FYS-GEO 4500 19 Oct 2009

Before we start:
Questions over the reading?

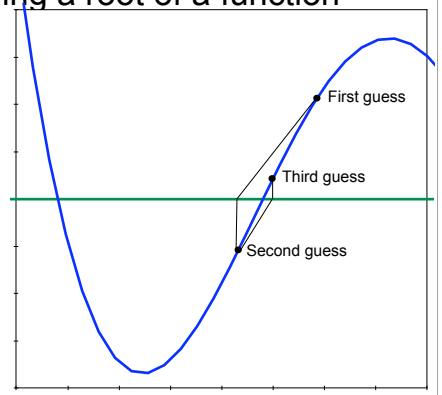
The problem set

Our schedule

	date		Topic	Chapter in LeVeque
1	17 Aug 2009	Monday 13.15-15.00	introduction to conservation laws, Clawpack	1 & 2 & 5
2	24 Aug 2009	Monday 13.15-15.00	the Riemann problem, characteristics	3
3	**28 Aug 2009**	**Friday 13.15-15.00**	finite volume methods for linear systems	4
4	**8 Sep 2009**	**Tuesday 13.15-15.00**	high resolution methods	6
5	21 Sep 2009	Monday 13.15-15.00	boundary conditions, accuracy, variable coeff.	7,8, part of 9
6	**29 Sep 2009**	**Tuesday 13.15-15.00**	nonlinear conservation laws, finite volume methods	11 & 12
7	5 Oct 2009	Monday 13.15-15.00	nonlinear equations & systems	13 & 14
8	12 Oct 2009	Monday 13.15-15.00	finite volume methods for nonlinear systems	15
9	19 Oct 2009	Monday 13.15-15.00	source terms and multidimensions	16,17,18,19
10	26 Oct 2009	Monday 13.15-15.00	multidimensional systems	20 & 21
11	2 Nov 2009	Monday 13.15-15.00	any other topics; project planning	
12	16 Nov 2009	Monday 13.15-15.00	applications: tsunamis, pockmarks, venting, impacts	
13	23 Nov 2009	Monday 13.15-15.00	applications: volcanic jets, pyroclastic flows, lahars	
14	30 Nov 2009	Monday 13.15-15.00	review; progress, problems & projects	
16	7 Dec 2009	Monday 13.15-15.00	FINAL PROJECT REPORTS DUE	

Newton's Method for finding a root of a function

Newton's method aims at finding a root by extrapolating from the function's slope at the point where each successive guess is taken. This is very robust, provided there are no intervening extrema.



Each successive guess is given by

$$x_{n+1} = x_n - \frac{f(x_n)}{k}$$

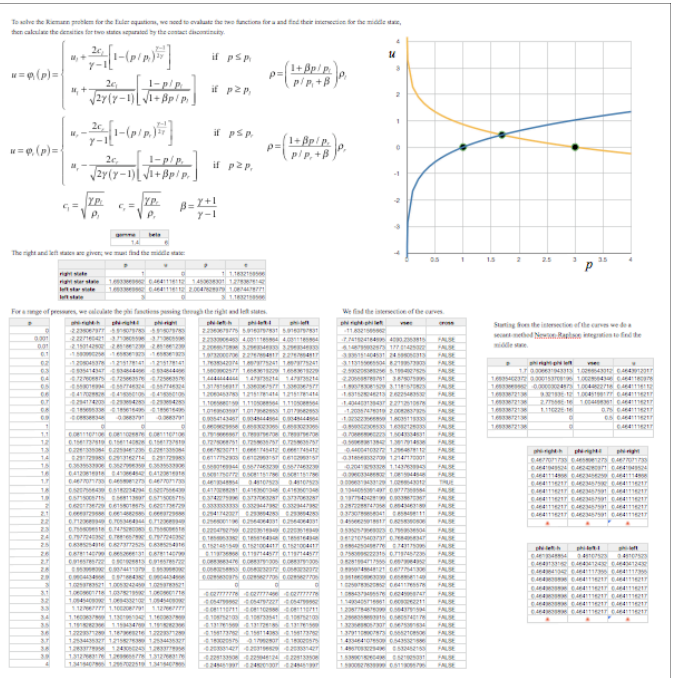
where the slope k is chosen from

(a) $k = f'(x_n)$ *tangent method*

(b) $k = \dfrac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}$ *secant method*; more convenient than the tangent method when the derivative is unavailable or difficult to compute.

(c) $k = \dfrac{f(x_n) - f(x_m)}{x_n - x_m}$ *regula falsi; m* is the most recent guess in which the function has the opposite sign. This method always brackets the root and is therefore most robust, but generally a bit slower.

