Clawpack Tutorial
Part 3

Randall J. LeVeque
Applied Mathematics
University of Washington

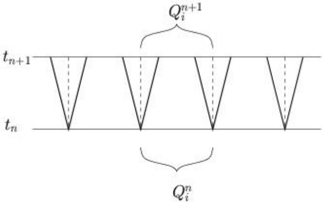
Slides posted at
www.clawpack.org/links/tutorials

Outline

- High-resolution methods
- Limiters

- Python plotting tools
- Specifying plotting parameters
- Options for viewing plots:
  Web pages
  Interactive

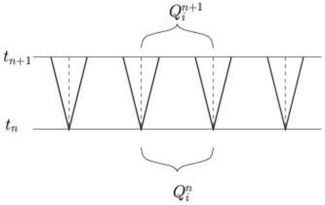
Godunov's Method for $q_t + f(q)_x = 0$



Then either:

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

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

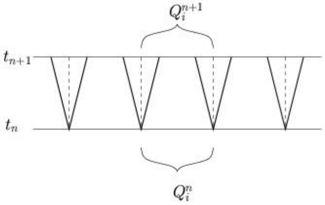


Then either:

1. Compute new cell averages by integrating over cell at t_{n+1},
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 $q_t + f(q)_x = 0$



Then either:

1. Compute new cell averages by integrating over cell at t_{n+1},
2. Compute fluxes at interfaces and flux-difference:

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

3. Update cell averages by contributions from all waves entering 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 $\mathcal{A}^\pm \Delta Q_{i-1/2} = \sum_{i=1}^{m} (s_{i-1/2}^p)^\pm \mathcal{W}_{i-1/2}^p$.

First-order REA Algorithm

1. Reconstruct a piecewise constant function $\tilde{q}^n(x, t_n)$ defined for all x, from the cell averages Q_i^n.

$$\tilde{q}^n(x, t_n) = Q_i^n \quad \text{for all } x \in \mathcal{C}_i.$$

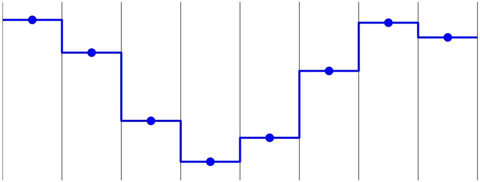
2. Evolve the hyperbolic equation exactly (or approximately) with this initial data to obtain $\tilde{q}^n(x, t_{n+1})$ a time Δt later.

3. Average this function over each grid cell to obtain new cell averages

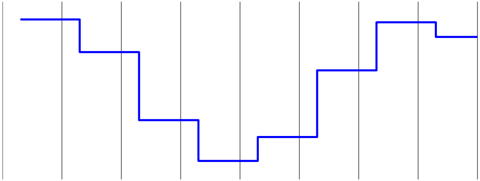
$$Q_i^{n+1} = \frac{1}{\Delta x} \int_{\mathcal{C}_i} \tilde{q}^n(x, t_{n+1})\, dx.$$

First-order REA Algorithm

Cell averages and piecewise constant reconstruction:



After evolution:



Second-order REA Algorithm

1. Reconstruct a piecewise linear function $\tilde{q}^n(x, t_n)$ defined for all x, from the cell averages Q_i^n.

$$\tilde{q}^n(x, t_n) = Q_i^n + \sigma_i^n (x - x_i) \quad \text{for all } x \in \mathcal{C}_i.$$

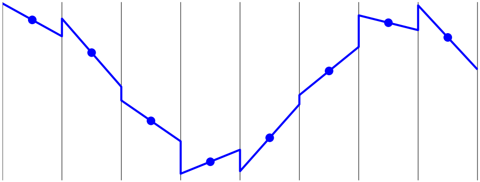
2. Evolve the hyperbolic equation exactly (or approximately) with this initial data to obtain $\tilde{q}^n(x, t_{n+1})$ a time k later.

