Equations of hydrodynamics

Gases and liquids are ensembles of atoms and molecules. Ideally, one wants to know the locations $\mathbf{r}_i$, the velocity $\boldsymbol{v}_i$, and the specific force $\mathbf{f}_i$, which acts on each particle i in the fluid containing N particles to integrate its motion and determine its future location and velocity via

$$\frac{\mathrm{d}\mathbf{r}_i}{\mathrm{d}t} = \boldsymbol{v}_i, \quad \frac{\mathrm{d}\boldsymbol{v}_i}{\mathrm{d}t} = \mathbf{f}_i(\mathbf{r}_j, \boldsymbol{v}_j, t), \quad j = 1 \ldots N. \tag{1}$$

Unfortunately, the number N gets boisterously high even for a modest amount of gas or fluid. One mole of a gas with a volume of $22.4 \times 10^{-3}\,\mathrm{m}^3$ at standard conditions (273.15 K and 10^5 Pa) contains 6.022×10^{23} particles. Following the trajectories of such a high number of particles is not feasible and beyond the capabilities of modern (and all known) future hardware.

In general, one does not need to know the exact locations and velocities of all these particles at any particular time. By the use of statistical physics, one can study the evolution of an ensemble of almost identical locations and velocities of almost identical particles. Fortunately, in many problems in astrophysics these particles can be treated as continuous media.

Equations of hydrodynamics 2

Continuous media can be described by the equations of hydrodynamics. The basic statistical equation from which the hydrodynamic equations are derived is the **Boltzmann equation**.

Here, we will introduce the hydrodynamical equations as conservation laws of mass, momentum and energy. For a derivation from the microphysics of gas see, e.g., Mihalas & Mihalas.

Several conditions have to be fulfilled so that an ensemble of particles can be treated as a continuous medium:

- The mean free path λ of a particle between two collisions has to be much smaller than the characteristic length scale l_{sc} of the problem

$$\lambda \ll l_{sc}. \tag{2}$$

- The material has to be in local thermal equilibrium.

Equations of hydrodynamics 3

Then, we can describe the fluid with the following set of variables

$$\rho, \mathbf{u}, \epsilon, \tag{3}$$

which are the five independent variables of the density, the three components of the velocity and the specific internal energy.

› In order to define a local mean of these values of $\rho(\mathbf{r})$, $\boldsymbol{v}(\mathbf{r})$, and $u(\mathbf{r})$, an element of the medium with a finite size must be chosen centered on $\mathbf{r}$ such that its size l_{el} satisfies the condition

$$\lambda \ll l_{el} \ll l_{sc}. \tag{4}$$

› The independent variables should not vary over l_{el} and the number of particles within l_{el} has to be large enough to do statistics.

› Another condition is that the inter-particle forces should be short range, i.e. no interaction should occur between particles separated by distances larger than $\sim l_{el}$. **Gravitational terms** in the hydrodynamical equations are obtained from the Boltzmann equation only if they are from sources external to the medium.

Equations of hydrodynamics 4

Let's check if the $\lambda \ll l_{sc}$ condition is fulfilled under typical astrophysical conditions.

In a neutral gas, the mean free path is given by

$$\lambda = \frac{1}{n\sigma}, \tag{5}$$

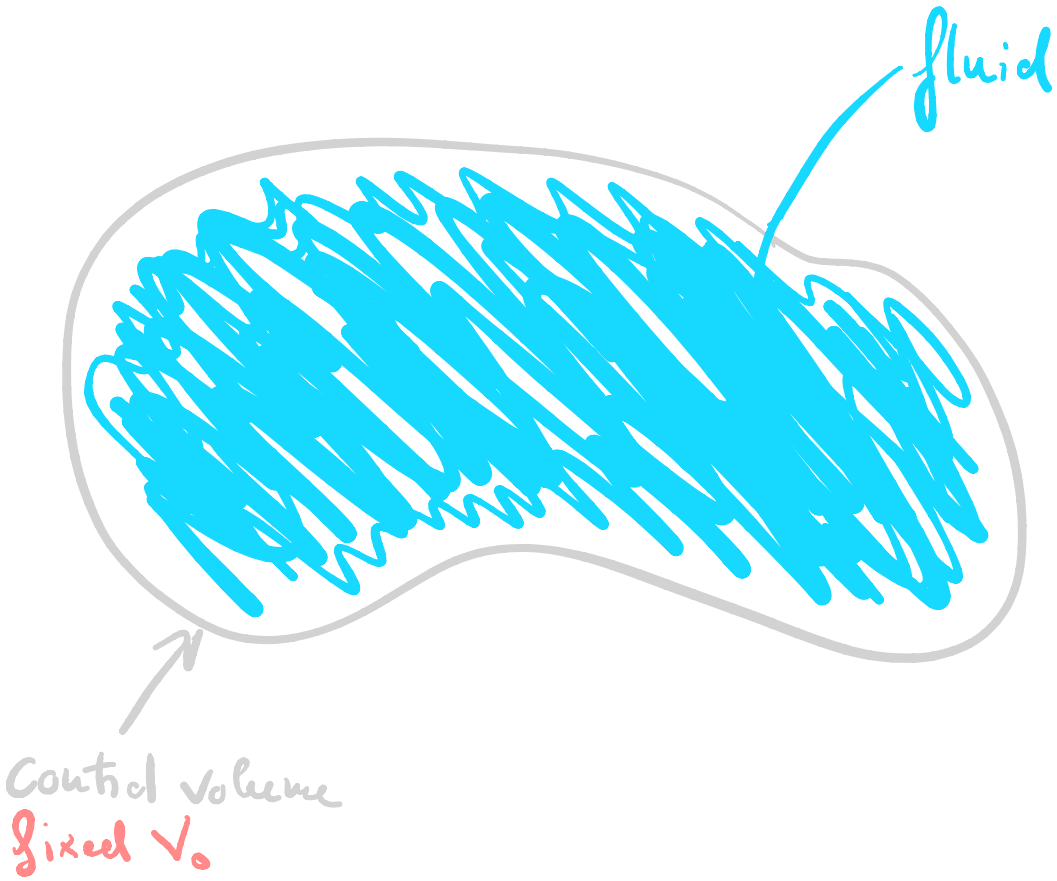
where σ is the scattering cross section (typically in the order of $\sim 10^{-15}\,\mathrm{cm}^2$), and n is the number density.

In interstellar clouds (so called HI clouds) with the main component neutral hydrogen, we find densities of a few atoms per cm^3. This gives us a mean free path of $\sim 10^{14}\,\mathrm{cm}$ (distance Sun to Jupiter is $7.8 \times 10^{13}\,\mathrm{cm}$).

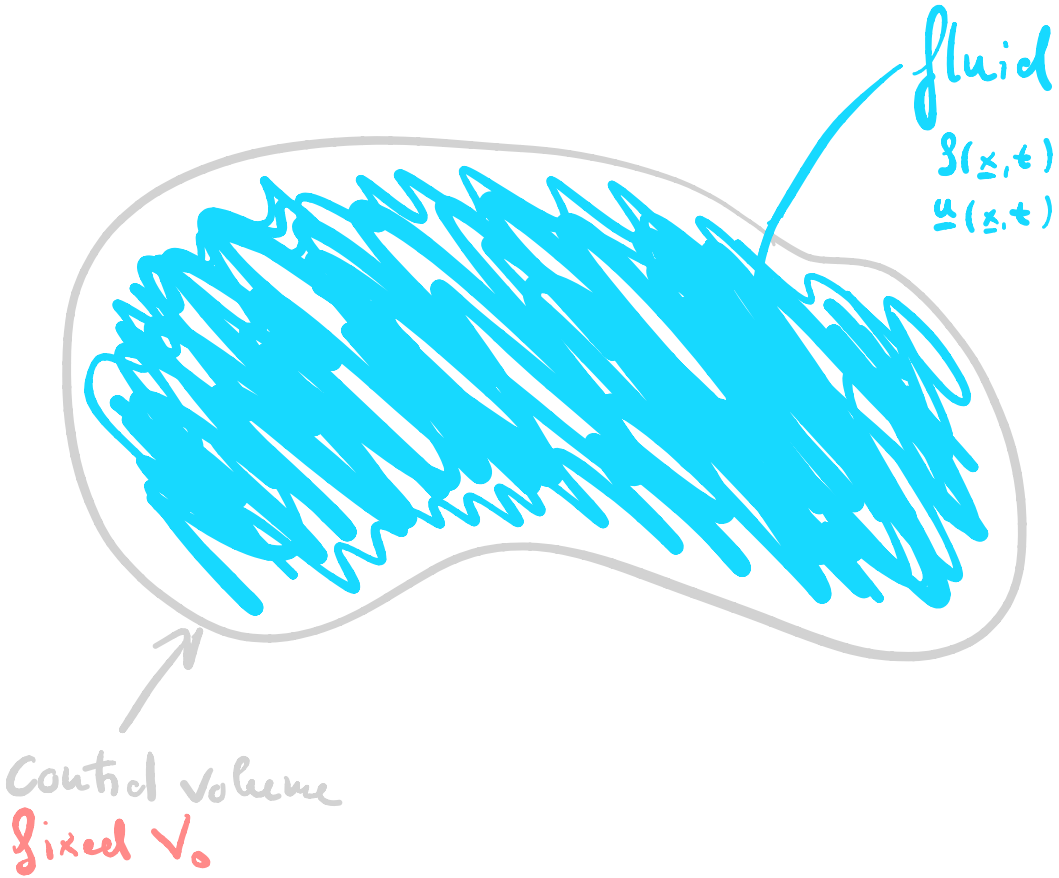
But these clouds have sizes that extend over tens of parsecs or $10^{19}\,\mathrm{cm}$ to $10^{20}\,\mathrm{cm}$, and it holds $\lambda \ll l_{sc}$.

We will see later, however, that strong gradients in density or temperatures, eventually resulting in shock fronts, which are just some few mean free paths wide, violate the condition. The hydrodynamical approach neglects this violation and shocks are treated as discontinuities.

Conservation of mass

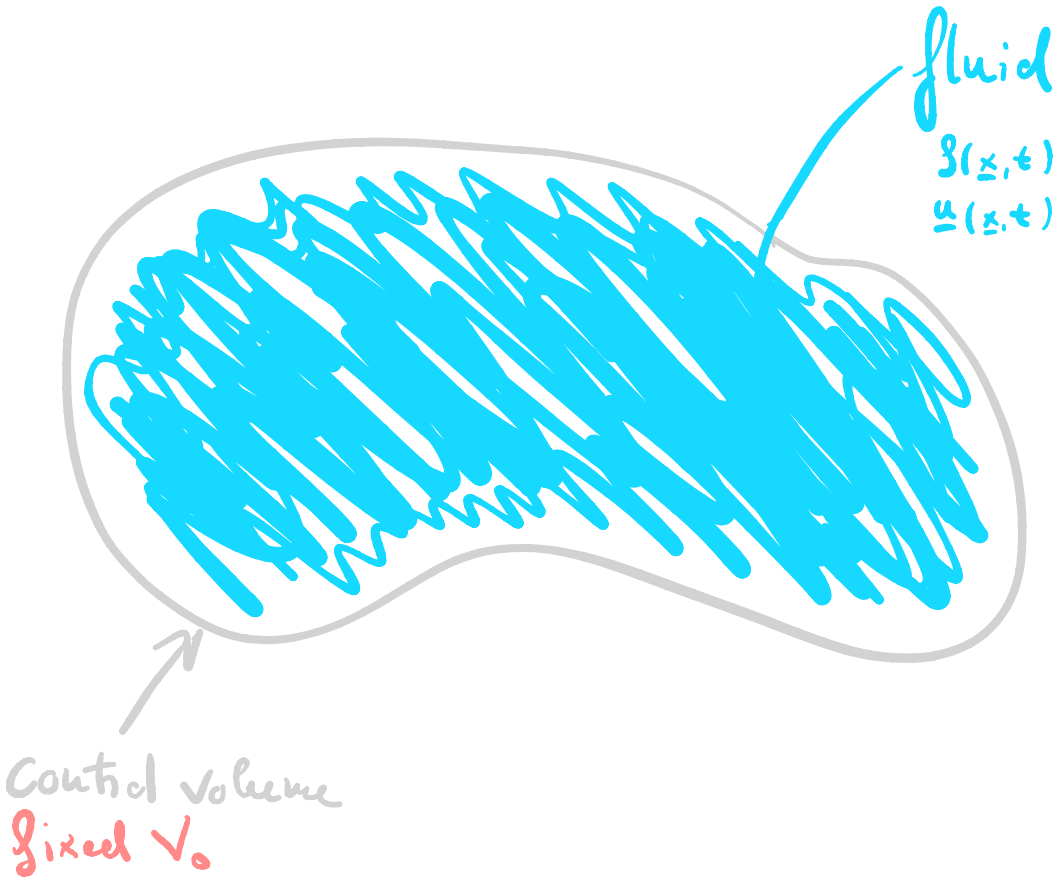


Conservation of mass



Mass in V_0 is given by $m(V_0, t) = \int_{V_0} \rho(\mathbf{x}, t) dV$.