An exact Riemann solver on a spreadsheet for the Euler equations



How to use an approximate Riemann solver

Since we only need the solution at the cell interface, we determine the state along $x/t = 0$, calling it Q^*. Thus with

$$Q^* = Q_{i-1} + \sum_{p:s^p<0} \mathcal{W}^p_{i-1/2}, \quad F_{i-1/2} = f(Q^*),$$

we form the fluctuations as

$$\mathcal{A}^-\Delta Q = F_{i-1/2} - f(Q_{i-1}), \quad \mathcal{A}^+\Delta Q = f(Q_i) - F_{i-1/2}.$$

You can sometimes use $\mathcal{A}^-\Delta Q = \sum_{p:s^p<0} s^p\mathcal{W}^p$, $\mathcal{A}^+\Delta Q = \sum_{p:s^p>0} s^p\mathcal{W}^p$,

which is conservative if $\mathcal{A}^-\Delta Q + \mathcal{A}^+\Delta Q = f(Q_i) - f(Q_{i-1})$,

as is true for the Roe solver.

The Roe solver

The most widely used approximate Riemann solver is the one developed by Phil Roe at the Royal Aircraft Establishment. The approach is to solve the linearised equation

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

where $\hat{A}$ is an average matrix such that $\hat{A}(q_r, q_l) \equiv \hat{A}_{rl} \approx f'(q_{ave})$.

and $\hat{A}_{rl}(q_r - q_l) = f(q_r) - f(q_l) = s(q_r - q_l)$

The Roe-average matrix can be determined analytically for many important nonlinear systems, including the shallow-water equations and the Euler equations.

Roe's approximate Riemann solver

Roe suggested these constraints for $\hat{A}$:

1. $\hat{A}_{rl}(q_r - q_l) = f(q_r) - f(q_l)$ Cf. Rankine-Hugoniot condition.
2. $\hat{A}$ is diagonalisable with real eigenvalues.
3. $\hat{A}_{rl} \to f'(\overline{q})$ smoothly as $q_l, q_r \to \overline{q}$

A single shock is captured exactly because (1.) is essentially the Rankine-Hugoniot jump condition.

$f(q_r) - f(q_l) = s(q_r - q_l)$ implies that $q_r - q_l$ is an eigenvector of $\hat{A}$.

It is a good approximation for weak waves, or smooth flow.

The wave-propagation algorithm is also conservative since

$$\mathcal{A}^-\Delta Q_{i-1/2} + \mathcal{A}^+\Delta Q_{i-1/2} = \sum_p s^p_{i-1/2}\mathcal{W}^p_{i-1/2} = \hat{A}\sum_p \mathcal{W}^p_{i-1/2}.$$

Roe solver for the Euler equations

The eigensystem of the Euler equations for a polytropic gas is:

$$\lambda^1 = u - c \qquad \lambda^2 = u \qquad \lambda^3 = u + c$$

$$r^1 = \begin{bmatrix} 1 \\ u - c \\ H - uc \end{bmatrix} \quad r^2 = \begin{bmatrix} 1 \\ u \\ \frac{1}{2}u^2 \end{bmatrix} \quad r^3 = \begin{bmatrix} 1 \\ u + c \\ H + uc \end{bmatrix}$$

These need to be evaluated at the Roe-averaged state, so we need the Roe averages for u, H, c. These are:

$$\hat{u} = \frac{\sqrt{\rho_{i-1}}u_{i-1} + \sqrt{\rho_i}u_i}{\sqrt{\rho_{i-1}} + \sqrt{\rho_i}}$$

$$\hat{H} = \frac{\sqrt{\rho_{i-1}}H_{i-1} + \sqrt{\rho_i}H_i}{\sqrt{\rho_{i-1}} + \sqrt{\rho_i}}$$

$$\hat{c} = \sqrt{(\gamma - 1)\left(\hat{H} - \frac{1}{2}\hat{u}^2\right)}$$

Roe solver for the Euler equations

Then the wave decomposition between the left and right states is

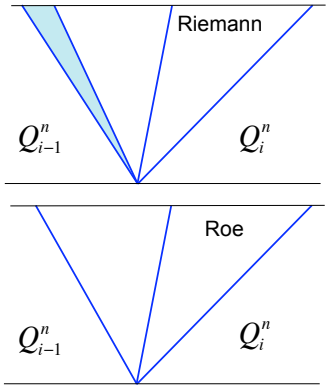
$$\delta \equiv Q_i - Q_{i-1} = \alpha^1 \hat{r}^1 + \alpha^2 \hat{r}^2 + \alpha^3 \hat{r}^3$$

where

$$\alpha^2 = (\gamma - 1)\frac{\left(\hat{H} - \hat{u}^2\right)\delta^1 + \hat{u}\delta^2 - \delta^3}{\hat{c}^2}$$

$$\alpha^3 = \frac{\delta^2 + (\hat{c} - \hat{u})\delta^1 - \hat{c}\alpha^2}{2\hat{c}}$$

$$\alpha^1 = \delta^1 - \alpha^2 - \alpha^3$$



But note that, while the Riemann solution consists of three waves, one of which is a rarefaction fan, the Roe solution only consists of three waves. In most cases this does not matter, since the desired solution at $x/t=0$ will be the same intermediate state. In the case of a transonic rarefaction a modification (in the form of an entropy fix) is necessary.

