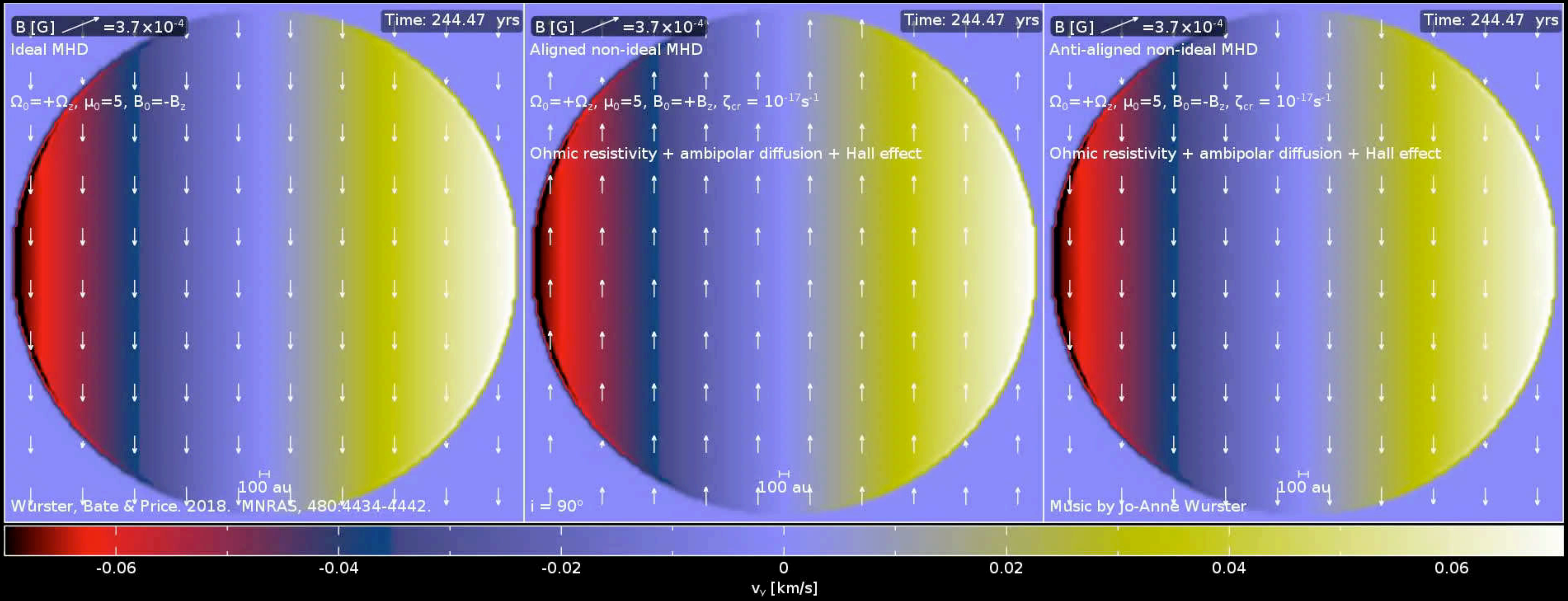
Numerical Hydrodynamics

James Wurster

Fluids: PH4031

15 February 2021

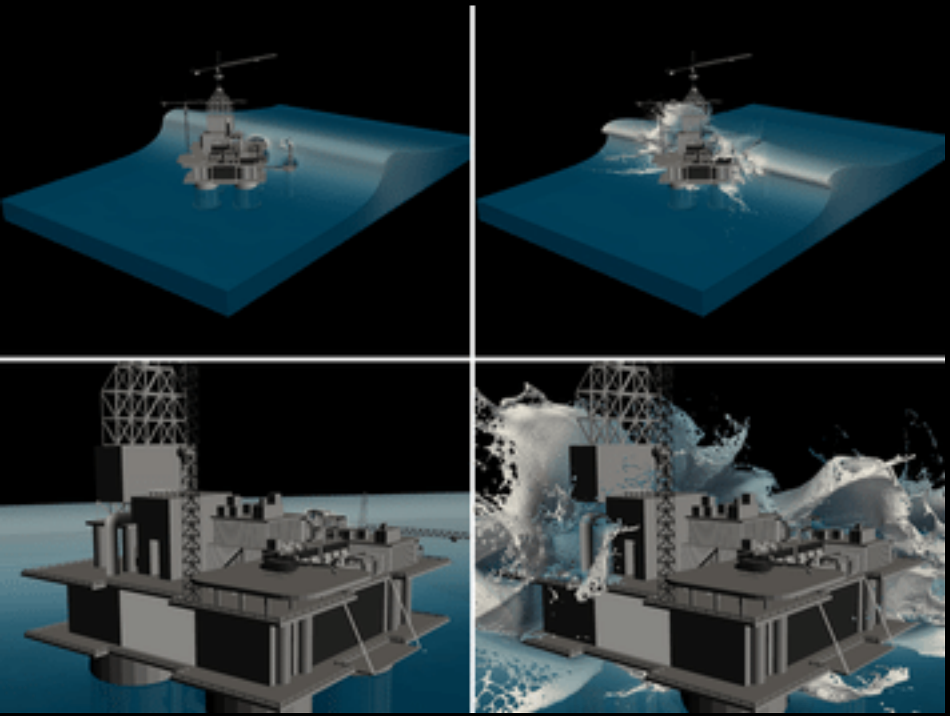
Example: Astrophysics: Star formation



Wurster, Bate & Price (2018): https://www.youtube.com/watch?v=2SQxgXbdJyg&t=22s

Example: Engineering

Wave on Oil Rig:



Urban Flooding:

https://www.youtube.com/watch?v=B8mP9E75D08

https://www.youtube.com/watch?v=jwz0stG4K9o

Example: Movies!

The Day After Tomorrow (2004): https://www.youtube.com/watch?v=GmjAp2eRDH0

2 + 2 = 4

Simple Calculation

- Fluid equations are often rate equations
  (e.g. the Parker wind model: equation for a spherically-symmetric, steady, isothermal outflow from a star of mass M_*):

$$\left(u^2 - c_s^2\right)\frac{1}{u}\frac{du}{dr} = \left(\frac{2c_s^2}{r} - \frac{GM_*}{r^2}\right)$$

- When possible, equations should be analytically simplified:

$$\bar{u}^2 - \ln(\bar{u}^2) - 4\ln(\bar{r}) - \frac{4}{\bar{r}} = K$$

- Although simplified, we still cannot analytically solve for u(r)
- To solve *u(r)*, numerical methods are required

2 + 2 = 4

Simple Calculation: Using the Newton-Raphson Method

- The Newton-Raphson Method (commonly referred to as Newton's Method) is a very powerful tool to numerically find roots of an equation. In general, assume we are given an equation, $f(x) = 0$ and an initial guess for $x = x_0$. Then,

$$x_1 = x_0 - \frac{f(x_0)}{df(x_0)/dx}$$

- Using the new x_1, we can repeat this method such that

$$x_2 = x_1 - \frac{f(x_1)}{df(x_1)/dx}$$

- and in general

$$x_{i+1} = x_i - \frac{f(x_i)}{df(x_i)/dx}$$

- This process is iterated until $|x_{i+1}/x_i - 1| < \epsilon$, where ϵ is a pre-determined tolerance. The choice of x_0 can be important; the closer to the actual value, the more stable the algorithm.