Conservation of mass

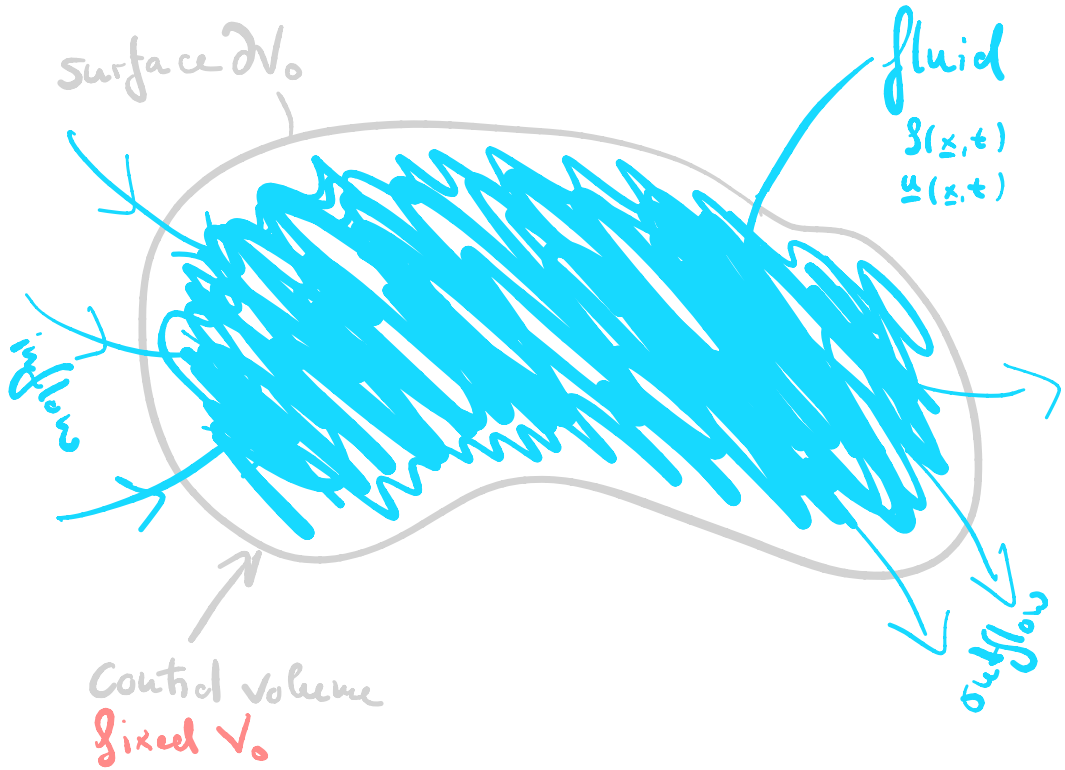


We are interested in the change of mass in Volume V_0, which is $\frac{dm}{dt}(V_0, t)$:

$$\frac{d}{dt} m(V_0, t) = \frac{d}{dt} \int_{V_0} \rho(\mathbf{x}, t) dV = \int_{V_0} \frac{\partial}{\partial t} \rho(\mathbf{x}, t) dV. \tag{6}$$

Conservation of mass

With no sinks or sources in V_0, the mass can only change by flow through the surface ∂V_0. The mass flow is given by $\rho\mathbf{u}$.

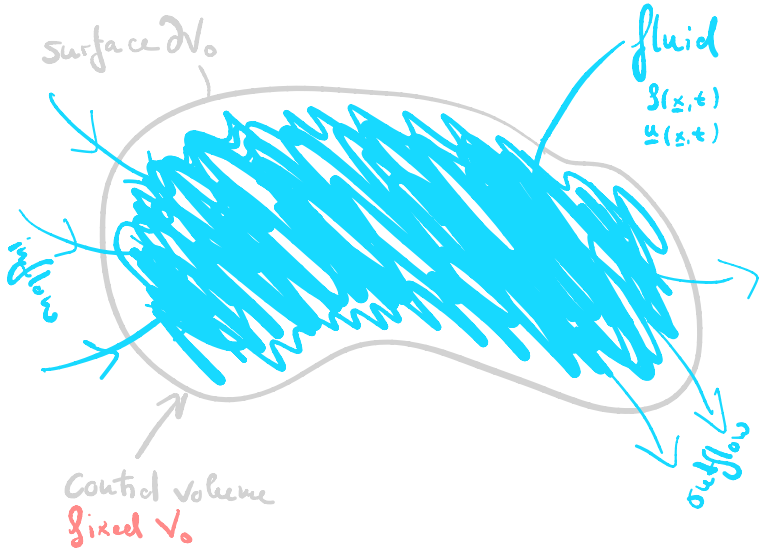


Hence, we postulate that the change of mass is given by the mass flow through the surface

$$\frac{\mathrm{d}}{\mathrm{dt}}\mathrm{m}(V_0,\mathrm{t}) = \frac{\mathrm{d}}{\mathrm{dt}}\int_{V_0}\rho(\mathbf{x},\mathrm{t})\mathrm{d}V = -\oint_{\partial V_0}\rho\mathbf{u}\cdot\mathbf{n}\mathrm{d}S. \tag{7}$$

Conservation of mass

With no sinks or sources in V_0, the mass can only change by flow through the surface ∂V_0. The mass flow is given by $\rho\mathbf{u}$.



With Gauss's theorem ($\oint_{\partial V} \mathbf{a} \cdot \mathbf{n} dS = \int_{V_0} \nabla \cdot \mathbf{a} dV$), we can transform the surface integral in an integral over the volume V_0 and it follows

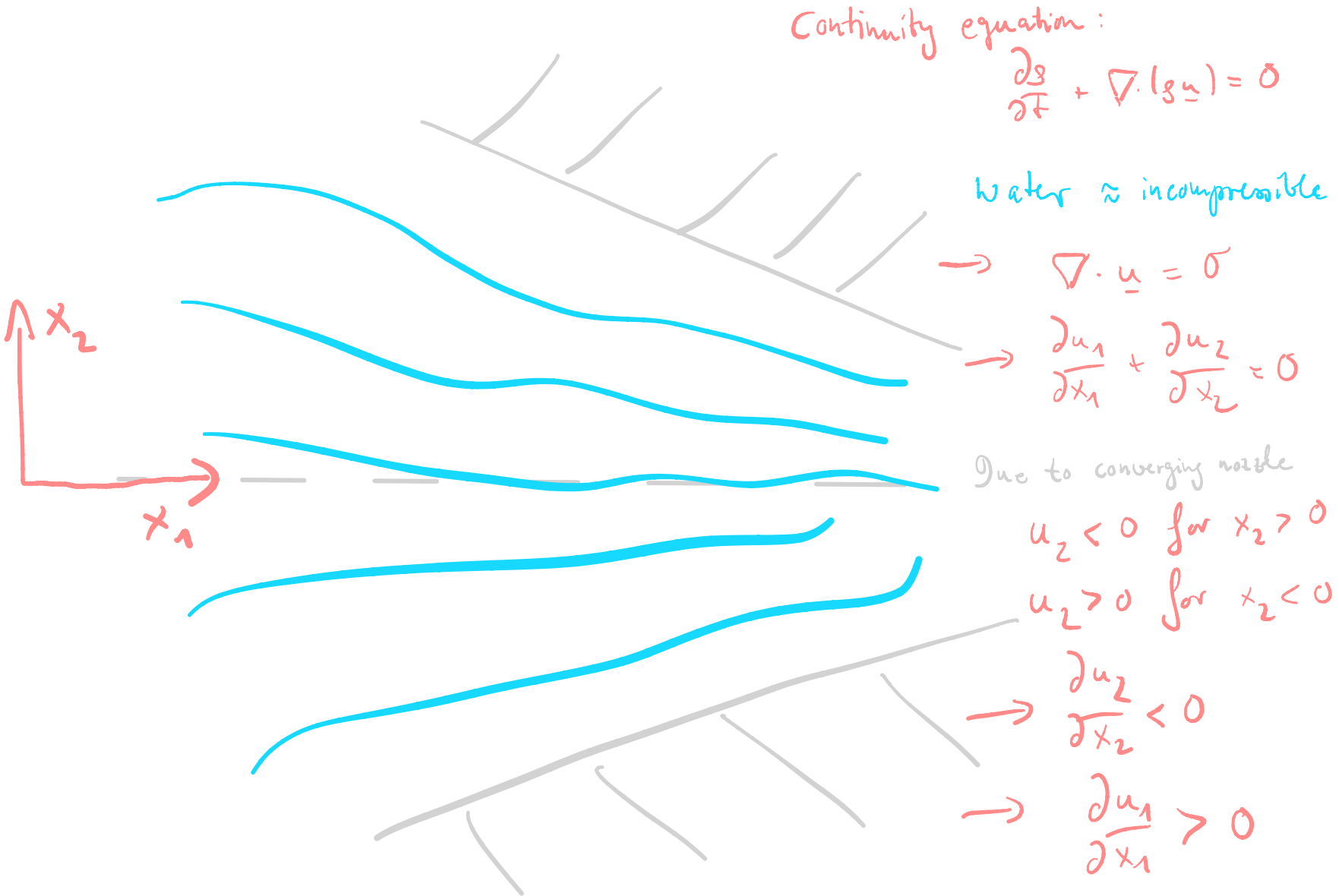
$$\int_{V_0} \left[\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) \right] dV = 0. \tag{8}$$

Since this is true for arbitrary V_0, we finally obtain

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \tag{9}$$

which is also known as the continuity equation.

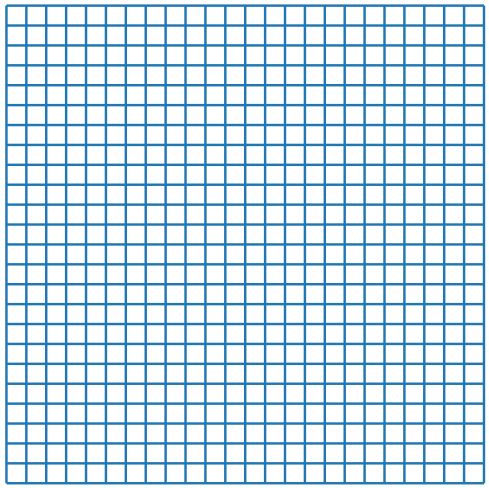
Simple example



Eulerian vs. Lagrangian formulation

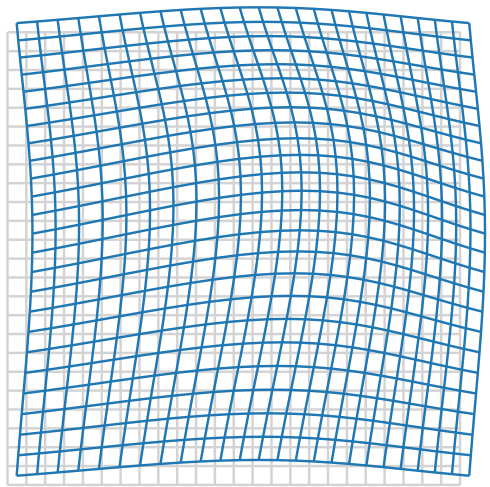
Eulerian vs. Lagrangian formulation

Eulerian view
fixed grid/coordinate system
laboratory frame of reference
matter flows through the grid



grid methods (finite differences, finite volume)

Lagrangian view
moving grid/particle
grid/coordinate system moves with the flow
material coordinate system



moving mesh methods, particle methods (Smoothed Particle Hydrodynamics)

Eulerian vs. Lagrangian formulation

Consider a scalar quantity $\mathrm{b} = \mathrm{b}(\mathbf{x}, \mathrm{t})$, which depends on space and time. The total time derivative of this quantity is given by

$$\frac{\mathrm{d}}{\mathrm{dt}}\mathrm{b}(\mathbf{x}, \mathrm{t}) = \frac{\partial \mathrm{b}}{\partial \mathrm{t}} + \frac{\mathrm{d}\mathbf{x}}{\mathrm{dt}} \cdot \nabla \mathrm{b}, \tag{22}$$

where $\dot{\mathbf{x}}(\mathrm{t})$ describes a chosen path in space. If we do not move ($\dot{\mathbf{x}} = 0$), the total time derivative reduces to the partial time derivative and we stay at a constant position.

If we choose a path that moves exactly with the fluid velocity $\mathbf{u}$, we set $\dot{\mathbf{x}} = \mathbf{u}$ and obtain

$$\frac{\mathrm{d}}{\mathrm{dt}}\mathrm{b}(\mathbf{x}, \mathrm{t}) = \frac{\partial \mathrm{b}}{\partial \mathrm{t}} + \mathbf{u} \cdot \nabla \mathrm{b}. \tag{23}$$

This is called the *material*, *Lagrangian*, *advective*, *comoving* or *substantial* derivative, and is the link between Eulerian and Lagrangian view on the flow. Often, the following notations are used

$$\mathrm{D}_\mathrm{t} \equiv \frac{\mathrm{D}}{\mathrm{Dt}} \equiv \frac{\mathrm{d}}{\mathrm{dt}} \tag{24}$$

Continuity equation

We start with the continuity equation in Eulerian representation

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0. \tag{25}$$

And transform the equation into Lagrangian representation by the following steps

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = \frac{\partial \rho}{\partial t} + \mathbf{u} \cdot \nabla \rho + \rho \nabla \cdot \mathbf{u} \tag{26}$$

$$= \frac{\mathrm{d}\rho}{\mathrm{d}t} + \rho \nabla \cdot \mathbf{u}, \tag{27}$$

to obtain

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \qquad \text{Euler}, \tag{28}$$

$$\frac{\mathrm{d}\rho}{\mathrm{d}t} + \rho \nabla \cdot \mathbf{u} = 0 \qquad \text{Lagrange}. \tag{29}$$

Euler equations on one slide

Hydroequations for an ideal fluid

Euler representation

$$\partial_t \rho + \nabla \cdot (\rho \mathbf{u}) = 0 \qquad \text{mass} \tag{53}$$

$$\partial_t (\rho \mathbf{u}) + \nabla \cdot (\rho \mathbf{u}\mathbf{u}) = -\nabla p + \rho \mathbf{f} \qquad \text{momentum} \tag{54}$$

$$\partial_t (\rho \epsilon) + \nabla \cdot (\rho \epsilon \mathbf{u}) = -p \nabla \cdot \mathbf{u} \qquad \text{internal energy} \tag{55}$$

Lagrange representation

$$\frac{d\rho}{dt} + \rho \nabla \cdot \mathbf{u} = 0 \qquad \text{mass} \tag{56}$$

$$\rho \frac{d\mathbf{u}}{dt} = -\nabla p + \rho \mathbf{f} \qquad \text{momentum} \tag{57}$$

$$\rho \frac{d\epsilon}{dt} = -p \nabla \cdot \mathbf{u} \qquad \text{internal energy} \tag{58}$$

The Euler equations in conserved form

We can write these equations (after some reformulation) combined in the following form ($I_{ij} \equiv \delta_{ij}$)

$$\partial_t \begin{pmatrix} \rho \\ \rho\mathbf{u} \\ \rho(\epsilon + \frac{1}{2}\mathbf{u}^2) \end{pmatrix} + \nabla \cdot \begin{pmatrix} \rho\mathbf{u} \\ \rho\mathbf{u}\mathbf{u} + pI \\ \left[\rho(\epsilon + \frac{1}{2}\mathbf{u}^2) + p\right]\mathbf{u} \end{pmatrix} = \begin{pmatrix} 0 \\ \rho\mathbf{f} \\ \rho\mathbf{u}\cdot\mathbf{f} \end{pmatrix}. \quad (59)$$

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{U}) = \mathbf{q}. \quad (60)$$

The vector $\mathbf{U}$ contains the conserved values $(\rho, \rho\mathbf{u}, \rho\epsilon_{tot})$, $\mathbf{F}$ denotes the transported fluxed, and $\mathbf{q}$ the sources.

