原行星盘与行星形成暑期学校, 北京, 2024/07/23

Finite Volume Method for (Magneto)-Hydrodynamics

Xuening Bai (白雪宁)

Pinghui Huang (黄平辉)

Institute for Advanced Study (IASTU) &

Department of Astronomy (DoA)

Tsinghua University

Types of PDEs

Hydro and MHD equations are a system of partial differential equations (PDEs).

There are in general 3 types of PDEs. For a 2nd order PDE of the form

$$a\partial^2_{xx}u + b\partial^2_{xy}u + c\partial^2_{yy}u + d\partial_x u + e\partial_y u + fu = g$$

it can be categorized based on the discriminant:

$$b^2 - 4ac \begin{cases} < 0 & \rightarrow \quad \text{elliptic,} \\ = 0 & \rightarrow \quad \text{parabolic,} \\ > 0 & \rightarrow \quad \text{hyperbolic,} \end{cases}$$

Hydro/ideal MHD equations are hyperbolic PDEs, but source terms (resistivity/viscosity/self-gravity) can be of other types.

Types of PDEs

Prototype of elliptic PDE:

Poisson equation: $\nabla^2 u = f$ (self-gravity)

Prototype of parabolic PDE:

Diffusion equation: $\partial_t u = D\partial_{xx}^2 u$ (viscosity, resistivity, heat conduction)

Prototype of hyperbolic PDE:

Wave equation: $\partial_{tt}^2 u - c^2 \partial_{xx}^2 u = 0$

Linear advection equation: $\partial_t u + A\partial_x u = 0$

Outline

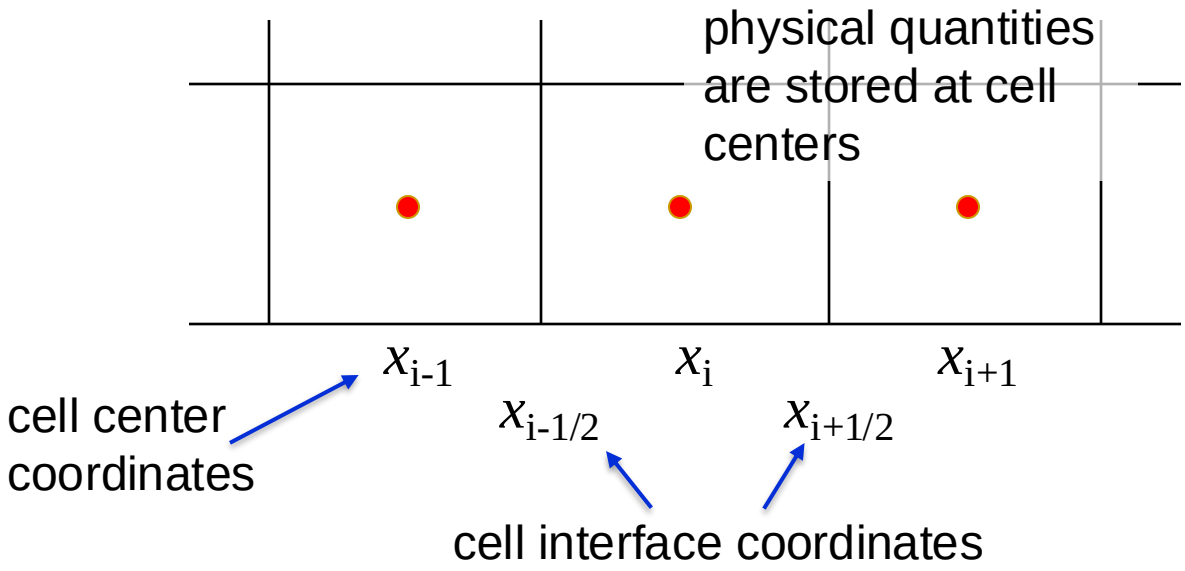
- Solving the linear advection equation
- Finite volume methods (for scalar conservation law)
- Godunov method for solving MHD equations

Useful reference: *Finite Volume Methods for Hyperbolic Problems*, LeVeque, 2002, Cambridge University Press.

Discretization

Solving PDEs by grid-based methods inevitably involves discretization and use finite differencing to approximate time/spatial derivatives.

For simplicity, we focus on Cartesian grid with uniform grid spacing Δx.



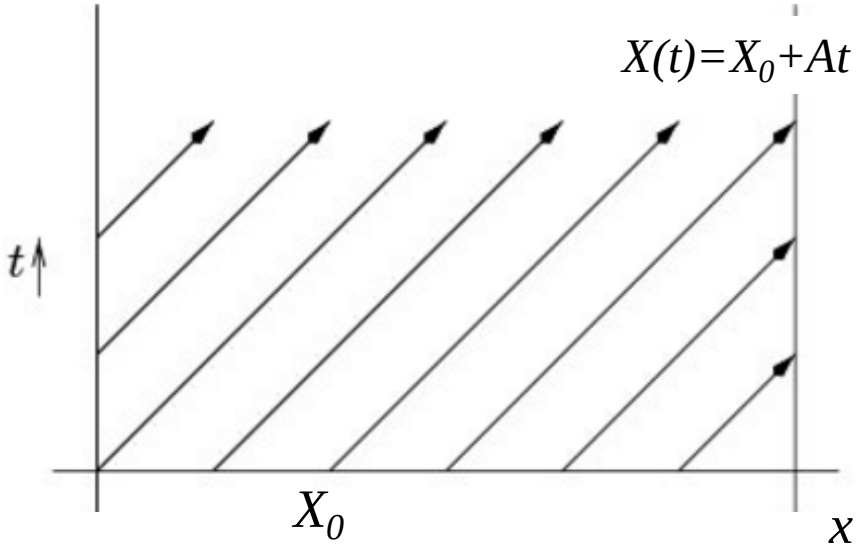
The linear advection equation (const coeff)

Consider linear advection eqs with constant A: $\partial_t u + A\partial_x u = 0$

Solution: $u(x,t) = u_0(x - At)$

The solution is constant along the ray (called the characteristic curve):

$$X(t) = X_0 + At$$



Proof:

$$\frac{d}{dt}u(X(t),t) = \partial_t u(X(t),t) + X'(t)\partial_x u(X(t),t)$$

$$= \partial_t u(X,t) + A\partial_x u(X,t) = 0$$

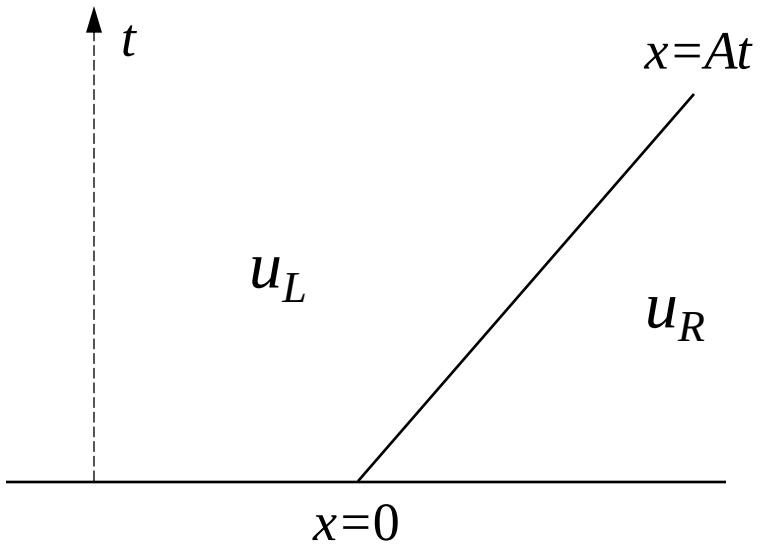
The Riemann problem (for linear advection eq)

$$\partial_t u + A\partial_x u = 0$$

Initial condition: $u=u_L$, $(x<0)$

$u=u_R$, $(x\geq 0)$

Result: discontinuity propagates along the characteristic curve.