2 + 2 = 4

Simple Calculation: Using the Newton-Raphson Method

- Using a simple quadratic example:

$$f(x) = x^2 - 4 = 0$$

- From inspection, we know the roots are $x = \pm 2$
- Using Newton's method, the equation to iterate is

$$x_{i+1} = x_i - \frac{x_i^2 - 4}{2x_i}$$

- Assuming initial guesses of $x_0 = \pm 6$

$$x_1 = 6.00 - \frac{6.00^2 - 4}{2 \cdot 6.00} = 3.33 \qquad x_1 = -6.00 - \frac{-6.00^2 - 4}{2 \cdot (-6.00)} = -3.33$$

$$x_2 = 3.33 - \frac{3.33^2 - 4}{2 \cdot 3.33} = 2.27 \qquad x_2 = -3.33 - \frac{3.33^2 - 4}{2 \cdot (-3.33)} = -2.27$$

$$x_3 = 2.27 - \frac{2.27^2 - 4}{2 \cdot 2.27} = 2.02 \qquad x_3 = -2.27 - \frac{2.27^2 - 4}{2 \cdot (-2.27)} = -2.02$$

$$x_4 = 2.02 - \frac{2.02^2 - 4}{2 \cdot 2.02} = 2.00 \qquad x_4 = -2.02 - \frac{2.02^2 - 4}{2 \cdot (-2.02)} = -2.00$$

- And we rapidly converge to $x = \pm 2$

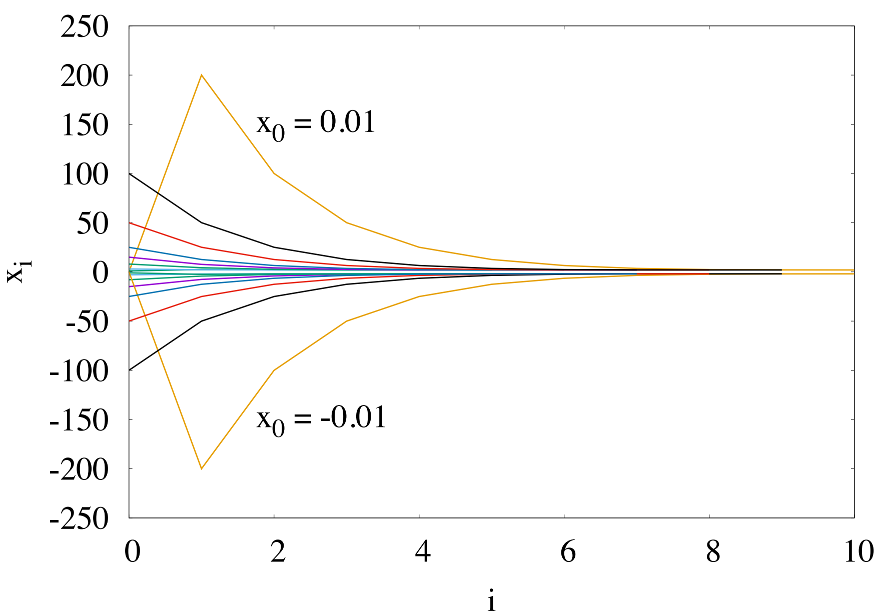
2 + 2 = 4

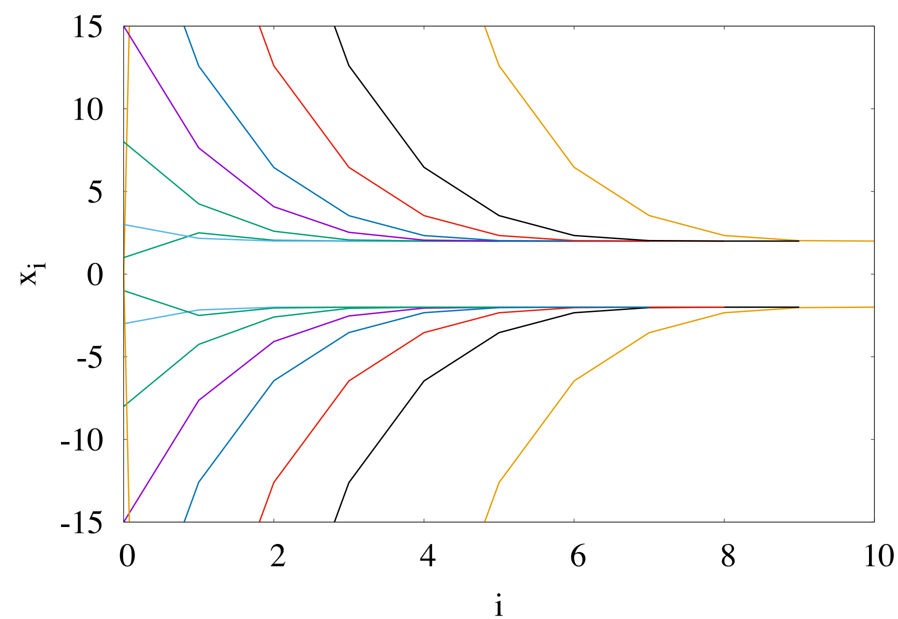
Simple Calculation: *Using the Newton-Raphson Method*

- Using a simple quadratic example:

$$f(x) = x^2 - 4 = 0$$

- All $x_0 > 0$ converge to $x = +2$ and all $x_0 < 0$ converge to $x = -2$.
- Initial guess will determine how quickly the solution converges and to which root it converges (same plot, just different vertical scales)

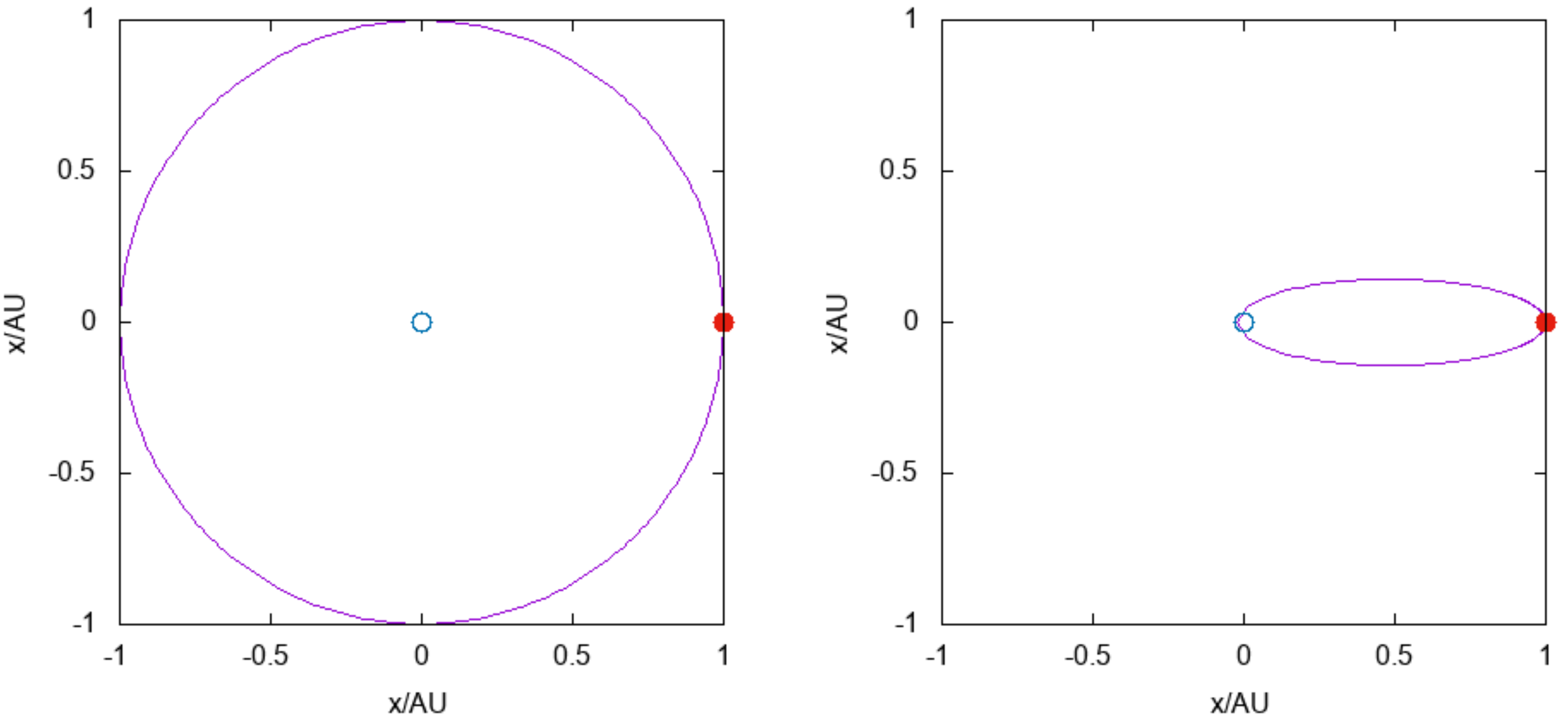




- This problem is a steady-state, and does not evolve in time
- Although steady-state, this does represent the power of Numerical Methods

