The Riemann problem

Very important for hyperbolic cons. laws. Key element for solving the equation(s). If you can solve it, and show the solution behaves properly, you are half there. Also key building block for many schemes for computing in problems with shocks

What it is: Solve equations with piecewise initial data

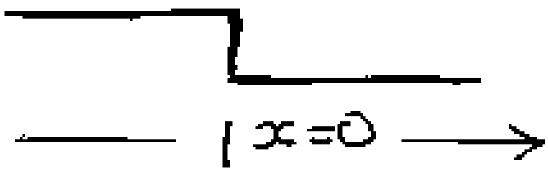
E.g. Solve $\rho_t + q_x = 0$, $q = Q(\rho)$

with
$$\left.\begin{array}{ll} \rho = \rho_1 & \text{for } x<0 \\ \rho = \rho_2 & \text{for } x>0 \end{array}\right\} \text{ at } t=0$$

where ρ_1 and ρ_2 are arbitrary const.

works for hyperbolic systems of conservation laws† as well

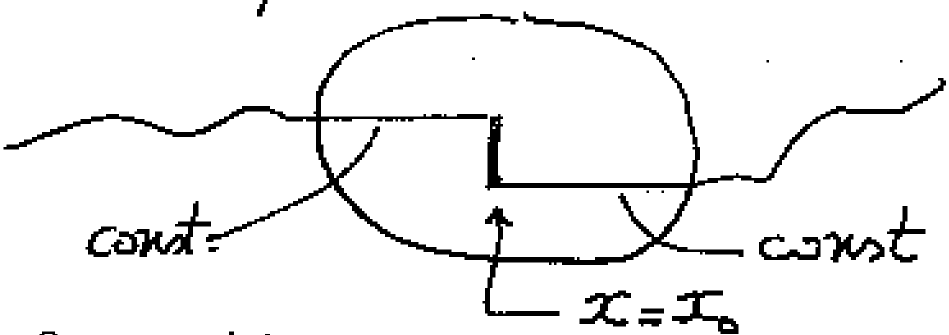
$$\vec{\rho}_t + \vec{q}(\vec{\rho})_x = 0 \text{ with } \begin{array}{ll} \vec{\rho} = \vec{\rho}_1 & x<0,\ t=0 \\ \vec{\rho} = \vec{\rho}_2 & x>0,\ t=0 \end{array}$$



ρ_j at $t=0$

<u>Why does it matter</u> Because for hyperbolic systems information travels at <u>finite speed</u>

So imagine data like this



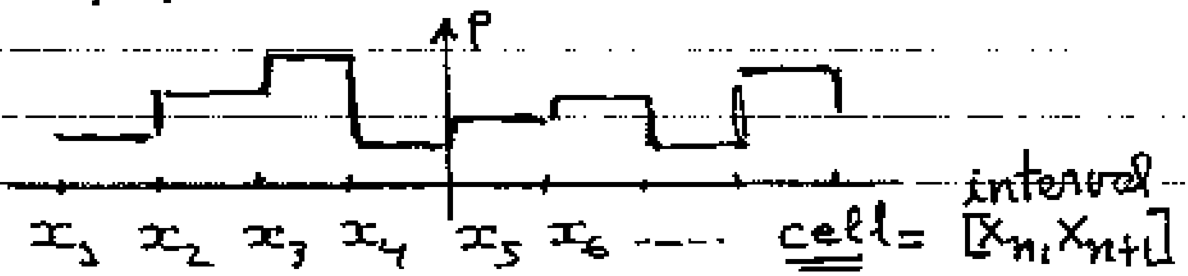
Then the solution to the Riemann problem

† Not yet defined. We will see this soon

will be the solution to the problem near x_0 for a short while!

Godunov's method

Take initial data and approximate it by piecewise constant data

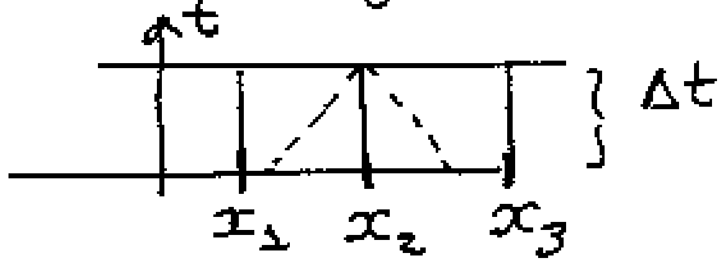


In each cell keep only the average value of solution. Because it is a conservation law, to update the average you only need to know the flux q through the cell boundary, which follows from knowing the solution to the Riemann problem

<u>So proceed as follows</u>

① Replace data by piecewise data.

