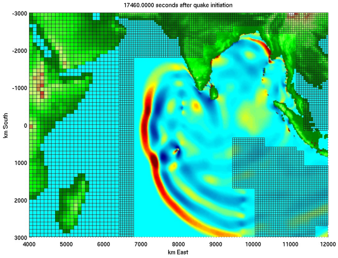
Multiscale Modeling of Tsunami Propagation and Inundation

Randall J. LeVeque
Department of Applied Mathematics
University of Washington



CLAWPACK Software:

`http://www.amath.washington.edu/~claw`

Supported in part by NSF, DOE

Outline

- Tsunami modeling, shallow water equations
- Finite volume methods for hyperbolic equations
- Riemann problems and Godunov's method
- Wave limiters and high-resolution methods
- Software: CLAWPACK
- Riemann problems for tsunamis: bathymetry and dry cells
- Adaptive mesh refinement
- AMR issues for tsunamis
- Validation and benchmarks

Some collaborators on these projects

Algorithms, software

Marsha Berger, NYU
Donna Calhoun, UW
Phil Colella, UC-Berkeley
Jan Olav Langseth, Oslo

Tsunamis

David George, UW grad student
Harry Yeh, OSU

Supported in part by NSF, DOE

Tsunamis

Generated by

- Earthquakes,
- Landslides,
- Submarine landslides,
- Volcanoes,
- Meteorite or asteroid impact

There were 97 significant tsunamis during the 1990's, causing 16,000 casualties.

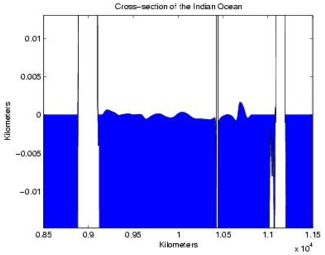
There have been approximately 28 tsunamis with run-up greater than 1m on the west coast of the U.S. since 1812.

Tsunamis

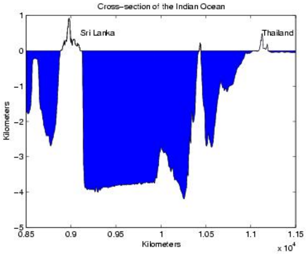
- Small amplitude in ocean (< 1 meter) but can grow to 10s of meters at shore.
- Run-up along shore can inundate 100s of meters inland
- Long wavelength (as much as 200 km)
- Propagation speed $\sqrt{gh}$ (bunching up at shore)
- Average depth of Pacific or Indian Ocean is 4km
  $\Longrightarrow$ average speed 200 m/s $\approx$ 450 mph

Cross section of Indian Ocean & tsunami

Surface elevation
on scale of 10 meters:



Cross-section
on scale of kilometers:



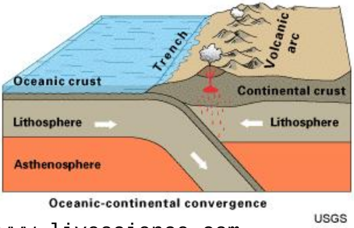
Sumatra event of December 26, 2004

Magnitude 9.1 quake near Sumatra, where Indian tectonic plate is being subducted under the Burma platelet.

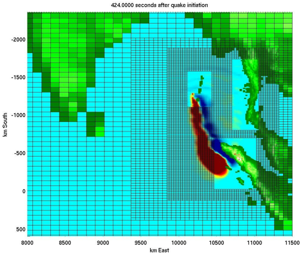
Rupture along subduction zone
$\approx$ 1200 km long, 150 km wide

Propagating at $\approx$ 2 km/sec (for $\approx$ 10 minutes)

Fault slip up to 15 m, uplift of several meters.

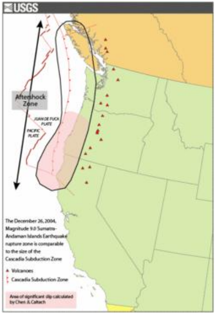


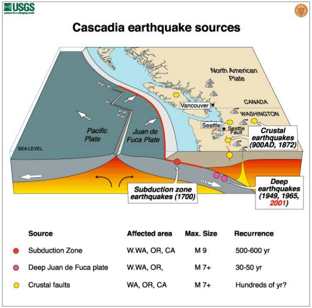
www.livescience.com



(Similar to Cascadia subduction zone off WA coast)

Cascadia subduction fault





- 1200 km long off-shore fault stretching from northern California to southern Canada.
- Last major rupture: magnitude 9.0 earthquake on January 26, 1700.
- Tsunami recorded in Japan with run-up of up to 5 meters.
- Historically there appear to be magnitude 8 or larger quakes every 500 years on average.

Our work on tsunami modeling

- Original thrust:
  NSF grant with Harry Yeh (OSU) and Joe Hammack / Diane Henderson (PSU) to do 1D and 2D simulations to complement wave tank experiments.
  Small-scale computations near shore, uniform grids.

- After December 26, 2004:
  Focus on Sumatra event.
  Model of Bay of Bengal, Indian Ocean, initially with uniform coarse grid.
  Addition of AMR for propagating wave.
  AMR near coastline to capture run-up and inundation.

- Latest results:
  Zoom in on Madras harbor area.
  Factor of 1024 refinement from coarsest to finest grids.
  Next: Compare with field data taken by Harry Yeh.