Roe solver for the Euler equations

Then the wave decomposition between the left and right states is

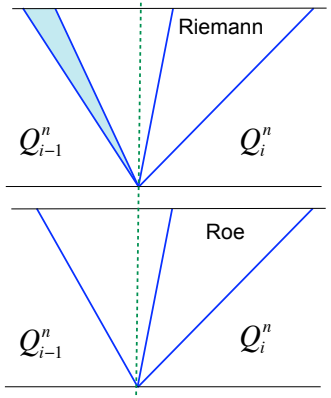
$$\delta \equiv Q_i - Q_{i-1} = \alpha^1 \hat{r}^1 + \alpha^2 \hat{r}^2 + \alpha^3 \hat{r}^3$$

where

$$\alpha^2 = (\gamma - 1)\frac{\left(\hat{H} - \hat{u}^2\right)\delta^1 + \hat{u}\delta^2 - \delta^3}{\hat{c}^2}$$

$$\alpha^3 = \frac{\delta^2 + (\hat{c} - \hat{u})\delta^1 - \hat{c}\alpha^2}{2\hat{c}}$$

$$\alpha^1 = \delta^1 - \alpha^2 - \alpha^3$$



But note that, while the Riemann solution consists of three waves, one of which is a rarefaction fan, the Roe solution only consists of three waves. In most cases this does not matter, since the desired solution at $x/t=0$ will be the same intermediate state. In the case of a transonic rarefaction a modification (in the form of an entropy fix) is necessary.

FYS-GEO 4500 Galen Gisler, Physics of Geological Processes, University of Oslo Autumn 2009
Monday, 26 October 2009

Entropy fix for transonic rarefactions

Suppose there is a transonic rarefaction in the k wave:

$$\lambda_l^k < 0 < \lambda_r^k, \quad q_l^k = Q_{i-1} + \sum_{p=1}^{k-1} \mathcal{W}^p, \quad q_r^k = q_l^k + \mathcal{W}^k$$

The method proposed by Harten and Hyman, modified slightly by Leveque, and implemented in Clawpack, is the following. Define

$$\beta = \frac{\lambda_k^r - \hat{\lambda}_k}{\lambda_k^r - \lambda_k^l}$$

where $\hat{\lambda}_k$ is the Roe-averaged eigenvalue for this wave. Then in computing the fluctuations

$$\mathcal{A}^- \Delta Q_{i-1/2} = \sum_p \left(s^p_{i-1/2}\right)^- \mathcal{W}^p_{i-1/2}, \quad \mathcal{A}^+ \Delta Q_{i-1/2} = \sum_p \left(s^p_{i-1/2}\right)^+ \mathcal{W}^p_{i-1/2}$$

for the speed of the k wave use

$$\left(\hat{\lambda}^k\right)^- = \beta\lambda_l^k, \quad \left(\hat{\lambda}^k\right)^+ = (1-\beta)\lambda_r^k$$

FYS-GEO 4500 Galen Gisler, Physics of Geological Processes, University of Oslo Autumn 2009
Monday, 26 October 2009

Ch 16: Nonclassical hyperbolic problems

Chapter 16 covers, not in very much detail, some situations that may be of interest to some of you:

- flow in porous media
- nearly singular equations
- phase changes, van der Waals gases
- nonconservative transport

I recommend skimming this chapter so you'll know where to look for this information if you need it later.

FYS-GEO 4500 Galen Gisler, Physics of Geological Processes, University of Oslo Autumn 2009
Monday, 26 October 2009

Source Terms, etc. (Chapter 17 in Leveque)

Source terms

Many of the situations we will want to study, especially in geophysics, are conservation laws with source terms:

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

Of course this equation arises from the more fundamental integral form:

$$\frac{\partial}{\partial t}\int_{x_1}^{x_2} q(x,t)\,dx = f\left(q\left(x_1,t\right)\right) - f\left(q\left(x_2,t\right)\right) + \int_{x_1}^{x_2} \psi\left(q\left(x,t\right)\right)dx$$

Examples:
- external forces, for example gravity
- reacting flow (combustion, dissolution, exsolution)
- conductive or radiative heat transfer
- drag, viscosity
- varying depth in shallow-water equations
- varying pipe shape in Euler equations
- systems with symmetries (geometrical source terms, see section 18.9)

A given system may have more than one of these sources!

Fractional-Step Methods

In the system $q_t + f(q)_x = \psi(q)$:

If the homogeneous part $q_t + f(q)_x = 0$ is hyperbolic, and the source term part can be expressed as $q_t = \psi\left(q(x,t),x\right)$ (that is, without derivatives of q), then it is possible to alternate between solving the homogeneous hyperbolic equation and the source term equation.

Using the example of an advection-reaction equation, Leveque shows in some detail how fractional-step and “unsplit” methods relate to one another.

Fractional step methods are easier to implement in code, they are generally faster than unsplit methods, they do nearly as well, and are readily extended to high resolution.

However, there is a *splitting error* one must be aware of.

