown in Figure 11.2. For any fixed amplitude A_0, the gauge responses should be the same in Test 1 and Test 2, since the island and gauges are simply rotated together to a new position. This is illustrated in Figure 11.4, where the solid curve is from Test 1 and the dashed curve is from Test 2. The good agreement indicates that propagation is handled properly on the surface of the sphere and that the topography of the island and shore are well approximated on the grid, regardless of orientation.

The tsunami propagation over the deep ocean is essentially linear for all of these amplitudes. For the two smaller-amplitude cases the propagation on the shelf and around the island also shows nearly linear dependence on the data, especially for gauges 1 and 4, where the undisturbed ocean depth is 100 m. Gauges 2 and 3 are at locations where the depth is about 11 m, and some nonlinear effects can be observed. Compare the gauge plots in Figure 11.4(a,b) (for $A_0 = 0.005$) with those in Figure 11.4(c,d) (for $A_0 = 5$), and note that the vertical axis has been rescaled by a factor of 100. Also note that for the $A_0 = 0.005$ case, the maximum amplitude seen at any of the gauges is below 1 cm. Before hitting the continental shelf, this tsunami had even smaller amplitude. This test illustrates that it is possible to accurately capture even very small-amplitude tsunamis.

Figure 11.4(e,f) shows a much larger-amplitude tsunami, using the same initial data (11.1) but with $A0 = 150$. Propagation across the ocean is still essentially linear and so the arrival time is nearly the same, but the wave amplitude is large enough that steepening occurs on the shelf. The nonlinear effects are evident in these gauge plots, which are no longer simply scaled up linearly from the first two cases.

In Figure 11.5 we show surface plots of the run-up at different times, comparing the two computations with the island in different locations. Figure 11.3 shows contour plots for the first of these times, $t = 10000$ s. Four gauge locations are shown in Figure 11.2 and the surface elevation at these gauges is shown in Figure 11.4. The agreement is quite good.

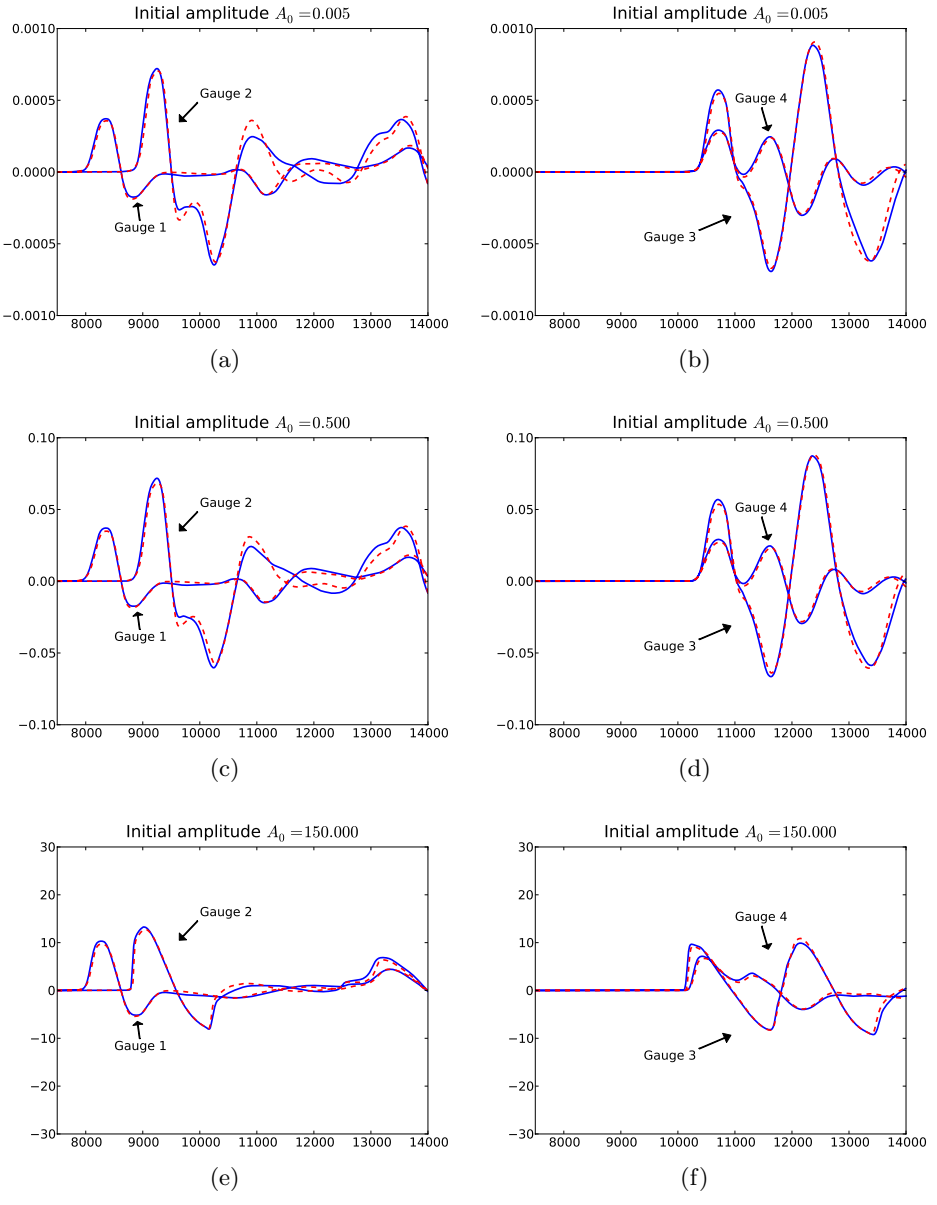


Figure 11.4. Comparison of gauge output from Test 1 and Test 2, showing the surface elevation as a function of time (in seconds) for the gauges shown in Figure 11.2. In each figure, the solid curve is from Test 1 and the dashed curve is from Test 2. (a,b) Very small-amplitude tsunami, with $A_0 = 0.005$ in (11.1). (c,d) $A_0 = 0.5$. (e,f) $A_0 = 150$, giving the large-amplitude tsunami seen in Figure 11.5. Note the difference in vertical scale in each set of figures (metres in each case).