Our work on tsunami modeling

- Original thrust:
  NSF grant with Harry Yeh (OSU) and Joe Hammack / Diane Henderson (PSU) to do 1D and 2D simulations to complement wave tank experiments.
  Small-scale computations near shore, uniform grids.

- After December 26, 2004:
  Focus on Sumatra event.
  Model of Bay of Bengal, Indian Ocean, initially with uniform coarse grid.
  Addition of AMR for propagating wave.
  AMR near coastline to capture run-up and inundation.

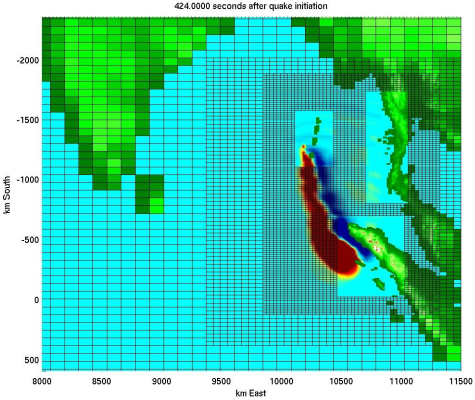
- Latest results:
  Zoom in on Madras harbor area.
  Factor of 1024 refinement from coarsest to finest grids.
  Next: Compare with field data taken by Harry Yeh.

Our work on tsunami modeling

- Original thrust:
  NSF grant with Harry Yeh (OSU) and Joe Hammack / Diane Henderson (PSU) to do 1D and 2D simulations to complement wave tank experiments.
  Small-scale computations near shore, uniform grids.
- After December 26, 2004:
  Focus on Sumatra event.
  Model of Bay of Bengal, Indian Ocean, initially with uniform coarse grid.
  Addition of AMR for propagating wave.
  AMR near coastline to capture run-up and inundation.
- Latest results:
  Zoom in on Madras harbor area.
  Factor of 1024 refinement from coarsest to finest grids.
  Next: Compare with field data taken by Harry Yeh.

Tsunami simulations

- 2D shallow water + bathymetry
- Finite volume method
- Cartesian grid
- Cells can be dry ($h = 0$)
- Cells become wet/dry as wave moves on shore
- Mesh refinement on rectangular patches
- Adaptive — follows wave, more levels near shore



Tsunami simulations

Movies:

Fault area

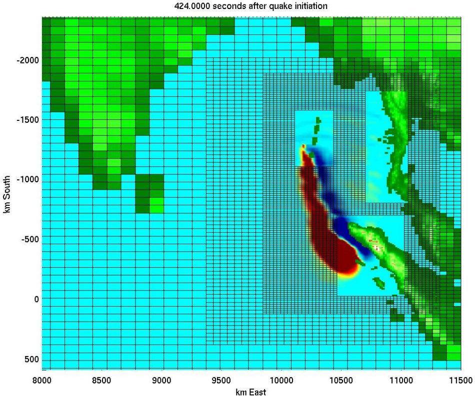
Bay of Bengal

Sri Lanka

Indian Ocean

Zoom on Madras

Slice of Madras Harbor



For movies, see

`http://www.amath.washington.edu/∼dgeorge/research.html`

Tsunami simulations

Latest simulation:

- 4 levels of refinement.
- Level 1: 1 degree resolution ($\Delta x \approx 60$ nautical miles)
- Level 2 refined by 8.
- Level 3 refined by 8: $\Delta x \approx 1$ nautical mile (only near coast)
- Level 4 refined by 16: $\Delta x \approx 100$ meters (only near Madras)

≈ 6 hours on single CPU PC. (< 2 hours with only 3 levels)

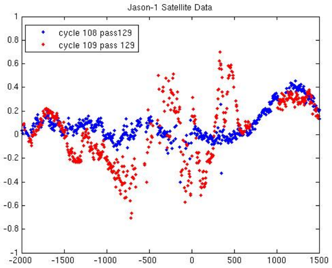
Next:

- Obtain better topography (and bathymetry) data.
- Add seawall, buildings.
- Compare to field data collected by Harry Yeh in January.
- Global model on sphere, compare to tide gauge data.

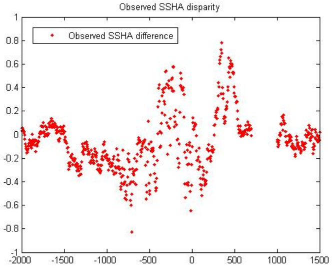
Satellite data

Jason-1 Satellite passed over the Indian Ocean during the tsunami event.