N-Body Calculation

- Numerical methods are excellent at calculating N-body motion, such as planetary/cometary orbits (e.g. Exercise 3 in Computational Astrophysics)
- N-body calculations are integrated in time, and include gravity only but no fluid dynamics



Complex Calculation

- Rather than a steady flow, or a gravity-only simulation, assume we have a dynamically evolving situation that includes fluids rather than discrete bodies (e.g. rolling clouds):
- We do not have a single equation that can describe this motion
    - Motion cannot be described analytically
- To solve this numerically, we require
    - The initial properties of the system (i.e. initial conditions)
    - A method to divide the region (e.g. grids)
    - A method to describe the edge of the region (i.e. boundary conditions)
    - A method to describe the evolution of the region (i.e. the set of fluid dynamics equations)

Fluid equations

- Continuum Equations:

$$\text{Continuity equation:} \quad \frac{\mathrm{D}\rho}{\mathrm{D}t} = -\rho\nabla\cdot\boldsymbol{v}$$

$$\text{Equation of motion:} \quad \frac{\mathrm{D}\boldsymbol{v}}{\mathrm{Dt}} = -\frac{1}{\rho}\nabla P$$

$$\text{Energy equation:} \quad \frac{\mathrm{D}u}{\mathrm{D}t} = -\frac{P}{\rho}\nabla\cdot\boldsymbol{v}$$

$$\text{Equation of state:} \quad P = (\gamma-1)\,\rho u$$

- Where

$$\frac{\mathrm{D}}{\mathrm{Dt}} \equiv \frac{\partial}{\partial t} + \boldsymbol{v}\cdot\boldsymbol{\nabla}$$

is the Lagrangian (or co-moving) derivative

Fluid equations

- Continuum Equations:

$$\text{Continuity equation:} \quad \frac{\mathrm{D}\rho}{\mathrm{D}t} = -\rho \nabla \cdot \boldsymbol{v}$$

$$\text{Equation of motion:} \quad \frac{\mathrm{D}\boldsymbol{v}}{\mathrm{Dt}} = -\frac{1}{\rho} \nabla P$$

$$\text{Energy equation:} \quad \frac{\mathrm{D}u}{\mathrm{D}t} = -\frac{P}{\rho} \nabla \cdot \boldsymbol{v}$$

$$\text{Equation of state:} \quad P = (\gamma - 1)\, \rho u$$

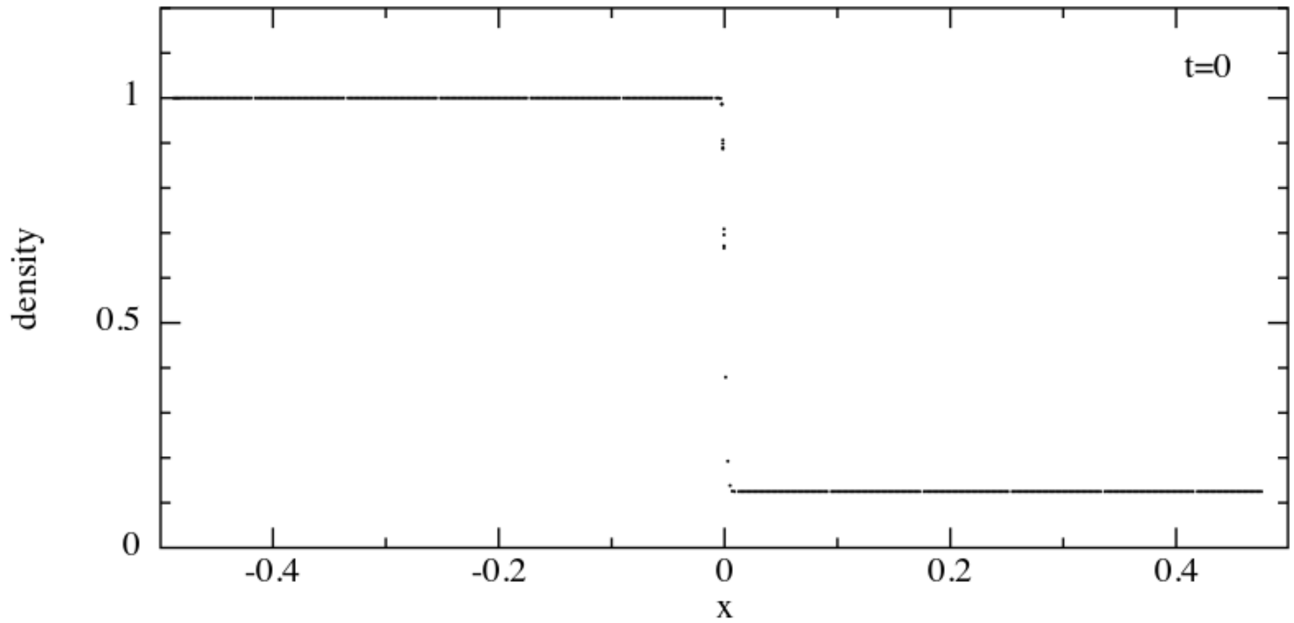
- Where

$$\frac{\mathrm{D}}{\mathrm{Dt}} \equiv \frac{\partial}{\partial t} + \boldsymbol{v} \cdot \boldsymbol{\nabla}$$

- This is a closed set of equations: 4 equations 4 four unknowns
- The system evolves in time (i.e. $\partial / \partial t$) & position i.e. (∇)
- To convert to numerical equations, must first choose a grid

Defining your problem

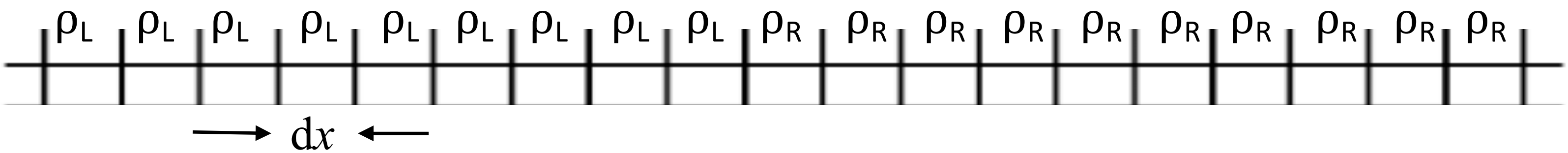
- Assume we have a simple 1D problem where the density is as follows:



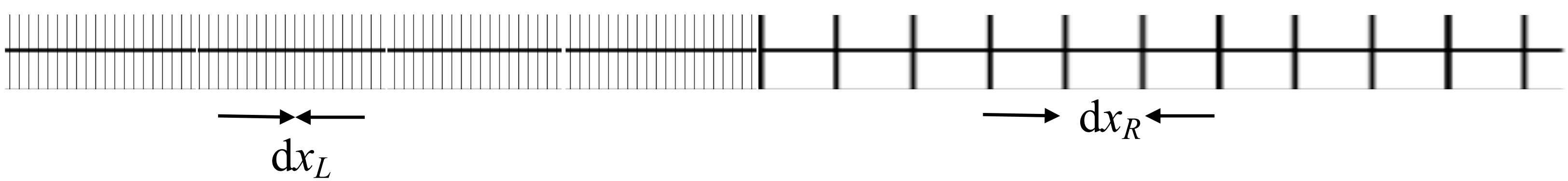
- How do we divide up the region?

