Clawpack Tutorial
Part I

Randall J. LeVeque
Applied Mathematics
University of Washington

Conservation Laws Package

www.clawpack.org

(pdf's will be posted and green links can be clicked)

Some collaborators

Marsha Berger, NYU
Jan Olav Langseth, FFI, Oslo
David George, USGS CVO
Donna Calhoun, CEA, Paris
Christiane Helzel, Bochum
David Ketcheson, KAUST
Sorin Mitran, UNC

Current students:

Kyle Mandli (PyClaw, GeoClaw, storm surges)
Jonathan Claridge (Implicit AMR)
Grady Lemoine (Cut cells, bone modeling)
Jihwan Kim (multi-layer, submarine landslides)

Funded in part by: NSF, DOE, NCAR, NIH, ONR, AFOSR
Founders Term Professorship

Outline

Monday:

- Overview of Clawpack software
- What are hyperbolic problems?
- Finite volume methods
- Riemann problems and Godunov's method
- Downloading and installing
- Running and plotting

Tuesday:

- Specifying boundary conditions
- Riemann solvers
- Limiters

Wednesday:

- Plotting with the Python modules
- Multidimensional, Adaptive mesh refinement

Options for using Clawpack

1. Use IMA computers platinum, carbon, or tan.
   Install from tar file or Subversion: **Instructions**.
2. Install on your own computer.
   Requires some **prerequisites:** Fortran, Python modules.
3. Use the **VirtualClaw** virtual machine.
4. For some applications, use **EagleClaw**
   (Easy Access Graphical Laboratory for Exploring Conservation Laws)

Also perhaps useful:
Class notes on Python, Fortran, version control, etc.

First order hyperbolic PDE in 1 space dimension

Linear: $q_t + Aq_x = 0, \qquad q(x,t) \in \mathbb{R}^m, \ A \in \mathbb{R}^{m \times m}$

Conservation law: $q_t + f(q)_x = 0, \qquad f : \mathbb{R}^m \to \mathbb{R}^m$ (flux)

Quasilinear form: $q_t + f'(q)q_x = 0$

Hyperbolic if A or $f'(q)$ is diagonalizable with real eigenvalues.

Models wave motion or advective transport.

Eigenvalues are wave speeds.

Note: Second order wave equation $p_{tt} = c^2 p_{xx}$ can be written as a first-order system (acoustics).

Some applications where CLAWPACK has been used

- Aerodynamics, supersonic flows
- Seismic waves, tsunamis, flow on the sphere
- Volcanic flows, dusty gas jets, pyroclastic surges
- Ultrasound, lithotripsy, shock wave therapy
- Plasticity, nonlinear elasticity
- Chemotaxis and pattern formation
- Semiconductor modeling
- Multi-fluids, multi-phase flows, bubbly flow
- Combustion, detonation waves
- Astrophysics: binary stars, planetary nebulae, jets,
- Magnetohydrodynamics, plasmas, relativistic flow
- Numerical relativity — gravitational waves, cosmology

Finite differences vs. finite volumes

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.

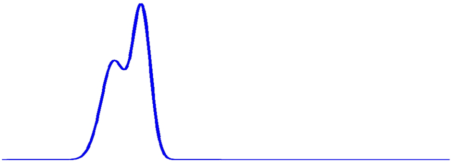
Advection equation

$u =$ constant flow velocity

$q(x,t) =$ tracer concentration, $\quad f(q) = uq$

$\implies \quad q_t + uq_x = 0.$

True solution: $q(x,t) = q(x - ut, 0)$



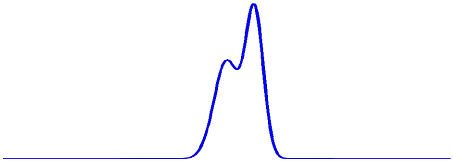
Advection equation

$u =$ constant flow velocity

$q(x,t) =$ tracer concentration, $\quad f(q) = uq$

$\implies \quad q_t + uq_x = 0.$

True solution: $q(x,t) = q(x - ut, 0)$



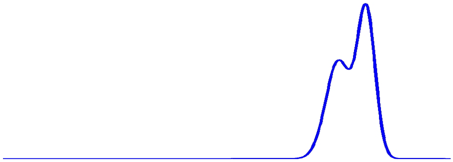
Advection equation

$u =$ constant flow velocity

$q(x,t) =$ tracer concentration, $\quad f(q) = uq$

$\implies \quad q_t + uq_x = 0.$

True solution: $q(x,t) = q(x - ut, 0)$



The Riemann problem

The Riemann problem consists of the hyperbolic equation under study together with initial data of the form

$$q(x,0) = \begin{cases} q_l & \text{if } x < 0 \\ q_r & \text{if } x \geq 0 \end{cases}$$

Piecewise constant with a single jump discontinuity from q_l to q_r.