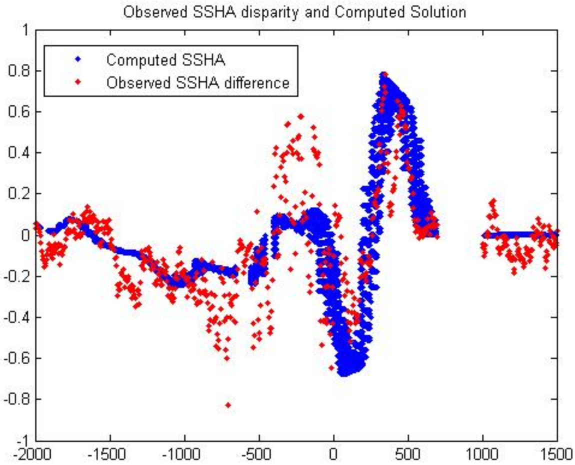
Surface height on two passes
(one a week before)



Disparity shows tsunami:



Comparison of simulation with satellite data



Shallow water equations with topography $B(x)$

$$\begin{aligned} h_t + (hu)_x &= 0 \\ (hu)_t + \left(hu^2 + \frac{1}{2}gh^2\right)_x &= -ghB_x(x) \end{aligned}$$

$h(x,t) =$ depth of water
$u(x,t) =$ horizontal velocity

This has the form of a conservation law with a source term:

$$q_t + f(q)_x = \psi(q,x),$$

where

$$q = \begin{bmatrix} h \\ hu \end{bmatrix}, \quad f(q) = \begin{bmatrix} hu \\ hu^2 + gh^2/2 \end{bmatrix}, \quad \psi(q,x) = \begin{bmatrix} 0 \\ -ghB'(x) \end{bmatrix}.$$

Shallow water equations with topography $B(x,y)$

$$
\begin{aligned}
h_t + (hu)_x + (hv)_y &= 0 \\
(hu)_t + \left(hu^2 + \frac{1}{2}gh^2\right)_x + (huv)_y &= -ghB_x(x,y) \\
(hv)_t + (huv)_x + \left(hv^2 + \frac{1}{2}gh^2\right)_y &= -ghB_y(x,y)
\end{aligned}
$$

Applications:

- Tsunamis
- Estuaries
- River flooding, dam breaks
- Debris flows from volocanic eruptions

High resolution finite volume methods

Hyperbolic conservation law:

$$1D: q_t + f(q)_x = 0 \qquad 2D: q_t + f(q)_x + g(q)_y = 0$$

$$1D: q_t + f'(q)q_x = 0 \qquad 2D: q_t + f'(q)q_x + g'(q)q_y = 0$$

Variable coefficient linear hyperbolic system:

$$1D: q_t + A(x)q_x = 0 \qquad 2D: q_t + A(x,y)q_x + B(x,y)q_y = 0$$

Def: Hyperbolic if eigenvalues of Jacobian $f'(q)$ in 1D or $\alpha f'(q) + \beta g'(q)$ in 2D are real and there exists a complete set of eigenvectors.

Eigenvalues are wave speeds, eigenvectors yield decomposition of data into waves.

Finite-difference Methods

- Pointwise values $Q_i^n \approx q(x_i, t_n)$
- Approximate derivatives by finite differences
- Assumes smoothness

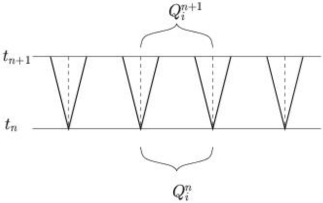
Finite-volume Methods

- Approximate cell averages: $Q_i^n \approx \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} q(x, t_n)\, dx$
- Integral form of conservation law,

$$\frac{\partial}{\partial t} \int_{x_{i-1/2}}^{x_{i+1/2}} q(x, t)\, dx \quad = \quad f(q(x_{i-1/2}, t)) - f(q(x_{i+1/2}, t))$$

leads to conservation law $q_t + f_x = 0$ but also directly to numerical method.

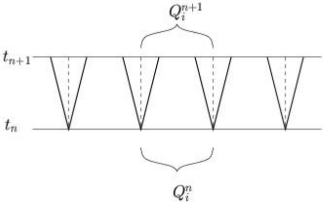
Godunov's Method for $q_t + f(q)_x = 0$



1. Solve Riemann problems at all interfaces, yielding waves $\mathcal{W}_{i-1/2}^p$ and speeds $s_{i-1/2}^p$, for $p = 1,\ 2,\ \ldots,\ m$.

Riemann problem: Original equation with piecewise constant data.

Godunov's Method for $q_t + f(q)_x = 0$

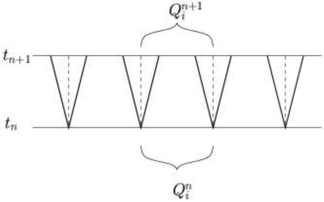


Then either:

1. Compute new cell averages by integrating over cell at time t_{n+1},

or ...

Godunov's Method for $q_t + f(q)_x = 0$



Then either:

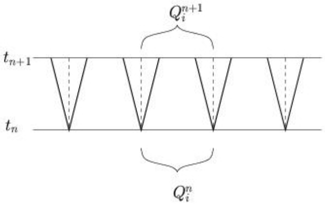
1. Compute new cell averages by integrating over cell at time t_{n+1},

or...

2. Compute fluxes at interfaces and flux-difference,

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

Godunov's Method for $q_t + f(q)_x = 0$



or ...

3. Update old cell averages by contributions from all waves entering the cell.

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x}[\mathcal{A}^+\Delta Q_{i-1/2} + \mathcal{A}^-\Delta Q_{i+1/2}]$$

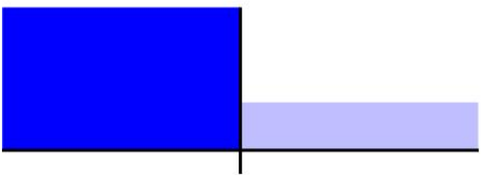
where $\mathcal{A}^\pm \Delta Q_{i-1/2} = \sum_{i=1}^{m} (s_{i-1/2}^p)^\pm \mathcal{W}_{i-1/2}^p$.