Defining your problem: Dividing your region

- Eulerian grid:
  grid of constant spacing, density varies in each cell



- Lagrangian grid:
  grid of varying spacing, mass is constant



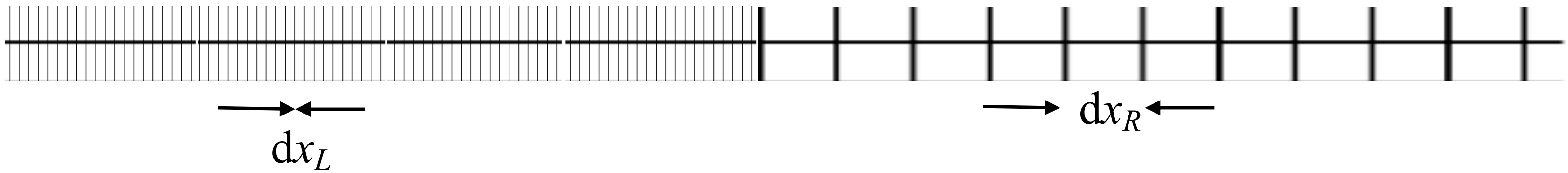
- Smoothed Particle Hydrodynamics:
  Spheres of constant mass represent ‘packets’ of fluid;
  density is dependent on proximity of neighbours

Defining your problem: Dividing your region

For Lagrangian systems, the co-moving derivative is simply

$$\frac{\mathrm{D}}{\mathrm{Dt}} \equiv \frac{\partial}{\partial t}$$

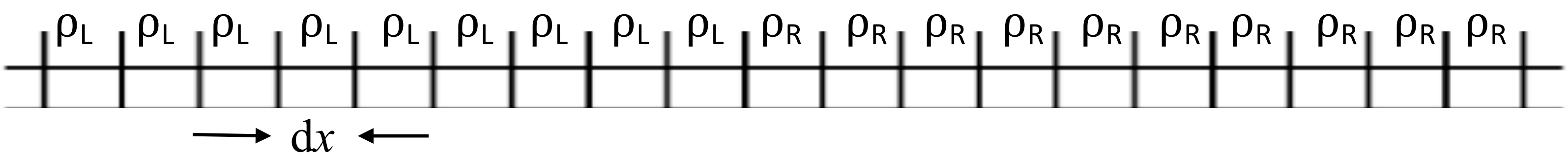
- Lagrangian grid:
  grid of varying spacing, mass is constant



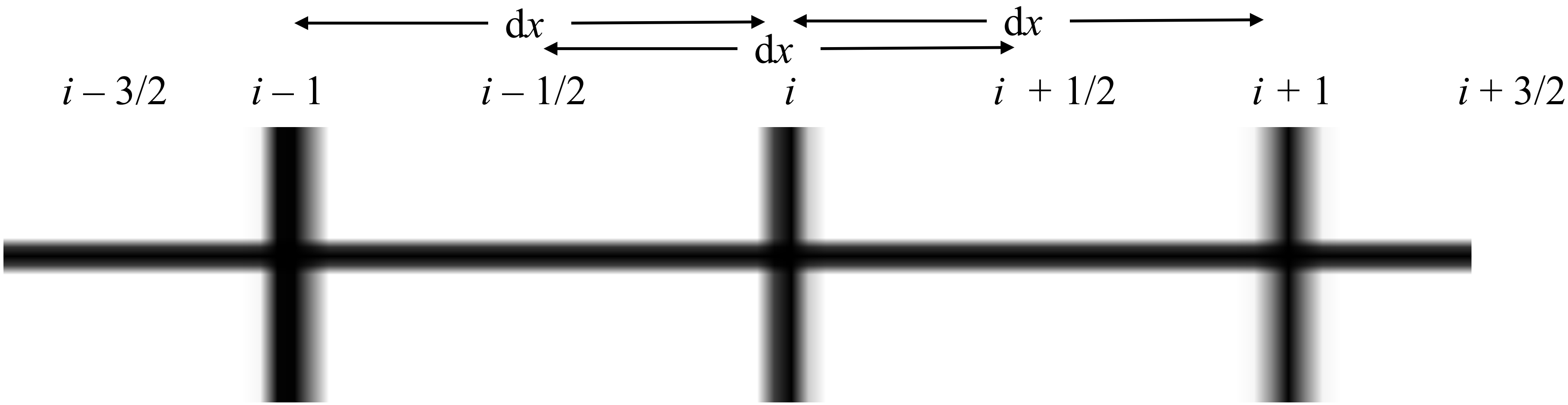
- Smoothed Particle Hydrodynamics:
  Spheres of constant mass represent ‘packets’ of fluid;
  density is dependent on proximity of neighbours

Defining your problem: Defining quantities

- Eulerian grid: grid of constant spacing



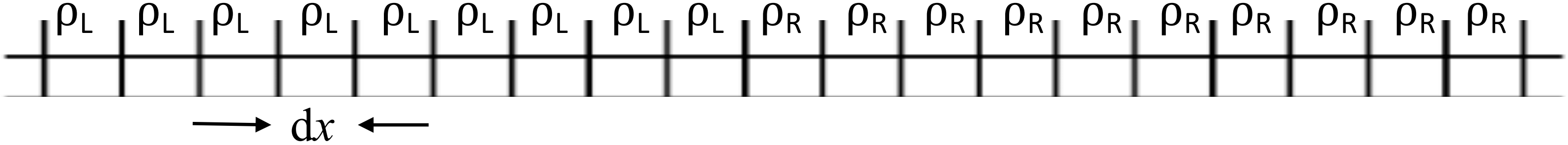
- A few cells:



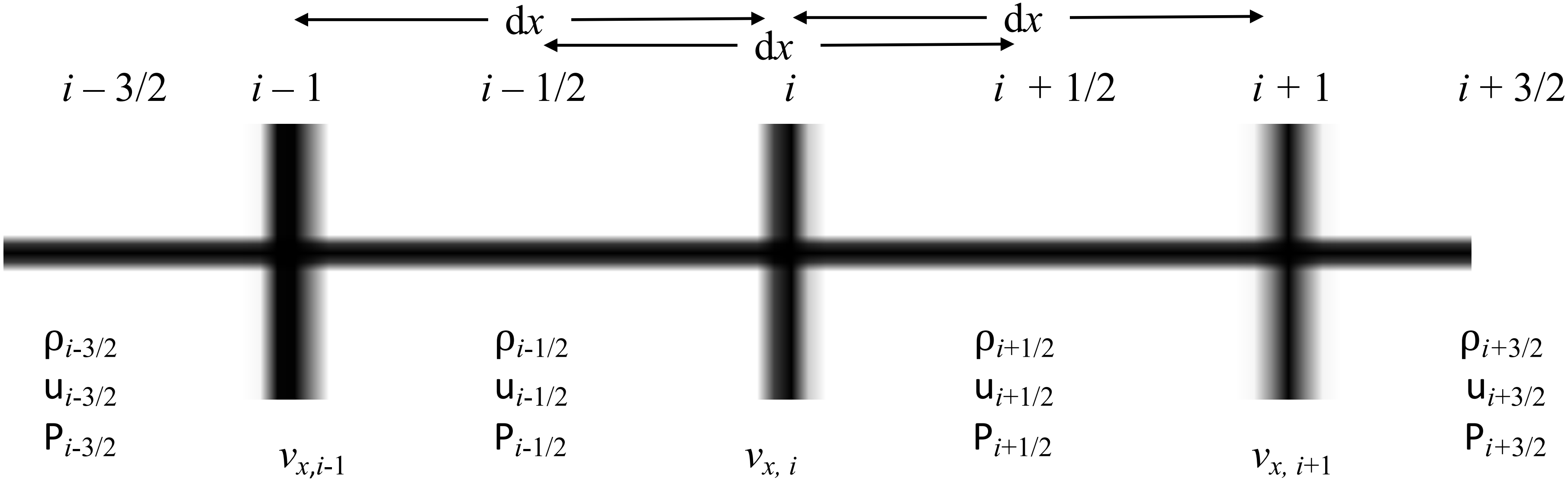
- Quantities need to be defined at a given position.
  - Scalars: density, internal energy, pressure
  - Vectors: velocity