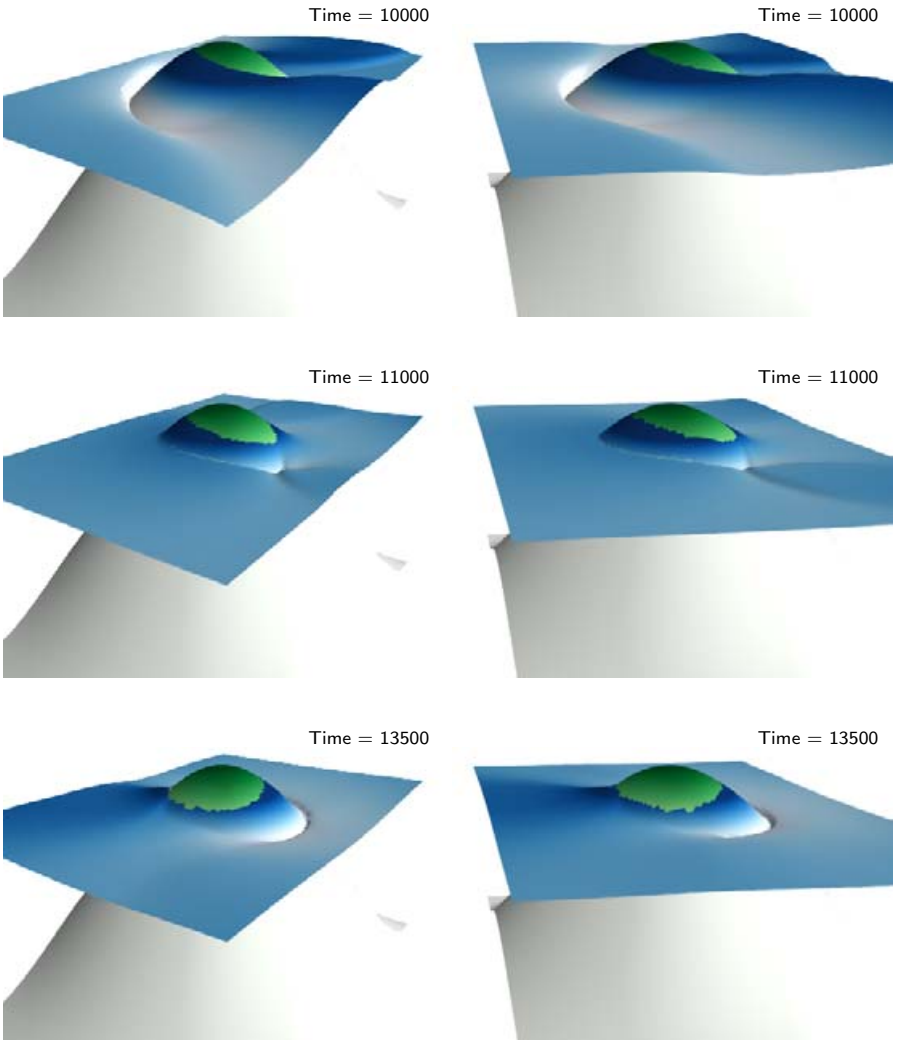


Figure 11.5. Surface plots of the tsunami interacting with the island for Test 1 and Test 2, in the case $A_0 = 150$ in (11.1). At time $t = 10000$ seconds the tsunami is just wrapping around the island, as shown also in Figure 11.3. At time 13500 the reflected wave from the mainland has run up the lee side of the island and is flowing back down. For an animation of these results, see the webpage for this paper (www13).

12. The 27 February 2010 Chile tsunami

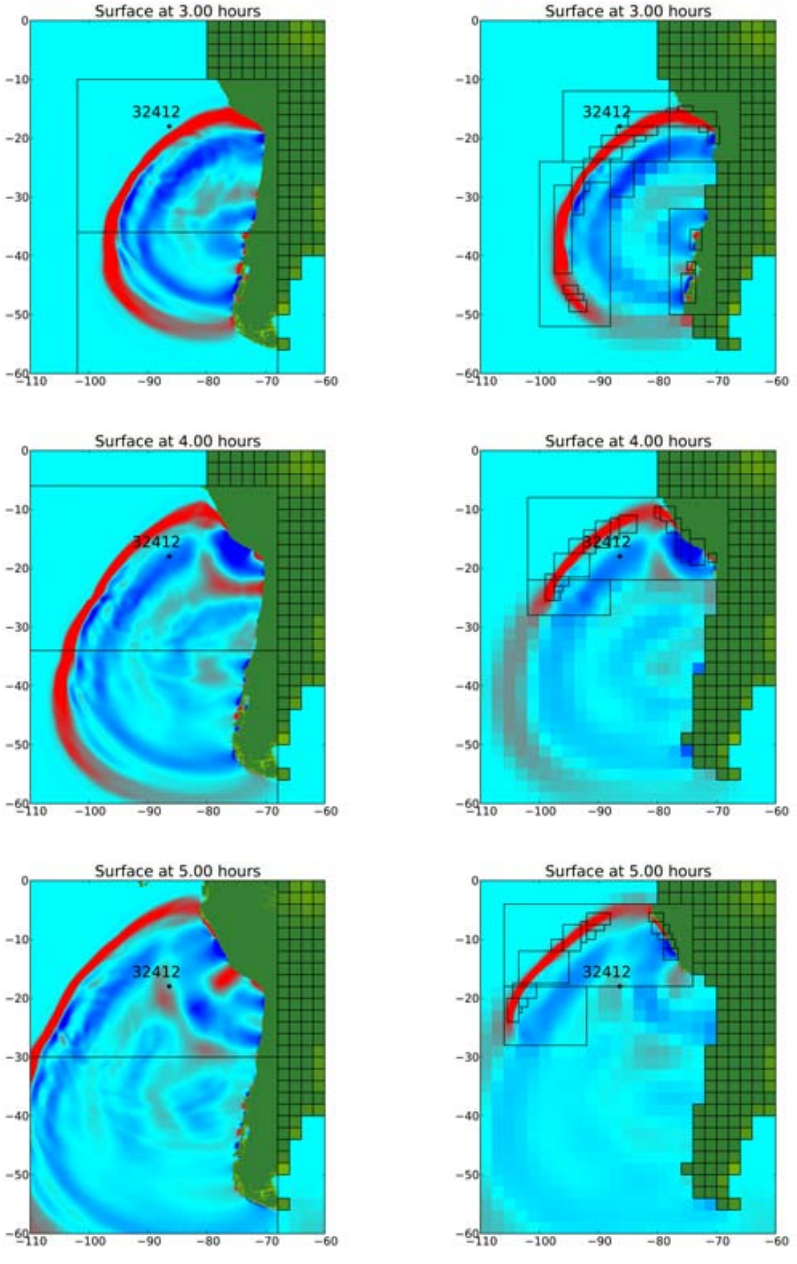
As an illustration of the use of adaptive mesh refinement to explore real-world tsunamis, we will show some results obtained using GeoClaw for tsunamis similar to the one generated by the 27 February 2010 earthquake near Maule, Chile. Some computations of the 2004 Indian Ocean tsunami calculated with this software can be found in LeVeque and George (2004) and George and LeVeque (2006).

We will use relatively coarse grids so that the computations can be easily repeated by the interested reader using the source code available on the webpage for this paper (www13). Several source mechanisms have been proposed for this event. Here we use a simple source computed by applying the Okada model to the fault parameters given by USGS earthquake data (www12). We use bathymetry at a resolution of 10 minutes (1/6 degree) in latitude and longitude, obtained from the ETOPO2 data set at the National Geophysical Data Center (NGDC) GEODAS Grid Translator (www9).

Figures 12.1 and 12.2 show comparisons of results obtained with two simulations using different AMR strategies. In both cases, the level 1 grid has a 2° resolution in each direction (roughly 222 km in latitude), and grid cells of this grid can be seen on the South American continent. The finest grid level also has the same resolution in both cases, a factor of 20 smaller in each direction.