The Riemann problem

The Riemann problem for $q_t + f(q)_x = 0$ has special initial data

$$q(x,0) = \begin{cases} q_l & \text{if } x < x_{i-1/2} \\ q_r & \text{if } x > x_{i-1/2} \end{cases}$$

Dam break problem for shallow water equations

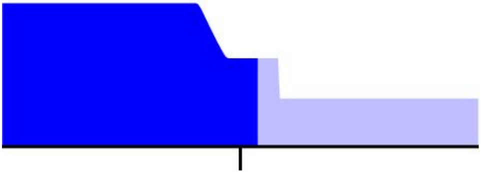


The Riemann problem

The Riemann problem for $q_t + f(q)_x = 0$ has special initial data

$$q(x,0) = \begin{cases} q_l & \text{if } x < x_{i-1/2} \\ q_r & \text{if } x > x_{i-1/2} \end{cases}$$

Dam break problem for shallow water equations

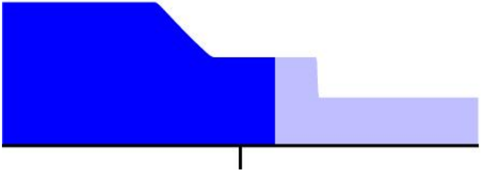


The Riemann problem

The Riemann problem for $q_t + f(q)_x = 0$ has special initial data

$$q(x,0) = \begin{cases} q_l & \text{if } x < x_{i-1/2} \\ q_r & \text{if } x > x_{i-1/2} \end{cases}$$

Dam break problem for shallow water equations

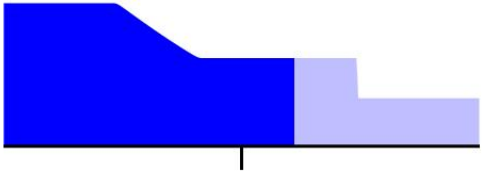


The Riemann problem

The Riemann problem for $q_t + f(q)_x = 0$ has special initial data

$$q(x,0) = \begin{cases} q_l & \text{if } x < x_{i-1/2} \\ q_r & \text{if } x > x_{i-1/2} \end{cases}$$

Dam break problem for shallow water equations

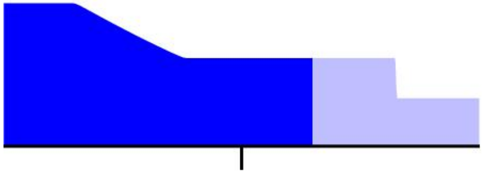


The Riemann problem

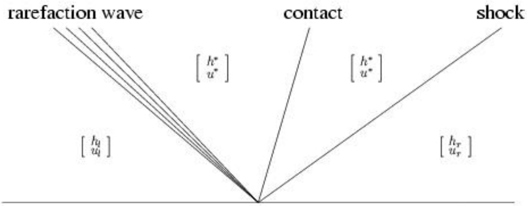
The Riemann problem for $q_t + f(q)_x = 0$ has special initial data

$$q(x,0) = \begin{cases} q_l & \text{if } x < x_{i-1/2} \\ q_r & \text{if } x > x_{i-1/2} \end{cases}$$

Dam break problem for shallow water equations



Riemann solution for the SW equations



The Roe solver uses the solution to a linear system

$$q_t + \hat{A}_{i-1/2} q_x = 0, \qquad \hat{A}_{i-1/2} = f'(q_{\text{ave}}).$$

All waves are simply discontinuities.

Typically a fine approximation if jumps are approximately correct.

Upwind wave-propagation algorithm

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x}\left[\sum_{p=1}^{m}(s_{i-1/2}^p)^+\mathcal{W}_{i-1/2}^p + \sum_{p=1}^{m}(s_{i+1/2}^p)^-\mathcal{W}_{i+1/2}^p\right]$$

where

$$s^+ = \max(s,0), \qquad s^- = \min(s,0).$$

Note: Requires only waves and speeds.

Applicable also to hyperbolic problems not in conservation form.

For $q_t + f(q)_x = 0$, conservative if waves chosen properly,
e.g. using Roe-average of Jacobians.

Great for general software, but only first-order accurate (upwind method for linear systems).

Wave-propagation form of high-resolution method

$$
\begin{aligned}
Q_i^{n+1} &= Q_i^n - \frac{\Delta t}{\Delta x}\left[\sum_{p=1}^{m}(s_{i-1/2}^p)^+\mathcal{W}_{i-1/2}^p + \sum_{p=1}^{m}(s_{i+1/2}^p)^-\mathcal{W}_{i+1/2}^p\right] \\
&\quad - \frac{\Delta t}{\Delta x}(\tilde{F}_{i+1/2} - \tilde{F}_{i-1/2})
\end{aligned}
$$

Correction flux:

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

where $\widetilde{\mathcal{W}}_{i-1/2}^p$ is a limited version of $\mathcal{W}_{i-1/2}^p$ to avoid oscillations.