Fractional Step Methods

The system $q_t + f(q)_x = \psi(q)$ is split into two parts:

A-part: Use a high-resolution method to solve $q_t + f(q)_x = 0$

B-part: Use a high-order method to solve $q_t = \psi\left(q\right)$

There are two popular ways of doing this:

Godunov splitting: (first-order accurate at best) `method(5)=1`
- full time step Δt on the A-part
- full time step Δt on the B-part

Strang splitting: (second-order accurate at best) `method(5)=2`
- time step $\Delta t / 2$ on the B-part
- time step Δt on the A-part
- time step $\Delta t / 2$ on the B-part

Although the order of accuracy suffers, Godunov splitting is often preferred because it is usually faster, depending on the complexity of the source term.

Fractional Step Methods

The system $q_t + f(q)_x = \psi(q)$ is split into two parts:

A-part: Use a high-resolution method to solve $q_t + f(q)_x = 0$

B-part: Use a high-order method to solve $q_t = \psi(q)$

in Clawpack

There are two popular ways of doing this:

Godunov splitting: (first-order accurate at best) `method(5)=1`
- full time step Δt on the A-part
- full time step Δt on the B-part

Strang splitting: (second-order accurate at best) `method(5)=2`
- time step $\Delta t / 2$ on the B-part
- time step Δt on the A-part
- time step $\Delta t / 2$ on the B-part

Although the order of accuracy suffers, Godunov splitting is often preferred because it is usually faster, depending on the complexity of the source term.

Operator splitting for a general PDE: how the splitting error arises

A general form for a linear partial differential equation is

$$q_t = (\mathcal{A} + \mathcal{B})q.$$

(For a conservation law in quasilinear form, $\mathcal{A} = f'(q)\partial_x$; $\mathcal{B} = \psi(\,)$.)

Further time derivatives can be computed by

$$q_{tt} = (\mathcal{A} + \mathcal{B})q_t = (\mathcal{A} + \mathcal{B})^2 q,$$

and we can form the Taylor series expansion

$$q(x,\Delta t) = q(x,0) + \Delta t(\mathcal{A} + \mathcal{B})q(x,0) + \tfrac{1}{2}\Delta t^2(\mathcal{A} + \mathcal{B})^2 q(x,0) + \cdots$$

which we can write in short-hand (*solution operator form*) as

$$q(x,\Delta t) = e^{\Delta t(\mathcal{A}+\mathcal{B})}q(x,0).$$

Using *operator splitting* we compute

$$\hat{q}(x,\Delta t) = e^{\Delta t\mathcal{B}}e^{\Delta t\mathcal{A}}q(x,0).$$

The difference between these two, in second order, is

$$q(x,\Delta t) - \hat{q}(x,\Delta t) = \tfrac{1}{2}\Delta t^2(\mathcal{A}\mathcal{B} - \mathcal{B}\mathcal{A})q(x,0) + \mathcal{O}(\Delta t^3)$$

which is zero *only when the differential operators commute*.

Multidimensional problems can also use a kind of operator splitting

The two-dimensional conservation law $q_t + f(q)_x + g(q)_y = 0$

is similar to the one-dimensional conservation law with a source term, and can be solved in a similar way. In Clawpack (see $CLAW/2d/lib/claw2.f):

`method(3)=-1` dimensional splitting, Godunov style
`method(3)=-2` dimensional splitting, Strang style

This is the easiest way to extend good 1-D methods to 2-D and 3-D, and it is usually very effective and efficient.

Other (unsplit) methods are also available in Clawpack (we'll talk about these later):

`method(3)= 0` no transverse propagation, only normal waves
`method(3)=+1` transverse propagation without correction waves
`method(3)=+2` transverse propagation with correction waves

The dimensionally split Godunov method

The two steps of a dimensionally-split Godunov method are:

$$Q_{ij}^* = Q_{ij}^n - \frac{\Delta t}{\Delta x}\left(\mathcal{A}^+\Delta Q_{i-1/2,j} + \mathcal{A}^-\Delta Q_{i+1/2,j}\right)$$

$$Q_{ij}^{n+1} = Q_{ij}^* - \frac{\Delta t}{\Delta y}\left(\mathcal{B}^+\Delta Q_{i,j-1/2}^* + \mathcal{B}^-\Delta Q_{i,j+1/2}^*\right)$$

For the linear system, $q_t + Aq_x + Bq_y = 0$, the fluctuations are

$$\mathcal{A}^\pm\Delta Q_{i-1/2,j} = A^\pm\left(Q_{ij} - Q_{i-1,j}\right)$$
$$\mathcal{B}^\pm\Delta Q_{i-1/2,j} = B^\pm\left(Q_{ij} - Q_{i,j-1}\right)$$

where $A^\pm = R^x\left(\Lambda^x\right)^\pm\left(R^x\right)^{-1}$ and $B^\pm = R^y\left(\Lambda^y\right)^\pm\left(R^y\right)^{-1}$.

Multidimensional Hyperbolic Problems (Chapter 18 in Leveque)

The world has more than one dimension!