Defining your problem: Defining quantities

- Eulerian grid: grid of constant spacing



- A few cells:



- Scalars are calculated at *cell-centre*
- Vectors are calculated at *cell-interface*

Complex Calculation: Required Components

- To solve any system numerically, we require

A method to divide the region (e.g. grids)
A method to describe the evolution of the region
(i.e. the set of fluid dynamics equations)
A method to describe the edge of the region (i.e. boundary conditions)
The initial properties of the system (i.e. initial conditions)

Fluid equations: Continuum vs 1D-Numerical

- Continuity Equation

$$\frac{\mathrm{D}\rho}{\mathrm{D}t} = -\rho\nabla\cdot\boldsymbol{v}$$

$$\frac{\partial\rho}{\partial t} + \boldsymbol{v}\cdot\boldsymbol{\nabla}\rho = -\rho\nabla\cdot\boldsymbol{v}$$

$$\frac{\partial\rho}{\partial t} = -\rho\nabla\cdot\boldsymbol{v} - \boldsymbol{v}\cdot\boldsymbol{\nabla}\rho$$

Fluid equations: Continuum vs 1D-Numerical

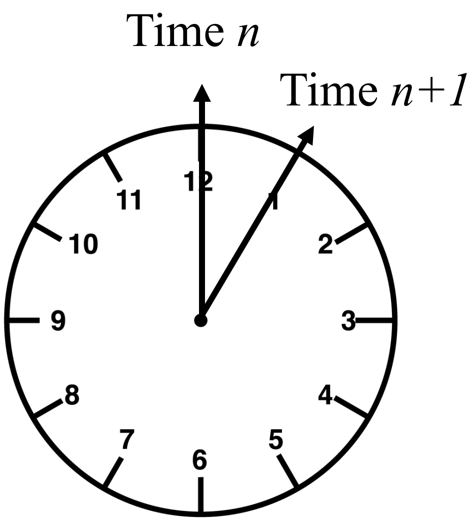
- Continuity Equation

$$\frac{\mathrm{D}\rho}{\mathrm{D}t} = -\rho\nabla\cdot\boldsymbol{v}$$

$$\frac{\partial\rho}{\partial t} + \boldsymbol{v}\cdot\boldsymbol{\nabla}\rho = -\rho\nabla\cdot\boldsymbol{v}$$

$$\frac{\partial\rho}{\partial t} = -\rho\nabla\cdot\boldsymbol{v} - \boldsymbol{v}\cdot\boldsymbol{\nabla}\rho$$

$$\frac{\rho^{n+1}_{i+\frac{1}{2}} - \rho^{n}_{i+\frac{1}{2}}}{dt} = -\rho\nabla\cdot\boldsymbol{v} - \boldsymbol{v}\cdot\boldsymbol{\nabla}\rho$$



Fluid equations: Continuum vs 1D-Numerical

- Continuity Equation

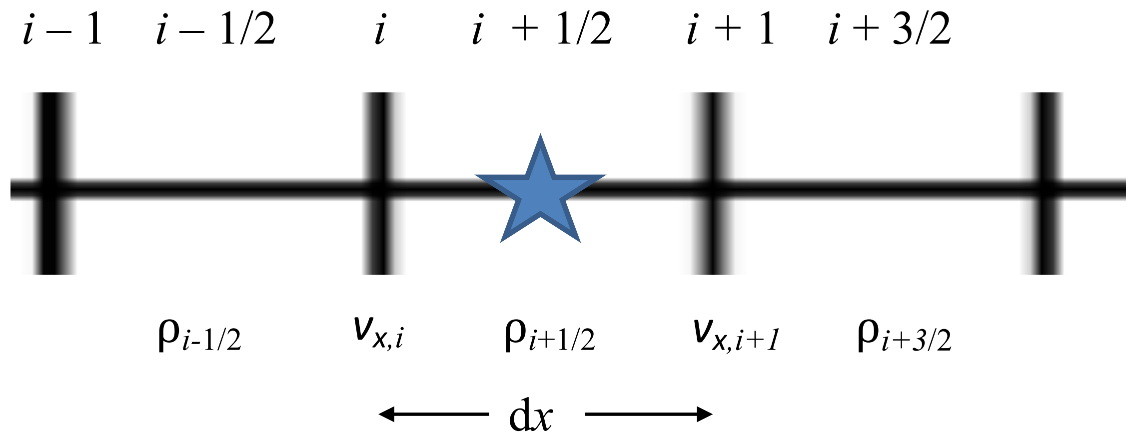
$$\frac{\mathrm{D}\rho}{\mathrm{D}t} = -\rho\nabla\cdot\boldsymbol{v}$$

$$\frac{\partial\rho}{\partial t} + \boldsymbol{v}\cdot\boldsymbol{\nabla}\rho = -\rho\nabla\cdot\boldsymbol{v}$$

$$\frac{\partial\rho}{\partial t} = -\rho\nabla\cdot\boldsymbol{v} - \boldsymbol{v}\cdot\boldsymbol{\nabla}\rho$$

$$\frac{\rho^{n+1}_{i+\frac{1}{2}} - \rho^{n}_{i+\frac{1}{2}}}{dt} = -\rho\nabla\cdot\boldsymbol{v} - \boldsymbol{v}\cdot\boldsymbol{\nabla}\rho$$

$$\frac{\rho^{n+1}_{i+\frac{1}{2}} - \rho^{n}_{i+\frac{1}{2}}}{dt} = -\rho^{n}_{i+\frac{1}{2}}\frac{v^{n}_{x,i+1} - v^{n}_{x,i}}{dx} - \boldsymbol{v}\cdot\boldsymbol{\nabla}\rho$$