The Riemann problem is fundamental to understanding

- The mathematical theory of hyperbolic problems,
- Godunov-type finite volume methods

Why? Even for nonlinear systems of conservation laws, the Riemann problem can often be solved for general q_l and q_r, and consists of a set of waves propagating at constant speeds.

The Riemann problem for advection

The Riemann problem for the advection equation $q_t + uq_x = 0$ with

$$q(x,0) = \begin{cases} q_l & \text{if } x < 0 \\ q_r & \text{if } x \ge 0 \end{cases}$$

has solution

$$q(x,t) = q(x-ut,0) = \begin{cases} q_l & \text{if } x < ut \\ q_r & \text{if } x \ge ut \end{cases}$$

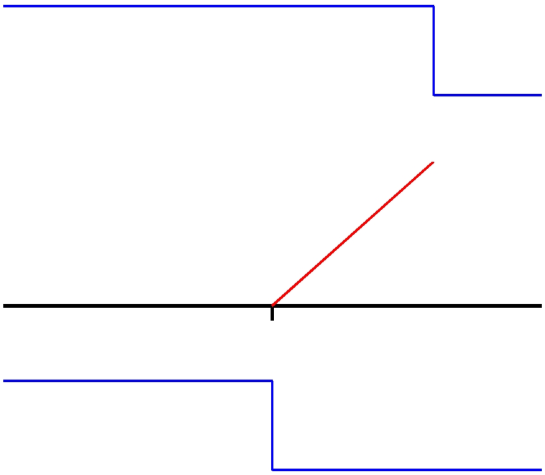
consisting of a single wave of strength $\mathcal{W}^1 = q_r - q_l$ propagating with speed $s^1 = u$.

Riemann solution for advection

$q(x,T)$

x–t plane

$q(x,0)$



Advection examples

- **$CLAW/apps/advection/1d/example1/README.html**
- **Advection in EagleClaw**

Example: Linear acoustics in a 1d tube

$$q = \begin{bmatrix} p \\ u \end{bmatrix} \qquad \begin{aligned} p(x,t) &= \text{pressure perturbation} \\ u(x,t) &= \text{velocity} \end{aligned}$$

Equations:

$$\begin{aligned} p_t + \kappa u_x &= 0 \qquad & \kappa &= \text{bulk modulus} \\ \rho u_t + p_x &= 0 \qquad & \rho &= \text{density} \end{aligned}$$

or

$$\begin{bmatrix} p \\ u \end{bmatrix}_t + \begin{bmatrix} 0 & \kappa \\ 1/\rho & 0 \end{bmatrix} \begin{bmatrix} p \\ u \end{bmatrix}_x = 0.$$

Eigenvalues: $\lambda = \pm c$, where $c = \sqrt{\kappa/\rho}$ = sound speed

Second order form: Can combine equations to obtain

$$p_{tt} = c^2 p_{xx}$$

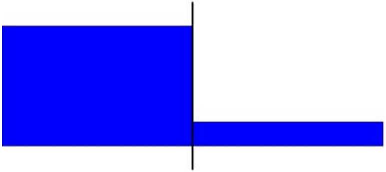
Riemann Problem

Special initial data:

$$q(x,0) = \begin{cases} q_l & \text{if } x < 0 \\ q_r & \text{if } x > 0 \end{cases}$$

Example: Acoustics with bursting diaphram

Pressure:



Acoustic waves propagate with speeds $\pm c$.