3. Average this function over each grid cell to obtain new cell averages

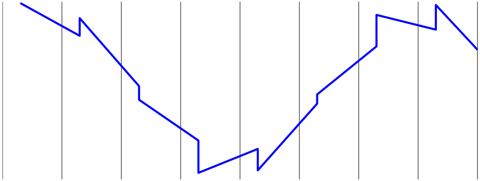
$$Q_i^{n+1} = \frac{1}{\Delta x} \int_{\mathcal{C}_i} \tilde{q}^n(x, t_{n+1})\, dx.$$

Second-order REA Algorithm

Cell averages and piecewise linear reconstruction:



After evolution:



Choice of slopes

$$\tilde{Q}^n(x,t_n) = Q_i^n + \sigma_i^n(x - x_i) \qquad \text{for } x_{i-1/2} \le x < x_{i+1/2}.$$

Applying REA algorithm gives:

$$Q_i^{n+1} = Q_i^n - \frac{u\Delta t}{\Delta x}(Q_i^n - Q_{i-1}^n) - \frac{1}{2}\frac{u\Delta t}{\Delta x}(\Delta x - \bar{u}\Delta t)(\sigma_i^n - \sigma_{i-1}^n)$$

Choice of slopes:

Centered slope: $\sigma_i^n = \dfrac{Q_{i+1}^n - Q_{i-1}^n}{2\Delta x}$ (Fromm)

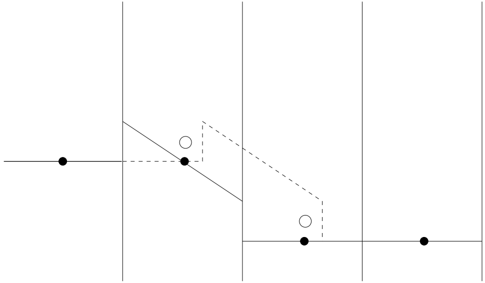
Upwind slope: $\sigma_i^n = \dfrac{Q_i^n - Q_{i-1}^n}{\Delta x}$ (Beam-Warming)

Downwind slope: $\sigma_i^n = \dfrac{Q_{i+1}^n - Q_i^n}{\Delta x}$ (Lax-Wendroff)

Oscillations

Any of these slope choices will give oscillations near discontinuities.

Ex: Lax-Wendroff:



High-resolution methods

Want to use slope where solution is smooth for "second-order" accuracy.

Where solution is not smooth, adding slope corrections gives oscillations.

Limit the slope based on the behavior of the solution.

$$\sigma_i^n = \left(\frac{Q_{i+1}^n - Q_i^n}{\Delta x}\right)\Phi_i^n.$$

$\Phi = 1 \implies$ Lax-Wendroff,

$\Phi = 0 \implies$ upwind.

Minmod slope

$$\text{minmod}(a,b) = \begin{cases} a & \text{if } |a| < |b| \text{ and } ab > 0 \\ b & \text{if } |b| < |a| \text{ and } ab > 0 \\ 0 & \text{if } ab \le 0 \end{cases}$$

Slope:

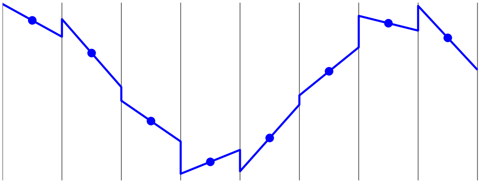
$$\begin{aligned} \sigma_i^n &= \text{minmod}((Q_i^n - Q_{i-1}^n)/\Delta x, \;\; (Q_{i+1}^n - Q_i^n)/\Delta x) \\ &= \left(\frac{Q_{i+1}^n - Q_i^n}{\Delta x}\right)\Phi(\theta_i^n) \end{aligned}$$

where

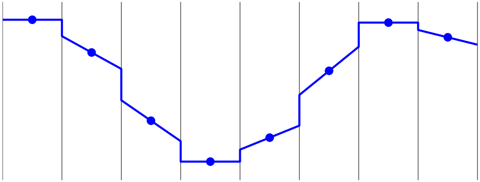
$$\begin{aligned} \theta_i^n &= \frac{Q_i^n - Q_{i-1}^n}{Q_{i+1}^n - Q_i^n} \\ \Phi(\theta) &= \text{minmod}(\theta, 1) \end{aligned}$$

Piecewise linear reconstructions

Lax-Wendroff reconstruction:



Minmod reconstruction:



Some popular limiters

Linear methods:

$$\begin{aligned} \text{upwind}: &\quad \phi(\theta) = 0 \\ \text{Lax-Wendroff}: &\quad \phi(\theta) = 1 \\ \text{Beam-Warming}: &\quad \phi(\theta) = \theta \\ \text{Fromm}: &\quad \phi(\theta) = \frac{1}{2}(1+\theta) \end{aligned}$$

High-resolution limiters:

$$\begin{aligned} \text{minmod}: &\quad \phi(\theta) = \text{minmod}(1,\theta) \\ \text{superbee}: &\quad \phi(\theta) = \max(0,\ \min(1,2\theta),\ \min(2,\theta)) \\ \text{MC}: &\quad \phi(\theta) = \max(0,\ \min((1+\theta)/2,\ 2,\ 2\theta)) \\ \text{van Leer}: &\quad \phi(\theta) = \frac{\theta+|\theta|}{1+|\theta|} \end{aligned}$$

Wave limiters

Let $\mathcal{W}_{i-1/2} = Q_i^n - Q_{i-1}^n$.