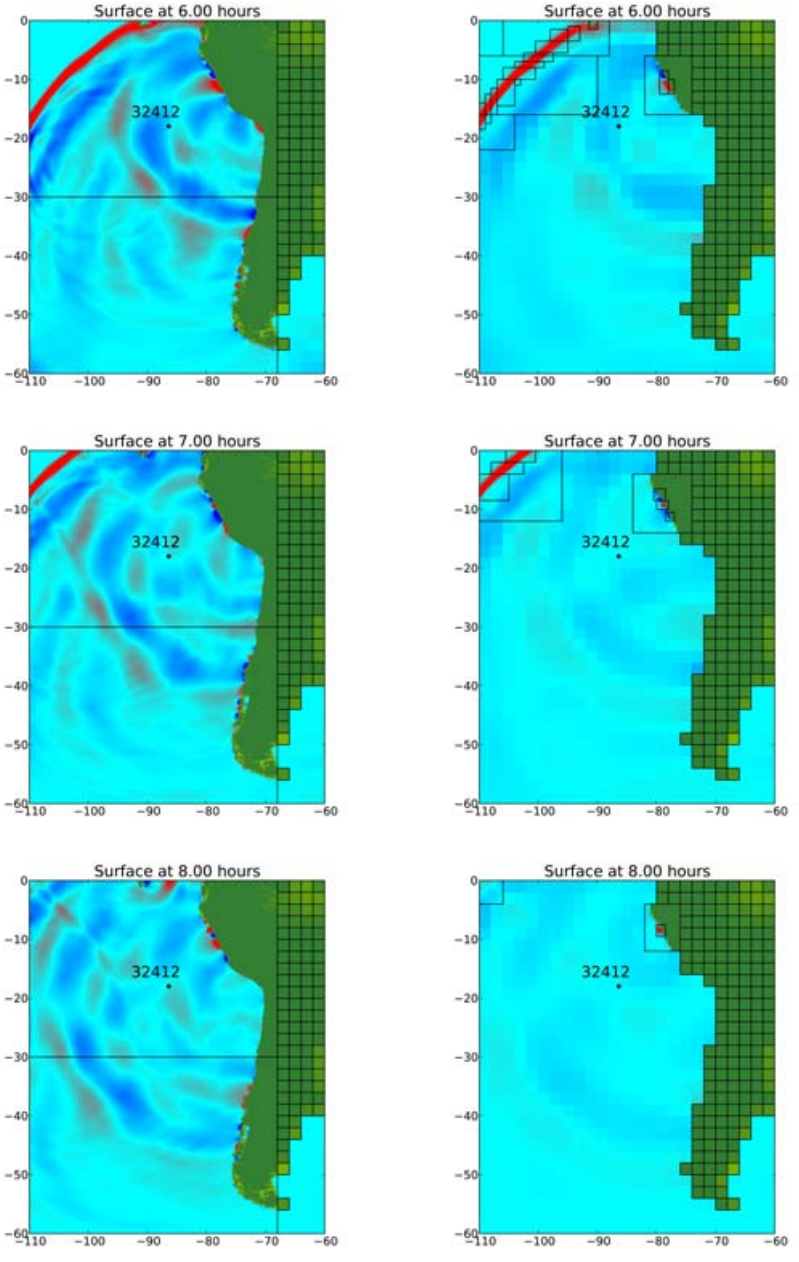
For the calculation in Figures 12.1(a) and 12.2(a), only two AMR levels are used, with a refinement factor of $r_x^1 = r_y^1 = 20$ in each direction and flagging all cells where the water is disturbed from sea level. The level 2 grid grows as the tsunami propagates until it covers the full domain at 5 hours. (It is split into 4 level 2 grids at this point because of restrictions in the software on the size of any single grid to reduce the memory overhead.) Results of this calculation agree exactly with what would be obtained with a single grid with a uniform grid size of 0.1°.

For the calculation in Figures 12.1(b) and 12.2(b), three AMR levels are used and cells are only flagged for refinement if the deviation from sea level is greater than 0.1 m. Moreover, after $t = 3$ hours, we flag points only if the latitude is greater than −25°S. The refinement factors are $r_x^1 = r_y^1 = 4$ from level 1 to level 2 and $r_x^2 = r_y^2 = 5$ from level 2 to level 3, so the finest grid has the same resolution as in (a). Ideally the results in the regions covered by the finest grid (the smallest rectangles in the figure) would be identical to those in (a). Visually they agree quite well. In particular, it should be noted that there is no apparent difficulty with spurious wave generation at the interfaces between patches at different refinement levels. Also note that in both calculations the wave leaves the computational domain cleanly with little spurious reflection at the open boundaries.



(a) (b)

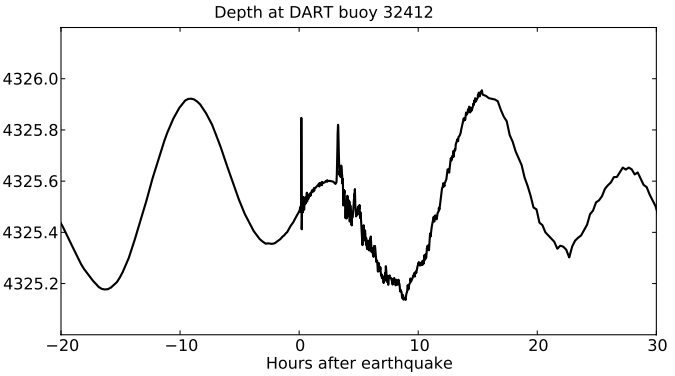
Figure 12.1. The 27 February 2010 tsunami as computed using GeoClaw. The rectangles show the edges of refinement patches. (a) Two levels of AMR, with refinement everywhere around and behind the wave. (b) Three levels of AMR, with the same finest grid resolution but refinement in limited regions. The colour scale is the same as in Figure 1.1, ranging from −0.2 to 0.2 m.



(a) (b)

Figure 12.2. Continuation of Figure 12.1. Note the reflection from the Galapagos on the equator at 6 hours, and that elsewhere the wave leaves the computational domain cleanly. For an animation of these results, see the webpage for this paper (www13).

(a)



(b)

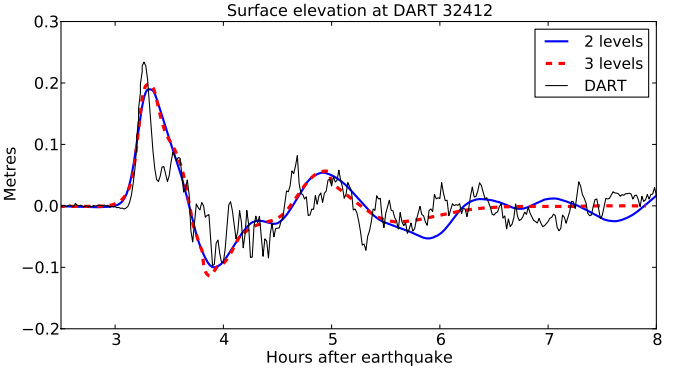


Figure 12.3. (a) Data from DART buoy 32412 before removing the tides. The first blip about 15 minutes after the earthquake is the seismic wave. The tsunami arrives roughly 3 hours later. (b) Comparison of de-tided DART buoy data with two GeoClaw computations. The resolution of the finest grid was the same in both cases.

For a more quantitative comparison of these results, Figure 12.3(b) shows a comparison of the computed surface elevation as a function of time at the location of DART buoy 32412. The solid line is from the level 2 (uniform fine grid) computation, while the dashed line is from the level 3 computation in Figures 12.1(b) and 12.2(b). The agreement is very good up to about 5 hours, after which the DART buoy is in a region that is not refined.

Note from Figure 12.2 that at 8 hours there is still wave action visible on the coast of Peru to the northeast of the DART buoy. This is a region where the continental shelf is very wide and shallow, and waves become trapped in this region due to the slow propagation speed and reflections for the steep shelf slope. In Figure 12.4 we show a zoomed view of these regions from another computation in which a fourth level of AMR has been added, refining by an additional factor of 4. (We have also used finer bathymetry

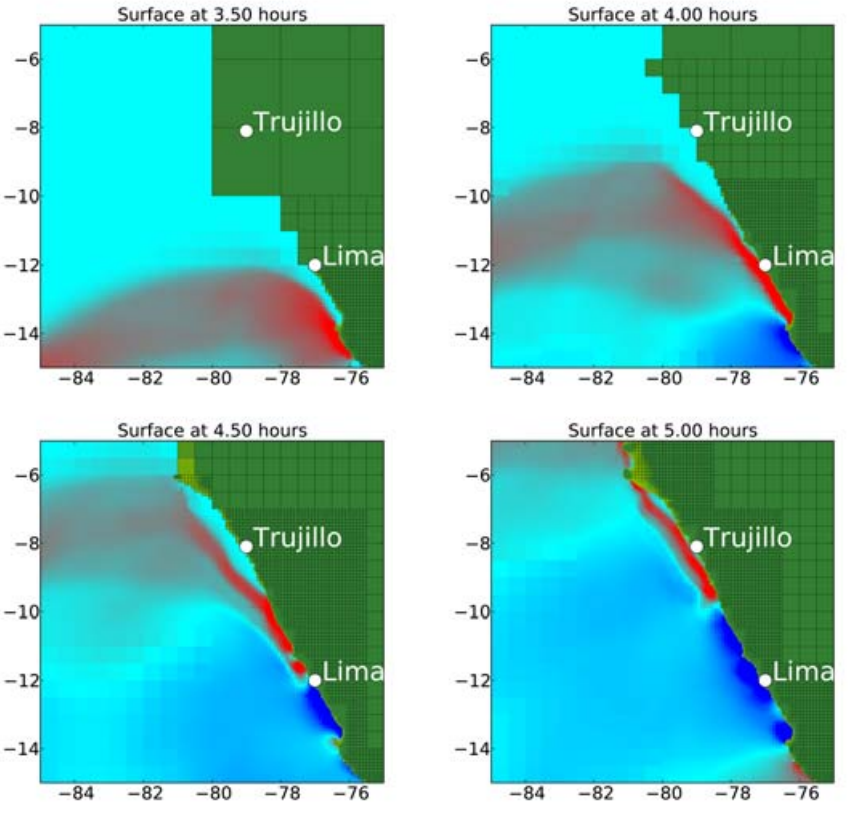


Figure 12.4. Tsunami interaction with the broad continental shelf off the coast of Peru. A fourth level of refinement has been added beyond the levels shown on Figure 12.2. Grid patch edges are not shown and grid lines are shown only for levels 1–3 on land. For an animation of these results, see the webpage for this paper (www13).