[Note: Using the total specific energy $\epsilon_{tot} = \varepsilon + \frac{1}{2}\mathbf{u}^2$, one finds the conservation equation for the total energy

$$\partial_t(\rho\epsilon_{tot}) + \nabla \cdot (\rho\epsilon_{tot}\mathbf{u}) = -\nabla \cdot (p\mathbf{u}) + \rho\mathbf{u} \cdot \mathbf{f},$$

where the term $\rho\mathbf{u} \cdot \mathbf{f}$ originates from external forces acting on the fluid element.]

Equation of state

These are the equations that we want to solve using different numerical methods. They describe the flow of an ideal fluid. Ideal means no viscosity, no shear stress, no heat conduction.

The motion is described by the Euler equations, which are the conservation (or better *balance*) equations for mass, momentum and internal energy.

They form a set of partial differential equations. In order to close the system of the PDEs, we need an additional relation between the thermodynamical variables internal energy and density. Normally, this is achieved by the use of an equation of state (EOS) that relates pressure, temperature and density

$$p = p(\rho, \epsilon). \tag{61}$$

In most of our applications we will use the EOS of a perfect gas, which is also known as the ideal gas equation.

Experiment shows that the pressure exerted by a gas at constant T is $\sim 1/V$, and the pressure exerted by a fixed V is $\sim T$ (Boyle's law (1662) and law of Gay-Lussac (1802)).

Equation of state

For an ideal gas, the EOS reads

$$pV = n\mathcal{R}T, \tag{62}$$

where n is the number of moles and $\mathcal{R}$ the universal gas constant. The ideal gas equation is an excellent approximation for low-density (*dilute*) gases. Since n moles contain $n \times N_A$ (Avogadro number N_A) particles, we can rewrite the ideal gas equation as

$$p = NkT, \tag{63}$$

with the number density of the particles $N = nN_A/V$ and Boltzmann's constant $k = \mathcal{R}/N_A$. We can write the density of the gas using the mean molecular weight per particle μ as $\rho = N\mu m_H$, where m_H denotes the mass of the hydrogen atom. All individual particles have to be considered: pure neutral hydrogen gas (H_2) has $\mu = 2$, fully ionized hydrogen gas has $\mu = 0.5$. Using these terms, the ideal gas law reads

$$p = \frac{k}{\mu m_H}\rho T = \text{const.} \times \rho T = R\rho T, \tag{64}$$

with the material dependent gas constant R.

Equation of state

Using the specific heat capacity at constant volume, one can calculate the specific internal energy via

$$\epsilon = \int c_v dV.$$

For an ideal gas c_v is constant, and we find $\epsilon = c_v T$. One can show (see exercises) that the specific heat capacities are related via

$$c_p - c_v = R,$$

with $R = \frac{k}{\mu m_H}$.

From the first law of thermodynamics, we find the following relation for the specific entropy of an ideal gas

$$ds = c_v \frac{dT}{T} - R\frac{d\rho}{\rho}.$$

Equation of state

With the assumption that our gas does not exchange heat $\delta Q = 0$ (adiabatic), we find with $\gamma = c_p/c_v$

$$c_v \frac{dT}{T} = (c_p - c_v)\frac{d\rho}{\rho}$$

and by integrating

$$\frac{T}{T_0} = \left(\frac{\rho}{\rho_0}\right)^{\gamma-1} \qquad \text{and} \qquad \frac{p}{p_0} = \left(\frac{\rho}{\rho_0}\right)^{\gamma}.$$

These are the relations for a polytropic gas. The EOS can be written as

$$p = K\rho^{\gamma}, \tag{65}$$

with the adiabatic exponent γ. Compare to the isothermal case with $T = \text{const.}$

$$p = \text{const.} \times \rho. \tag{66}$$

EOS where the pressure depends only on the density are called *barotropic*.

Equation of state

A high number of different EOS are used in modern simulations depending on the modelled material. A simple EOS for water is the so-called liquid EOS

$$p = c_s^2(\rho - \rho_0), \tag{67}$$

with the sound speed c_s and the density at zero pressure ρ_0.

A generalized version of the liquid EOS is the Murnaghan EOS for isothermal compression, which is often used for solid bodies

$$p = \frac{K_0}{n}\left[\left(\frac{\rho}{\rho_0}\right)^n - 1\right], \tag{68}$$

where K_0 is the zero pressure bulk modulus and n a constant. For example, basalt has the following material parameters $K_0 = 19.3 \times 10^9\,\mathrm{Pa}$, $n = 5.5$, and $\rho_0 = 2.86 \times 10^3\,\mathrm{kg\,m^{-3}}$.

Modern EOS are often semi-analytic or even only tabulated.

Further reading, links and literature

- First chapters in Mihalas & Mihalas, Foundations of Radiation Hydrodynamics
  Derivation of Euler equations from microphysics of gas
- First chapter in Landau & Lifshitz, Fluid Mechanics

Numerical spatial derivatives

To convert the PDEs into a discrete form, we also need to transform the spatial derivatives in discrete form. A derivative is defined as follows

$$\frac{\partial u}{\partial x} = \lim_{\Delta x \to 0} \frac{u(x + \Delta x) - u(x)}{\Delta x}.$$

In our discrete form, we could take $x_{i+1} - x_i$ for Δx, but in a numerical calculation we cannot execute the limit $\Delta x \to 0$, since it would require an infinite number of grid points. We can instead write

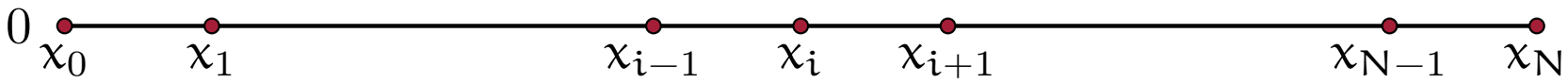
$$\left.\frac{\partial u}{\partial x}\right|_{i+1/2} = \frac{u_{i+1} - u_i}{x_{i+1} - x_i} + \mathcal{O}(\Delta x^2) \approx \frac{u_{i+1} - u_i}{x_{i+1} - x_i},$$

where we assume an equally spaced grid. Note, that this is an approximation for the derivative in between the gridpoints x_{i+1} and x_i. Normally, we do not want to know the derivative inbetween the grid cells, but precisely on a grid point x_i. This leads to the formula

$$\left.\frac{\partial u}{\partial x}\right|_{i} = \frac{u_{i+1} - u_{i-1}}{x_{i+1} - x_{i-1}} + \mathcal{O}(\Delta x^2) \approx \frac{u_{i+1} - u_{i-1}}{x_{i+1} - x_{i-1}}.$$

Numerical spatial derivatives

Our grid has $N+1$ data points x_i, $0 = 1, \ldots, N$, the grid points are located at $x_i = i\,\Delta x$ with the mesh size $\Delta x = \frac{1}{N}(x_N - x_0)$, the values of $u(x_i) = u_i$.



The forward difference is given by

$$\left.\frac{\partial u}{\partial x}\right|_i \approx \frac{u_{i+1} - u_i}{\Delta x},$$

the backward difference is given by

$$\left.\frac{\partial u}{\partial x}\right|_i \approx \frac{u_i - u_{i-1}}{\Delta x},$$

the central difference is given by

$$\left.\frac{\partial u}{\partial x}\right|_i \approx \frac{u_{i+1} - u_{i-1}}{2\Delta x}.$$

Numerical spatial derivatives

Analysis of truncation error for the three different difference schemes

Consider the Taylor expansion of $u(x)$ around x_i

$$u(x) = \sum_{n=0}^{\infty} \frac{1}{n!}(x - x_i)^n \left.\frac{\partial^n u}{\partial x^n}\right|_{x=x_i} .$$

Expansion for $u(x_{i+1}) = u_{i+1}$ reads

$$u(x_{i+1}) = u(x_i) + (x_{i+1} - x_i) \left.\frac{\partial u}{\partial x}\right|_{x_i} + \frac{1}{2}(x_{i+1} - x_i)^2 \left.\frac{\partial^2 u}{\partial x^2}\right|_{x_i} + \frac{1}{6}(x_{i+1} - x_i)^3 \left.\frac{\partial^3 u}{\partial x^3}\right|_{x_i} + \dots .$$

in other notation

$$u_{i+1} = u_i + \Delta x \left.\frac{\partial u}{\partial x}\right|_i + \frac{(\Delta x)^2}{2} \left.\frac{\partial^2 u}{\partial x^2}\right|_i + \frac{(\Delta x)^3}{6} \left.\frac{\partial^3 u}{\partial x^3}\right|_i + \dots \qquad (2)$$

Numerical spatial derivatives

And for $u(x_{i-1}) = u_{i-1}$

$$u_{i-1} = u_i - \Delta x \left.\frac{\partial u}{\partial x}\right|_i + \frac{(\Delta x)^2}{2} \left.\frac{\partial^2 u}{\partial x^2}\right|_i - \frac{(\Delta x)^3}{6} \left.\frac{\partial^3 u}{\partial x^3}\right|_i + \dots \tag{3}$$

Reorganising eq. (2) yields

$$\left.\frac{\partial u}{\partial x}\right|_i = \frac{u_{i+1} - u_i}{\Delta x} - \frac{\Delta x}{2} \left.\frac{\partial^2 u}{\partial x^2}\right|_i - \frac{(\Delta x)^2}{6} \left.\frac{\partial^3 u}{\partial x^3}\right|_i + \dots$$

with a truncation error $\mathcal{O}(\Delta x)$.
Reorganising eq. (3) yields

$$\left.\frac{\partial u}{\partial x}\right|_i = \frac{u_i - u_{i-1}}{\Delta x} + \frac{\Delta x}{2} \left.\frac{\partial^2 u}{\partial x^2}\right|_i - \frac{(\Delta x)^2}{6} \left.\frac{\partial^3 u}{\partial x^3}\right|_i + \dots$$

with a truncation error $\mathcal{O}(\Delta x)$.
Substracting eqs. (2) and (3) or adding the two equations above yields

$$\left.\frac{\partial u}{\partial x}\right|_i = \frac{u_{i+1} - u_{i-1}}{2\Delta x} - \frac{(\Delta x)^2}{6} \left.\frac{\partial^3 u}{\partial x^3}\right|_i + \dots$$

with a truncation error $\mathcal{O}(\Delta x^2)$.

Numerical 2nd order spatial derivatives

Adding eqs. (2) and (3) yields the following expression for the 2nd derivative of $u(x)$ at x_i

$$\left.\frac{\partial^2 u}{\partial x^2}\right|_i = \frac{u_{i+1} - 2u_i + u_{i-1}}{(\Delta x)^2} + \mathcal{O}(\Delta x^2).$$

An alternative derivation of this formula is as follows

$$\left.\frac{\partial^2 u}{\partial x^2}\right|_i = \left[\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial x}\right)\right]_i \approx \frac{\left.\frac{\partial u}{\partial x}\right|_{i+1/2} - \left.\frac{\partial u}{\partial x}\right|_{i-1/2}}{\Delta x} \approx \frac{\frac{u_{i+1}-u_i}{\Delta x} - \frac{u_i-u_{i-1}}{\Delta x}}{\Delta x}$$
$$= \frac{u_{i+1} - 2u_i + u_{i-1}}{(\Delta x)^2}.$$

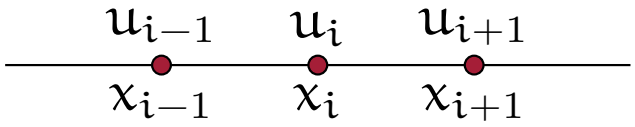
Dealing with variable coefficients, e.g. $f(x) = d(x)\frac{\partial u}{\partial x}$ (diffusive flux).

$$\left.\frac{\partial f}{\partial x}\right|_i \approx \frac{f_{i+1/2} - f_{i-1/2}}{\Delta x} = \frac{d_{i+1/2}\frac{u_{i+1}-u_i}{\Delta x} - d_{i-1/2}\frac{u_i-u_{i-1}}{\Delta x}}{\Delta x}$$
$$= \frac{d_{i+1/2}u_{i+1} - (d_{i+1/2} + d_{i-1/2})u_i + d_{i-1/2}u_{i-1}}{(\Delta x)^2}.$$

Advection schemes

Consider the 1D linear advection equation and the regular grid

$$u_t + a u_x = 0,$$



We discretise the PDE in time using an Euler step

$$\left.\frac{\partial u}{\partial t}\right|_i^t = \frac{u_i^{n+1} - u_i^n}{\Delta t}$$

and in space using the central difference

$$\left.\frac{\partial u}{\partial x}\right|_i^t = \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x},$$