Solve the linear advection equation on a grid

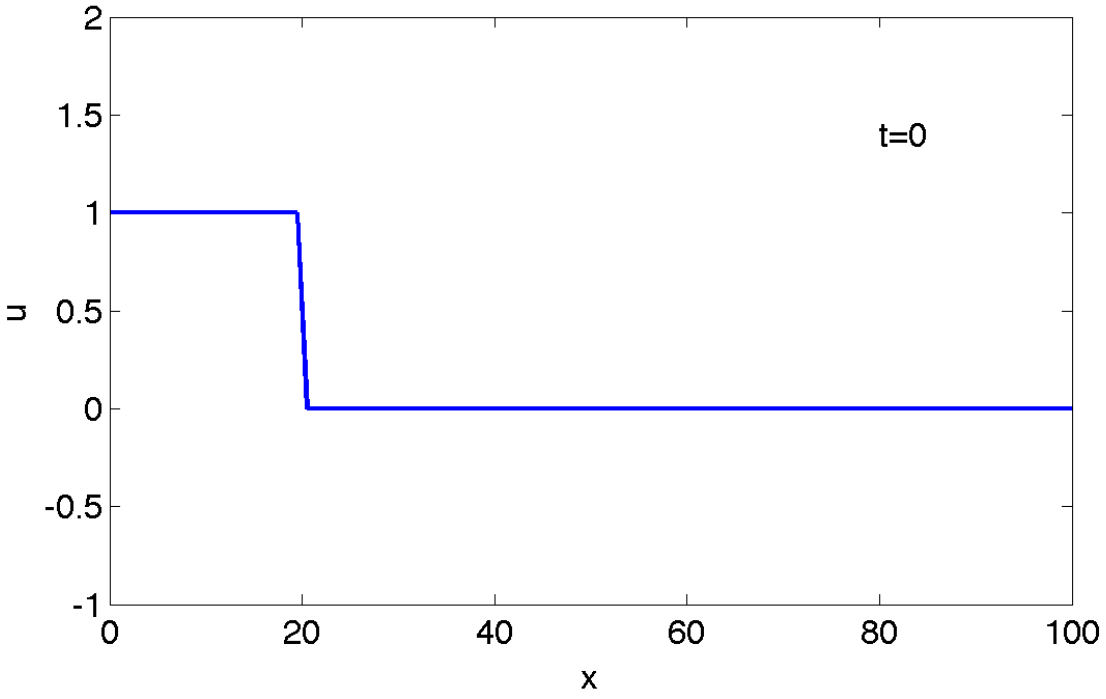
$$\partial_t u + A\partial_x u = 0$$

Initial condition: u=1 (x<50), u=0 (x>50), A=1.

1. Forward-time central-space (FTCS):

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = -A\left(\frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x}\right)$$

1st order in time
2nd order in space



The (finite-difference) method is unconditionally unstable!

von Neumann stability analysis

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = -A\left(\frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x}\right)$$

Represent the discretized solution by a finite Fourier series (strictly speaking, this is appropriate only for linear problem in a periodic domain).

Pick up one mode of the form: $u_i^n = ae^{\mathrm{i}k(i\Delta x)}$

Solution at the next timestep is: $u_i^{n+1} = A_k u_i^n$

A_k: amplification factor (complex number)

For FTCS method, it is straightforward to obtain:

$$A_k = 1 - \mathrm{i}\frac{A\Delta t}{\Delta x}\sin(k\Delta x)$$

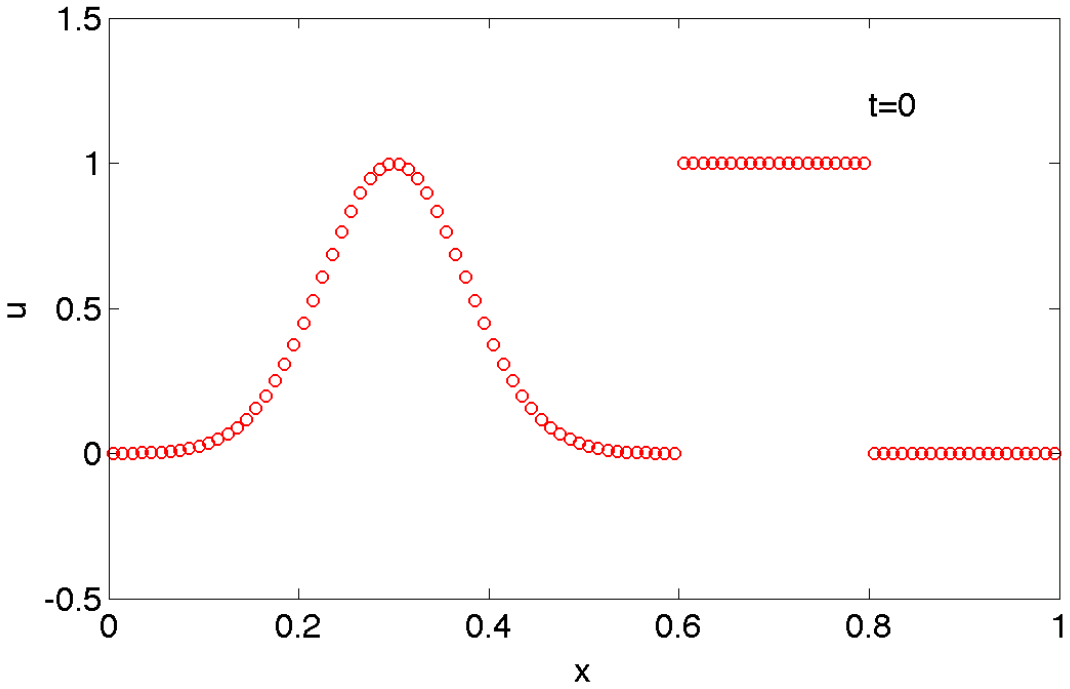
We see that no matter what timestep we choose, $|A_k|>1$. This means that the solution is exponentially amplified in time, being unconditionally unstable!

Lax-Friedrichs (LF) method

$$\partial_t u + A\partial_x u = 0$$

IC: one Gaussian, one square waves, A=1, periodic BC.

$$\frac{u_i^{n+1} - (u_{i-1}^n + u_{i+1}^n)/2}{\Delta t} = -A\left(\frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x}\right)$$



Effectively, added numerical diffusion to stabilize FTCS.

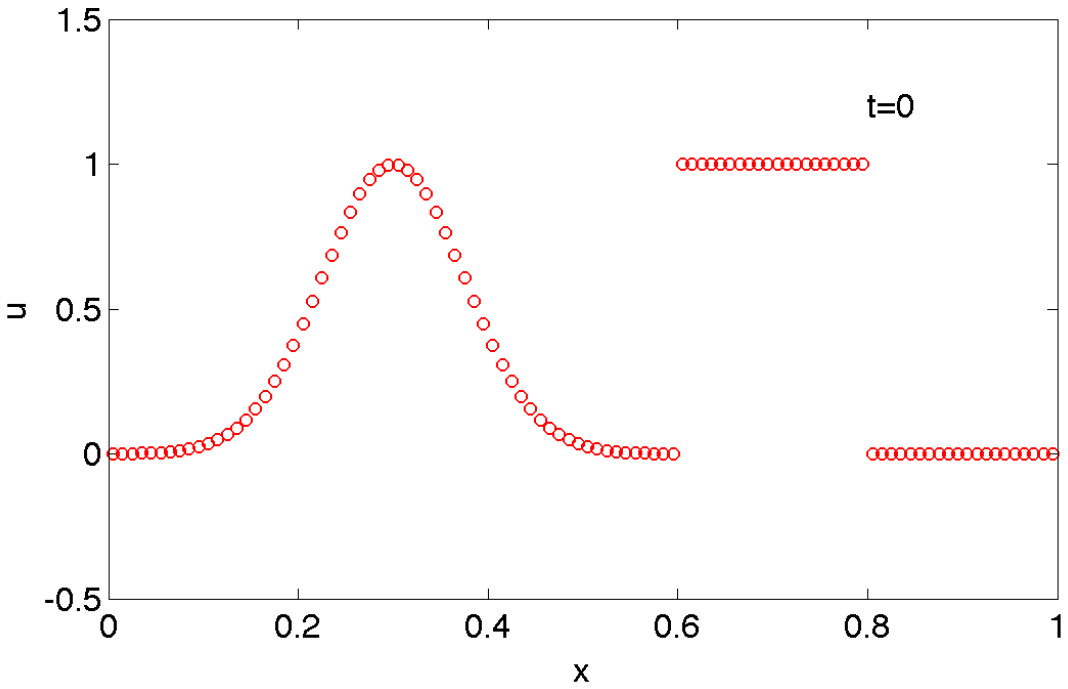
The method is stable, but VERY diffusive!

Upwind method

$$\partial_t u + A\partial_x u = 0$$

IC: one Gaussian, one square waves, A=1, periodic BC.