in this region, 4-minute data from (www9).) In these figures one can clearly see the fast and broad tsunami sweeping northwards at times 4 and 4.5 hours, and the manner in which this wave is refracted at the continental slope to become a narrower wave of larger amplitude moving towards the coast. These waves continue to propagate up and down the coast in this region for more than 24 hours after the tsunami has passed.

12.1. Inundation of Hilo

Some bays have the dubious distinction of being *tsunami magnets* due to local bathymetry that tends to focus and amplify tsunamis. Notable examples on the US coastline are Crescent City, CA and Hilo, HI. In this section

we use adaptive refinement to track the 27 February 2010 tsunami originating in Chile across the Pacific Ocean and then add several additional levels of refinement to resolve the region near Hilo Harbor. The two simulations described in this section had the same computational parameters such as refinement criteria, but with two different source mechanisms. The computational domain for these simulations spanned 120 degrees of latitude and 100 degrees of longitude, from the source region near Maule, Chile in the southeastern corner of the domain to the Hawaiian Islands in the northwestern corner. We used five levels of grids, using refinement ratios of 8, 4, 16 and 32. The coarsest level consisted of a 60×50 grid with 2° grid cells, yielding a very coarse grid over the ocean at rest. Transoceanic propagating waves were resolved in level 2 grids. Level 3 grids were allowed only near the Hawaiian island chain, with the refinement ratio chosen to roughly match the resolution of bathymetric data used (1 minute). Level 4 grids were allowed surrounding the Big Island of Hawaii, where Hilo is located. In this region we used 3-arcsecond bathymetry from (www8). Finally, level 5 grids were allowed only near Hilo Harbor, where 1/3-arcsecond data from the same source were used. Figure 12.5 shows the the domain of the simulation at four times, as the waves propagate across the Pacific. The outlines of level 3–5 grids can be seen in the final frame, and appear just as the waves reach Hawaii. Figure 12.6 shows a close-up of the grids surrounding Hilo. The finest fifth-level grids were sufficient to resolve the small-scale features necessary to model inundation, such as shoreline structures and a sea wall in Hilo Harbor. The computational grids on the fifth level had grid cells with roughly 10 m grid resolution. Note that the finest grids are refined by a factor $2^{14} = 16\,384$ in each coordinate direction relative to level 1 grids. Each grid cell on the coarsest level would contain roughly 286 million grid cells if a uniform fine grid were used, far more than the total number of grid cells used at any one time with adaptive refinement.

The two source mechanisms used were identical except for the magnitude of slip. We first modelled the actual 27 February 2010 event described in Section 12, by applying the Okada model to fault parameters given by (www12). USGS earthquake data (www12). For the second source mechanism we used the same fault parameters, but increased each subfault dislocation by a factor of 10. The motivation for this second source model is twofold. First, the actual 27 February 2010 tsunami produced little or no inundation in Hilo, so to demonstrate inundation modelling with GeoClaw we created a much larger hypothetical tsunami. Second, amplifying the dislocation while keeping all other parameters fixed allows us to examine linearity in the off-shore region versus nonlinearity for tsunami inundation.

The waves produced in Hilo Harbor by the larger hypothetical source are comparable to those arising from more tsunamigenic events in the past (for

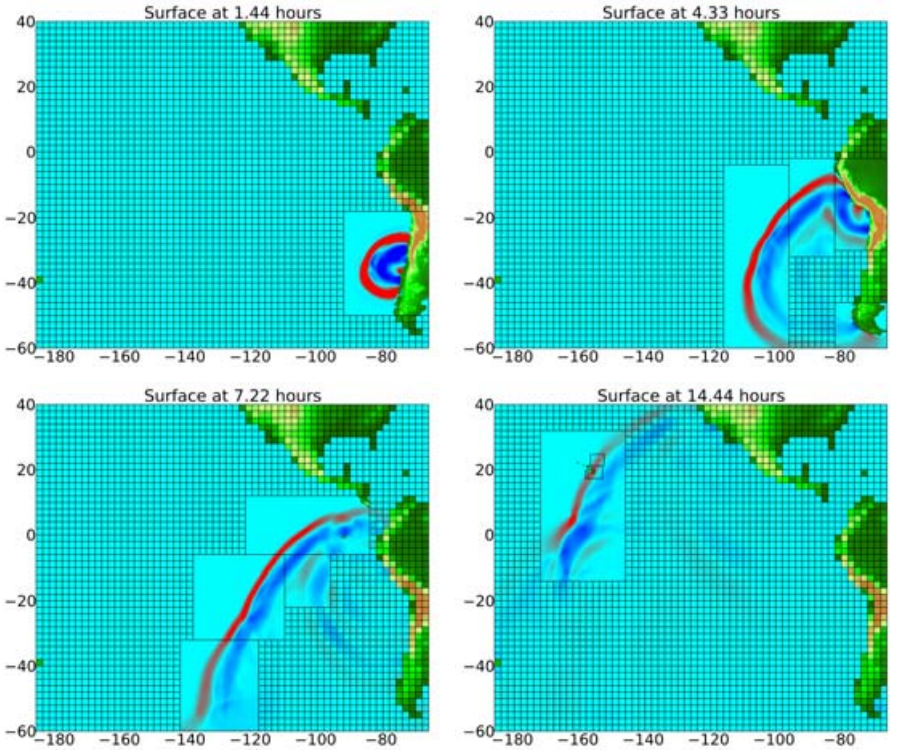
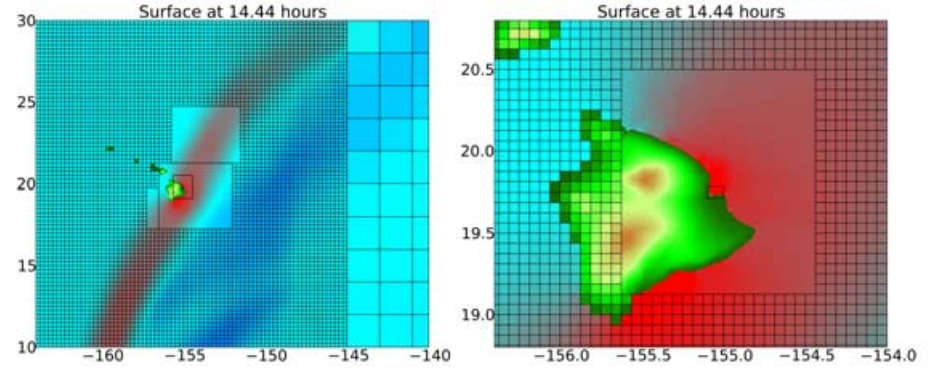


Figure 12.5. The 27 February 2010 tsunami propagates toward the Hawaiian Islands on level 1 and 2 grids. Higher-level grids appear around Hawaii as the waves arrive.