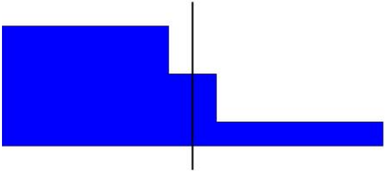
Riemann Problem

Special initial data:

$$q(x,0) = \begin{cases} q_l & \text{if } x < 0 \\ q_r & \text{if } x > 0 \end{cases}$$

Example: Acoustics with bursting diaphram

Pressure:



Acoustic waves propagate with speeds $\pm c$.

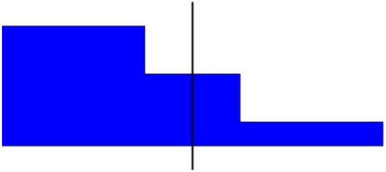
Riemann Problem

Special initial data:

$$q(x,0) = \begin{cases} q_l & \text{if } x < 0 \\ q_r & \text{if } x > 0 \end{cases}$$

Example: Acoustics with bursting diaphram

Pressure:



Acoustic waves propagate with speeds $\pm c$.

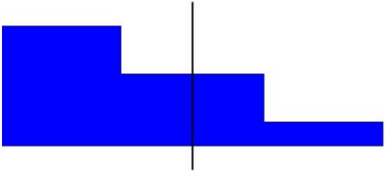
Riemann Problem

Special initial data:

$$q(x,0) = \begin{cases} q_l & \text{if } x < 0 \\ q_r & \text{if } x > 0 \end{cases}$$

Example: Acoustics with bursting diaphram

Pressure:



Acoustic waves propagate with speeds $\pm c$.

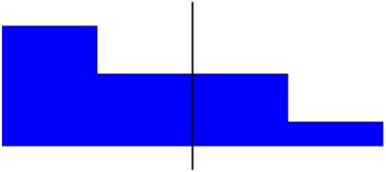
Riemann Problem

Special initial data:

$$q(x,0) = \begin{cases} q_l & \text{if } x < 0 \\ q_r & \text{if } x > 0 \end{cases}$$

Example: Acoustics with bursting diaphram

Pressure:



Acoustic waves propagate with speeds $\pm c$.

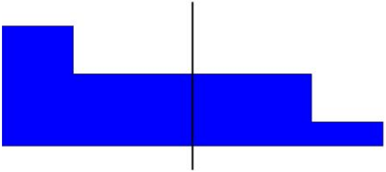
Riemann Problem

Special initial data:

$$q(x,0) = \begin{cases} q_l & \text{if } x < 0 \\ q_r & \text{if } x > 0 \end{cases}$$

Example: Acoustics with bursting diaphram

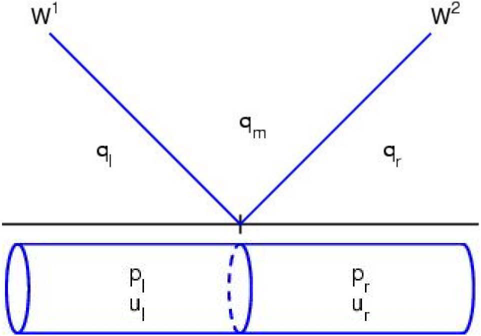
Pressure:



Acoustic waves propagate with speeds $\pm c$.

Riemann Problem for acoustics

Waves propagating in x–t space:



Left-going wave $\mathcal{W}^1 = q_m - q_l$ and
right-going wave $\mathcal{W}^2 = q_r - q_m$ are eigenvectors of A.

Acoustics examples

- **$CLAW/apps/acoustics/1d/example2/README.html**
- **Acoustics in EagleClaw**

CLAWPACK — www.clawpack.org

- Open source, 1d, 2d, (3d in V4.3, soon to be ported)
- Originally f77 with Matlab graphics (V4.3).
- Now use Python for user interface, graphics
- Adaptive mesh refinement, GeoClaw.
- Coming: OpenMP and MPI.

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`

Options for using Clawpack

1. Use IMA computers platinum, carbon, or tan.
   Install from tar file or Subversion: **Instructions**.
2. Install on your own computer.
   Requires some **prerequisites:** Fortran, Python modules.
3. Use the **VirtualClaw** virtual machine.
4. For some applications, use **EagleClaw**
   (Easy Access Graphical Laboratory for Exploring Conservation Laws)

Also perhaps useful:

Class notes on Python, Fortran, version control, etc.