A conservation law in two dimensions:

$$q_t + f(q)_x + g(q)_y = 0$$

and in three dimensions:

$$q_t + f(q)_x + g(q)_y + h(q)_z = 0$$

where $f(q)$, $g(q)$, and $h(q)$ are the fluxes in the x, y, and z directions respectively.

More generally we write:

$$q_t + \vec{\nabla} \cdot \vec{f}(q) = 0$$

where $\vec{f}(q)$ is a vector function representing the flux of q, and the divergence operator is defined as

$$\text{for } \vec{f} = [f, g, h], \quad \vec{\nabla} \cdot \vec{f} = \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} + \frac{\partial h}{\partial z} = f_x + g_y + h_z$$

A word about notation

In general we will refer to spatial vectors and their components as:

position	$\vec{x} = [x, y, z]$
velocity	$\vec{u} = [u(x,y,z,t), v(x,y,z,t), w(x,y,z,t)]$
flux	$\vec{f}(q) = [f(q), g(q), h(q)]$
vector of matrices	$\vec{A} = [A, B, C]$
unit normal vector	$\vec{n} = [n^x, n^y, n^z]$

Hyperbolicity

The linear system in one dimension, $q_t + Aq_x = 0$, is hyperbolic if the matrix A is diagonalisable, with real eigenvalues.

In two dimensions, we need a stronger condition. For the system

$$q_t + Aq_x + Bq_y = 0$$

to be hyperbolic, not only must the matrices A and B be diagonalisable, with real eigenvalues, but so must every projection of these matrices in all spatial directions.

Defining a unit vector $\vec{n} = (n^x, n^y)$, we define

$$\breve{A} = n^x A + n^y B$$

and require that this combination be diagonalisable, with real eigenvalues for any choice of $\vec{n}$. Problem 18.3 is an example in which each matrix is separately hyperbolic, but the combination is not! **For extra credit, do this.**

This requirement is easily extended to a three-dimensional system.

Coupling among dimensions

We cannot in general decouple a linear multidimensional system into separate advection equations as we could in the one-dimensional case.

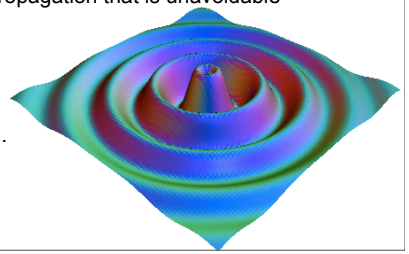
We can diagonalise each matrix separately:

$$A = R^x \Lambda^x (R^x)^{-1}, \quad B = R^y \Lambda^y (R^y)^{-1}$$

but unless $BA=AB$, performing these operations does not produce decoupling.

The point is that hyperbolic equations in two or more dimensions produce waves that can travel in *any* direction, not just in the x or y directions. Thus there is coupling between the x and the y propagation that is unavoidable except under special circumstances.

Dimensional splitting may not adequately deal with the coupling between dimensions. Effectively, the order of accuracy is reduced.



Compressible barotropic flow in 2 dimensions

With the velocity vector defined as $\vec{u} = [u, v]$

the equations of compressible flow in 2 dimensions are:

$$\rho_t + (\rho u)_x + (\rho v)_y = 0$$
$$(\rho u)_t + \left(\rho u^2 + p\right)_x + (\rho uv)_y = 0$$
$$(\rho v)_t + (\rho uv)_x + \left(\rho v^2 + p\right)_y = 0$$

We can write these as $q_t + f(q)_x + g(q)_y = 0$

by putting

$$q = \begin{bmatrix} \rho \\ \rho u \\ \rho v \end{bmatrix}, f(q) = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \end{bmatrix}, g(q) = \begin{bmatrix} \rho u \\ \rho uv \\ \rho v^2 + p \end{bmatrix}$$

The Jacobians for 2-d compressible flow

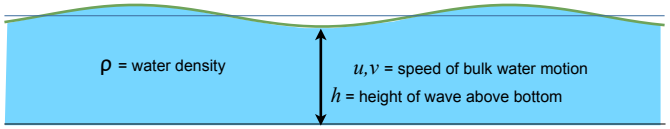
There are now two Jacobian matrices, for the quasilinear form

$$q_t + f'(q)q_x + g'(q)q_y = 0$$

with the barotropic equation of state $p = P(\rho)$ these are:

$$f'(q) = \begin{bmatrix} 0 & 1 & 0 \\ -u^2 + P'(\rho) & 2u & 0 \\ -uv & v & u \end{bmatrix}, \; g'(q) = \begin{bmatrix} 0 & 0 & 1 \\ -uv & v & u \\ -v^2 + P'(\rho) & 0 & 2v \end{bmatrix}$$

Shallow-water equations in 2 dimensions



In two dimensions the shallow-water equations are:

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

and the flux Jacobians are (similar to barotropic compressible flow):

$$f'(q) = \begin{bmatrix} 0 & 1 & 0 \\ -u^2 + gh & 2u & 0 \\ -uv & v & u \end{bmatrix}, \; g'(q) = \begin{bmatrix} 0 & 0 & 1 \\ -uv & v & u \\ -v^2 + gh & 0 & 2v \end{bmatrix}$$