(a) (b)

Figure 12.6. Close-up of the higher-level grids appearing as the tsunami reaches the Hawaiian Islands. (a) All levels of grids, with grid lines omitted from levels 3–5. (b) Close-up of the Island of Hawaii, where an outline of level 5 grids surrounding the city of Hilo can be seen on the east coast of Hawaii. Grid lines are omitted from levels 4–5.

(a)

(b)

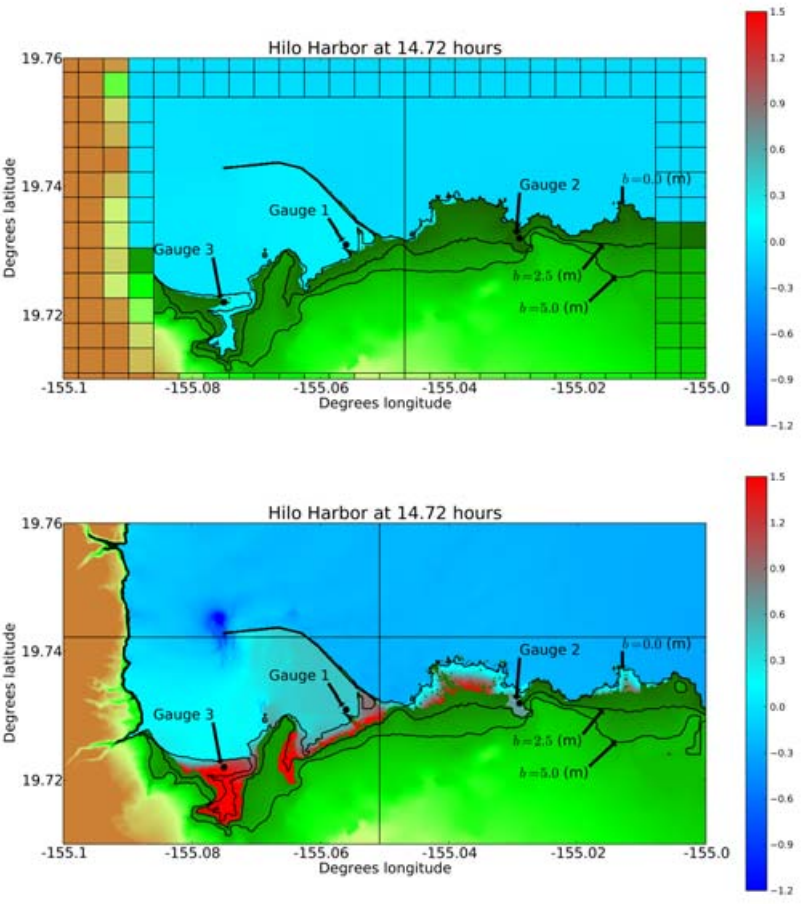


Figure 12.7. Computed inundation maps of Hilo, HI, based on two source mechanisms. (a) Inundation using an actual USGS fault model for the February 2010 event. (b) Hypothetical source mechanism, with the dislocations amplified by a factor of 10, in order to show inundation.

example, the 1960 Chile quake and the 1964 Alaska event). Figure 12.7 shows the two different inundation patterns from the original and amplified source. The location of simulated water level gauges are indicated in the figures. The output of the gauges is shown in Figure 12.8. Gauge 1 is in Hilo Harbor and shows the incoming waves. Note that the waves in the harbour are still close to linear with respect to the source dislocation, several thousand kilometres away. However, in the nearer-shore and onshore regions inundation becomes strongly nonlinear, and determining the patterns of inundation cannot be done by a linear scaling of solutions from different source models.

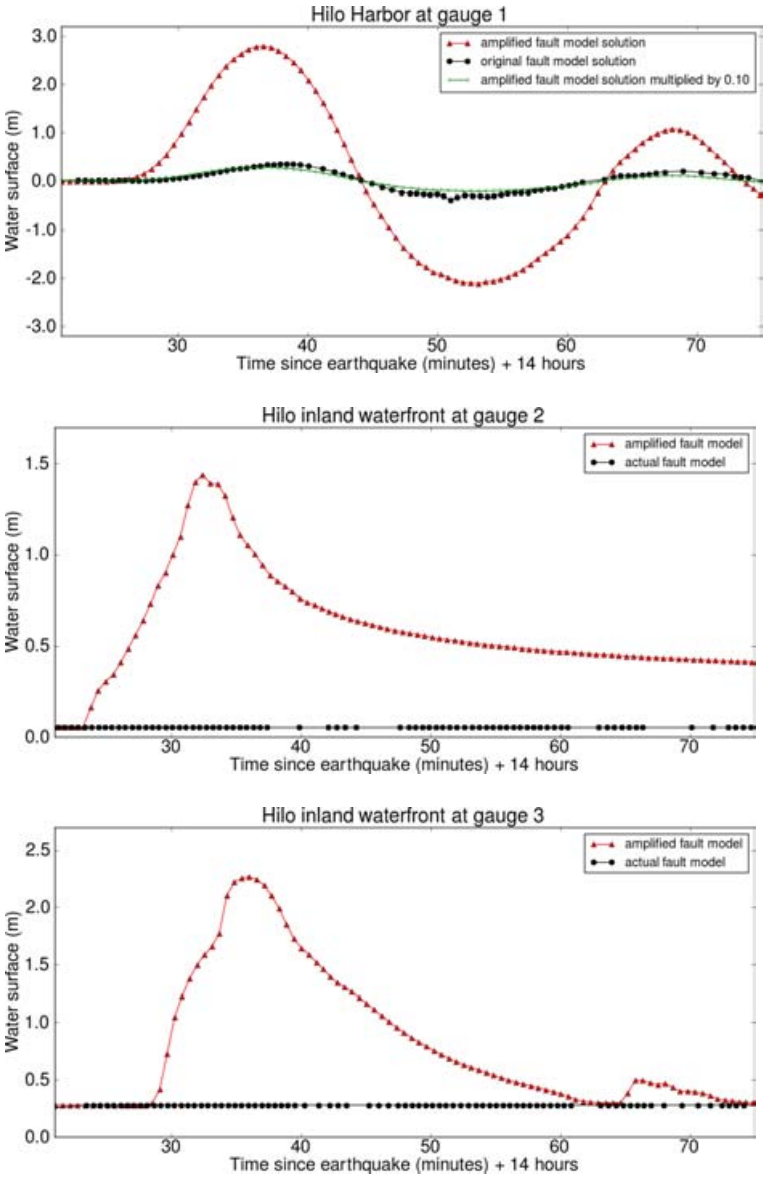


Figure 12.8. Time series of the surface elevation at three different simulated gauge locations near Hilo, HI. The locations of the gauges are shown in Figure 12.7. The results for the 'amplified fault model' were obtained by increasing the slip displacement at the source by a factor of 10. The solid curve in (a) shows the solution for the amplified source mechanism multiplied by 0.10. This lies nearly on top of the curve from the original source model, indicating that the response is nearly linear in the harbour. The inland gauges in (b,c) exhibit nonlinear behaviour: they remain dry for the smaller tsunami but show large inundation waves for the amplified fault model.

13. Final remarks

In this paper we have focused on tsunami modelling using Godunov-type finite volume methods, and the issues that arise when simulating the propagation of a very small-amplitude wave across the ocean, followed by modelling the nonlinear run-up and inundation of specific regions remote from the initial event. The scale of these problems makes the use of adaptive mesh refinement crucial.

The same techniques are applicable to a variety of other geophysical flow problems such as those listed in Section 1. The GeoClaw software has recently been applied to other problems, such as modelling the failure of the Malpasset dam in 1959 (George 2010). This has often been used as a test problem for validation of codes modelling dam breaks. Storm surge associated with tropical storms is another application where shallow water equations are often used, and some preliminary results on this topic have been obtained by Mandli (2010) with the GeoClaw code.