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \begin{cases} -A(u_i^n - u_{i-1}^n)/\Delta x & (A \geq 0) \\ -A(u_{i+1}^n - u_i^n)/\Delta x & (A < 0) \end{cases}$$



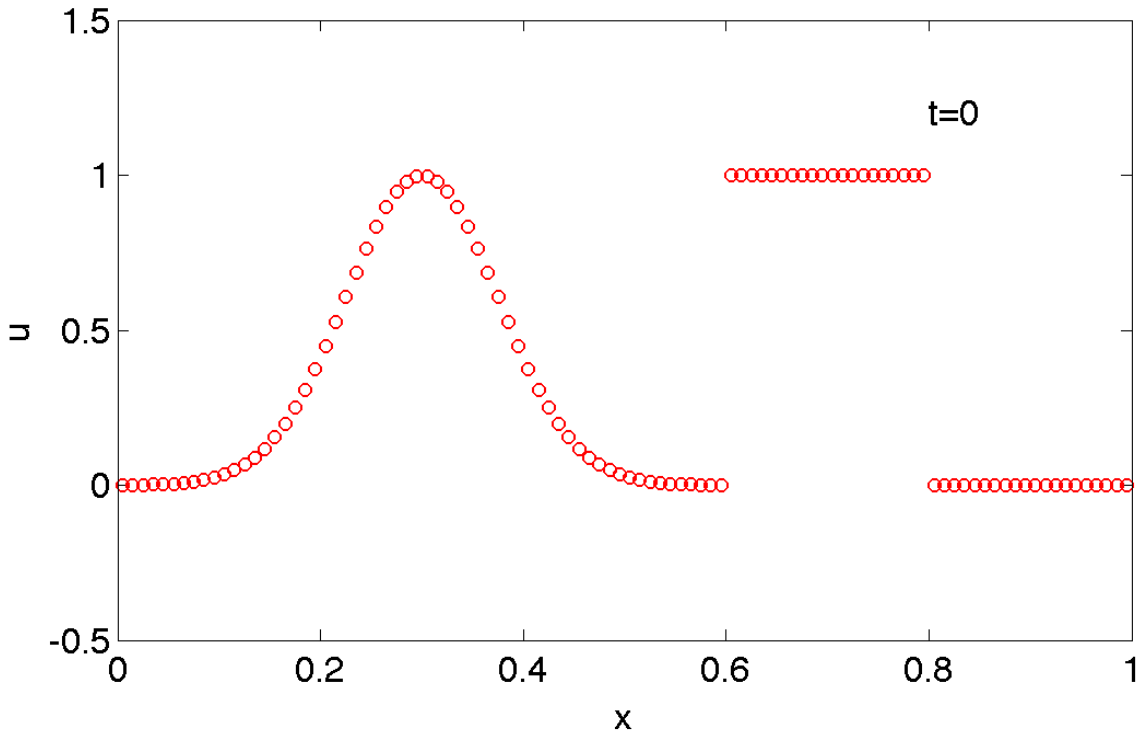
The method is stable and less diffusive, though still only first order accurate.

Lax-Wendroff method

$$\partial_t u + A\partial_x u = 0$$

IC: one Gaussian, one square waves, A=1, periodic BC.

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = -A\left(\frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x}\right) + \frac{A^2\Delta t}{2}\left(\frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{\Delta x^2}\right)$$



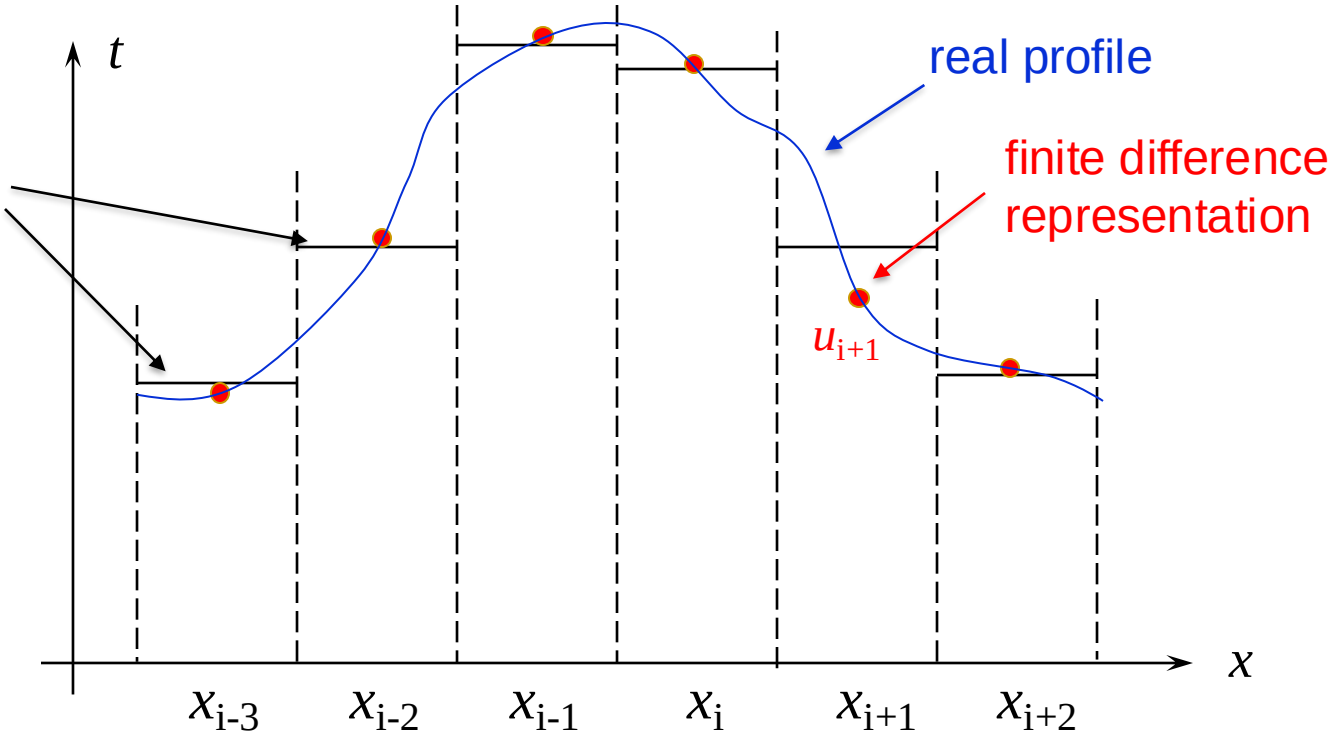
Method is stable but:

1). Oscillatory solution at discontinuities.

2). Phase shift in the smooth region.

Finite volume vs finite difference methods

FVM interprets grid discretization very differently.



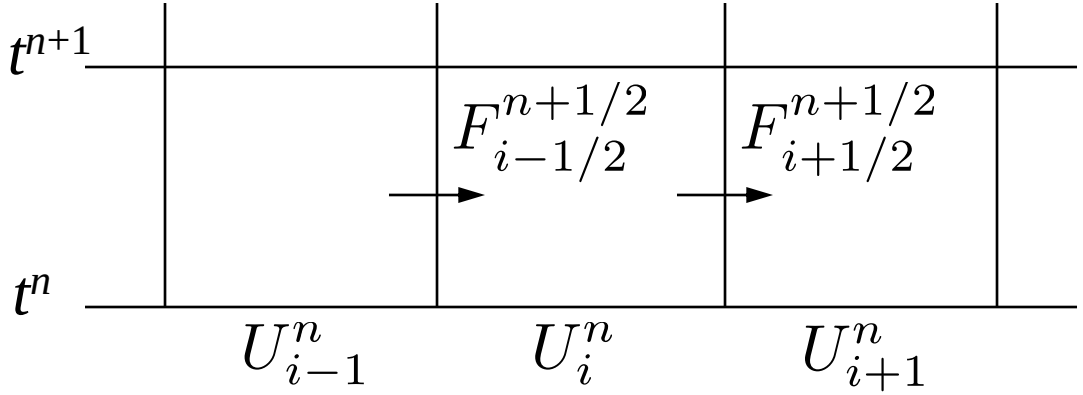
- FVM works with the integral form of the conservation laws.

Conserved variables are volume-averaged:

$$U_i^n = \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} u(x, t_n) dx$$

Finite volume method

- FVM works with the integral form of the conservation laws.



Conserved variables are volume-averaged:

$$U_i^n = \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_{i+1/2}} u(x,t_n)dx$$

Interface fluxes are time-averaged:

$$F_{i-1/2}^{n+1/2} = \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} f(u(x_{i-1/2},t))dt$$

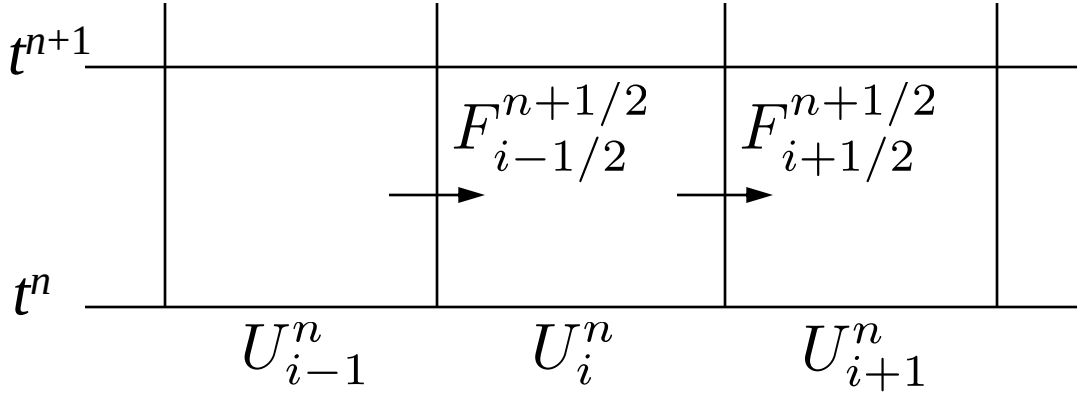
Finite volume update:

$$U_i^{n+1} = U_i^n + \frac{\Delta t}{\Delta x}(F_{i+1/2}^{n+1/2} - F_{i-1/2}^{n+1/2})$$

Conserved variables are conserved to machine accuracy.