Upwind: $Q_i^{n+1} = Q_i^n - \frac{u\Delta t}{\Delta x}\mathcal{W}_{i-1/2}$.

Lax-Wendroff:

$$Q_i^{n+1} = Q_i^n - \frac{u\Delta t}{\Delta x}\mathcal{W}_{i-1/2} - \frac{\Delta t}{\Delta x}(\tilde{F}_{i+1/2} - \tilde{F}_{i-1/2})$$

$$\tilde{F}_{i-1/2} = \frac{1}{2}\left(1 - \left|\frac{u\Delta t}{\Delta x}\right|\right)|u|\mathcal{W}_{i-1/2}$$

High-resolution method:

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

where $\widetilde{\mathcal{W}}_{i-1/2} = \Phi_{i-1/2}\mathcal{W}_{i-1/2}$.

Extensions

These methods extend naturally to:

Linear systems of equations:

Solve Riemann problem to decompose each jump into waves,
Apply same technique to each wave.

Nonlinear problems:

Use approximate Riemann solver to decompose jump,
Apply same technique to each wave.

Multidimensional problems:

Waves propagate normal to interfaces,
Can add in transverse propagation.

Exercise to illustrate limiters

Start with example in **$CLAW/apps/acoustics/1d/example2/**

Modify to use periodic boundary conditions

Modify to go up to time 1, with 12 output times.
(Note: solution at $t = 1$ should agree with data at $t = 0$)

Compare different limiters: `clawdata.mthlim = [k,k]`

- `k = 0`: No limiter (Lax-Wendroff)
- `k = 1`: Minmod limiter
- `k = 2`: Superbee limiter
- `k = 4`: MC limiter

Can also try `clawdata.order = 1` (First order Godunov)

Python plotting tools

Directory `_output` contains files `fort.t000N, fort.q000N` of data at frame N (N'th output time).

`fort.t000N`: Information about this time,
`fort.q000N`: Solution on all grids at this time

There may be many grids at each output time.

Python tools provide a way to specify what plots to produce for each frame:

- One or more figures,
- Each figure has one or more axes,
- Each axes has one or more items,
  (Curve, contour, pcolor, etc.)

`setplot` function for speciying plots

The file `setplot.py` contains a function setplot
Takes an object `plotdata` of class ClawPlotData,
Sets various attributes, and returns the object.

Documentation: **www.clawpack.org/users/setplot.html**

Example: 1 figure with 1 axes showing 1 item:

```
def setplot(plotdata):
    plotfigure = plotdata.new_plotfigure(name,num)
    plotaxes = plotfigure.new_plotaxes(title)
    plotitem = plotaxes.new_plotitem(plot_type)
    # set attributes of these objects
    return plotdata
```

setplot function for speciying plots

Example: plot first component of q as blue curve, red circles.

```
plotfigure = plotdata.new_plotfigure('Q', 1)
plotaxes = plotfigure.new_plotaxes('axes1')

plotitem = plotaxes.new_plotitem('1d_plot')
plotitem.plotvar = 0  # Python indexing!
plotitem.plotstyle = '-'
plotitem.color = 'b'  # or [0,0,1] or '#0000ff'

plotitem = plotaxes.new_plotitem('1d_plot')
# plotitem now points to a new object!
plotitem.plotvar = 0
plotitem.plotstyle = 'ro'
```

Plotting examples and documentation

General plotting information:
www.clawpack.org/users/plotting.html

Use of setplot, possible attributes:
www.clawpack.org/users/setplot.html

Examples:

1d: **www.clawpack.org/users/plotexamples.html**

2d: **www.clawpack.org/users/plotexamples2d.html**

FAQ: **www.clawpack.org/users/plotting_faq.html**

Gallery of applications:
www.clawpack.org/users/apps.html