Eigensystem for shallow-water equations

The Jacobian matrix $f'(q)$ has these eigenvalues and eigenvectors:

$$\lambda^{x1} = u - c, \qquad \lambda^{x2} = u, \qquad \lambda^{x3} = u + c, \quad c = \sqrt{gh}$$

$$r^{x1} = \begin{bmatrix} 1 \\ u-c \\ v \end{bmatrix}, r^{x2} = \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}, r^{x3} = \begin{bmatrix} 1 \\ u+c \\ v \end{bmatrix}$$

and the other Jacobian matrix $g'(q)$ has similar pattern, reversing u and v:

$$\lambda^{y1} = v - c, \qquad \lambda^{y2} = v, \qquad \lambda^{y3} = v + c$$

$$r^{y1} = \begin{bmatrix} 1 \\ u \\ v-c \end{bmatrix}, r^{y2} = \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}, r^{y3} = \begin{bmatrix} 1 \\ u \\ v+c \end{bmatrix}$$

Notice the eigenvectors are different, but in general we have two nonlinear fields with the eigenvalues $\vec{n}\cdot\vec{u} \pm c$ and the linearly degenerate field $\vec{n}\cdot\vec{u}$.

The Euler equations in 3 dimensions

In 2 dimensions, we have 4 equations, and in 3 dimensions 5 equations:

$$q_t + f(q)_x + g(q)_y + h(q)_z = 0$$

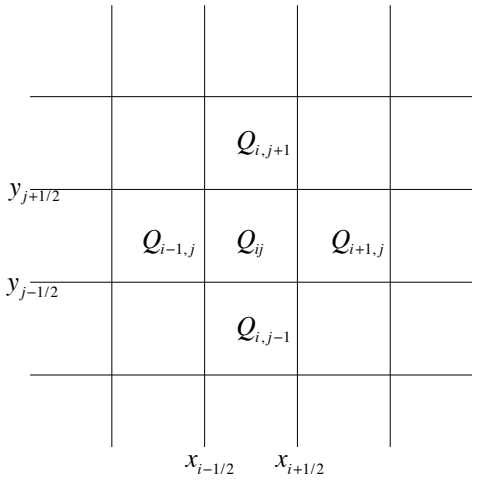
$$q = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{bmatrix}, f(q) = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ (E+p)u \end{bmatrix}, g(q) = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ (E+p)v \end{bmatrix}, h(q) = \begin{bmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ (E+p)w \end{bmatrix}$$

The energy is $E = \rho e + \frac{1}{2}\rho\vec{u}\cdot\vec{u} = \rho e + \frac{1}{2}\rho\left(u^2 + v^2 + w^2\right)$ and this system still needs to be supplemented with an equation of state $e = e(p,\rho)$. Again there will be eigenvalues of the form $\vec{n}\cdot\vec{u}$ and $\vec{n}\cdot\vec{u} \pm c$, where $c = \sqrt{\frac{dp}{d\rho}}$.

In any direction, there are two nonlinear acoustic fields and three nonlinearly degenerate fields.

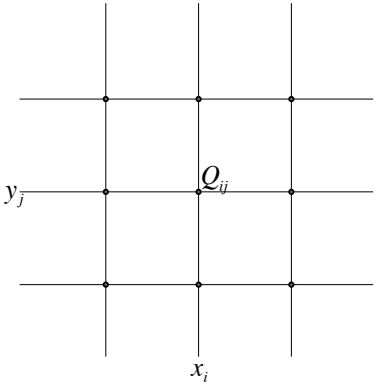
Multidimensional Numerical Methods (Chapter 19 in Leveque)

Finite Volume vs Finite Difference



The values are considered to be averages within cells:

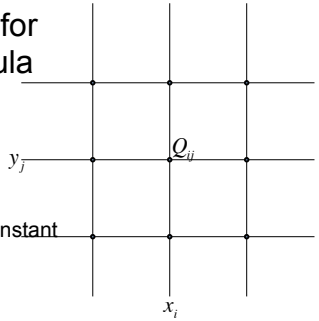
$$Q_{ij}^n \approx \frac{1}{\Delta x \Delta y}\int_{y_{j-1/2}}^{y_{j+1/2}}\int_{x_{i-1/2}}^{x_{i+1/2}} q(x,y,t_n)\,dx\,dy$$



The values are considered to be evaluated on a grid:

$$Q_{ij}^n \approx q(x_i, y_j, t_n)$$

The finite-difference form is useful for developing the Lax-Wendroff formula via a Taylor-series expansion



We consider first the linear system, with A and B constant (though noncommutative) matrices:

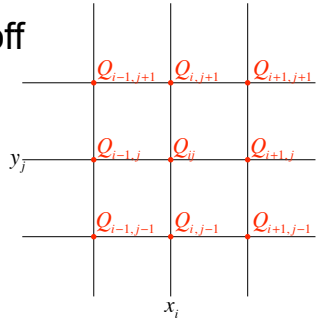
$$q_t + Aq_x + Bq_y = 0$$

We make a Taylor-series expansion of q at the point (x_i, y_j) at time $t_n + \Delta t$ in terms of its value at time t_n:

$$q(x_i, y_j, t_n + \Delta t) = q + \Delta t q_t + \frac{1}{2}\Delta t^2 q_{tt} + \ldots$$

$$= q - \Delta t(Aq_x + Bq_y) + \frac{1}{2}\Delta t^2 (A^2 q_{xx} + ABq_{yx} + BAq_{xy} + B^2 q_{yy}) + \ldots$$

Two-dimensional Lax-Wendroff



The pure double derivatives are approximated as in one dimension:

$$q_{xx} \approx \left(\frac{1}{\Delta x}\right)^2 \left(Q^n_{i-1,j} - 2Q^n_{i,j} + Q^n_{i+1,j}\right)$$

$$q_{yy} \approx \left(\frac{1}{\Delta y}\right)^2 \left(Q^n_{i,j-1} - 2Q^n_{i,j} + Q^n_{i,j+1}\right)$$

and there are also cross-double-derivative terms:

$$q_{xy} = q_{yx} \approx \frac{1}{4\Delta x \Delta y}\left[\left(Q^n_{i+1,j+1} - Q^n_{i-1,j+1}\right) - \left(Q^n_{i+1,j-1} - Q^n_{i-1,j-1}\right)\right]$$

giving the method:

$$Q^{n+1}_{ij} = Q^n_{ij} - \frac{\Delta t}{2\Delta x} A\left(Q^n_{i+1,j} - Q^n_{i-1,j}\right) - \frac{\Delta t}{2\Delta y} B\left(Q^n_{i,j+1} - Q^n_{i,j-1}\right)$$

$$+ \frac{\Delta t^2}{2\Delta x^2} A^2 \left(Q^n_{i+1,j} - 2Q^n_{ij} + Q^n_{i-1,j}\right) + \frac{\Delta t^2}{2\Delta y^2} B^2 \left(Q^n_{i,j+1} - 2Q^n_{ij} + Q^n_{i,j-1}\right)$$

$$+ \frac{\Delta t^2}{8\Delta x \Delta y}(AB + BA)\left[\left(Q^n_{i+1,j+1} - Q^n_{i-1,j+1}\right) - \left(Q^n_{i+1,j-1} - Q^n_{i-1,j-1}\right)\right]$$

Again, Lax-Wendroff is a good starting point, but we need to do better…

In a *finite volume* approach, using upwind biasing and flux limiting we can achieve second-order accuracy and high-resolution in multi-dimensions as we did in one dimension.

Note: the Riemann problem at a cell edge is essentially one-dimensional; we can do the problem as before, and bring in multidimensional information to improve the solution.

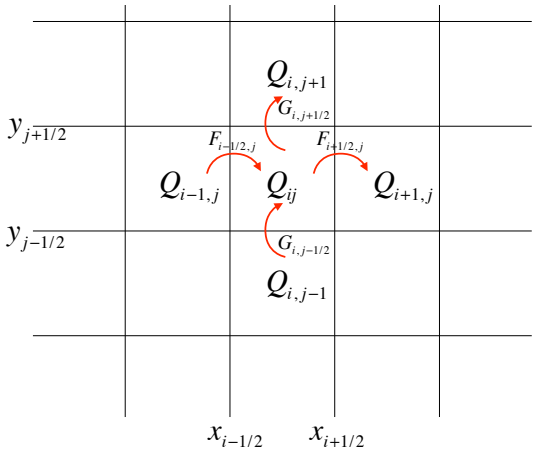
You can

use a full flux-differencing (dimensionally unsplit) approach (better) or

split the problem into a sequence of one-dimensional problems (easier, but you have to take care of the corners)

For conservation laws, finite volume methods are natural

We update the conserved quantity by keeping track of the fluxes into and out of each cell (and sources and sinks, as relevant)



The fully discrete flux differencing method to update Q for the next time step is:

$$Q^{n+1}_{ij} \approx Q^n_{ij} - \frac{\Delta t}{\Delta x}\left(F^n_{i+1/2,j} - F^n_{i-1/2,j}\right) - \frac{\Delta t}{\Delta y}\left(G^n_{i,j+1/2} - G^n_{i,j-1/2}\right)$$

The fluxes are found by integrating along the edges

$$F^n_{i-1/2,j} \approx \frac{1}{\Delta x \Delta y}\int_{t_n}^{t_{n+1}}\int_{y_{j-1/2}}^{y_{j+1/2}} f(q(x_{i-1/2},y,t))\,dy\,dt$$

$$G^n_{i,j-1/2} \approx \frac{1}{\Delta x \Delta y}\int_{t_n}^{t_{n+1}}\int_{x_{i-1/2}}^{x_{i+1/2}} g(q(x,y_{j-1/2},t))\,dx\,dt$$