How to compute the fluxes?

Consider a general scalar conservation law: $\partial_t u + \partial_x f(u) = 0$



We only know the volume-averaged values U.

To get the flux at cell interfaces, we essentially need to know the value of u at $x_{i+1/2}$ through some sort of averaging/interpolation.

Alternatively, need to find ways to approximate interface fluxes directly.

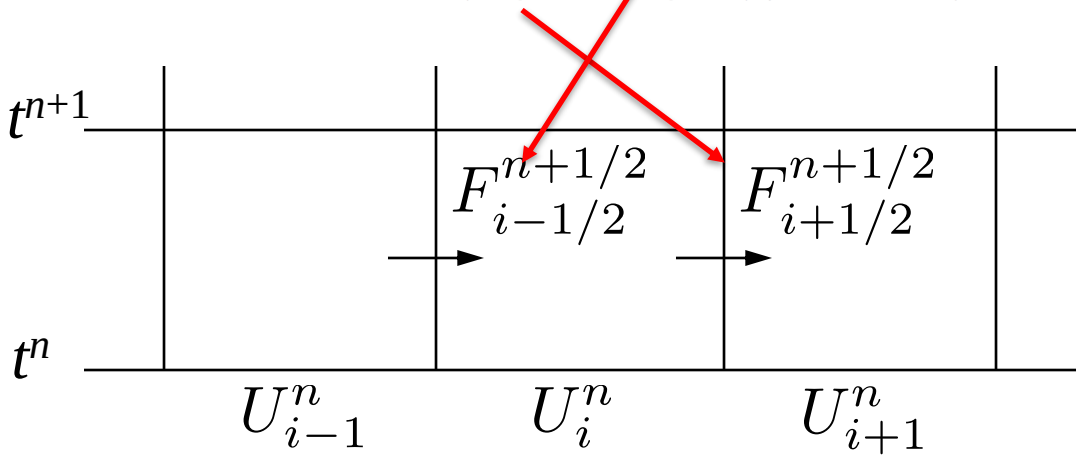
This is the key to finite volume methods.

Linear advection equation

For the linear advection equation, the aforementioned finite-difference methods can be given finite-volume interpretations:

For the upwind method:

$$\frac{U_i^{n+1} - U_i^n}{\Delta t} = \begin{cases} -A(U_{i+1}^n - U_i^n)/\Delta x & (A < 0) \\ -A(U_i^n - U_{i-1}^n)/\Delta x & (A \geq 0) \end{cases}$$



Upwind flux: $F_{i-1/2} = \begin{cases} AU_{i-1} & (A \geq 0) \\ AU_i & (A < 0) \end{cases}$

Godunov method (Basic idea) (Godunov, 1959)

- 1. Given volume averaged values U_i^n (defined at each cell), reconstruct piecewise polynomial function $\tilde{u}^n(x)$ (defined at all x).

Simplest scenario (piecewise constant/donor cell):

$$\tilde{u}^n(x) = U_i^n \quad \text{for } x_{i-1/2} \leq x < x_{i_1/2}$$

- 2. Using $\tilde{u}^n(x)$ as initial condition, evolve the hyperbolic equation exactly (or approximately) for Δt to obtain $\tilde{u}^{n+1}(x)$.

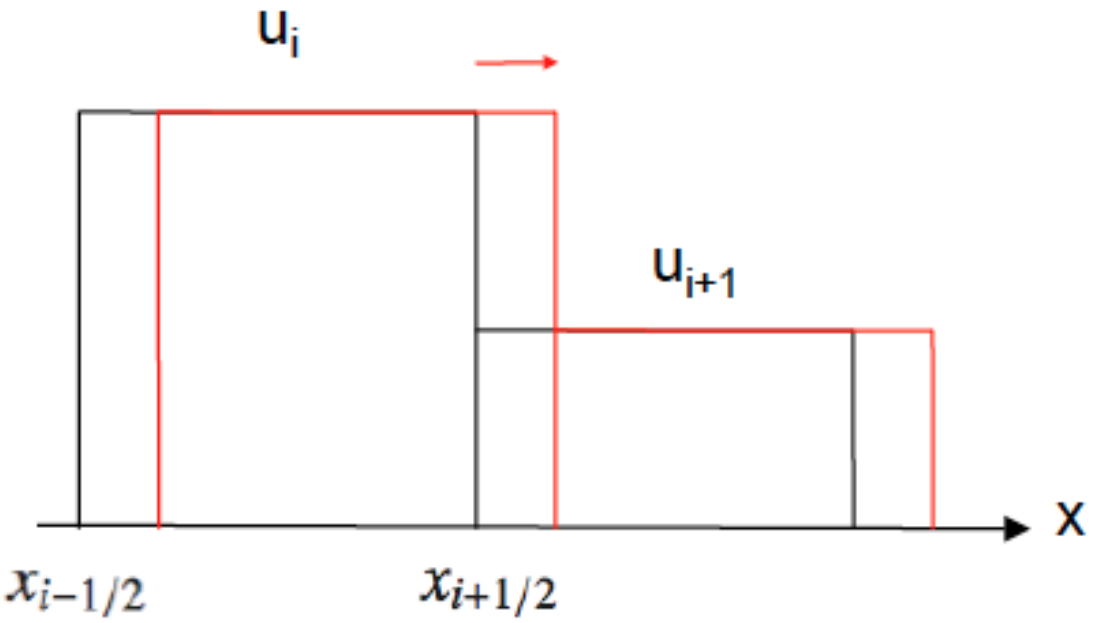
- 3. Average $\tilde{u}^{n+1}(x)$ over each cell to obtain new cell averages:

$$U_i^{n+1} = \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} \tilde{u}^{n+1}(x) dx$$

Godunov method (Basic idea)

A finite volume method originally proposed by Godunov (1959) for solving (non-linear) equations of gas dynamics.

Reconstruct ->
Evolve ->
Average



For linear advection equations, Godunov method with piecewise constant reconstruction = upwind method.

Key property: flux is properly upwinded to avoid spurious oscillations.

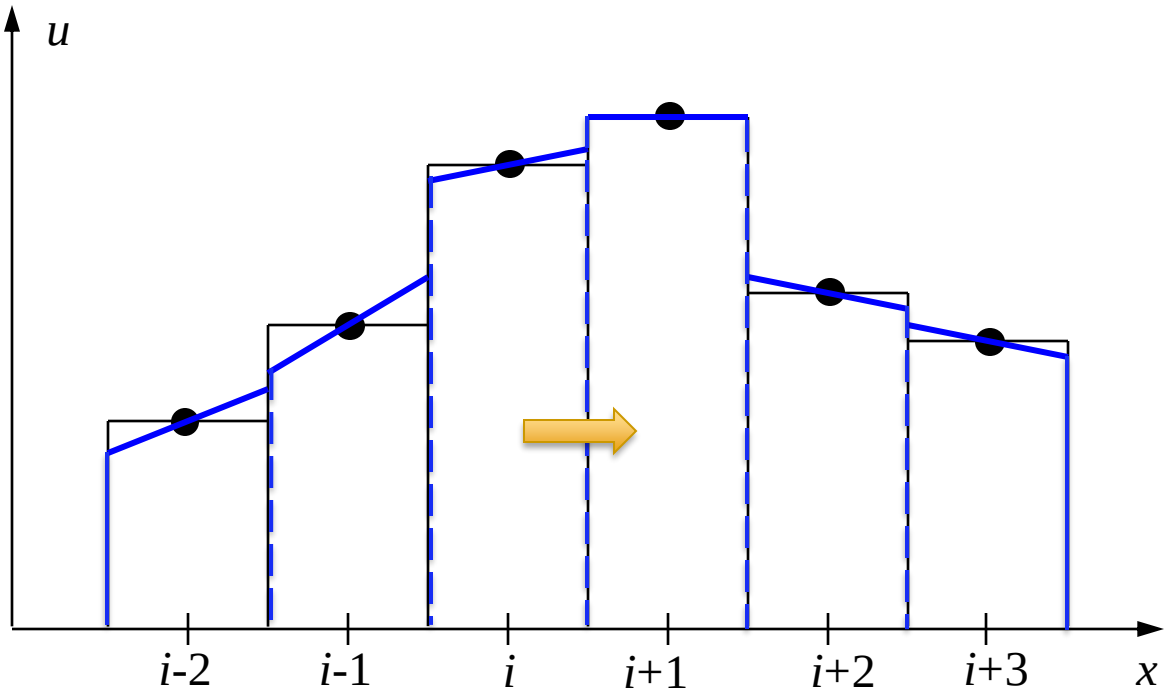
Toward higher order accuracy

Piecewise linear reconstruction:

$$u(x) = u_i + \sigma_i(x - x_i) \quad \text{for } (x_{i-1/2} \leq x < x_{i+1/2})$$

Evolve reconstructed profile according to the (linear advection) equation.