Fluid equations: Continuum vs 1D-Numerical

- Continuity Equation

$$\frac{\mathrm{D}\rho}{\mathrm{D}t} = -\rho\nabla\cdot\boldsymbol{v}$$

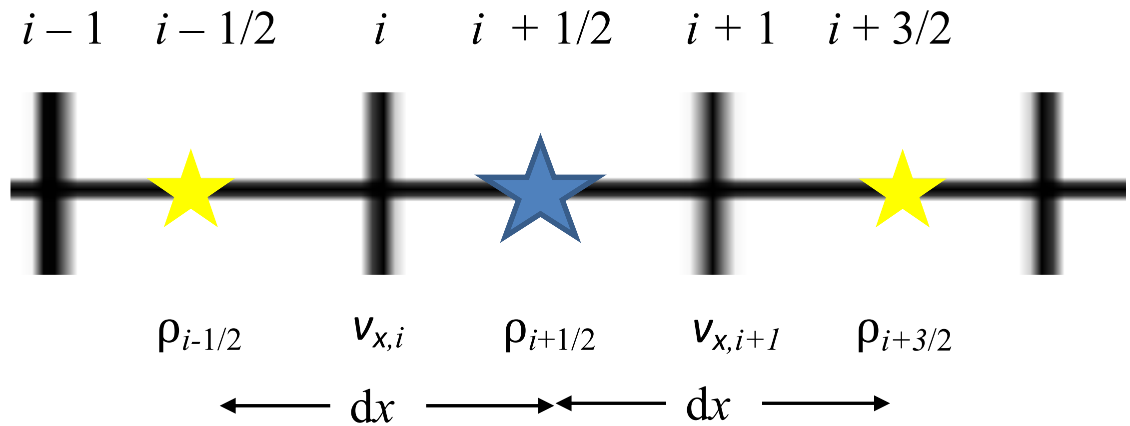
$$\frac{\partial\rho}{\partial t} + \boldsymbol{v}\cdot\boldsymbol{\nabla}\rho = -\rho\nabla\cdot\boldsymbol{v}$$

$$\frac{\partial\rho}{\partial t} = -\rho\nabla\cdot\boldsymbol{v} - \boldsymbol{v}\cdot\boldsymbol{\nabla}\rho$$

$$\frac{\rho^{n+1}_{i+\frac{1}{2}} - \rho^{n}_{i+\frac{1}{2}}}{dt} = -\rho\nabla\cdot\boldsymbol{v} - \boldsymbol{v}\cdot\boldsymbol{\nabla}\rho$$

$$\frac{\rho^{n+1}_{i+\frac{1}{2}} - \rho^{n}_{i+\frac{1}{2}}}{dt} = -\rho^{n}_{i+\frac{1}{2}}\frac{v^{n}_{x,i+1} - v^{n}_{x,i}}{dx} - \boldsymbol{v}\cdot\boldsymbol{\nabla}\rho$$

$$\frac{\rho^{n+1}_{i+\frac{1}{2}} - \rho^{n}_{i+\frac{1}{2}}}{dt} = -\rho^{n}_{i+\frac{1}{2}}\frac{v^{n}_{x,i+1} - v^{n}_{x,i}}{dx} - \frac{v^{n}_{x,i+1} + v^{n}_{x,i}}{2}\frac{\rho^{n}_{i+\frac{3}{2}} - \rho^{n}_{i-\frac{1}{2}}}{2dx}$$



‘zero-th order’ approximation

Fluid equations: Continuum vs 1D-Numerical

- Continuity Equation

$$\frac{\mathrm{D}\rho}{\mathrm{D}t} = -\rho\nabla\cdot\boldsymbol{v}$$

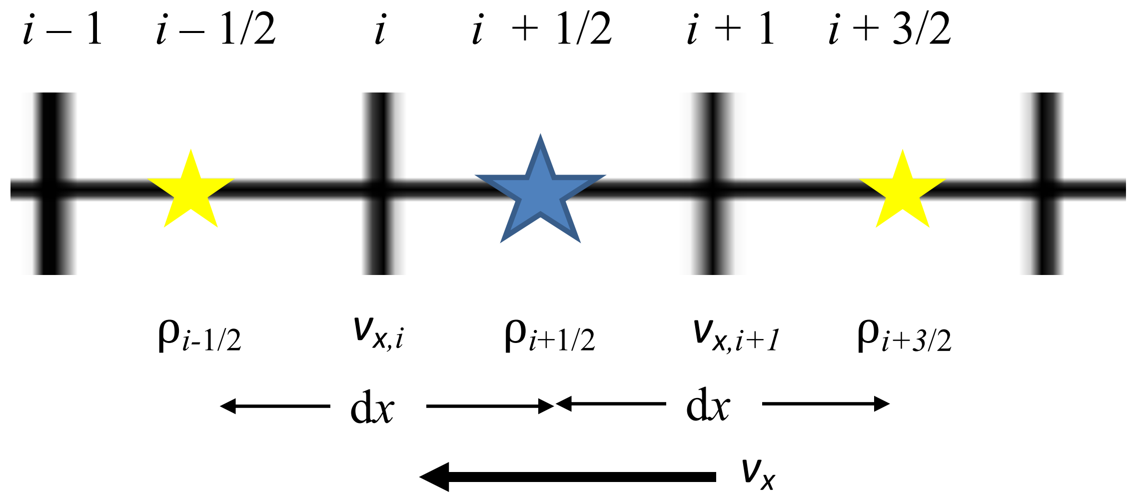
$$\frac{\partial\rho}{\partial t} + \boldsymbol{v}\cdot\boldsymbol{\nabla}\rho = -\rho\nabla\cdot\boldsymbol{v}$$

$$\frac{\partial\rho}{\partial t} = -\rho\nabla\cdot\boldsymbol{v} - \boldsymbol{v}\cdot\boldsymbol{\nabla}\rho$$

$$\frac{\rho^{n+1}_{i+\frac{1}{2}} - \rho^{n}_{i+\frac{1}{2}}}{dt} = -\rho\nabla\cdot\boldsymbol{v} - \boldsymbol{v}\cdot\boldsymbol{\nabla}\rho$$

$$\frac{\rho^{n+1}_{i+\frac{1}{2}} - \rho^{n}_{i+\frac{1}{2}}}{dt} = -\rho^{n}_{i+\frac{1}{2}}\frac{v^{n}_{x,i+1} - v^{n}_{x,i}}{dx} - \boldsymbol{v}\cdot\boldsymbol{\nabla}\rho$$

$$\frac{\rho^{n+1}_{i+\frac{1}{2}} - \rho^{n}_{i+\frac{1}{2}}}{dt} = -\rho^{n}_{i+\frac{1}{2}}\frac{v^{n}_{x,i+1} - v^{n}_{x,i}}{dx} - \frac{v^{n}_{x,i+1} + v^{n}_{x,i}}{2}\begin{cases}\frac{\rho^{n}_{i+\frac{1}{2}} - \rho^{n}_{i-\frac{1}{2}}}{dx} & \text{for } \frac{1}{2}(v^{n}_{x,i+1} + v^{n}_{x,i}) > 0\\ \frac{\rho^{n}_{i+\frac{3}{2}} - \rho^{n}_{i+\frac{1}{2}}}{dx} & \text{for } \frac{1}{2}(v^{n}_{x,i+1} + v^{n}_{x,i}) < 0\end{cases}$$



First order donor cell method

Fluid equations: Continuum vs 1D-Numerical

- The discrete fluid dynamic equations for an Eulerian grid:

$$
\begin{aligned}
\rho^{n+1}_{i+\frac{1}{2}} &= \rho^{n}_{i+\frac{1}{2}} - dt\left(\rho^{n}_{i+\frac{1}{2}}\frac{v^{n}_{x,i+1}-v^{n}_{x,i}}{dx}+\frac{v^{n}_{x,i+1}+v^{n}_{x,i}}{2}f(\rho)\right)\\
v^{n+1}_{x,i} &= v^{n}_{x,i} - dt\left(\frac{2}{\rho^{n}_{i+\frac{1}{2}}+\rho^{n}_{i-\frac{1}{2}}}\frac{P^{n}_{i+\frac{1}{2}}-P^{n}_{i-\frac{1}{2}}}{dx}+v^{n}_{x,i}f(v)\right)\\
u^{n+1}_{i+\frac{1}{2}} &= u^{n}_{i+\frac{1}{2}} - dt\left(\frac{P^{n}_{i+\frac{1}{2}}}{\rho^{n}_{i+\frac{1}{2}}}\frac{v^{n}_{x,i+1}-v^{n}_{x,i}}{dx}+\frac{v^{n}_{x,i+1}+v^{n}_{x,i}}{2}f(u)\right)\\
P^{n+1}_{i+\frac{1}{2}} &= (\gamma-1)\,\rho^{n+1}_{i+\frac{1}{2}}u^{n+1}_{i+\frac{1}{2}}
\end{aligned}
$$

- where $f(a) = \nabla a$, and can be 0^{th} order, 1^{st} order (Donor cell) or even higher order (e.g. 2^{nd} order van Leer; 3^{rd} order piecewise parabolic advection; etc...)