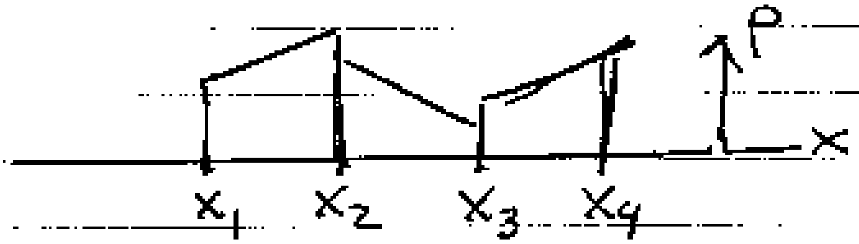
② Use Riemann problem to update value in each cell by computing flux through ends over time Δt (Δt limited by CFL)



Δt so characteristics from either x_1 or x_3 have not arrived at x_2!

③ Recompute averages and proceed.

For higher accuracy one can use higher order interpolants in each cell; e.g. linear



Then flux given by "generalized" Riemann problem. However we need the solution to it only approx. and once sln. standard problem known, this is possible.

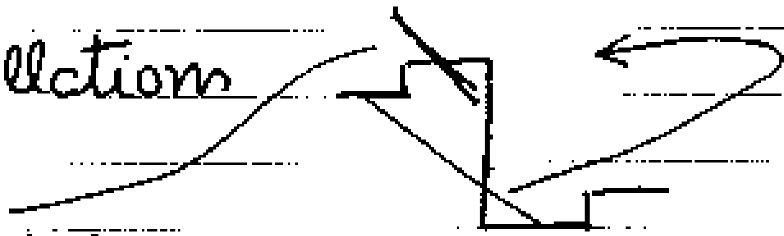
Alternative "Instantaneous" flow for [A] is known. Can combine this with Runge-Kutta type ode solver (that uses only instantaneous flows

Once you get slopes, in each cell sln.

looks like

Note: at x_n sln. still discontinuous

Limiters Want to avoid spurious oscillations



note large overshoot

$\therefore$ need "slope limiters", etc.

Ideas can be extended to higher order. The RA is the key step where there has been a lot of research

1. Godunov/van Leer schemes
2. ENO (Essentially non-oscillatory)
3. WENO (Weighted ENO)

etc etc

at various times) to get higher order.

Note: note that knowing the instantaneous flow allows the pde, for solutions like [A] to be written as

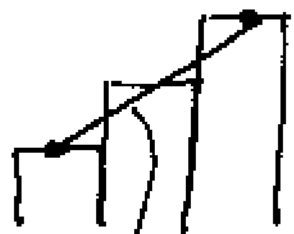
$$\frac{d}{dt}\vec{\rho}_a = \vec{F}(\vec{\rho}_a)$$

vector of flow averages.

Need way to go from $\vec{\rho}_a$ to detailed $\vec{\rho}$ with slopes, etc. $\left\{\begin{array}{l}\text{Reconstruction}\\ \text{Algorithm (RA)}\end{array}\right.$

RA : how do you get, say, slopes from just knowing averages? Example:

Use estimators based in finite differences



slope in middle cell

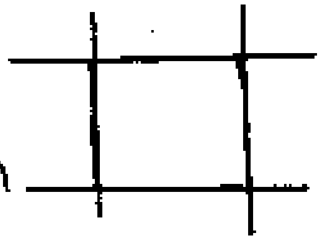
Examples

Can use this to give meaning to

$$u_t + (u^2)_x = a(t)\,\delta(x-\sigma(t)),$$

as long as $\sigma=\sigma(t)$ has derivative

Multi-D Multi-D Riemann problem has no known sln., even for "simple" problems. Thus



Solve 1-D Riemann for each side to compute flow and ignore corners. If Δt small enough this "should" be OK.

Theory still "open".