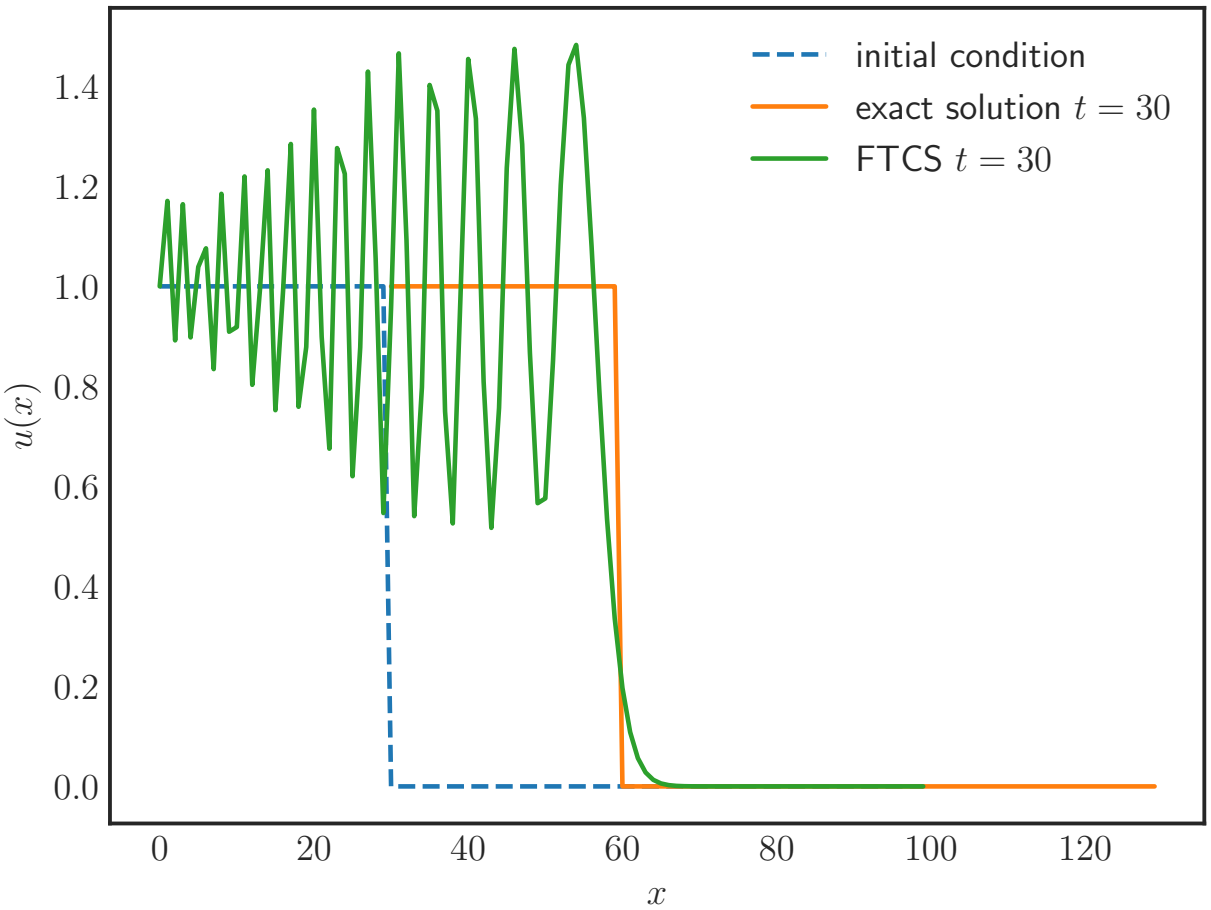
to obtain the following scheme

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

which is called the centered-differencing scheme or FTCS (forward time centered space).

Let's look how it performs!

Advection schemes



! not so good ... we will understand later why ... let's try different schemes!

Advection schemes

(i) Upwind Use the forward difference for the time derivative and the backward difference for the spatial derivative aka FTBS (forward time backward space)

$$\frac{1}{\Delta t}(u_i^{n+1} - u_i^n) + \frac{a}{\Delta x}(u_i^n - u_{i-1}^n) = 0,$$

re-arranging yields the explicit scheme

$$u_i^{n+1} = u_i^n - \sigma(u_i^n - u_{i-1}^n),$$

with $\sigma = a\frac{\Delta t}{\Delta x}$.

(ii) Downwind Use the forward difference for the time derivative and the forward difference for the spatial derivative aka FTFS (forward time forward space)

$$\frac{1}{\Delta t}(u_i^{n+1} - u_i^n) + \frac{a}{\Delta x}(u_{i+1}^n - u_i^n) = 0,$$

re-arranging yields the explicit scheme

$$u_i^{n+1} = u_i^n - \sigma(u_{i+1}^n - u_i^n).$$

Advection schemes

(iii) Leapfrog Use the center difference for the time derivative and the spatial derivative, but at different points in time

$$\frac{1}{2\Delta t}(u_i^{n+1} - u_i^{n-1}) + \frac{a}{2\Delta x}(u_{i+1}^n - u_{i-1}^n) = 0,$$

re-arranging yields the explicit scheme

$$u_i^{n+1} = u_i^{n-1} - \sigma(u_{i+1}^n - u_{i-1}^n).$$

(iv) Lax-Wendroff Consider the Taylor expansion of $u(x, t + \Delta t)$

$$u(x, t + \Delta t) = u(x, t) + \Delta t \frac{\partial u}{\partial t}(x, t) + \frac{(\Delta t)^2}{2} \frac{\partial^2 u}{\partial t^2}(x, t) + \mathcal{O}(\Delta t^3).$$

From $u_t = -a u_x$ follows $u_{tt} = a^2 u_{xx}$ (using $u_{xt} = u_{tx}$ and $a = \text{const.}$), which yields after inserting into the above expansion

$$u(x, t + \Delta t) = u(x, t) - \Delta t a \frac{\partial u}{\partial x}(x, t) + \frac{(\Delta t)^2}{2} a^2 \frac{\partial^2 u}{\partial x^2}(x, t) + \mathcal{O}(\Delta t^3).$$

Advection schemes

Now we use forward differencing in time and centered differencing in space to obtain the so-called Lax-Wendroff explicit scheme

$$u_i^{n+1} = u_i^n - \frac{\sigma}{2}(u_{i+1}^n - u_{i-1}^n) + \frac{\sigma^2}{2}(u_{i+1}^n - 2u_i^n + u_{i-1}^n) + \mathcal{O}(\Delta t^3) + \mathcal{O}(\Delta t \Delta x^2). \tag{5}$$

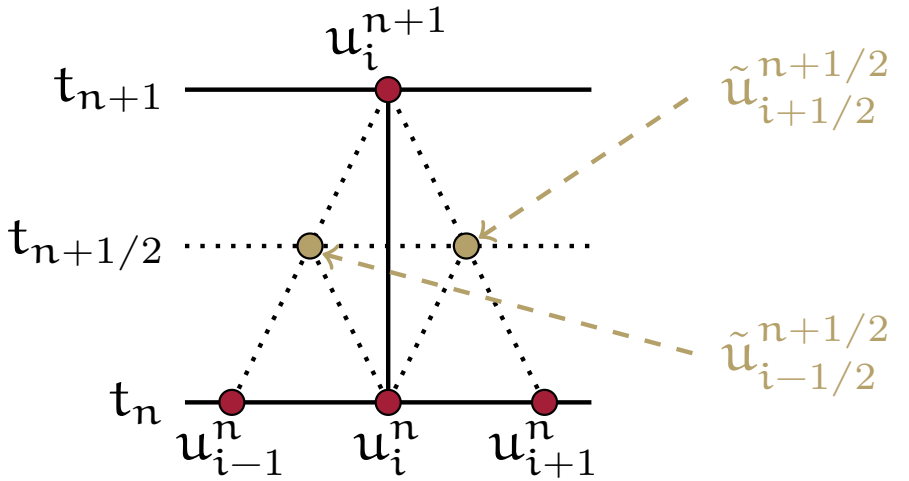
In principle this is equivalent to the following procedure: The timestep is subdivided into two substeps: a predictor step for $\tilde{u}_{i+1/2}^{n+1/2}$

$$\tilde{u}_{i+1/2}^{n+1/2} = \frac{1}{2}(u_i^n + u_{i+1}^n) - \frac{\sigma}{2}(u_{i+1}^n - u_i^n)$$

followed by a corrector step

$$u_i^{n+1} = u_i^n - \sigma(\tilde{u}_{i+1/2}^{n+1/2} - \tilde{u}_{i-1/2}^{n+1/2})$$

which yields eventually eq. (5).



Advection schemes

(v) Beam-Warming The derivation of the Beam-Warming scheme (1976) makes also use of the trick to replace the time derivative by the spatial derivative (see LW derivation above). The Taylor expansion of $u(x, t + \Delta t)$ then reads

$$u(x, t + \Delta t) = u(x, t) + a\Delta t \frac{\partial u}{\partial x}(x, t) + a^2 \frac{(\Delta t)^2}{2} \frac{\partial^2 u}{\partial x^2}(x, t) + \mathcal{O}(\Delta t^3).$$

We replace now the first spatial derivative with a 2nd order backward difference

$$\left.\frac{\partial u}{\partial x}\right|_{i,n} = \frac{3u_i^n - 4u_{i-1}^n + u_{i-2}^n}{2\Delta x} + \mathcal{O}(\Delta x^2),$$

and the second spatial derivative with a centered difference around x_{i-1}

$$u_i^{n+1} = u_i^n - \frac{\sigma}{2}(3u_i^n - 4u_{i-1}^n + u_{i-2}^n) + \frac{\sigma^2}{2}(u_i^n - 2u_{i-1}^n + u_{i-2}^n) + \mathcal{O}(\Delta t^2 + \mathcal{O}(\Delta t \Delta x^2),$$

for $a > 0$. For $a < 0$, the indices switch from $(i, i-1, i-2)$ to $(i, i+1, i+2)$.

Advection schemes

(vi) Lax-Friedrich To derive the Lax-Friedrich scheme (1954), one starts with the centered difference

$$u_i^{n+1} = u_i^n + \frac{\sigma}{2}(u_{i-1}^n - u_{i+1}^n),$$

and replaces u_i^n by the average of u at its neighbouring grid points

$$u_i^{n+1} = \frac{1}{2}(u_{i+1}^n + u_{i-1}^n) + \frac{\sigma}{2}(u_{i-1}^n - u_{i+1}^n),$$

The methods (i)-(vi) are all *explicit* methods and can be implemented relatively easy for the 1D advection equation (see exercises).

On the next slides we will take a quick look on implicit schemes.

Advection schemes

In general a finite difference scheme for an equation with the linear operator L

$$u_t = L[u]$$

reads

$$u_i^{n+1} = u_i^n + L[u_i^n]\Delta t = u_i^n + \sum_{|k|\leqslant K} c_k u_{i+k}^n \Delta t,$$

with the integer number K that depends on the order of the spatial derivative. The coefficients c_k do not depend on u or i.

We can rewrite the finite difference scheme in a more general form as

$$u^{n+1} = u^n + L[u^n](1-\beta)\Delta t + \beta L[u^{n+1}]\Delta t.$$

Now, β is a parameter with $0 \leqslant \beta \leqslant 1$, which denotes the grade of *implicity*.
For $\beta = 0$, one can write down u^{n+1} as a function of u^n only.
For $\beta = 1$, the scheme is full implicit.
For $\beta = 1/2$, 2nd order in time (centered), semi-implicit scheme.

Advection schemes

For all schemes with $\beta > 0$, the values of u_i are coupled, and one has to deal with matrix equations of the following form

$$\mathbf{u}^{n+1} = A\mathbf{u}^n,$$

where $\mathbf{u}$ denotes the vector of the u_i. In the 1D case, A is tridiagonal.

Let's look at two examples:

(i) Backward Time Centered Space The BTCS scheme is an analog to the FTCS scheme with the spatial derivative now taken at the new time t^{n+1}, making the scheme implicit

$$u_i^{n+1} = u_i^n - \frac{\sigma}{2}(u_{i+1}^{n+1} - u_{i-1}^{n+1}).$$

For each timestep, one has to solve a tridiagonal matrix, i.e. solve the system of the following linear equations

$$u_i^{n+1} - \frac{\sigma}{2}u_{i-1}^{n+1} + \frac{\sigma}{2}u_{i+1}^{n+1} = u_i^n.$$

(ii) Crank-Nicolson The CN scheme follows from the FTCS scheme

$$u_i^{n+1} = u_i^n - \frac{\sigma}{2}(u_{i+1}^n - u_{i-1}^n),$$

by writing the spatial derivative on the rhs centered in time

$$u_i^{n+1} = u_i^n - \frac{\sigma}{4}(u_{i+1}^n - u_{i-1}^n + u_{i+1}^{n+1} - u_{i-1}^{n+1}),$$

which yields a semi-implicit scheme with $\beta = 0.5$.

The standard use-case for the CN scheme is the heat equation (linear diffusion/parabolic PDE) $q_t = q_{xx}$ and not a hyperbolic PDE. Here, q denotes the temperature in the case of thermal conduction. One can write a possible explicit discretisation

$$q_i^{n+1} = q_i^n + \frac{\Delta t}{\Delta x^2}(q_{i-1}^n - 2q_i^n + q_{i+1}^n),$$

or an implicit version with CN

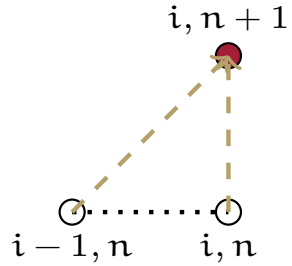
$$q_i^{n+1} = q_i^n + \frac{\Delta t}{2\Delta x^2}((q_{i-1}^n - 2q_i^n + q_{i+1^n}) + (q_{i-1}^{n+1} - 2q_i^{n+1} + q_{i+1}^{n+1})).$$

Advection schemes for linear advection

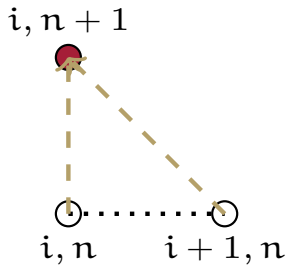
Name	Finite Differences equation
Upwind	$u_i^{n+1} = u_i^n - \sigma(u_i^n - u_{i-1}^n)$
Downwind	$u_i^{n+1} = u_i^n - \sigma(u_{i+1}^n - u_i^n)$
Centered	$u_i^{n+1} = u_i^n - \frac{\sigma}{2}(u_{i+1}^n - u_{i-1}^n)$
Lax-Friedrichs	$u_i^{n+1} = \frac{1}{2}(u_{i-1}^n + u_{i+1}^n) - \frac{\sigma}{2}(u_{i+1}^n - u_{i-1}^n)$
Leapfrog	$u_i^{n+1} = u_i^{n-1} + \sigma(u_{i-1}^n - u_{i+1}^n)$
Lax-Wendroff	$u_i^{n+1} = u_i^n - \frac{\sigma}{2}(u_{i+1}^n - u_{i-1}^n) + \frac{\sigma^2}{2}(u_{i+1}^n - 2u_i^n + u_{i-1}^n)$
Beam-Warming	$u_i^{n+1} = u_i^n - \frac{\sigma}{2}(3u_i^n - 4u_{i-1}^n + u_{i-2}^n) + \frac{\sigma^2}{2}(u_i^n - 2u_{i-1}^n + u_{i-2}^n)$

$\sigma = a\Delta t/\Delta x$, $a > 0$.