For many problems the shallow water equations are insufficient and other depth-averaged models must be developed. Even in the context of tsunami modelling, there are some situations where it may be important to include dispersive terms (González and Kulikov 1993, Saito, Matsuzawa, Obara and Baba 2010), particularly for shorter-wavelength waves arising from submarine landslides, as discussed for example by Watts *et al.* (2003) or Lynett and Liu (2002). A variety of dispersive terms might be added; see Bona, Chen and Saut (2002) for a recent survey of Boussinesq models and other alternatives. Dispersive terms generally arise in the form of third-order derivatives in the equation, generally requiring implicit algorithms in order to obtain stable results with reasonable time steps. This adds significant complication in the context of AMR and this option is not yet available in GeoClaw. An alternative is to use a dispersive numerical method, tuned to mimic the true dispersion; see for example Burwell, Tolkova and Chawla (2007). This seems problematic in the context of AMR.

Other flows require the use of more complex rheologies than water, for example landslides, debris flows, or lava flows. We are currently extending GeoClaw to handle debris flows using a variant of the models of Denlinger and Iverson (2004*a*, 2004*b*). For some related work with similar finite volume algorithms, see Pelanti, Bouchut and Mangeney (2008, 2011) and Costa and Macedonio (2005).

A variety of numerical approaches have been used for modelling tsunamis and other depth-averaged flows in recent years. There is a large literature on topics such as well-balanced methods and wetting-and-drying algorithms, not only for finite volume methods but also for finite difference, finite element, discontinuous Galerkin, and other methodologies, on both structured and unstructured grids. We have not attempted a full literature survey in this paper, either on numerical methods or tsunami science, but hope that

the references provided give the reader some entry into this field. There are a host of challenging problems remaining in the quest to better understand and protect against these hazards.

Acknowledgements

The GeoClaw software is an extension and generalization of the Tsunami-Claw software originally developed from Clawpack (www3) as part of the PhD thesis of one of the authors (George 2006). Our work on tsunami modelling has benefited greatly from the generous help of many researchers in this community, in particular Harry Yeh, who first encouraged us to work on this problem, members of the tsunami sedimentology group at the University of Washington, and members of the NOAA Center for Tsunami Research in Seattle. Several people provided valuable feedback on this paper, including B. Atwater, J. Bourgeois, G. Gelfenbaum, F. González, B. MacInnes, K. Mandli, and M. Martin. Thanks to Jody Bourgeois for providing Figures 2.1 and 2.2.

This work was supported in part by NSF grants CMS-0245206, DMS-0106511, DMS-0609661 and DMS-0914942, ONR grant N00014-09-1-0649, DOE grant DE-FG02-88ER25053, AFOSR grant FA9550-06-1-0203, the Founders Term Professorship in Applied Mathematics at the University of Washington, and a USGS Mendenhall Postdoctoral Fellowship.

REFERENCES[1]

B. F. Atwater *et al.* (2005), *The Orphan Tsunami of 1700*, University of Washington Press, Seattle.

D. S. Bale, R. J. LeVeque, S. Mitran and J. A. Rossmanith (2002), 'A wave propagation method for conservation laws and balance laws with spatially varying flux functions', *SIAM J. Sci. Comput.* **24**, 955–978.

J. P. Bardet, C. E. Synolakis, H. L. Davies, F. Imamura and E. A. Okal (2003), 'Landslide tsunamis: Recent findings and research directions', *Pure Appl. Geophys.* **160**, 1793–1809.

M. Berger and J. Oliger (1984), 'Adaptive mesh refinement for hyperbolic partial differential equations', *J. Comput. Phys.* **53**, 484–512.

M. J. Berger and P. Colella (1989), 'Local adaptive mesh refinement for shock hydrodynamics', *J. Comput. Phys.* **82**, 64–84.

M. J. Berger and R. J. LeVeque (1998), 'Adaptive mesh refinement using wave-propagation algorithms for hyperbolic systems', *SIAM J. Numer. Anal.* **35**, 2298–2316.

M. J. Berger and I. Rigoutsos (1991), 'An algorithm for point clustering and grid generation', *IEEE Trans. Sys. Man & Cyber.* **21**, 1278–1286.

[1] The URLs cited in this work were correct at the time of going to press, but the publisher and the authors make no undertaking that the citations remain live or are accurate or appropriate.

M. J. Berger, D. A. Calhoun, C. Helzel and R. J. LeVeque (2009), 'Logically rectangular finite volume methods with adaptive refinement on the sphere', *Phil. Trans. R. Soc. A* **367**, 4483–4496.

M. J. Berger, D. L. George, R. J. LeVeque and K. T. Mandli (2010), The GeoClaw software for depth-averaged flows with adaptive refinement. To appear in *Advances in Water Resources*. Available at: www.clawpack.org/links/awr11.

J. L. Bona, M. Chen and J.-C. Saut (2002), 'Boussinesq equations and other systems for small-amplitude long waves in nonlinear dispersive media I: Derivation and linear theory', *J. Nonlinear Sci.* **12**, 283–318.

N. Botta, R. Klein, S. Langenberg and S. Lützenkirchen (2004), 'Well balanced finite volume methods for nearly hydrostatic flows', *J. Comput. Phys.* **196**, 539–565.

F. Bouchut (2004), *Nonlinear Stability of Finite Volume Methods for Hyperbolic Conservation Laws and Well-Balanced Schemes for Sources*, Birkhäuser.

J. Bourgeois (2009), Geologic effects and records of tsunamis. In *The Sea*, Vol. 15 (E. N. Bernard and A. R. Robinson, eds), Harvard University Press, pp. 55–92.

D. Burwell, E. Tolkova and A. Chawla (2007), 'Diffusion and dispersion characterization of a numerical tsunami model', *Ocean Modelling* **19**, 10–30.

G. F. Carrier and H. Yeh (2005), 'Tsunami propagation from a finite source', *CMES* **10**, 113–121.

G. F. Carrier, T. T. Wu and H. Yeh (2003), 'Tsunami run-up and draw-down on a plane beach', *J. Fluid Mech.* **475**, 79–99.

M. J. Castro, P. G. LeFloch, M. L. Munoz and C. Parés (2008), 'Why many theories of shock waves are necessary: Convergence error in formally path-consistent schemes', *J. Comput. Phys.* **227**, 8107–8129.

A. Costa and G. Macedonio (2005), 'Numerical simulation of lava flows based on depth-averaged equations', *Geophys. Res. Lett.* **32**, L05304.

A. Dawson, D. Long and D. Smith (1988), 'The Storegga Slides: Evidence from eastern Scotland for a possible tsunami', *Marine Geology*, January 1988.

R. P. Denlinger and R. M. Iverson (2004*a*), 'Granular avalanches across irregular three-dimensional terrain 1: Theory and computation', *J. Geophys. Res.* **109**, F01014.

R. P. Denlinger and R. M. Iverson (2004*b*), 'Granular avalanches across irregular three-dimensional terrain 2: Experimental tests', *J. Geophys. Res.* **109**, F01015.

B. Einfeldt (1988), 'On Godunov-type methods for gas dynamics', *SIAM J. Numer. Anal.* **25**, 294–318.

B. Einfeldt, C. D. Munz, P. L. Roe and B. Sjogreen (1991), 'On Godunov type methods near low densities', *J. Comput. Phys.* **92**, 273–295.

S. Fomel and J. F. Claerbout (2009), 'Guest editors' introduction: Reproducible research', *Comput. Sci. Engrg* **11**, 5–7.

J. M. Gallardo, C. Parés and M. Castro (2007), 'On a well-balanced high-order finite volume scheme for shallow water equations with topography and dry areas', *J. Comput. Phys.* **227**, 574–601.