Volume average the evolved profile to the grid structure.



Can also be done at 3rd order: Piecewise-parabolic method (Colella & Woodward, 1984)

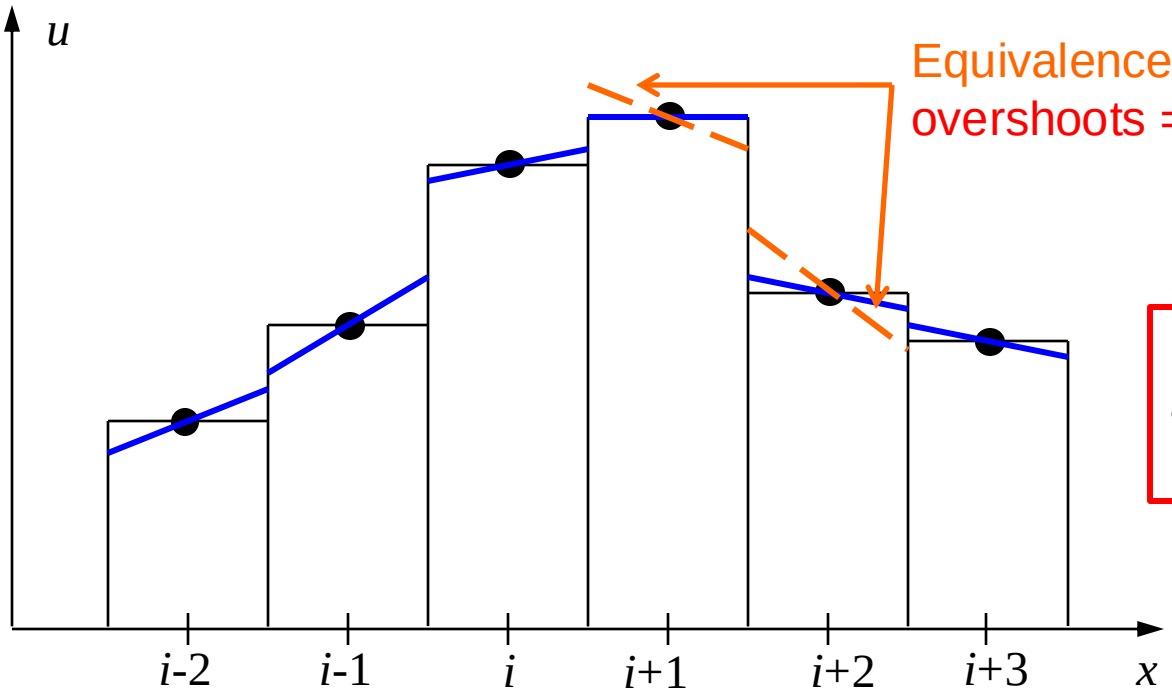
Toward higher order accuracy

How to choose the slopes?

$$u(x) = u_i + \sigma_i (x - x_i) \quad \text{for } (x_{i-1/2} \le x < x_{i+1/2})$$

Simplest choice: $\sigma_i = (u_{i+1} - u_{i-1})/2\Delta x$

This is in fact equivalent to the Lax-Wendroff method.



Equivalence of the Lax-Wendroff: overshoots => oscillations

Need slope limiters so as to be "monotonicity preserving"

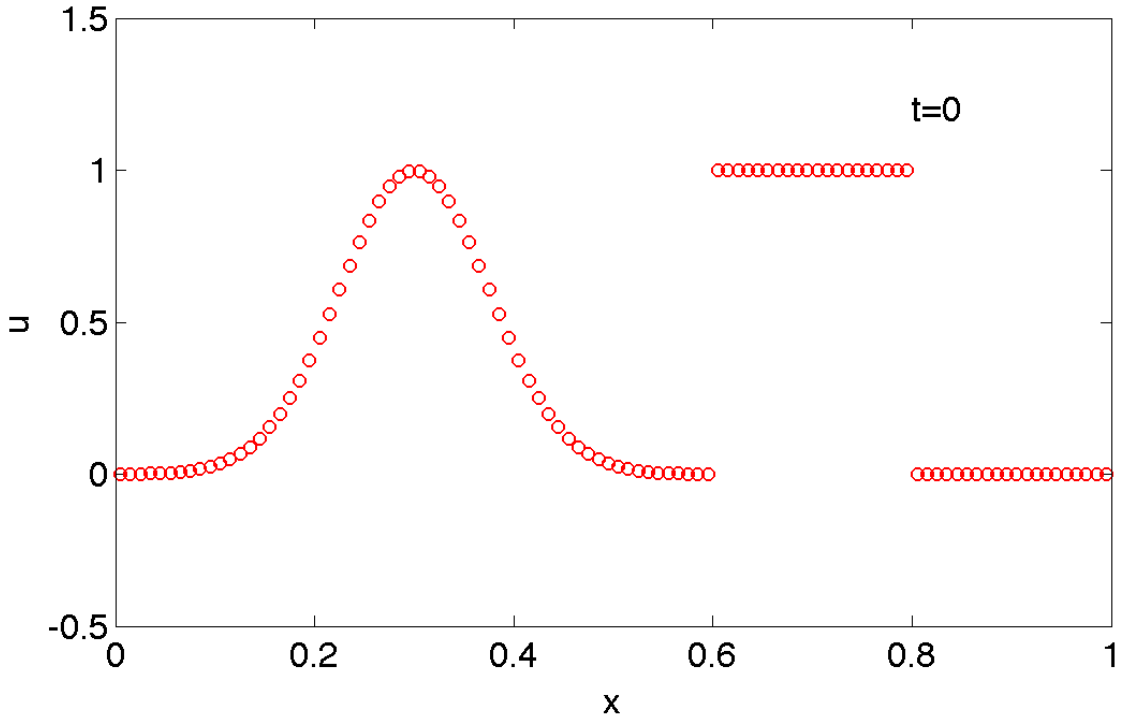
Solve the linear advection equation

Higher-order Godunov method with piecewise linear reconstruction

+the MC slope limiter.

$$\partial_t u + A\partial_x u = 0$$

Initial condition: one Gaussian, one square waves, A=1, periodic BC.



The method is stable and much more accurate:

2nd order accurate for smooth flow

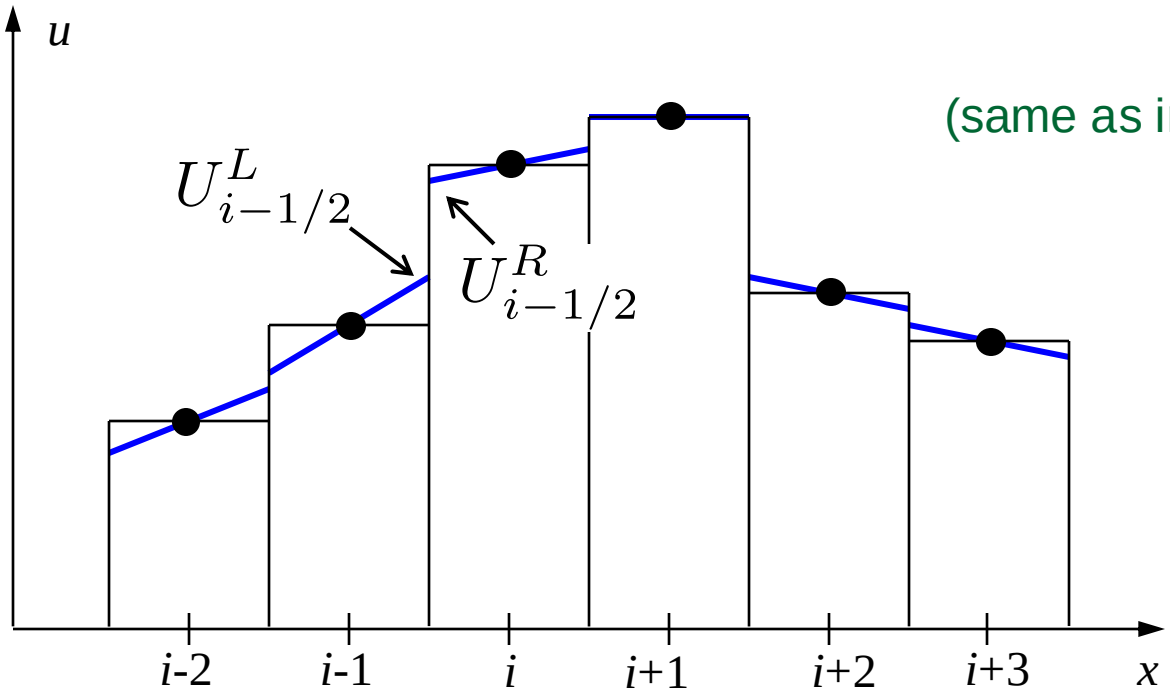
1st order accurate at discontinuities.