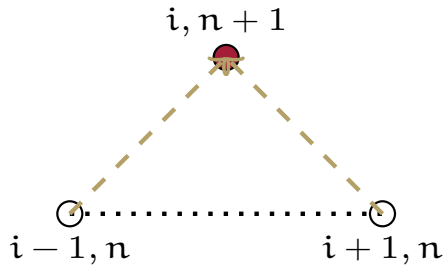
Advection schemes for linear advection



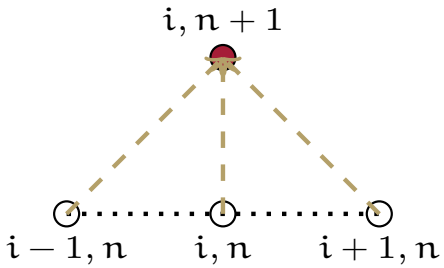
Upwind



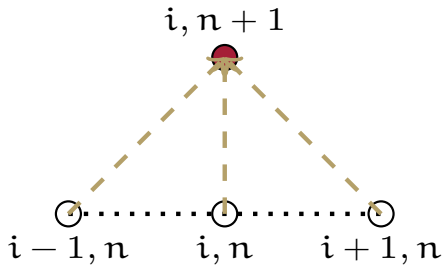
Downwind



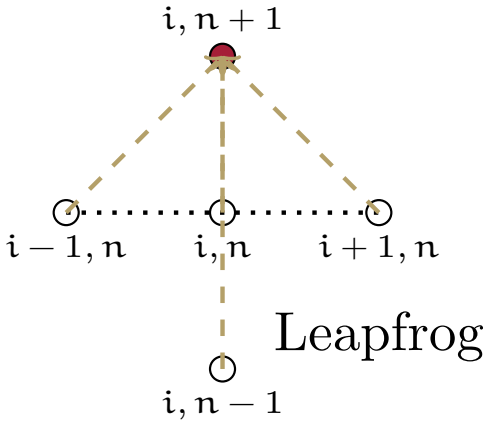
Centered



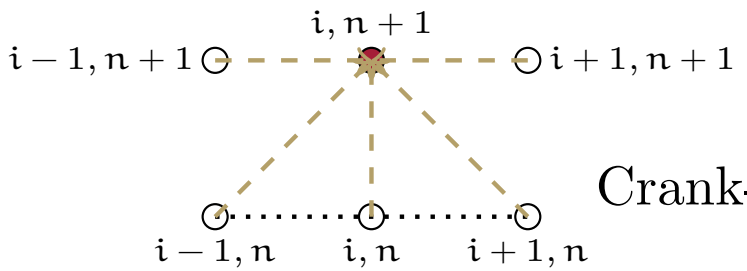
Lax-Friedrichs



Lax-Wendroff



Leapfrog



Crank-Nicolson

Von Neumann stability analysis

During the Manhatten project people found the unstable behaviour of the FTCS scheme. John von Neumann developed in this time in Los Alamos a method to analyse the stability of numerical advection schemes. His research was confidential and the method was published first by Crank and Nicolson in 1947. On the following slides we will take a look at this technique.

Since the typical instabilities in the advection scheme show oscillations, von Neumann came up with the idea to expand the solution in a Fourier series and look at the time evolution of its coefficients.

Let u_j^n represent an arbitrary grid function with grid index j, so that $x_j = j\Delta x$. It can be expressed as a Fourier series

$$u_j^n = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \tilde{u}(k) e^{ikx_j} dk = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \tilde{u}(k) e^{ikj\Delta x} dk.$$

Von Neumann stability analysis

If we apply a linear finite difference method to u_j^n, we typically obtain an expression of the form

$$u_j^{n+1} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \tilde{u}(k) g(k, \Delta x, \Delta t) e^{ikj\Delta x} dk,$$

since the advection equation $u_t + a u_x = 0$ moves the function u with velocity a, which translates in Fourier space to a phase rotation only

$$u(x,t) = u(x - at) \qquad \longleftrightarrow \qquad \tilde{u}(k,t) = \tilde{u}(k) e^{-iakt}.$$

Hence, it is

$$\tilde{u}^{n+1}(k) = g(k, \Delta x, \Delta t) \tilde{u}^n(k), \tag{12}$$

where $g(k, \Delta x, \Delta t)$ is called the *amplification factor* for wave number k. In the following, we choose the 2- or L_2-norm, since it holds (Parseval's relation)

$$\|u^n\|_2 = \|\tilde{u}^n\|_2 .$$

with $\|u^n\|_2 = (\Delta x \sum_{j=-\infty}^{\infty} |u_j^n|^2)^{1/2}$ and $\|\tilde{u}^n\|_2 = (\int_{-\infty}^{\infty} |\tilde{u}^n|^2 dk)^{1/2}$.

Von Neumann stability analysis

Therefore, it is sufficient to show that the 2-norm of $\tilde{u}$ remains bounded. This is easier than looking at the u^n, since the time evolution of all elements u^n are coupled by the difference equations, but each element of $\tilde{u}^n$ satisfies eq. (12) that is decoupled from all other wave numbers. Hence, we have only to consider a single wave number k and data of the form

$$u_j^n = e^{ikj\Delta x},$$

and from this we can compute the amplification factor $g(k, \Delta x, \Delta t)$.

In Fourier space, the true operator $L_e[\cdot]$ is just the complex number $L_e = e^{-ia k \Delta t}$.

Finally, requiring that $|g(k, \Delta x, \Delta t)| \leqslant 1$ for all k gives a sufficient condition for stability.

In fact it suffices to have $|g(k, \Delta x, \Delta t)| \leqslant 1 + \alpha \Delta t$ for some constant α independent of k.

But let's do some examples!

Von Neumann stability analysis

Example 1: Stability analysis of the upwind scheme.

The upwind scheme reads

$$\begin{aligned} u_j^{n+1} &= u_j^n - \sigma(u_j^n - u_{j-1}^n), \\ &= (1-\sigma)u_j^n + \sigma u_{j-1}^n, \end{aligned}$$

with $\sigma = a\Delta t/\Delta x$.

Now, we use $u_j^n = e^{ikj\Delta x}$ as our test function and get

$$\begin{aligned} u_j^{n+1} &= (1-\sigma)e^{ikj\Delta x} + \sigma e^{ik(j-1)\Delta x} \\ &= \left[(1-\sigma) + \sigma e^{-ik\Delta x}\right] e^{ikj\Delta x} \\ &= g(k, \Delta x, \Delta t) u_j^n, \end{aligned}$$

with the amplification factor

$$g(k, \Delta x, \Delta t) = (1-\sigma) + \sigma e^{-ik\Delta x}.$$

As k varies, g lies on a circle of radius σ in the complex plan, centered on the real axis at $1-\sigma$. So, in order to obtain $|g| \leqslant 1$ for all k, it has to be $0 \leqslant \sigma \leqslant 1$, or $a\Delta t \leqslant \Delta x$. Let's try this!

Von Neumann stability analysis

Example 2: Stability analysis of the FTCS scheme.

The FTCS scheme reads

$$u_j^{n+1} = u_j^n - \frac{\sigma}{2}\left(u_{j+1}^n - u_{j-1}^n\right),$$

with $\sigma = a\Delta t/\Delta x$.

Now, we use $u_j^n = e^{ikj\Delta x}$ as our test function and get

$$\begin{aligned} u_j^{n+1} &= e^{ikj\Delta x} - \frac{\sigma}{2}\left(e^{ik(j+1)\Delta x} - e^{ik(j-1)\Delta x}\right) \\ &= e^{ikj\Delta x}\left[1 + \frac{\sigma}{2}e^{ik\Delta x} - \frac{\sigma}{2}e^{-ik\Delta x}\right] = u_j^n\left[1 - \frac{\sigma}{2}\left(e^{ik\Delta x} - e^{-ik\Delta x}\right)\right] \\ &= u_j^n\left[1 - i\sigma\sin k\Delta x\right]. \end{aligned}$$

Hence, $g = 1 - i\sigma\sin k\Delta x$. We compute the norm $\|g\| = \sqrt{g^* g} = \sqrt{(\mathrm{Re}(g))^2 + (\mathrm{Im}(g))^2}$ and find

$$\|g\| = \sqrt{1 + \sigma^2\sin^2 k\Delta x} \geqslant 1,$$

which means the FTCS is unstable for all choices of Δx and Δt.

Finite difference

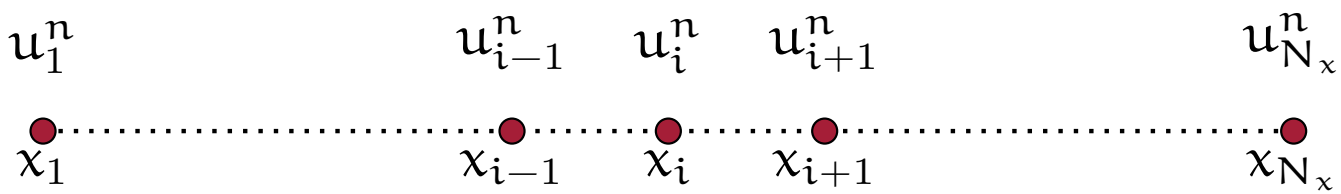
So far, we have discretised the time and the spatial coordinates with a finite grid, e.g.

$$t_n \in \{t_1, \ldots, t_{N_t}\} \quad \text{and} \quad x_i \in \{x_1, \ldots, x_{N_x}\},$$

with N_t grid points for the time and N_x grid points for the spatial coordinate. Then, we have looked at the numerical solution of the linear advection equation $u_t + a u_x = 0$, where a denotes a constant velocity. The values of the solution $u(x,t)$ are then given at the discrete points in time and space, c.f.

$$u(x_i, t_n) = u_i^n.$$

our grid looks like the following at time $t = t_n$



The time evolution of u was in this sense the motion of u over the grid.

Finite difference vs. finite volumes

Now, we want to change the perspective and re-focus on our main goal: the numerical solution of the equations of hydrodynamics. The equations of hydrodynamics are *conservation* equations, which is a property of fundamental importance: If we have systematic errors in the conservation of conserved quantities, our numerical solutions will be clearly unphysical. We have to make sure that the conserved quantities are really conserved in our numerical scheme.

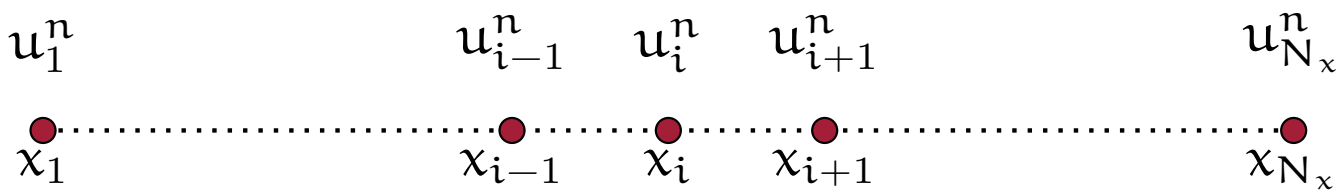
It is very much preferable that our schemes to solve the hydrodynamic equations are formulated in a *numerically flux conserving form*, or simply speaking: everything that leaves our grid cell has to enter another one.

We will now subdivide the spatial domain into finite intervals (the *finite volumes* aka grid cells) and keep track of an approximation to the integral of $\mathfrak{u}$ over each volume. In each time step we update these values in the grid cell using approximations to the flux through the endpoints of the intervals.