(Unlimited waves $\widetilde{\mathcal{W}}^p = \mathcal{W}^p \implies$ Lax-Wendroff for a linear system $\implies$ nonphysical oscillations near shocks.)

CLAWPACK

`http://www.amath.washington.edu/~claw/`

- Fortran codes with Matlab graphics routines.
- Many examples and applications to run or modify.
- 1d, 2d, and 3d.
- Adaptive mesh refinement, MPI for parallel computing.

User supplies:

- Riemann solver, splitting data into waves and speeds
  (Need not be in conservation form)
- Boundary condition routine to extend data to ghost cells
  Standard `bc1.f` routine includes many standard BC's
- Initial conditions — `qinit.f`
- Source terms — `src1.f`

Some other applications

- Volcanic flows, dusty gas jets, pyroclastic surges
- Acoustics, ultrasound, seismology, lithotripsy
- Elasticity, plasticity, nonlinear elasticity
- Electromagnetic waves, photonic crystals
- Flow in porous media, groundwater contamination
- Oil reservoir simulation
- Geophysical flow on the sphere
- Hyperbolic equations on general curved manifolds (CLAWMAN)
- Chemotaxis and pattern formation
- Semiconductor modeling
- Traffic flow
- Multi-fluid, multi-phase flows, bubbly flow
- Incompressible flow (projection methods or streamfunction vorticity)
- Combustion, detonation waves
- Astrophysics: binary stars, planetary nebulae, jets
- Magnetohydrodynamics, plasmas
- Relativistic flow, black hole accretion
- Numerical relativity — gravitational waves, cosmology

Riemann solver for tsunamis

Issues:

- Bottom topography varies on scale of 4km
- Wave amplitude on scale of 1m
- Some cells are dry
- Cells become wet or dry as wave moves along shore

Approximate Riemann Solvers

Approximate true Riemann solution by set of waves consisting of finite jumps propagating at constant speeds.

Local linearization:

Replace $q_t + f(q)_x = 0$ by

$$q_t + \hat{A} q_x = 0,$$

where $\hat{A} = \hat{A}(q_l, q_r) \approx f'(q_{ave})$.

Then decompose

$$q_r - q_l = \alpha^1 \hat{r}^1 + \cdots \alpha^m \hat{r}^m$$

to obtain waves $\mathcal{W}^p = \alpha^p \hat{r}^p$ with speeds $s^p = \hat{\lambda}^p$.

HLLE Solver

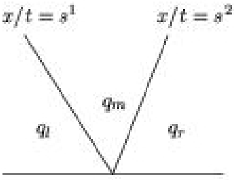
Harten – Lax – van Leer: Use only 2 waves with

$s^1 =$minimum characteristic speed

$s^2 =$maximum characteristic speed

Conservation implies unique value for middle state q_m.

Einfeldt: Also use Roe speeds in min and max.



Relaxation Schemes (Jin and Xin)

$u_t + f(u)_x = 0$ is replaced by the relaxation system

$$\begin{aligned} u_t + v_x &= 0 \\ v_t + D^2 u_x &= \frac{1}{\tau}(f(u) - v) \end{aligned}$$

where $D = \text{diag}(d^1, \ldots,\ d^m)$ (or more general...)

Gives linear hyperbolic system plus a relaxation source term:

$$\begin{bmatrix} u \\ v \end{bmatrix}_t + \begin{bmatrix} 0 & I \\ D^2 & 0 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}_x = \begin{bmatrix} 0 \\ (f(u) - v)/\tau \end{bmatrix}.$$

Eigenvalues are $\pm d^j$.

Convergence to original solution as $\tau \to 0$ if the *subcharacteristic condition* holds:

$$\min(-d_j) \le \lambda \le \max(d_j) \quad \text{for eigenvalues of } f'(u).$$

Relaxation Scheme

1. Given U^n and V^n, update over time Δt by solving the homogeneous linear hyperbolic system

$$\begin{bmatrix} u \\ v \end{bmatrix}_t + \begin{bmatrix} 0 & I \\ D^2 & 0 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}_x = 0.$$

   Call the new values U^* and V^*.

2. Update $U^*,\ V^*$ to $U^{n+1},\ V^{n+1}$ by solving the equations

$$\begin{aligned} u_t &= 0 \\ v_t &= \frac{1}{\tau}(f(u) - v) \end{aligned}$$

Note:

- $U^{n+1} = U^*$.
- For $\tau \to 0$ (the relaxed scheme of Jin and Xin),

$$V^{n+1} = f(U^*) = f(U^{n+1}).$$

Another view of relaxed scheme

1. Given U^n and V^n, update over time Δt by solving the linear system

$$\begin{bmatrix} u \\ v \end{bmatrix}_t + \begin{bmatrix} 0 & I \\ D^2 & 0 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}_x = 0.$$

   This gives the new value U^{n+1}.

2. Set $V^{n+1} = f(U^{n+1})$.

Even simpler: Store only U^n.

As "approximate Riemann solver", decompose

$$\begin{bmatrix} U_r - U_l \\ f(U_r) - f(U_l) \end{bmatrix}$$

into eigenvectors of

$$\begin{bmatrix} 0 & I \\ D^2 & 0 \end{bmatrix}.$$

Use resulting waves to update U. Note: There are now $2m$ waves.