Non-linear scalar conservation law

To solve a non-linear scalar conservation law with Godunov method:

$$\partial_t u + \partial_x f(u) = 0$$

1. Given volume averaged values U_i^n, compute the left and right states $U_{L,i-1/2}$/$U_{R,i-1/2}$ at cell interfaces based on a reconstruction method.



(same as in the linear problem)

Non-linear scalar conservation law

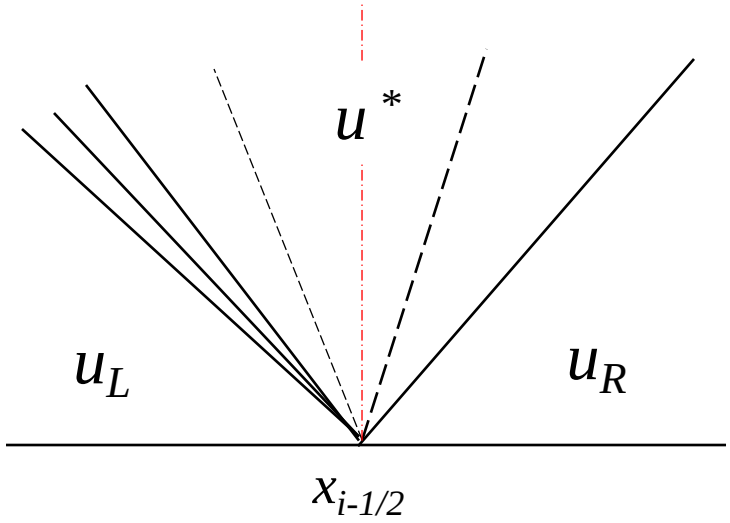
2. However, one generally can no longer directly evolve the system. This is replaced by a Riemann solver.

$$\partial_t u + \partial_x f(u) = 0 \quad \text{with} \quad \begin{array}{l} u=u_L,\ (x<0) \\ u=u_R,\ (x\geq 0) \end{array}$$

The key to a Riemann solver is to return the intermediate state u^* so that the interface flux is given by:

$$F_{i+1/2}=f(u^*)$$

(This approach automatically captures the shocks with dissipation)



3. Apply the flux-differencing formula:

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x}(F_{i+1/2} - F_{i-1/2})$$

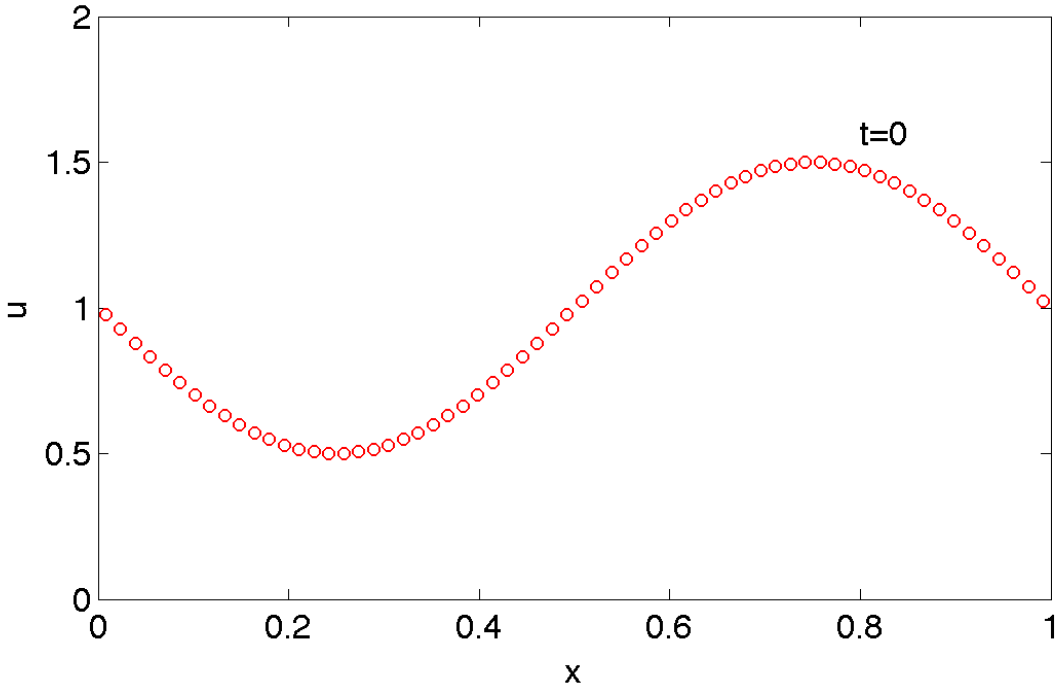
Solving non-linear equations

- Simplest example: Burger's Eqs $\partial_t u + u\partial_x u = 0$

  In conservative form: $\partial_t u + \partial_x \left(\frac{u^2}{2} \right) = 0$

Initial condition: u=1-sin(2πx)/2 in [0, 1], periodic BC.

Solved with Godunov method + 2nd order reconstruction



Hyperbolicity of linear systems

A linear system of the form

$$\partial_t \boldsymbol{u} + \mathbf{A} \cdot \partial_x \boldsymbol{u} = 0$$

is hyperbolic if matrix **A** is diagonalizable with real eigenvalues.

Let us denote the eigenvalues by $\lambda^1 \leq \lambda^2 \leq ... \leq \lambda^m$

The matrix is diagonalizable if there is a complete set of eigenvectors such that

$$A\boldsymbol{r}^p = \lambda^p \boldsymbol{r}^p$$

The right-eigenvectors jointly form a matrix: $R \equiv (\boldsymbol{r}^1, \boldsymbol{r}^2, ..., \boldsymbol{r}^m)$

so that $AR = \Lambda R$ where $\Lambda \equiv \mathrm{diag}(\lambda^1, \lambda^2, ..., \lambda^m)$.

In this way, the matrix A is diagonalized as: $\boxed{R^{-1}AR = \Lambda}$

Hyperbolicity of linear systems

A linear system of the form

$$\partial_t \boldsymbol{u} + \mathbf{A} \cdot \partial_x \boldsymbol{u} = 0$$

is hyperbolic if matrix **A** is diagonalizable with real eigenvalues.

For any vector $\boldsymbol{u}$, we can rewrite the original equation into:

$$\partial_t (R^{-1}\boldsymbol{u}) + (R^{-1}AR)\partial_x (R^{-1}\boldsymbol{u}) = 0$$

By defining characteristic variables as $\boldsymbol{w} = R^{-1}\boldsymbol{u}$, the linear system becomes

$$\partial_t \boldsymbol{w} + \Lambda \partial_x \boldsymbol{w} = 0$$

Or,

$$\partial_t w^p + \lambda^p \partial_x w^p = 0 \qquad (p=1,2,\ldots,m)$$

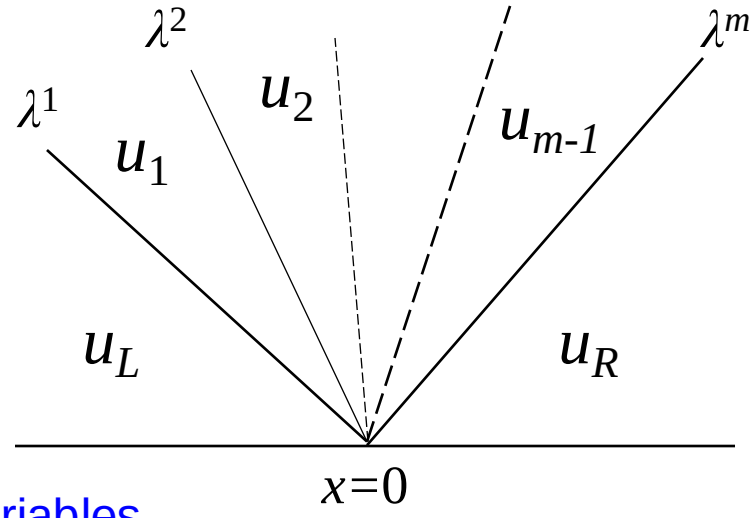
This is a set of decoupled linear advection equations, with λ^p being wave speeds.