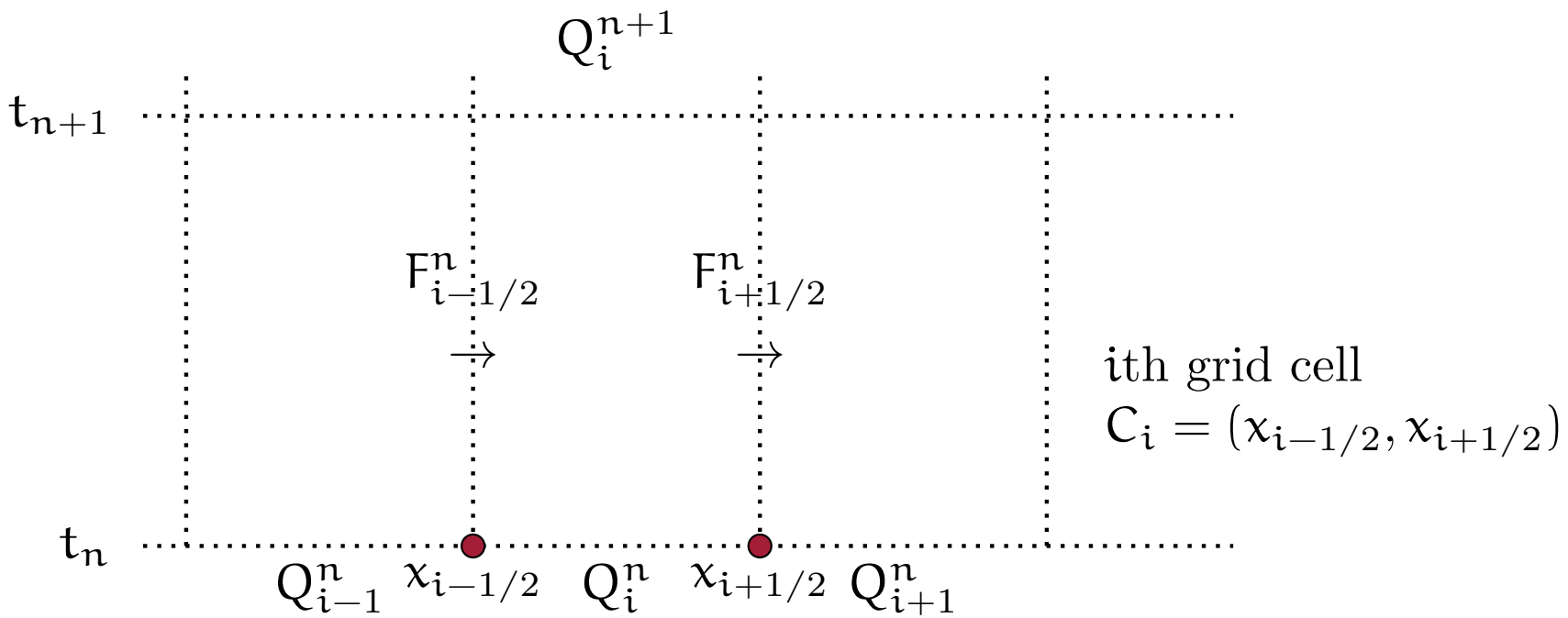
The main new goal will be to calculate the fluxes in a way that the flux entering a given volume is identical to that leaving the adjacent volume.

Finite volume

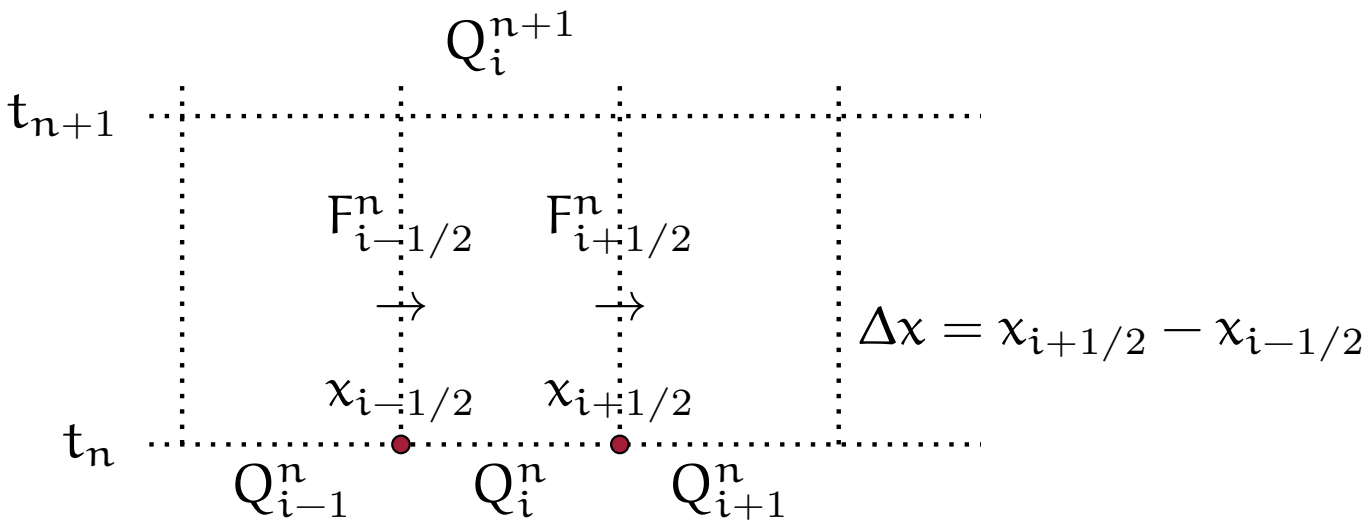
In principle, we leave this picture



for this one



Finite volume

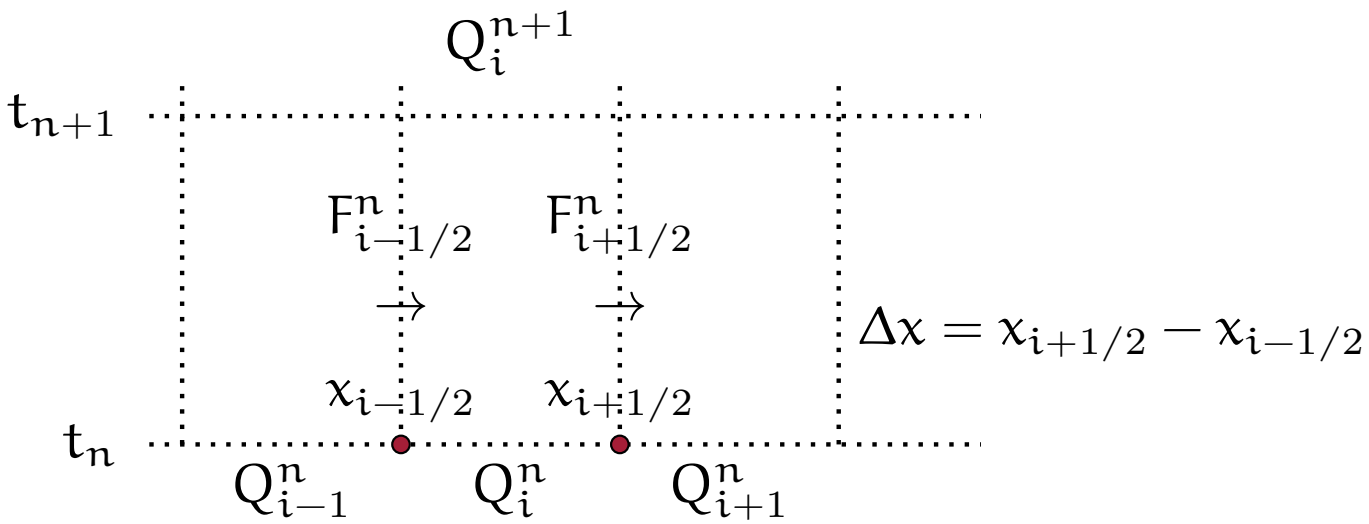


We used to have grid points x_i, and now we create cells out of our grid. The grid points will become now the centers of cells located at x_i, and we have *cell interfaces* or *cell edges* located at

$$x_{i+1/2} = \frac{x_i + x_{i+1}}{2}.$$

If we have a grid of N cell centers, we will get $N-1$ cell interfaces. The left boundary (cell no. 1) and the right boundary (cell no. N) need also a cell interface at $i = 1/2$ and $i = N + 1/2$. We set these boundary interfaces at $x_{1/2} = x_1 - \frac{1}{2}(x_2 - x_1)$ and $x_{N+1/2} = x_N + \frac{1}{2}(x_N - x_{N-1})$, giving a total of $N+1$ interfaces, and cell spacing $\Delta x = x_{i+1/2} - x_{i-1/2}$.

Finite volume



The value Q_i^n will approximate the average value over the ith interval at time t_n

$$Q_i^n \approx \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} q(x, t_n) dx \equiv \frac{1}{\Delta x} \int_{C_i} q(x, t_n) dx.$$

The conservation law $q_t + f(q)_x = 0$ in integral form reads

$$\frac{d}{dt} \int_{x_1}^{x_2} q(x, t) dx = f(q(x_1, t)) - f(q(x_2, t)),$$

or for our ith cell

$$\frac{d}{dt}\int_{C_i} q(x,t)dx = f(q(x_{i-1/2},t)) - f(q(x_{i+1/2},t)), \tag{1}$$

We will use this expression to develop an explicit time marching algorithm. The averaged cell values Q_i^n at time t_n are known, we want to approximate Q_i^{n+1} at the next time t_{n+1} after the time step of length $\Delta t = t_{n+1} - t_n$. Integrating eq. (1) in time from t_n to t_{n+1} yields

$$\int_{C_1} q(x,t_{n+1})dx - \int_{C_1} q(x,t_n)dx = \int_{t_n}^{t_{n+1}} f(q(x_{i-1/2},t))dt - \int_{t_n}^{t_{n+1}} f(q(x_{i+1/2},t))dt.$$

Re-arranging and dividing by Δx gives

$$\frac{1}{\Delta x}\int_{C_1} q(x,t_{n+1}) = \frac{1}{\Delta x}\int_{C_1} q(x,t_n) + \frac{1}{\Delta x}\left[\int_{t_n}^{t_{n+1}} f(q(x_{i-1/2},t))dt - \int_{t_n}^{t_{n+1}} f(q(x_{i+1/2},t))dt\right] \tag{2}$$

Finite volume

Equation (2) tells us now how to update the cell average of q in one time step. However, we cannot evaluate the time integrals on the rhs of this equation exactly, since $q(x_{i\pm1/2}, t)$ varies with time along the cell edge. However, it gives the hint to search for numerical schemes in the form of

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x}\left(F_{i+1/2}^n - F_{i-1/2}^n\right), \tag{3}$$

where $F_{i-1/2}^n$ is some approximation to the average flux along the interface at $x_{i-1/2}$

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

The idea is by approximating this average flux based on the known values Q^n, we will obtain a fully discrete method.

Finite volume

Since information propagates with finite speed in hyperbolic problems, our first approach is that we obtain $F^n_{i-1/2}$ based only on the values Q^n_{i-1} and Q^n_i, the cell averages on either side of the interface, so

$$F^n_{i-1/2} = \mathcal{F}(Q^n_{i-1}, Q^n_i),$$

where $\mathcal{F}$ is a *numerical flux function*. We get the general formula

$$Q^{n+1}_i = Q^n_i - \frac{\Delta t}{\Delta x}\left(\mathcal{F}(Q^n_i, Q^n_{i+1}) - \mathcal{F}(Q^n_{i-1}, Q^n_i)\right). \tag{4}$$

The specific scheme is then obtained when the formula $\mathcal{F}$ is chosen, but all of these schemes are explicit methods with a three point stencil, and they are in conservation form. If we sum all $\Delta x Q^{n+1}_i$ from eq. (3) over any set of cells, we get

$$\Delta x \sum_{i=1}^{N} Q^{n+1}_i = \Delta x \sum_{i=1}^{N} Q^n_i - \Delta t \left(F^n_{1/2} - F^n_{N+1/2}\right)$$

Only the fluxes at the boundaries remain, all other fluxes cancel out, hence we have exact conservation over the full domain except at the boundaries.

Finite volume - finite difference

If one re-arranges eq. (3) into

$$\frac{Q_i^{n+1} - Q_i^n}{\Delta t} + \frac{F_{i+1/2}^n - F_{i-1/2}^n}{\Delta x} = 0,$$

one notices that the scheme can be equally well viewed as a finite difference approximation to

$$q_t + f(q)_x = 0.$$

However, now we know how to formulate our advection problem in flux-conserving form.

In the case of hydrodynamics, the equations we are interested in are not like our test equation for linear advection $q_t + aq_x = 0$, but rather of the form

$$q_t(x,t) + [u(x)q(x,t)]_x = 0.$$

The new question is now: how do we deal with the non-constant $u(x)$, especially how do we construct the fluxes $F_{i\pm1/2}^n$ from Q_i^n and the velocity field $u(x)$?

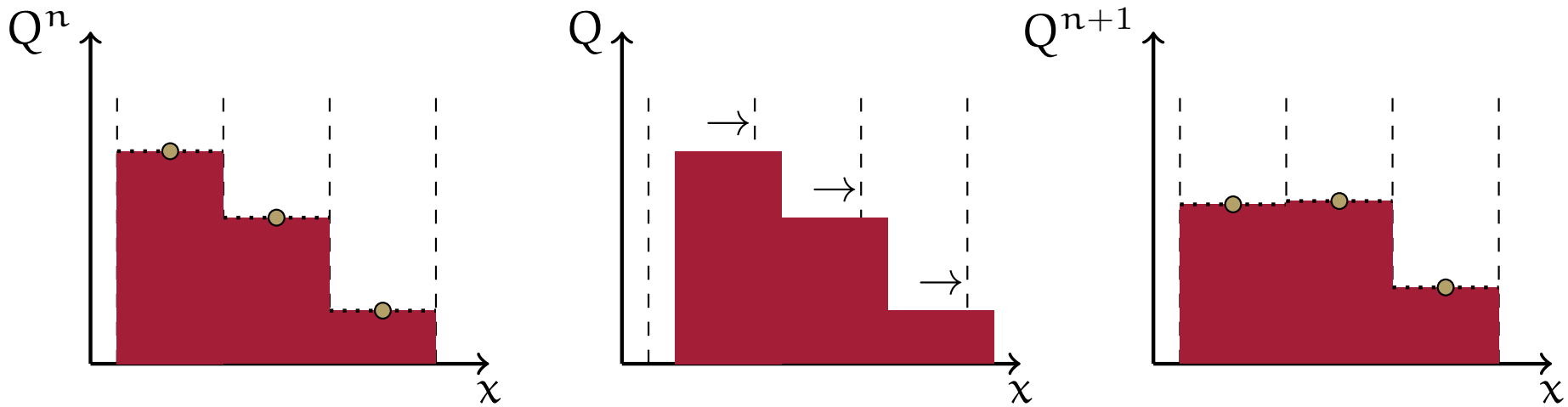
Donor-cell advection

The simplest flux conserving scheme is the donor-cell scheme. Here, the average interface state is simply

$$\tilde{q}^n = \begin{cases} Q_i^n & \text{for } u_{i+1/2} > 0 \\ Q_{i+1}^n & \text{for } u_{i+1/2} < 0, \end{cases}$$

which means that the fluxes at the interfaces are given by

$$F_{i+1/2}^n = \begin{cases} u_{i+1/2} Q_i^n & \text{for } u_{i+1/2} > 0 \\ u_{i+1/2} Q_{i+1}^n & \text{for } u_{i+1/2} < 0. \end{cases}$$



Reconstruct-evolve-average

The donor cell scheme assumes that at the beginning of each time step the state within each cell is constant throughout the whole cell, and the state at the grid center is the same as the state of the interfaces. This is the simplest case for a more general algorithm, the so-called REA algorithm. Moreover, it includes a so-called *subgrid model*, since we assume something for our material at a spatial scale that is smaller than the resolution of our grid. The basic steps of the more general algorithm REA

1. Reconstruct a piecewise polynomial function $\tilde{q}^n(x, t_n)$ defined for all x from the cell averages Q_i^n. In the simplest case this is a piecewise constant function that takes the value Q_i^n in the ith grid cell, i.e.
$$\tilde{q}^n(x, t_n) = Q_i^n.$$
2. Evolve the hyperbolic equations exactly (or approximately) with this initial data to obtain $\tilde{q}^n(x, t_{n+1})$ at the time $t_n + \Delta t$.
3. Average this function over each grid cell to obtain new cell averages
$$Q_i^{n+1} = \frac{1}{\Delta x} \int_{C_1} \tilde{q}^n(x, t_{n+1}) dx.$$

Piecewise linear scheme

In the following we will try the assumption that the state is assumed to be *piecewise linear*.