RJL and M. Pelanti: Relaxation Riemann Solvers (JCP 2001)

Riemann problem for spatially-varying flux

$$q_t + f(q,x)_x = 0$$

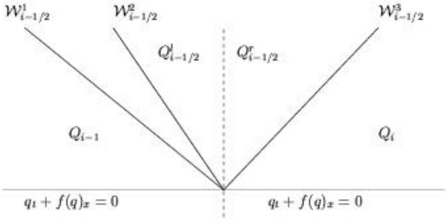
Applications:

- Wave propagation in heterogeneous nonlinear media
- Flow in heterogeneous porous media
- Traffic flow with varying road conditions
- Solving conservation laws on curved manifolds

Riemann problem for spatially-varying flux

$$q_t + f(q,x)_x = 0$$

Cell-centered discretization: Flux $f_i(q)$ defined in ith cell.



Need $f_{i-1}(Q^*_{i-1/2}) = f_i(Q^*_{i-1/2}) \implies m$ propagating waves plus jump in q ($= m$ waves for the m components of q).

Total of $2m$ waves, which can be found by decomposing

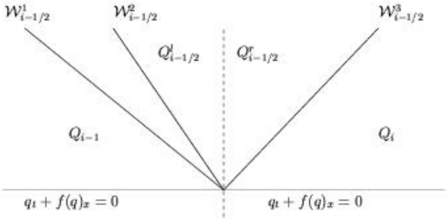
$$\begin{bmatrix} Q_i - Q_{i-1} \\ f_i(Q_i) - f_{i-1}(Q_{i-1}) \end{bmatrix} = \alpha^1 \begin{bmatrix} r^1 \\ s^1 r^1 \end{bmatrix} + \cdots + \alpha^m \begin{bmatrix} r^m \\ s^m r^m \end{bmatrix}$$

$$+ \alpha^{m+1} \begin{bmatrix} e^1 \\ 0 \end{bmatrix} + \cdots + \alpha^{2m} \begin{bmatrix} e^m \\ 0 \end{bmatrix}.$$

Note that this simplifies to first solving for $\alpha^1, \ldots, \alpha^m$ from

$$f_i(Q_i) - f_{i-1}(Q_{i-1}) = \alpha^1 s^1 r^1 + \cdots + \alpha^m s^m r^m$$

In fact this is all we need for the wave-propagation algorithms.

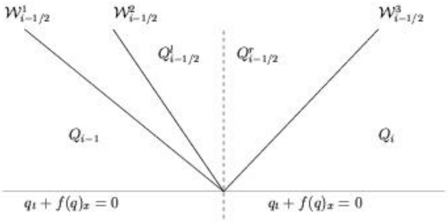


Flux-based wave decomposition (f-waves)

Choose waveforms r^p (e.g. eigenvectors of Jacobian on each side).

Then decompose flux difference:

$$f_i(Q_i) - f_{i-1}(Q_{i-1}) = \sum_{p=1}^{m} \beta^p r^p \equiv \sum_{p=1}^{m} \mathcal{Z}^p$$



Wave-propagation algorithm using f-waves

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x}[\mathcal{A}^+\Delta Q_{i-1/2} + \mathcal{A}^-\Delta Q_{i+1/2}] - \frac{\Delta t}{\Delta x}[\tilde{F}_{i+1/2} - \tilde{F}_{i-1/2}]$$

Standard version: $Q_i - Q_{i-1} = \sum_{p=1}^m \mathcal{W}_{i-1/2}^p$

$$\mathcal{A}^-\Delta Q_{i+1/2} = \sum_{p=1}^m (s_{i+1/2}^p)^- \mathcal{W}_{i+1/2}^p$$

$$\mathcal{A}^+\Delta Q_{i-1/2} = \sum_{p=1}^m (s_{i-1/2}^p)^+ \mathcal{W}_{i-1/2}^p$$

$$\tilde{F}_{i-1/2} = \frac{1}{2}\sum_{p=1}^m |s_{i-1/2}^p| \left(1 - \frac{\Delta t}{\Delta x}|s_{i-1/2}^p|\right) \widetilde{\mathcal{W}}_{i-1/2}^p.$$

Wave-propagation algorithm using f-waves

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x}[\mathcal{A}^+\Delta Q_{i-1/2} + \mathcal{A}^-\Delta Q_{i+1/2}] - \frac{\Delta t}{\Delta x}[\tilde{F}_{i+1/2} - \tilde{F}_{i-1/2}]$$

Using f-waves: $f_i(Q_i) - f_{i-1}(Q_{i-1}) = \sum_{p=1}^m \mathcal{Z}_{i-1/2}^p$

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

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

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

Source terms and quasi-steady solutions

$$q_t + f(q)_x = \psi(q)$$

Steady-state solution:

$$q_t = 0 \implies f(q)_x = \psi(q)$$

Balance between flux gradient and source.

Quasi-Steady solution:

Small perturbation propagating against steady-state background.

$$q_t \ll f(q)_x \approx \psi(q)$$

Want accurate calculation of perturbation.