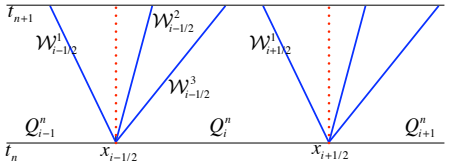
With a Taylor expansion we approximate these, to second-order in time, as:

$$F_{i-1/2,j} \approx Aq(x_{i-1/2},y_j,t_n) - \frac{\Delta t}{2}A^2 q_x(x_{i-1/2},y_j,t_n) - \frac{\Delta t}{2}ABq_y(x_{i-1/2},y_j,t_n)$$

$$G_{i,j-1/2} \approx Bq(x_i,y_{j-1/2},t_n) - \frac{\Delta t}{2}B^2 q_y(x_i,y_{j-1/2},t_n) - \frac{\Delta t}{2}BAq_x(x_i,y_{j-1/2},t_n)$$

These fluxes can be used to re-interpret the Lax-Wendroff formula as a finite volume method.

The Godunov method is simpler



Here we simply find the Riemann solution that propagates with zero speed (straight up the time axis) and evaluate the flux functions at these values:

$$F_{i-1/2,j} = f(Q^{\downarrow}_{i-1/2,j})$$

$$G_{i,j-1/2} = f(Q^{\downarrow}_{i,j-1/2})$$

$Q^{\downarrow}_{i-1/2,j}$ is obtained by solving the Riemann problem for $q_t + f(q)_x = 0$ and

$Q^{\downarrow}_{i,j-1/2}$ is obtained by solving the Riemann problem for $q_t + g(q)_y = 0$.

We can adopt the fluctuation form

$$Q^{n+1}_{ij} = Q^n_{ij} - \frac{\Delta t}{\Delta x}\left(\mathcal{A}^+\Delta Q_{i-1/2,j} + \mathcal{A}^-\Delta Q_{i+1/2,j}\right) - \frac{\Delta t}{\Delta y}\left(\mathcal{B}^+\Delta Q_{i,j-1/2} + \mathcal{B}^-\Delta Q_{i,j+1/2}\right)$$
$$- \frac{\Delta t}{\Delta x}\left(\tilde{F}_{i+1/2,j} - \tilde{F}_{i-1/2,j}\right) - \frac{\Delta t}{\Delta y}\left(\tilde{G}_{i,j+1/2} - \tilde{G}_{i,j-1/2}\right)$$

Here the high-resolution corrections in the second line can be omitted for the pure Godunov method, or they can be included with appropriate flux limiters for high-resolution techniques.

Dimensional Splitting - Godunov Splitting

We can split a multidimensional problem into a sequence of one-dimensional steps, for example for the two-dimensional linear problem:

$$q_t + Aq_x + Bq_y = 0 \quad \begin{cases} x \text{ sweeps: } & q_t + Aq_x = 0 \\ y \text{ sweeps: } & q_t + Bq_y = 0 \end{cases}$$

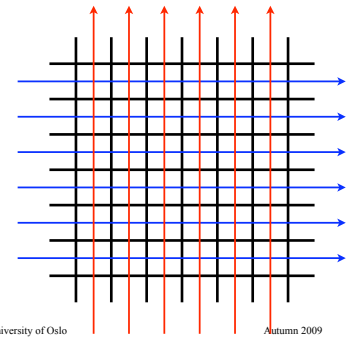
In the x sweeps, we march along in i, keeping j fixed, and update

$$Q^*_{ij} = Q^n_{ij} - \frac{\Delta t}{\Delta x}\left(F^n_{i+1/2,j} - F^n_{i,j-1/2}\right)$$

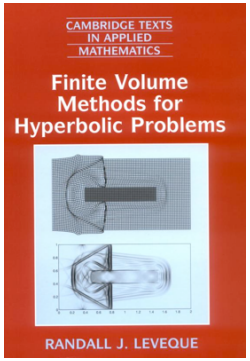
In the y sweeps, we march along in j, keeping i fixed, and update

$$Q^{n+1}_{ij} = Q^*_{ij} - \frac{\Delta t}{\Delta y}\left(G^*_{i,j+1/2} - G^*_{i,j-1/2}\right)$$

Alternating the order of the sweeps every time step is equivalent to Strang splitting



Assignment for next time



Skim Chapter 16, read Chapter 17 through 17.5 (more if you're interested) and read all of Chapters 18 and 19 (they are short!).

Write a one-paragraph ***draft*** description of a project you would like to do with Clawpack, including a description of the physical circumstances, and how you might implement it in code. Indicate the dimensionality of the problem, the equations you would like to solve (naming the equation set will be sufficient for this draft), whether there are source terms, what boundary conditions, etc.

Let me have this by Monday 26 October. Even better, come talk to me about it this week.

The purpose of this preliminary exercise is to help me prepare to help you with the projects, and to know what to focus on in the next few lectures. I expect actual project design and execution to start in November, with completion by 7 December.

Next: Multidimensional scalar equations and systems (Ch 20 & 21)