E. L. Geist and T. Parsons (2006), 'Probabilistic analysis of tsunami hazards', *Nat. Haz.* **37**, 277–314.

E. L. Geist, T. Parsons, U. S. ten Brink and H. J. Lee (2009), 'Tsunami probability', **15**, 201–235.

G. Gelfenbaum and B. Jaffe (2003), 'Erosion and sedimentation from the 17 July, 1998 Papua New Guinea tsunami', *Pure Appl. Geophys.* **160**, 1969–1999.

D. George (2010), 'Adaptive finite volume methods with well-balanced Riemann solvers for modeling floods in rugged terrain: Application to the Malpasset dam-break flood (France, 1959)', *Int. J. Numer. Meth. Fluids*.

D. L. George (2006), Finite volume methods and adaptive refinement for tsunami propagation and inundation. PhD thesis, University of Washington.

D. L. George (2008), 'Augmented Riemann solvers for the shallow water equations over variable topography with steady states and inundation', *J. Comput. Phys.* **227**, 3089–3113.

D. L. George and R. J. LeVeque (2006), 'Finite volume methods and adaptive refinement for global tsunami propagation and local inundation', *Science of Tsunami Hazards* **24**, 319–328.

F. I. González and Y. A. Kulikov (1993), Tsunami dispersion observed in the deep ocean. In *Tsunamis in the World* (S. Tinti, ed.), Vol. 1 of *Advances in Natural and Technological Hazards Research*, Kluwer, pp. 7–16.

F. I. González, E. L. Geist, B. Jaffe, U. Kanoglu *et al.* (2009), 'Probabilistic tsunami hazard assessment at Seaside, Oregon, for near- and far-field seismic sources', *J. Geophys. Res.* **114**, C11023.

L. Gosse (2000), 'A well-balanced flux-vector splitting scheme designed for hyperbolic systems of conservation laws with source terms', *Comput. Math. Appl.* **39**, 135–159.

L. Gosse (2001), 'A well-balanced scheme using non-conservative products designed for hyperbolic systems of conservation laws with source terms', *Math. Mod. Meth. Appl. Sci.* **11**, 339–365.

J. M. Greenberg and A. Y. LeRoux (1996), 'A well-balanced scheme for numerical processing of source terms in hyperbolic equations', *SIAM J. Numer. Anal.* **33**, 1–16.

S. T. Grilli, M. Ioualalen, J. Asavanant, F. Shi, J. T. Kirby and P. Watts (2007), 'Source constraints and model simulation of the December 26, 2004, Indian Ocean Tsunami', *J. Waterway, Port, Coastal, and Ocean Engineering* **133**, 414.

H. Haflidason, H. Sejrup, A. Nygård, J. Mienert and P. Bryn (2004), 'The Storegga Slide: Architecture, geometry and slide development', *Marine Geology*, January 2004.

J. Hammack and H. Segur (1978), 'Modelling criteria for long water waves', *J. Fluid Mech.* **84**, 337–358.

A. Harten, P. D. Lax and B. van Leer (1983), 'On upstream differencing and Godunov-type schemes for hyperbolic conservation laws', *SIAM Review* **25**, 35–61.

B. Higman, G. Gelfenbaum, P. Lynett, A. Moore and B. Jaffe (2007), Predicted sedimentary record of reflected bores. In *Proc. Sixth International Symposium on Coastal Engineering and Science of Coastal Sediment Processes*, ASCE, pp. 1–14.

K. Hirata, E. Geist, K. Satake, Y. Tanioka and S. Yamaki (2003), 'Slip distribution of the 1952 Tokachi-Oki earthquake (M 8.1) along the Kuril Trench deduced from tsunami waveform inversion', *J. Geophys. Res.*

K. Huntington, J. Bourgeois, G. Gelfenbaum, P. Lynett, B. Jaffe, H. Yeh and R. Weiss (2007), 'Sandy signs of a tsunami's onshore depth and speed', *EOS* **88**, 577–578. www.agu.org/journals/eo/eo0752/2007EO52_tabloid.pdf.

A. In (1999), 'Numerical evaluation of an energy relaxation method for inviscid real fluids', *SIAM J. Sci. Comput.* **21**, 340–365.

K. Jankaew, B. F. Atwater, Y. Sawai, M. Choowong, T. Charoentitirat, M. E. Martin and A. Prendergast (2008), 'Medieval forewarning of the 2004 Indian Ocean tsunami in Thailand', *Nature* **455**, 1228–1231.

H. M. Kelsey, A. R. Nelson, E. Hemphill-Haley and R. C. Witter (2005), 'Tsunami history of an Oregon coastal lake reveals a 4600 yr record of great earthquakes on the Cascadia subduction zone', *GSA Bulletin* **117**, 1009–1032.

Z. Kowalik, W. Knight, T. Logan and P. Whitmore (2005), 'Modeling of the global tsunami: Indonesian Tsunami of 26 December 2004.', *Science of Tsunami Hazards* **23**, 40–56.

J. O. Langseth and R. J. LeVeque (2000), 'A wave-propagation method for three-dimensional hyperbolic conservation laws', *J. Comput. Phys.* **165**, 126–166.

R. J. LeVeque (1996), 'High-resolution conservative algorithms for advection in incompressible flow', *SIAM J. Numer. Anal.* **33**, 627–665.

R. J. LeVeque (2002), *Finite Volume Methods for Hyperbolic Problems*, Cambridge University Press.

R. J. LeVeque (2009), 'Python tools for reproducible research on hyperbolic problems', *Comput. Sci. Engrg* **11**, 19–27.

R. J. LeVeque (2010), 'A well-balanced path-integral f-wave method for hyperbolic problems with source terms', *J. Sci. Comput.* doi:10.1007/s10915-010-9411-0. www.clawpack.org/links/wbfwave10.

R. J. LeVeque and D. L. George (2004), High-resolution finite volume methods for the shallow water equations with bathymetry and dry states. In *Proc. Long-Wave Workshop, Catalina* (P. L.-F. Liu, H. Yeh and C. Synolakis, eds), Vol. 10, World Scientific, pp. 43–73. www.amath.washington.edu/~rjl/pubs/catalina04/.

R. J. LeVeque and M. Pelanti (2001), 'A class of approximate Riemann solvers and their relation to relaxation schemes', *J. Comput. Phys.* **172**, 572–591.

P. L. Liu, H. Yeh and C. Synolakis, eds (2008), *Advanced Numerical Models for Simulating Tsunami Waves and Runup*, Vol. 10 of *Advances in Coastal and Ocean Engineering*, World Scientific.

P. Liu, P. Lynett, H. Fernando, B. Jaffe and H. Fritz (2005), 'Observations by the International Tsunami Survey Team in Sri Lanka', *Science* **308**, 1595.

P. Liu, S. Woo and Y. Cho (1998), 'Computer programs for tsunami propagation and inundation'. ceeserver.cee.cornell.edu/pll-group/comcot.htm.

P. Lynett and P. L. Liu (2002), 'A numerical study of submarine-landslide-generated waves and run-up', *Proc. Royal Soc. London Ser. A* **458**, 2885–2910.

C. L. Mader and M. L. Gittings (2002), 'Modeling the 1958 Lituya Bay mega tsunami, II', *Science of Tsunami Hazards* **20**, 241.

K. T. Mandli (2010), Personal communication.

L. Mansinha and D. Smylie (1971), 'The displacement fields of inclined faults', *Bull. Seism. Soc. Amer.* **61**, 1433–1438.

M. E. Martin, R. Weiss, J. Bourgeois, T. K. Pinegina, H. Houston and V. V. Titov (2008), 'Combining constraints from tsunami modeling and sedimentology to untangle the 1969 Ozernoi and 1971 Kamchatskii tsunamis', *Geophys. Res. Lett.* **35**, L01610.

D. G. Masson, C. B. Harbitz, R. B. Wynn, G. Pedersen and F. Løvholt (2006), 'Submarine landslides: processes, triggers and hazard prediction', *Philos. Trans. Royal Soc. A: Math. Phys. Engrg Sci.* **364**, 2009.