Examples:

- Shallow water equations with bottom topography and flat surface
- Stationary atmosphere where pressure gradient balances gravity

Fractional steps for a quasisteady problem

Alternate between solving homogeneous conservation law

$$q_t + f(q)_x = 0 \tag{1}$$

and source term

$$q_t = \psi(q). \tag{2}$$

When $q_t \ll f(q)_x \approx \psi(q)$:

- Solving (1) gives large change in q
- Solving (2) should essentially cancel this change.

Numerical difficulties:

- (1) and (2) are solved by very different methods. Generally will not have proper cancellation.
- Nonlinear limiters are applied to $f(q)_x$ term, not to small-perturbation waves. Large variation in steady state hides structure of waves.

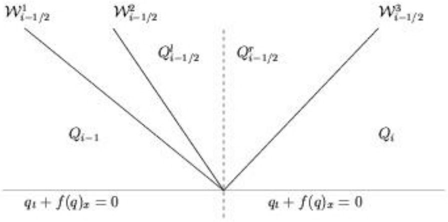
Incorporating source term in f-waves

$q_t + f(q)_x = \psi$ with $f(q)_x \approx \psi$.

Concentrate source at interfaces: $\Psi_{i-1/2}\,\delta(x - x_{i-1/2})$

Split $f(Q_i) - f(Q_{i-1}) - \Delta x \Psi_{i-1/2} = \sum_p \mathcal{Z}^p_{i-1/2}$

Use these waves in wave-propagation algorithm.



Incorporating source term in f-waves

$q_t + f(q)_x = \psi$ with $f(q)_x \approx \psi$.

Concentrate source at interfaces: $\Psi_{i-1/2}\,\delta(x - x_{i-1/2})$

Split $f(Q_i) - f(Q_{i-1}) - \Delta x \Psi_{i-1/2} = \sum_p \mathcal{Z}^p_{i-1/2}$

Use these waves in wave-propagation algorithm.

Steady state maintained:

If $\frac{f(Q_i)-f(Q_{i-1})}{\Delta x} = \Psi_{i-1/2}$ then $\mathcal{Z}^p \equiv 0$

Near steady state:

Deviation from steady state is split into waves and limited.

Riemann solver for shallow water with bathymetry

$$q = \begin{bmatrix} h \\ hu \end{bmatrix}, \quad f(q) = \begin{bmatrix} hu \\ hu^2 + \frac{1}{2}gh^2 \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}, \quad \psi = \begin{bmatrix} 0 \\ ghB'(x) \end{bmatrix}.$$

Robust solver (Dave George):

Split the vector $\begin{bmatrix} \Delta h \\ \Delta(hu) \\ \Delta f_2 \\ \Delta B \end{bmatrix}$ into 4 waves.

Modifications for dry cell problem (One of neighboring cells is dry or becomes dry).

Adaptive Mesh Refinement (AMR)

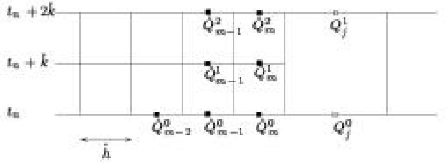
- Cluster grid points where needed
- Automatically adapt to solution
- Refined region moves in time-dependent problem

Basic approaches:

- Cell-by-cell refinement
  Quad-tree or Oct-tree data structure
  Structured or unstructured grid
- Refinement on “rectangular” patches
  Berger-Colella-Oliger style
  (AMRCLAW and CHOMBO-CLAW)

Time stepping algorithm for AMR

- Take 1 time step on coarse grid.
- Use space-time interpolation to set ghost cell values on fine grid near interface.
- Take K time steps on fine grid. ($K =$ refinement ratio)
- Replace coarse grid value by average of fine grid values on regions of overlap — better approximation and consistent representations.
- Conservative fix-up.



AMR time stepping for tsunami model

Normally $\Delta x,\ \Delta y,\ \Delta t$ are all refined by same factor K_L going from level L to $L+1$.

(Courant number is then the same on all grids)

For tsunami: Max wave speed in each cell is $|u| + \sqrt{gh}$.

In deep ocean: $\sqrt{gh} \approx 200m/s$, $\Delta t \approx 0.005\Delta x$.

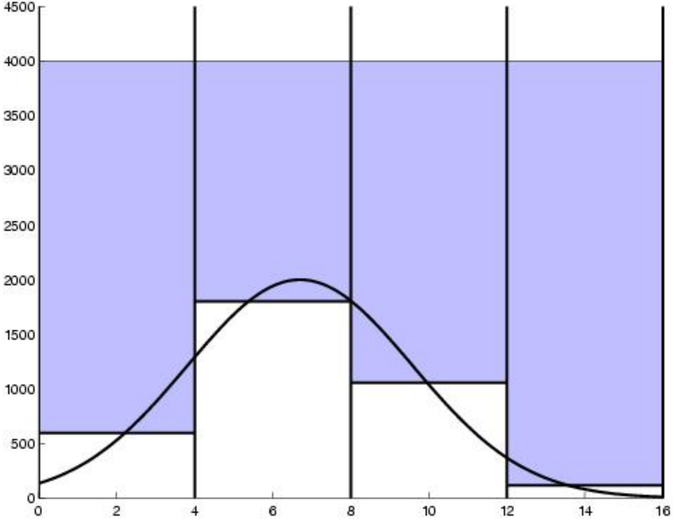
Suppose finest grids are only near shore, where $|u| + \sqrt{gh} < 10$, say. $\implies$ can take $\Delta t_f \approx 0.1\Delta x_f$