From the cell averages Q_i^n we construct a piecewise linear function of the form

$$\tilde{q}^n(x, t_n) = Q_i^n + \sigma_i^n(x - x_i) \quad \text{for } x_{i-1/2} \leqslant x < x_{i+1/2},$$

where

$$x_i = \frac{1}{2}\left(x_{i-1/2} + x_{i+1/2}\right) = x_{i-1/2} + \frac{\Delta x}{2}$$

denotes the center of the ith grid cell and σ_i^n is the slope on the ith cell. The function $\tilde{q}^n$ is defined in a way that its value at the cell center x_i is Q_i^n, and the average value of $\tilde{q}^n$ over cell C_i is Q_i^n regardless of the slope σ_i^n. Hence, the reconstructed function has the cell average Q_i^n. This is important in developing conservation methods for conservation laws.

Piecewise linear scheme

For the scalar advection equation $q_t + \bar{u} q_x = 0$, we can easily solve the equation with this data, and compute the new cell averages as required in step 3 of the REA algorithm. The solution is

$$\tilde{q}^n(x, t_{n+1}) = \tilde{q}^n(x - \bar{u}\Delta t, t_n).$$

Assuming $\bar{u} > 0$, we can calculate the fluxes $F^n_{i\pm 1/2}$ and using the conservative form

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

one obtains (see, e.g., LeVeque chapter 6.4 or some slides below)

$$Q_i^{n+1} = Q_i^n - \frac{\bar{u}\Delta t}{\Delta x}\left(Q_i^n - Q_{i-1}^n\right) - \frac{\bar{u}\Delta t}{2\Delta x}\left(\Delta x - \bar{u}\Delta t\right)\left(\sigma_i^n - \sigma_{i-1}^n\right), \qquad (5)$$

which is essentially the upwind method with a correction term that depends on the slopes. The fluxes $F_{i\pm 1/2}$ can be regarded as the averaged fluxes of the time interval $[t_n, t_{n+1}]$, i.e. "$F^{n+1/2}_{i\pm 1/2}$".

Piecewise linear scheme

We have now derived the higher-order version of the donor-cell advection scheme, and if the slopes are chosen to be zero, eq. (5) is identical to the donor-cell scheme.

We can now choose from different slopes σ_i^n of the linear function, and the three obvious slopes are

centered slope	$\sigma_i^n = \frac{Q_{i+1}^n - Q_{i-1}^n}{2\Delta x}$	Fromm's method
upwind slope	$\sigma_i^n = \frac{Q_i^n - Q_{i-1}^n}{\Delta x}$	Beam-Warming method
downwind slope	$\sigma_i^n = \frac{Q_{i+1}^n - Q_i^n}{\Delta x}$	Lax-Wendroff method

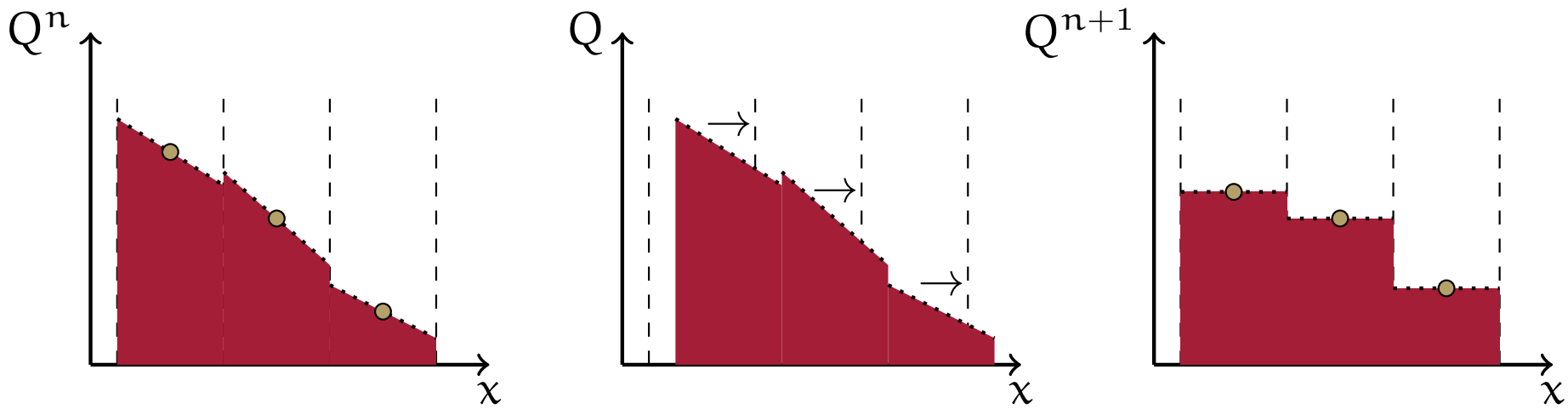
If we insert Fromm's method in eq. (5), we obtain the following explicit update of the state

$$Q_i^{n+1} = Q_i^n - \frac{\bar{u}\Delta t}{4\Delta x}\left(Q_{i+1}^n + 3Q_i^n - 5Q_{i-1}^n + Q_{i-2}^n\right) - \frac{\bar{u}^2\Delta t^2}{4\Delta x^2}\left(Q_{i+1}^n - Q_i^n - Q_{i-1}^n + Q_{i-2}^n\right)$$

Piecewise linear scheme

Sketch of the piecewise linear advection scheme that uses the Lax-Wendroff slope

$$\sigma_i^n = \frac{Q_{i+1}^n - Q_i^n}{\Delta x}$$

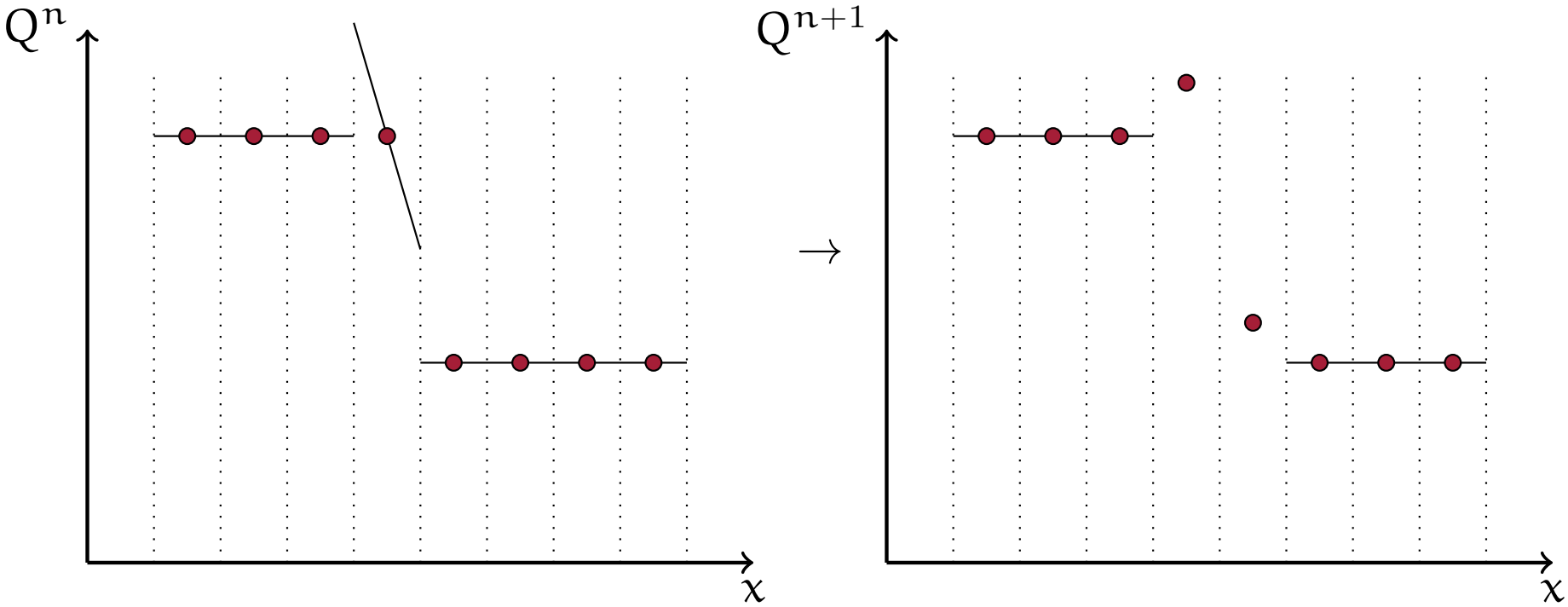


The total scheme for $\bar{u} > 0$ reads

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

Piecewise linear schemes

With our newly gained knowledge from the last slides, we can understand why higher-order schemes produce oscillations near discontinuities, here the Lax-Wendroff slope.



We see an example of a so-called overshooting. Fromm's method can also lead to undershooting, and we see oscillations in front and behind the discontinuity. Lax-Wendroff leads to oscillations behind the discontinuity.

Total variation

A successful method to prevent overshooting is the use of a *slope limiter*. A slope limiter is an additional non-linear condition that modifies σ_i^n if it is necessary to prevent overshooting. But before we define our first slope limiter, we look at the concept of *total variation* (TV), which we will use to measure the appearance of oscillations.

The total variation is defined as

$$TV(Q) = \sum_{i=1}^{N} |Q_i - Q_{i-1}|,$$

with the left and right boundaries $i = 1$ and $i = N$ of the grid. For a monotonically increasing function Q, it is $TV(Q) = |Q_1 - Q_N|$. If Q_1 and Q_N are constants, then, as long as Q is monotonic, $TV(Q)$ will remain constant. As soon as the Q_i will develop minima and maxima, $TV(Q)$ will increase. Therefore, by measuring $TV(Q)$ we see the formation of oscillations in the solution.

Total variation diminishing schemes

A numerical scheme is called a *total variation diminishing* (TVD) scheme if

$$\mathsf{TV}(\mathsf{Q}^{n+1}) \leqslant \mathsf{TV}(\mathsf{Q}^{n}).$$

Obviously such a scheme will not develop oscillations near a discontinuity, since a discontinuity is a monotonically increasing or decreasing function and a TVD scheme will not increase the TV. Hence, we will search for a numerical scheme that keeps the TV constant, since it appears to be a highly desirable property.

On the next slides we will take a look at some slope limiters that guarantee TVD. If we set all slopes to zero, we end up with the upwind scheme, which is only first order. So, we have to think about something more sophisticated. In principle, we want to prevent over- and undershooting at discontinuities, so we switch to upwind there, and use the slopes and hence second order where it is safe. These ideas were introduced first by van Leer in the seventies.

Slope limiters

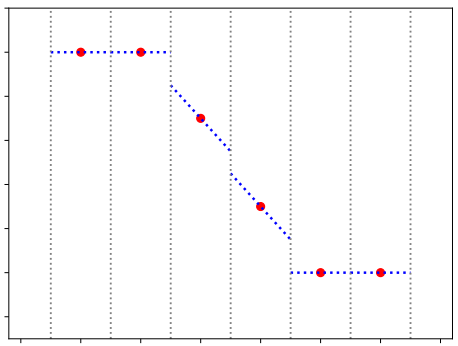
The *minmod slope limiter*

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

where the minmod function is defined as follows

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

This scheme chooses the smallest of two slopes if both have the same sign and zero otherwise.