D. S. McCulloch (1966), Slide-induced waves, seiching and ground fracturing caused by the earthquake of March 27, 1964, at Kenai Lake, Alaska. USGS Professional Paper 543-A. www.dggs.dnr.state.ak.us/pubs/pubs?reqtype=citation&ID=3884.

C. Meinig, S. E. Stalin, A. I. Nakamura, F. González and H. B. Milburn (2006), Technology developments in real-time tsunami measuring, monitoring and forecasting. In *OCEANS, 2005: Proc. MTS/IEEE*, pp. 1673–1679.

Z. Merali (2010), 'Why scientific computing does not compute', *Nature* **467**, 775–777.

D. J. Miller (1960), Giant waves in Lituya Bay, Alaska. USGS Professional Paper 354-C. www.dggs.dnr.state.ak.us/pubs/pubs?reqtype=citation&ID=3852.

S. Noelle, N. Pankrantz, G. Puppo and J. R. Natvig (2006), 'Well-balanced finite volume schemes of arbitrary order of accuracy for shallow water flows', *J. Comput. Phys.* **213**, 474–499.

Y. Okada (1985), 'Surface deformation due to shear and tensile faults in a half-space', *Bull. Seism. Soc. Amer.* **75**, 1135–1154.

Y. Okada (1992), 'Internal deformation due to shear and tensile faults in a half-space', *Bull. Seism. Soc. Amer.* **82**, 1018–1040.

V. V. Ostapenko (1999), 'Numerical simulation of wave flows caused by a shoreside landslide', *J. Applied Mech. Tech. Phys.* **40**, 647–654.

M. Pelanti, F. Bouchut and A. Mangeney (2008), 'A Roe-type scheme for two-phase shallow granular flows over variable topography', *M2AN* **42**, 851–885.

M. Pelanti, F. Bouchut and A. Mangeney (2011), 'A Riemann solver for single-phase and two-phase shallow flow models based on relaxation: Relations with Roe and VFRoe solvers', *J. Comput. Phys.* **230**, 515–550.

D. B. Percival, D. W. Denbo, M. C. Eblé, E. Gica, H. O. Mofjeld, M. C. Spillane, L. Tang and V. V. Titov (2010), 'Extraction of tsunami source coefficients via inversion of DART buoy data', *Natural Hazards* doi:10.1007/s11069-010-9688-1.

G. Plafker, R. Kachadoorian, E. B. Eckel and L. R. Mayo (1969), Effects of the earthquake of March 27, 1964 on various communities. USGS Professional Paper 542G. www.dggs.dnr.state.ak.us/pubs/pubs?reqtype=citation&ID=3883.

J. J. Quirk (2003), Computational science: 'Same old silence, same old mistakes' 'Something more is needed ...'. In *Adaptive Mesh Refinement: Theory and Applications* (T. Plewa, T. Linde and V. G. Weirs, eds), Vol. 41 of *Lecture Notes in Computational Science and Engineering*, Springer, pp. 3–28.

P. J. Roache (1998), *Verification and Validation in Computational Science and Engineering*, Hermosa Publishers, Albuquerque, NM.

T. Saito, T. Matsuzawa, K. Obara and T. Baba (2010), 'Dispersive tsunami of the 2010 Chile earthquake recorded by the high-sampling-rate ocean-bottom pressure gauges', *Geophys. Res. Lett.* **37**, L22303.

K. Satake, K. Shimazaki, Y. Tsuji and K. Ueda (1996), 'Time and size of a giant earthquake in Cascadia inferred from Japanese tsunami records of January 1700', *Nature* **379**, 246–249.

K. Satake, K. Wang and B. Atwater (2003), 'Fault slip and seismic moment of the 1700 Cascadia earthquake inferred from Japanese tsunami descriptions', *J. Geophys. Res.* **108**, 2535.

C. Synolakis, J. Bardet, J. Borrero, H. Davies, E. Okal, E. Silver, S. Sweet and D. Tappin (2002), 'The slump origin of the 1998 Papua New Guinea tsunami', *Proc. Royal Soc. London Ser. A: Math. Phys. Engrg Sci.* **458**, 763.

C. E. Synolakis and E. N. Bernard (2006), 'Tsunami science before and beyond Boxing Day 2004', *Philos. Trans. Royal Soc. A: Math. Phys. Engrg Sci.* **364**, 2231.

C. E. Synolakis, E. N. Bernard, V. V. Titov, U. Kânoğlu and F. I. González (2008), 'Validation and verification of tsunami numerical models', *Pure Appl. Geophys.* **165**, 2197–2228.

W. C. Thacker (1981), 'Some exact solutions to the nonlinear shallow water wave equations', *J. Fluid Mech.* **107**, 499–508.

V. V. Titov and C. E. Synolakis (1995), 'Modeling of breaking and nonbreaking long wave evolution and runup using VTCS-2', *J. Waterways, Ports, Coastal and Ocean Engineering* **121**, 308–316.

V. V. Titov and C. E. Synolakis (1998), 'Numerical modeling of tidal wave runup', *J. Waterways, Ports, Coastal and Ocean Engineering* **124**, 157–171.

V. V. Titov, F. I. Gonzalez, E. N. Bernard, M. C. Eble, H. O. Mofjeld, J. C. Newman and A. J. Venturato (2005), 'Real-time tsunami forecasting: Challenges and solutions', *Nat. Hazards* **35**, 35–41.

E. F. Toro (2001), *Shock-Capturing Methods for Free-Surface Shallow Flows*, Wiley.

X. Wang and P. L. Liu (2007), 'Numerical simulations of the 2004 Indian Ocean tsunamis: Coastal effects', *J. Earthquake Tsunami* **1**, 273–297.

P. Watts, S. Grilli, J. Kirby, G. J. Fryer and D. R. Tappin (2003), 'Landslide tsunami case studies using a Boussinesq model and a fully nonlinear tsunami generation model', *Nat. Haz. Earth Sys. Sci.* **3**, 391–402.

R. Weiss, H. Fritz and K. Wünnemann (2009), 'Hybrid modeling of the mega-tsunami runup in Lituya Bay after half a century', *Geophys. Res. Lett.* **36**, L09602.

H. Yeh, R. K. Chadha, M. Francis, T. Katada, G. Latha, C. Peterson, G. Raghuramani and J. P. Singh (2006), 'Tsunami runup survey along the southeast Indian coast', *Earthquake Spectra* **22**, S173–S186.

H. Yeh, P. L. Liu and C. Synolakis, eds (1996), *Long-Wave Runup Models*, World Scientific.

H. Yeh, P. Liu, M. Briggs and C. Synolakis (1994), 'Propagation and amplification of tsunamis at coastal boundaries', *Nature* **372**, 353–355.

Online references

www1: AMROC software: amroc.sourceforge.net/.
www2: Chombo software: seesar.lbl.gov/anag/chombo/.
www3: Clawpack software: www.clawpack.org.
www4: COMCOT software: ceeserver.cee.cornell.edu/pll-group/comcot.htm.
www5: DART data: www.ndbc.noaa.gov/.
www6: FLASH software: flash.uchicago.edu/website/home/.
www7: GeoClaw software: www.clawpack.org/geoclaw.
www8: Hilo, HI 1/3 arc-second MHW Tsunami Inundation DEM:
www.ngdc.noaa.gov/mgg/inundation/.
www9: National Geophysical Data Center (NGDC) GEODAS:
www.ngdc.noaa.gov/mgg/gdas/gd_designagrid.html.
www10: NOAA Tsunami Inundation Digital Elevation Models (DEMs):
www.ngdc.noaa.gov/mgg/inundation/tsunami/.
www11: SAMRAI: computation.llnl.gov/casc/SAMRAI/.
www12: USGS source for 27 February 2010 earthquake:
earthquake.usgs.gov/earthquakes/eqinthenews/2010/us2010tfan/.
www13: Webpage for this paper: www.clawpack.org/links/an11.