The Riemann problem (for a linear system)

$$\partial_t \boldsymbol{u} + \mathbf{A} \cdot \partial_x \boldsymbol{u} = 0$$

Initial condition: $\boldsymbol{u}=\boldsymbol{u}_L$, $(x<0)$

$\boldsymbol{u}=\boldsymbol{u}_R$, $(x\geq 0)$



Solution:

1). Decompose u_L, u_R into characteristic variables.

$$\boldsymbol{u}_{L,R} = \sum_p w^p_{L,R} \boldsymbol{r}^p$$

2). Each characteristic variable evolves according to its own characteristics.

$$w^p(x,t) = w^p_L \quad \text{if } x - \lambda^p t < 0\,, \text{ otherwise, } \quad w^p(x,t) = w^p_R$$

3). Convert back to original variables.

$$\boldsymbol{u}(x,t) = \sum_{p:\lambda^p < x/t} w^p_R \boldsymbol{r}^p + \sum_{p:\lambda^p > x/t} w^p_L \boldsymbol{r}^p$$

1D MHD Equations

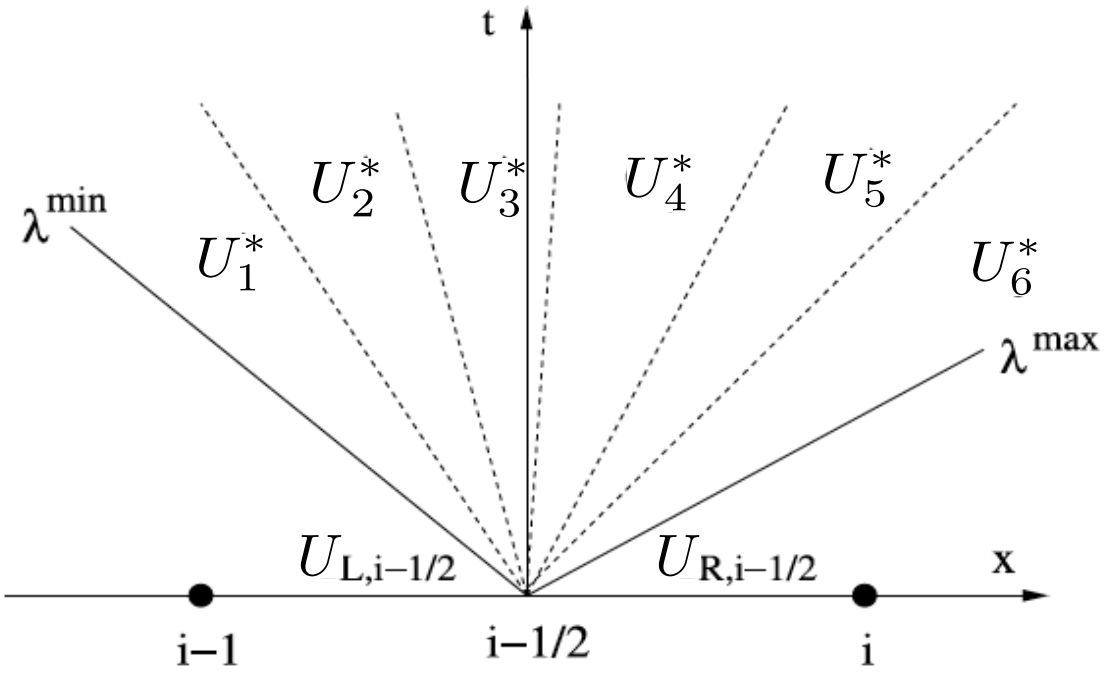
1D equations are plane-symmetric: $\nabla \cdot \boldsymbol{B} = 0$ => $B_x = \text{const}$

1D adiabatic MHD equations in conservative form: $\dfrac{\partial \boldsymbol{U}}{\partial t} + \dfrac{\partial \boldsymbol{F}}{\partial x} = 0$

$$\boldsymbol{U} = \begin{bmatrix} \rho \\ M_x \\ M_y \\ M_z \\ E \\ B_y \\ B_z \end{bmatrix} \qquad \boldsymbol{F} = \begin{bmatrix} \rho v_x \\ \rho v_x^2 + P + B^2/2 - B_x^2 \\ \rho v_x v_y - B_x B_y \\ \rho v_x v_z - B_x B_z \\ (E + P^*) v_x - (\boldsymbol{B} \cdot \boldsymbol{v}) B_x \\ B_y v_x - B_x v_y \\ B_z v_x - B_x v_z \end{bmatrix}$$

7 variables, 7 waves

The MHD Riemann problem



Goal: find the intermediate state at $x=x_{i-1/2}$.

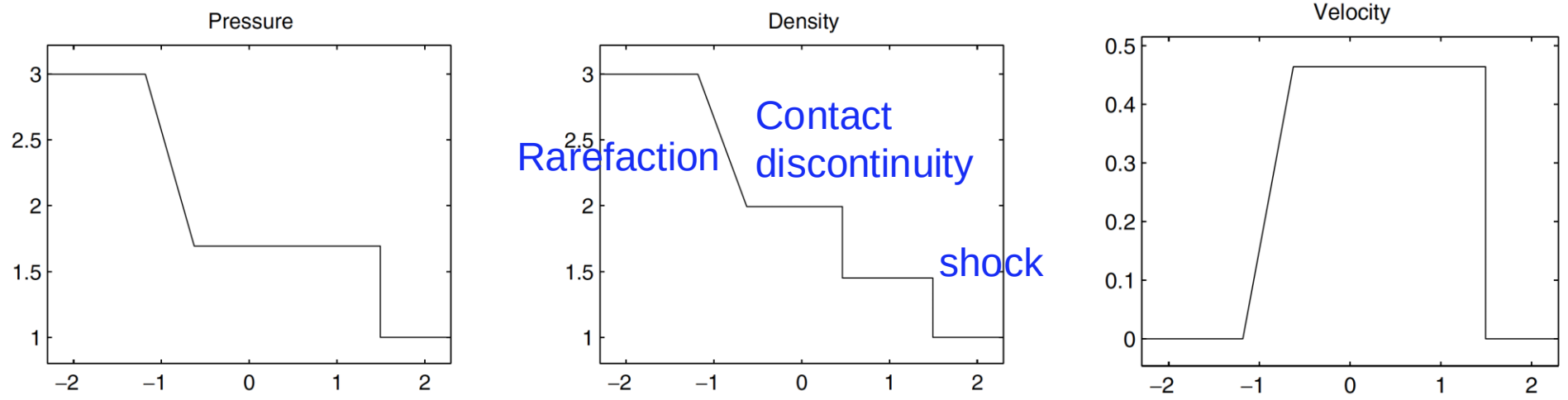
MHD Riemann solvers are much more complex, and in some cases, 2 of the 3 waves are degenerate (i.e., not strictly hyperbolic).

In practice, HD/MHD Godunov schemes use approximate Riemann solvers and/or linearized Riemann solvers.

Sod shock tube

Experimental shock tube: two different gas states separated by a membrane at $x=0$ => special Riemann problem with $u_l=u_r=0$.

Solution generally gives a shock + contact disc. + rarefaction:



For other initial conditions, it is possible to obtain two shocks or two rarefactions.

MHD Riemann solvers

- The Roe solver

  Exact solver for linearized equation with an approximate intermediate state.

  Good resolution for all 7 waves, generally less diffusive and more accurate.

  Expensive. Inaccurate/fail at strong discontinuity/rarefactions.

- The HLLE solver

  Only consider the fastest/slowest waves with only 1 intermediate state.

  Very simple and efficient; intermediate state is positive definite.

  Very diffusive, especially at contact discontinuities.

- The HLLC/HLLD solver

  Incorporate fast, Alfven waves and the contact discontinuity.

  Reasonably simple and efficient, guarantees positivity in 1D, better resolution at contact discontinuities.

Primitive vs. conserved variables

It is necessary to convert conserved variables $\boldsymbol{U}$ to primitive variables $\boldsymbol{W}$ in various stages of the computation.

Caveat: Due to the approximate nature of the Riemann solver, one might get negative density after one step of integration.