Slope limiters

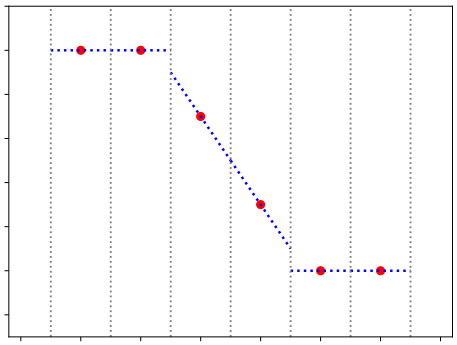
The *superbee slope limiter* (Roe 1986)

$$\sigma_i^n = \text{maxmod}(s_i^{(1)}, s_i^{(2)}),$$

with

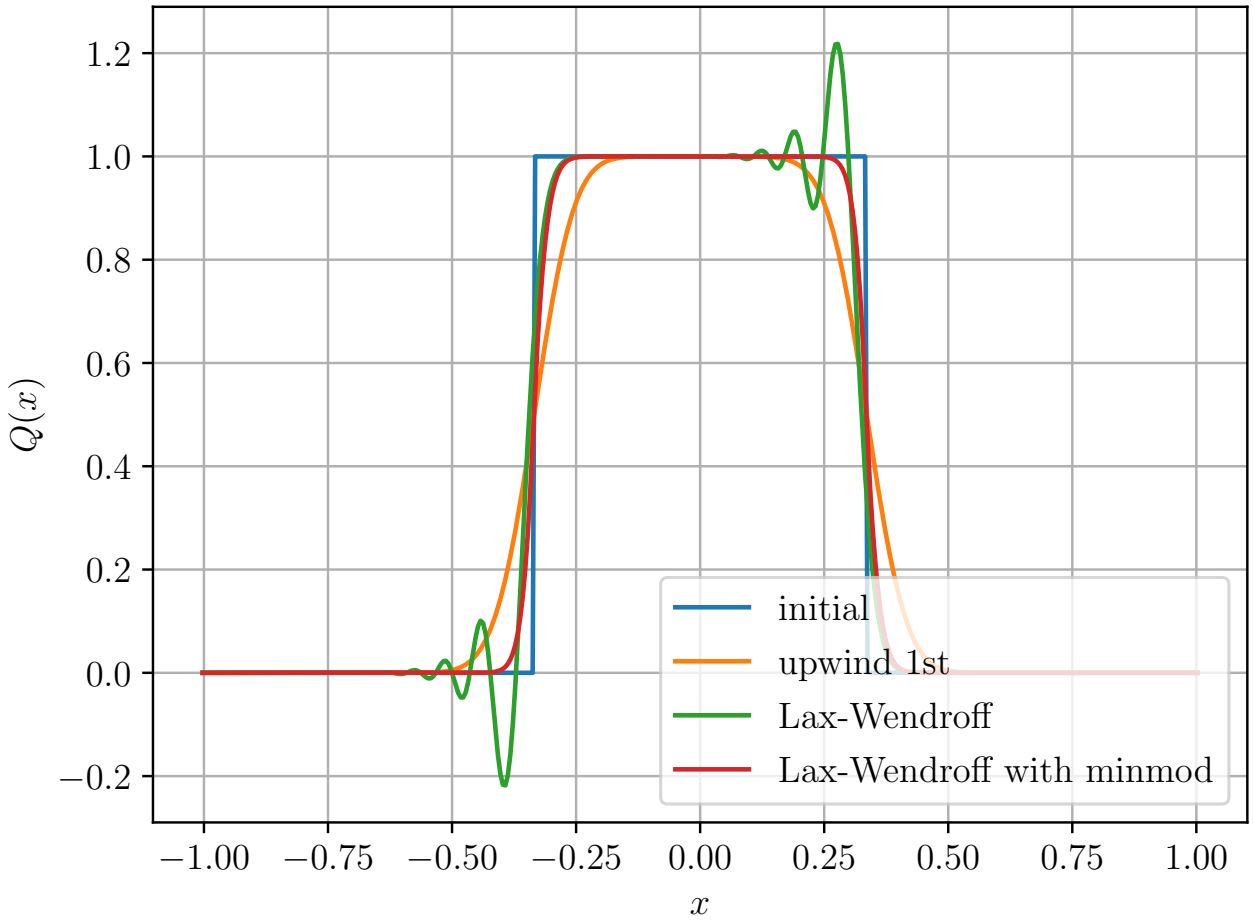
$$s_i^{(1)} = \text{minmod}\left(\frac{Q_{i+1}^n - Q_i^n}{\Delta x}, 2\frac{Q_i^n - Q_{i-1}^n}{\Delta x}\right),$$

$$s_i^{(2)} = \text{minmod}\left(2\frac{Q_{i+1}^n - Q_i^n}{\Delta x}, \frac{Q_i^n - Q_{i-1}^n}{\Delta x}\right).$$



Let's see how the slope limiter improves our solution!

minmod



see also the exercises...

Flux formulation with piecewise linear reconstruction

The slope limiter methods described on the last slides can also be written as flux differencing methods in the form

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

To derive the updating formulae and the numerical flux for eq. (5) above, we compute the exact flux through the interface $x_{i-1/2}$ using the piecewise linear solution $\tilde{q}^n(x,t)$ by integrating $\bar{u}\tilde{q}^n(x_{i-1/2},t)$ in time from t_n to t_{n+1}. For the linear advection equation, we can calculate this

$$\begin{aligned}
F_{i-1/2}^n &= \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} f(q(x_{i-1/2},t))dt \\
&= \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} \bar{u}\tilde{q}^n(x_{i-1/2},t)dt = \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} \bar{u}\tilde{q}^n(x_{i-1/2}-\bar{u}(t-t_n),t_n)dt \\
&= \frac{1}{\Delta t}\int_{t_n}^{t_{n+1}} \bar{u}\left[Q_{i-1}^n + \left(x_{i-1/2}-\bar{u}(t-t_n)-x_{i-1}\right)\sigma_{i-1}^n\right]dt \\
&= \bar{u}Q_{i-1}^n + \frac{1}{2}\bar{u}\left(\Delta x - \bar{u}\Delta t\right)\sigma_{i-1}^n.
\end{aligned}$$

Flux formulation with piecewise linear reconstruction

And for $F_{i+1/2}$ likewise

$$F^n_{i+1/2} = \bar{u} Q^n_i + \frac{1}{2}\bar{u}\left(\Delta x - \bar{u}\Delta t\right)\sigma^n_i,$$

finally giving

$$Q^{n+1}_i = Q^n_i - \frac{\bar{u}\Delta t}{\Delta x}\left(Q^n_i - Q^n_{i-1}\right) - \frac{\bar{u}\Delta t}{2\Delta x}\left(\Delta x - \bar{u}\Delta t\right)\left(\sigma^n_i - \sigma^n_{i-1}\right),$$

which is identical to eq. (5). If we also consider the case of $\bar{u} < 0$, we will find that the numerical flux for a slope limiter scheme is given by

$$F^n_{i-1/2} = \begin{cases} \bar{u} Q^n_{i-1} + \frac{1}{2}\bar{u}(\Delta x - \bar{u}\Delta t)\sigma^n_{i-1} & \text{if } \bar{u} \geqslant 0, \\ \bar{u} Q^n_i - \frac{1}{2}\bar{u}(\Delta x + \bar{u}\Delta t)\sigma^n_i & \text{if } \bar{u} < 0, \end{cases}$$

where σ^n_i is the slope in the ith cell C_i (e.g., Fromm, Beam-Warming, Lax-Wendroff,...). With the help of the flip-flop function θ

$$\theta(x) = \begin{cases} +1 & \text{for } x \geqslant 0 \\ -1 & \text{for } x < 0, \end{cases}$$

we can write down a combined formula for the numerical flux.

Flux limiters

Using $\theta_{i-1/2} = \theta(u_{i-1/2} = \bar{u})$, the above equations can be written in one formula, where $\theta_{i-1/2}$ is $+1$ for positive advection and -1 for negative advection

$$F^n_{i-1/2} = \frac{1}{2}\bar{u}\left[\left(1+\theta_{i-1/2}\right)Q^n_{i-1} + \left(1-\theta_{i-1/2}\right)Q^n_i\right] \qquad (6)$$
$$+ \frac{1}{2}|\bar{u}|\left(1-\left|\frac{\bar{u}\Delta t}{\Delta x}\right|\right)\frac{1}{2}\Delta x\left[\left(1+\theta_{i-1/2}\right)\sigma^n_{i-1} + \left(1-\theta_{i-1/2}\right)\sigma^n_i\right].$$

This expression is also valid for constant or variable advection velocities, since we only need the velocity at the interfaces and hence, we simply replace $\bar{u}$ with $u_{i-1/2}$. We will now replace the second part in the second term in eq. (6) with a slightly different term that includes the *flux limiter* ϕ

$$\frac{1}{2}\Delta x\left[\left(1+\theta_{i-1/2}\right)\sigma^n_{i-1} + \left(1-\theta_{i-1/2}\right)\sigma^n_i\right] \rightarrow \phi(r^n_{i-1/2})\left(Q^n_i - Q^n_{i-1}\right)$$

with

$$r^n_{i-1/2} = \begin{cases} \frac{Q^n_{i-1}-Q^n_{i-2}}{Q^n_i-Q^n_{i-1}} & \text{for } u_{i-1/2} \geqslant 0 \\ \frac{Q^n_{i+1}-Q^n_i}{Q^n_i-Q^n_{i-1}} & \text{for } u_{i-1/2} < 0 \end{cases}$$

Flux limiter

Equation (6) now becomes

$$F^n_{i-1/2} = \frac{1}{2} u_{i-1/2} \left[\left(1 + \theta_{i-1/2}\right) Q^n_{i-1} + \left(1 - \theta_{i-1/2}\right) Q^n_i \right] + \frac{1}{2} \left| u_{i-1/2} \right| \left(1 - \left| \frac{u_{i-1/2} \Delta t}{\Delta x} \right| \right) \phi(r^n_{i-1/2}) \left(Q^n_i - Q^n_{i-1} \right).$$

The first line gives the numerical flux for the simple donor-cell advection, and the second line are now additional correction terms.

The ratio $r^n_{i-1/2}$ can be thought as a measure of the smoothness of our data values near the interface $x_{i-1/2}$. Where the data is smooth, $r^n_{i-1/2}$ is approximately 1 and the correction terms are fully active, near a discontinuity $r^n_{i-1/2}$ will be far away from 1.

The flux limiter ϕ is a function of this ratio. By particular choices of this function, we obtain our known linear schemes and our two slope limiter schemes minmod and superbee.

Flux limiter

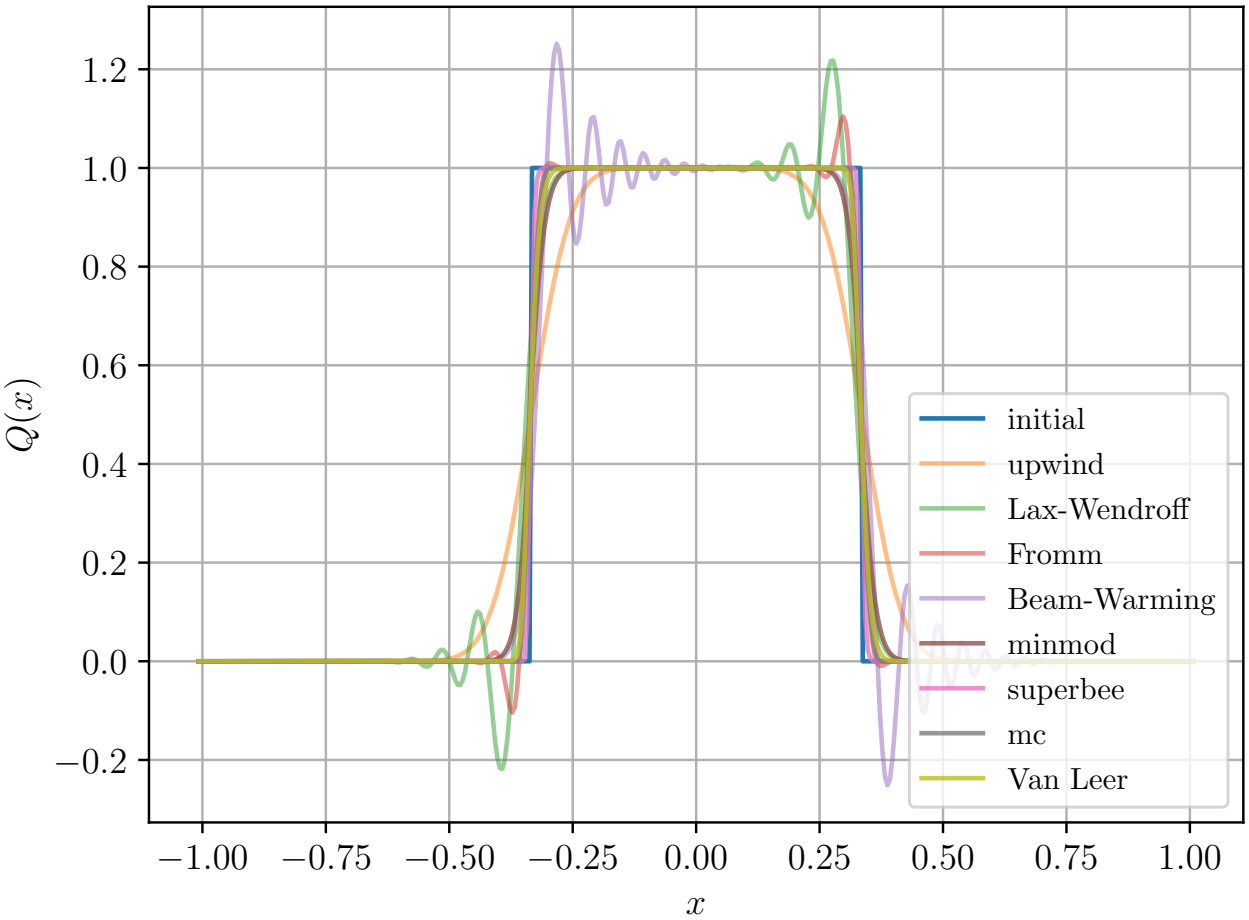
scheme	flux limiter
donor cell	$\phi(r) = 0$
Lax-Wendroff	$\phi(r) = 1$
Beam-Warming	$\phi(r) = r$
Fromm	$\phi(r) = \frac{1}{2}(1 + r)$
minmod	$\phi(r) = \text{minmod}(1, r)$
superbee	$\phi(r) = \max(0, \min(1, 2r), \min(2, r))$

Another often applied flux limiters are

MC	$\phi(r) = \max(0, \min((1 + r)/2, 2, 2r))$
van Leer	$\phi(r) = (r + \lvert r\rvert)/(1 + \lvert r\rvert)$

Flux limiters are constructed to have the following effects: (i) for smooth parts of a solution, they will do 2nd order accurate (flux conserved) advection, (ii) for regions near a discontinuity they switch to 1st order (donor cell/upwind) advection.

Flux limiter



see also the exercises...