Anisotropic refinement: In going from level L to $L+1$,

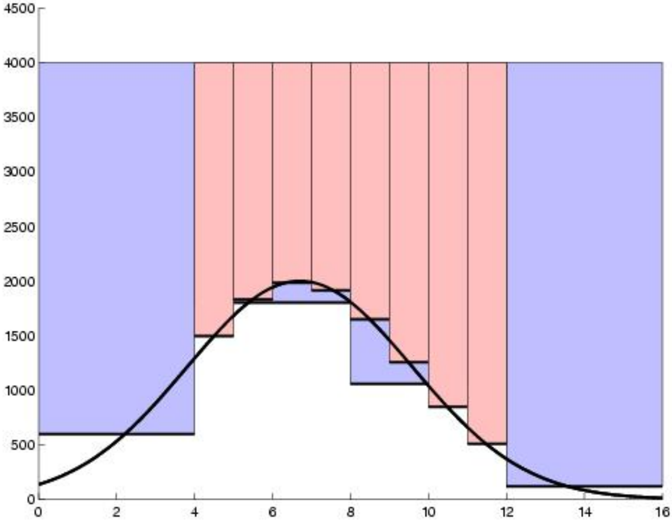
Refine Δx by K_{Lx}, Δy by K_{Ly}, Δt by K_{Lt}.

Near shore: $K_{Lx} = K_{Ly} = 8,\ K_{Lt} = 1$.

Grid refinement with bathymetry



Grid refinement with bathymetry



Grid refinement with bathymetry

Refining grid requires assigning interpolated Q values

- Simple linear interpolation of h doesn't work
- From h_i calculate $\eta_i = h_i + B_i$
- Interpolate η_i to fine grid to obtain η_j for $j = 1 : r$
  preserves constant surface!
- Compute h_j on fine grid from $h_j = \eta_j - B_j$
- Conservative provided $\frac{1}{r} \sum_{j=1}^{r} B_j = B_i$

Coarsening grid:

- Set $h_i = \frac{1}{r} \sum_{j=1}^{r} h_j$
- Then $h_i + B_i = \frac{1}{r} \sum (h_j + B_j)$

Validation of shallow water code

- 1d exact solutions of waves on a beach
- 2d benchmark problem with wave tank comparison
- Indian Ocean tsunami — lots of data
  - Jason-1 satellite data
  - run-up and inundation data from many coastal surveys
  - Data near Madras from Harry Yeh's survey
  - Tide gauge data from around the world

3rd Int’l workshop on long-wave runup models

Benchmark Problem 2: Scale model of part of coastline of Okushiri Island, site of 1993 tsunami.

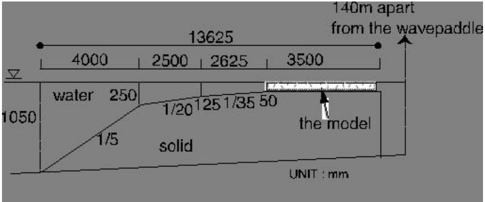
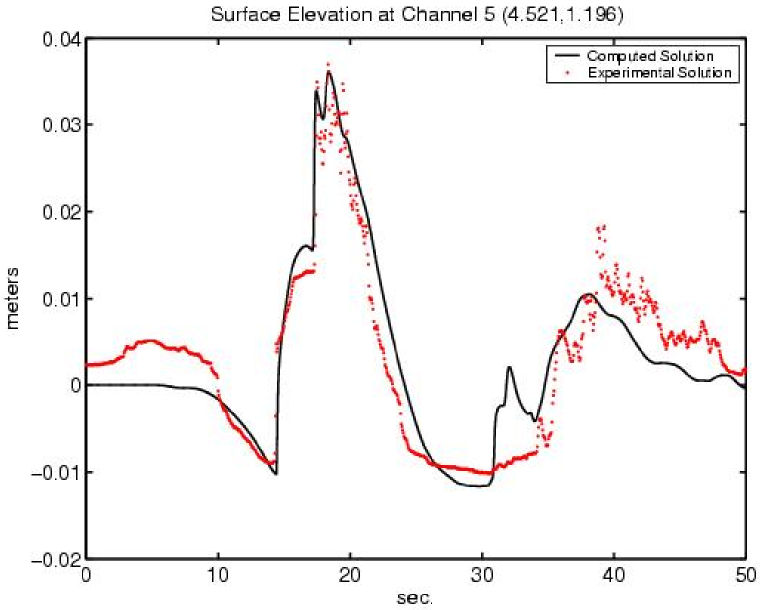


Fig. 1 Offshore Profile

Movie of wave tank ... Movie of simulation

Tide gauge data



Summary

- Tsunami model can handle global scale and local scale simultaneously.
- Preliminary validation looks good.

- Wave propagation problems are generally formulated as hyperbolic systems.
- Many practical applications in science and engineering.
- General software for high-resolution methods, AMR:
  `http://www.amath.washington.edu/~claw`
- Papers and simulations:

  `http://www.amath.washington.edu/~rjl/research.html`

  `http://www.amath.washington.edu/~rjl/research/tsunamis`