Similarly, with

$$E = \frac{P}{\gamma - 1} + \frac{1}{2}\rho v^2 + \frac{B^2}{8\pi}$$

one might obtain negative pressure following conversion from conserved to primitive variables.

$$\boldsymbol{U} = \begin{bmatrix} \rho \\ M_x \\ M_y \\ M_z \\ E \\ B_x \\ B_y \\ B_z \end{bmatrix}, \quad \boldsymbol{W} = \begin{bmatrix} \rho \\ v_x \\ v_y \\ v_z \\ P \\ B_x \\ B_y \\ B_z \end{bmatrix},$$

Solution:

1). Add density/pressure floors.

2). Use a more diffusive solver.

MHD integrator

- Second-order accuracy can be achieved using predictor-corrector type method (with a number of varieties).

Step 1: Donor-cell reconstruction to obtain interface L/R states.

Step 2: Use a Riemann solver to compute 1st order fluxes F^n.

Step 3: Advance the system for ½ time step (predict step).

$$U_i^{n+1/2} = U_i^n - \frac{\Delta t}{2\Delta x}(F_{i+1/2}^n - F_{i-1/2}^n)$$

Step 4: Use the second-order (piecewise-linear) reconstruction to compute the L/R states from $U^{n+1/2}$.

Step 5: Use a Riemann solver to compute 2nd order fluxes $F^{n+1/2}$.

Step 6: Update the system for a full time step.

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x}(F_{i+1/2}^{n+1/2} - F_{i-1/2}^{n+1/2})$$

This is one algorithm adopted in Athena++, following Falle (1991), modified from the MUSCL-Hancock ("van Leer") schemes.

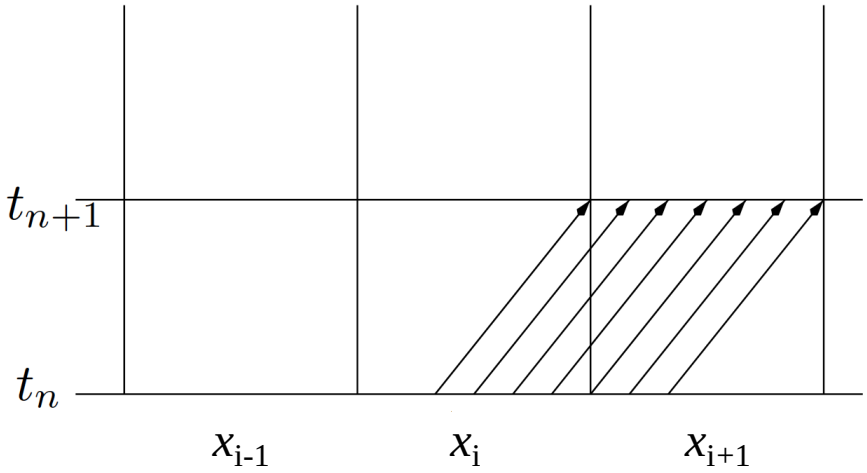
The Courant-Friedrichs-Lewy (CFL) condition

- A numerical method is convergent only if its domain of dependence contains the true domain of dependence of the PDE.
- In other words, the timestep Δt must be sufficiently small so that information propagates no more than one grid point per timestep.

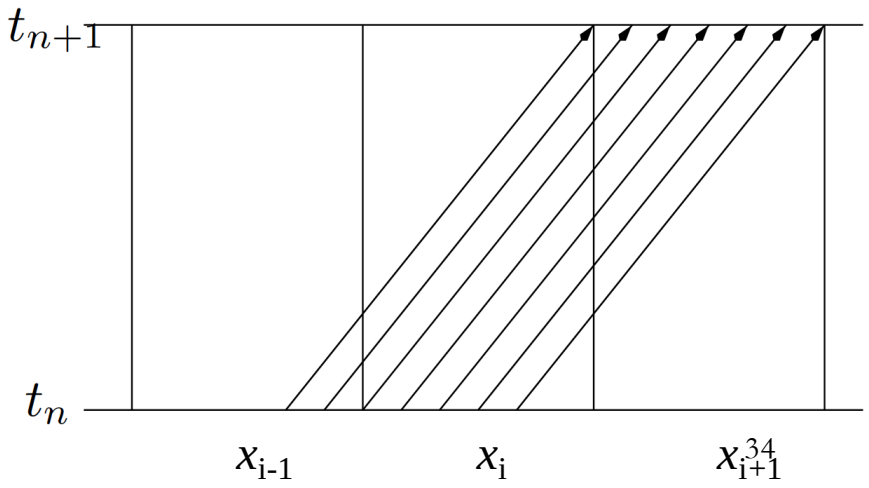
For the linear advection problem $\partial_t u + A\partial_x u = 0$,

$$\Delta t \equiv C\frac{\Delta x}{A} \quad , \quad \text{where the } \textbf{CFL number } C \leq 1.$$

When time step is appropriate (C<1)



When time step is too large (C>1)



The Courant-Friedrichs-Lewy (CFL) condition

- Numerical timestep Δt must be sufficiently small so that information propagates no more than one grid point per timestep.

For MHD equations, taking the fastest speed as:

$$V_{x,y,z} = \max[\mathrm{abs}(v_{x,y,z} \pm v_{f,\{x,y,z\}})]$$

where v is flow speed, v_f is the fast magnetosonic speed, in each direction.

The timestep is given by:

taken across the entire mesh

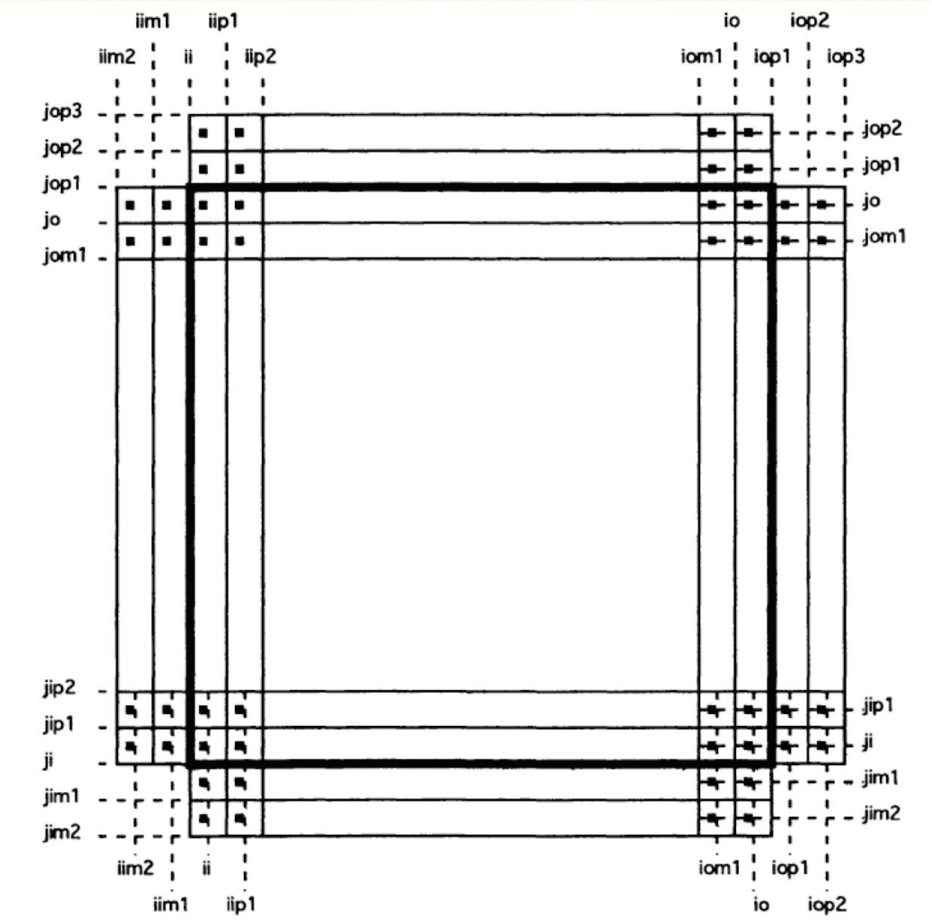
$$\Delta t = \mathrm{CFL} \cdot \min\left(\frac{\Delta x}{V_x}, \frac{\Delta y}{V_y}, \frac{\Delta z}{V_z}\right)$$

Different MHD integrators may have different requirement on the CFL number, which can also depend on the dimension of the problem.

e.g., the van Leer integrator in Athena++ requires the CFL # to < 1 for 1D and <0.5 for 2D/3D problems.

Boundary conditions (BCs)

Boundary conditions are implemented by properly filling the ghost zones.



Needed only to help determine the L/R states at the boundaries (and reconstruct EMFs for CT).

# of ghost zones depends on the order of reconstruction and numerical implementation (2 in Athena++).

Periodic and reflecting/conducting BCs are straightforward.

“zeroth-order extrapolation” generally gives best results serving for outflow/absorbing BCs