Fluid equations: Time integration

- Quantities are solved at different locations
- Should quantities also be solved at different times, where *n* is the current timestep?
  - Leapfrog
    - Update vectors to *n* + 1/2

$$v_{x,i}^{n+\frac{1}{2}} = v_{x,i}^{n-\frac{1}{2}} - dt\left(\frac{2}{\rho_{i+\frac{1}{2}}^{n} + \rho_{i-\frac{1}{2}}^{n}}\frac{P_{i+\frac{1}{2}}^{n} - P_{i-\frac{1}{2}}^{n}}{dx} + v_{x,i}^{n-\frac{1}{2}} f(v)\right)$$

    - Using updates vectors, update scalars to *n*+1

$$\rho_{i+\frac{1}{2}}^{n+1} = \rho_{i+\frac{1}{2}}^{n} - dt\left(\rho_{i+\frac{1}{2}}^{n}\frac{v_{x,i+1}^{n+\frac{1}{2}} - v_{x,i}^{n+\frac{1}{2}}}{dx} + \frac{v_{x,i+1}^{n+\frac{1}{2}} + v_{x,i}^{n+\frac{1}{2}}}{2} f(\rho)\right)$$

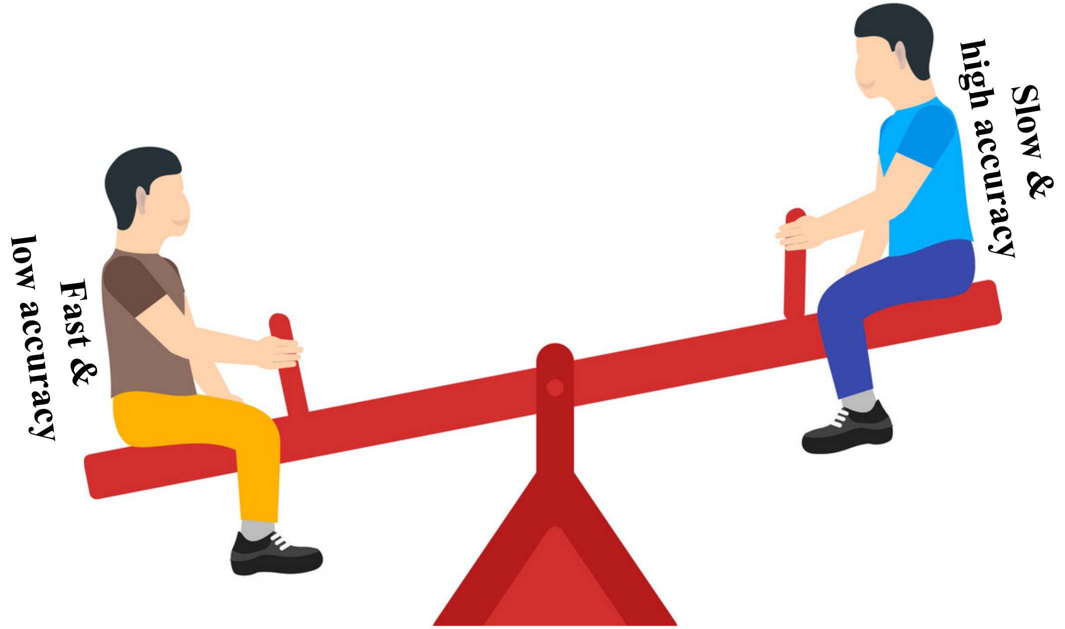
$$u_{i+\frac{1}{2}}^{n+1} = u_{i+\frac{1}{2}}^{n} - dt\left(\frac{P_{i+\frac{1}{2}}^{n}}{\rho_{i+\frac{1}{2}}^{n}}\frac{v_{x,i+1}^{n+\frac{1}{2}} - v_{x,i}^{n+\frac{1}{2}}}{dx} + \frac{v_{x,i+1}^{n+\frac{1}{2}} + v_{x,i}^{n+\frac{1}{2}}}{2} f(u)\right)$$

$$P_{i+\frac{1}{2}}^{n+1} = (\gamma - 1)\,\rho_{i+\frac{1}{2}}^{n+1} u_{i+\frac{1}{2}}^{n+1}$$

Fluid equations: Warning!

- There are several different time and spatial integration techniques
- The more advanced the technique…
  - the more accurate the result
  - the longer the computational time

In numerical studies, the user must always balance accuracy with time!



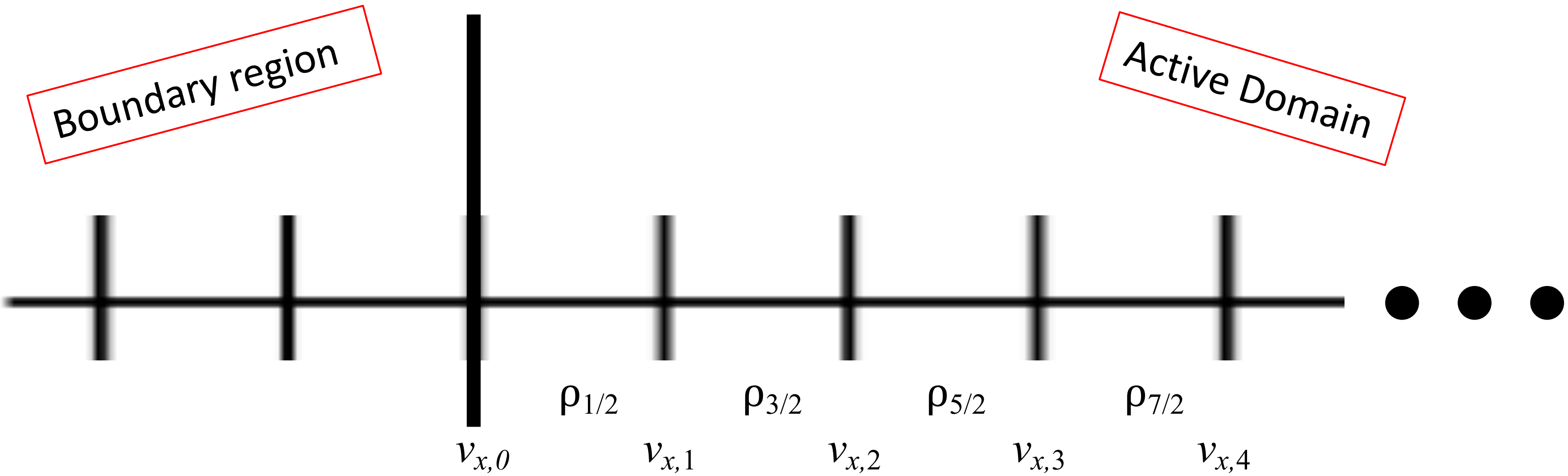
Complex Calculation: Required Components

- To solve any system numerically, we require

A method to divide the region (e.g. grids)
A method to describe the evolution of the region
(i.e. the set of fluid dynamics equations)
A method to describe the edge of the region (i.e. boundary conditions)
The initial properties of the system (i.e. initial conditions)

Boundaries

- We almost have enough information to run a simple simulation, but what happens at the edge of the simulation?
- Similar to solving differential equations, boundary conditions are required:

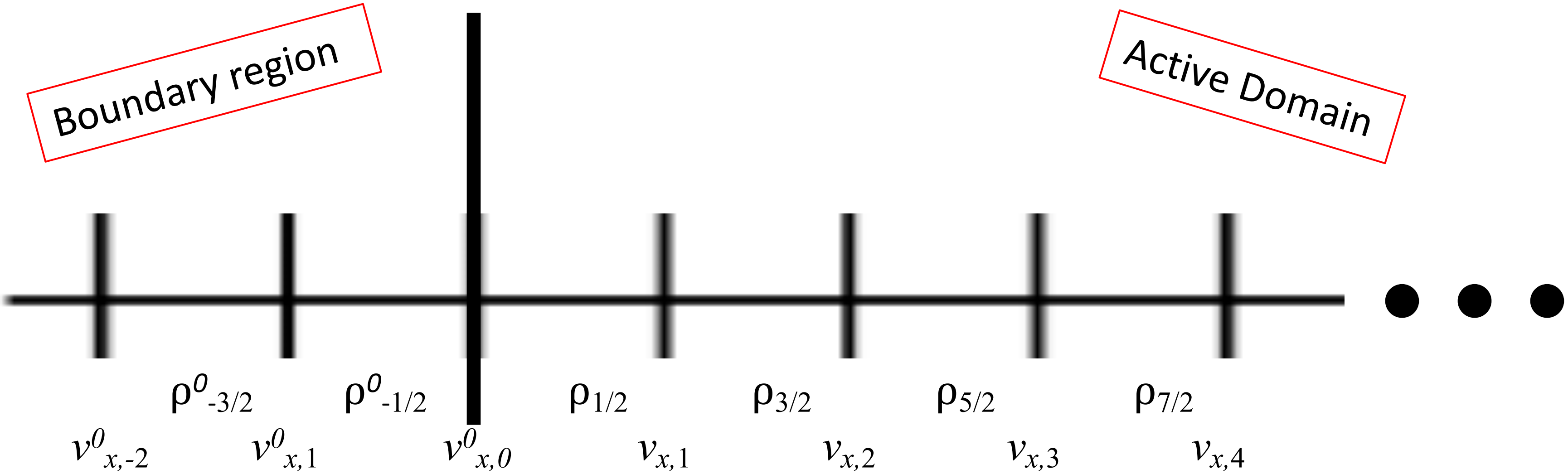


- Locations required to update scalars:

- Locations required to update vectors:

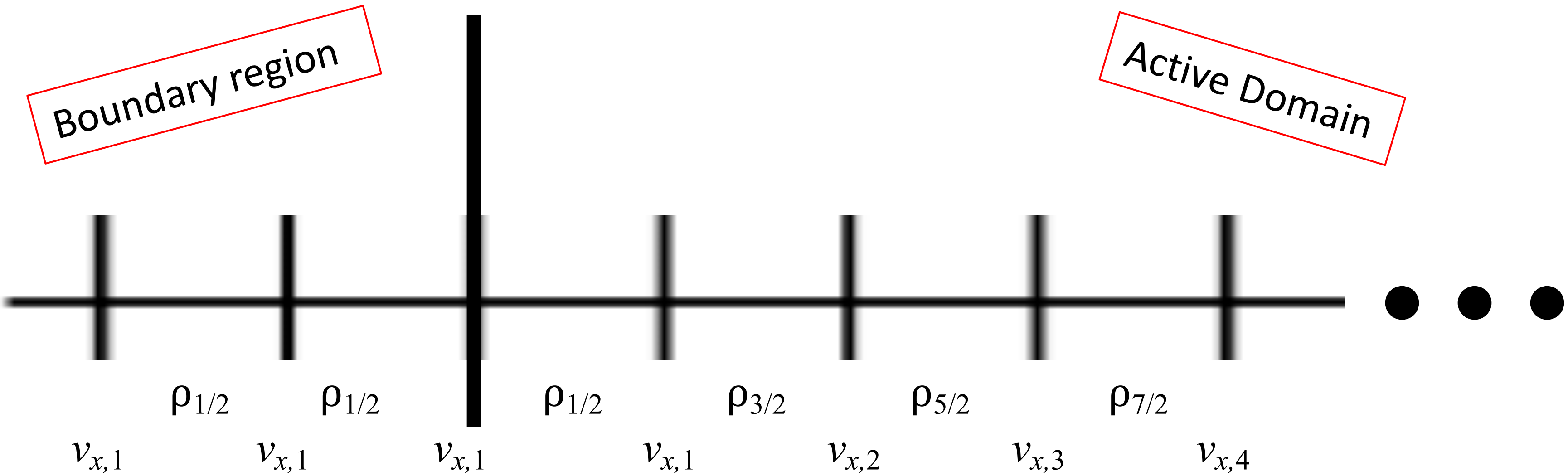
Boundaries

- We almost have enough information to run a simple simulation,
  but what happens at the edge of the simulation?
- Similar to solving differential equations, boundary conditions are required:
    - Fixed ($v^0 = 0$) / inflow ($v^0 > 0$):



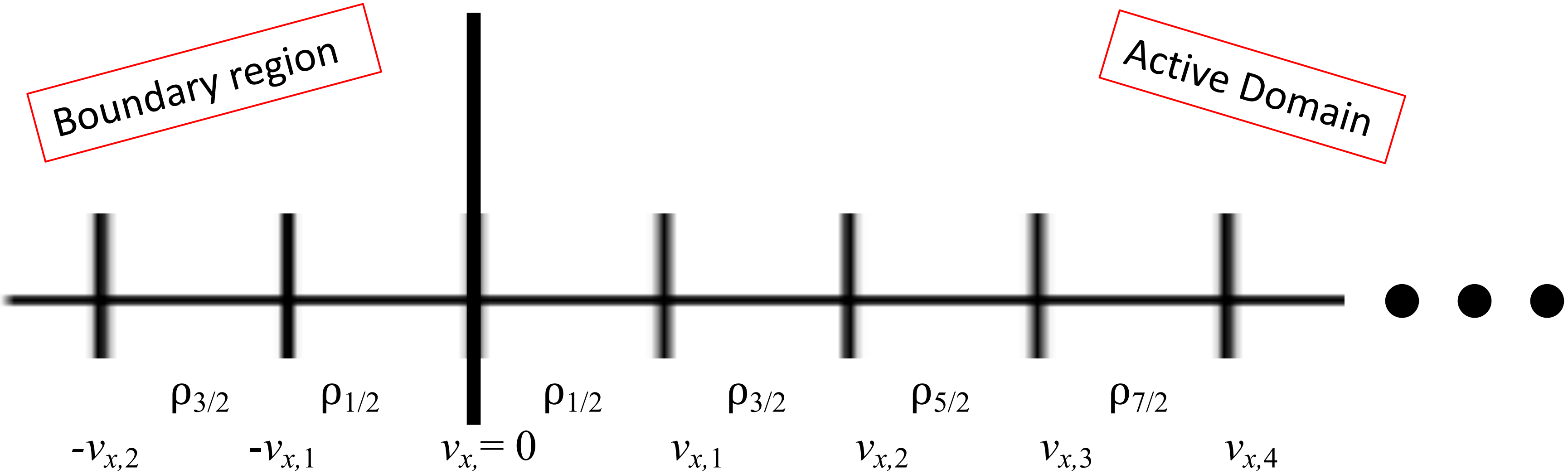
Boundaries

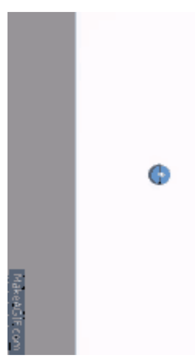
- We almost have enough information to run a simple simulation, but what happens at the edge of the simulation?
- Similar to solving differential equations, boundary conditions are required:
  - Outflow:



Boundaries

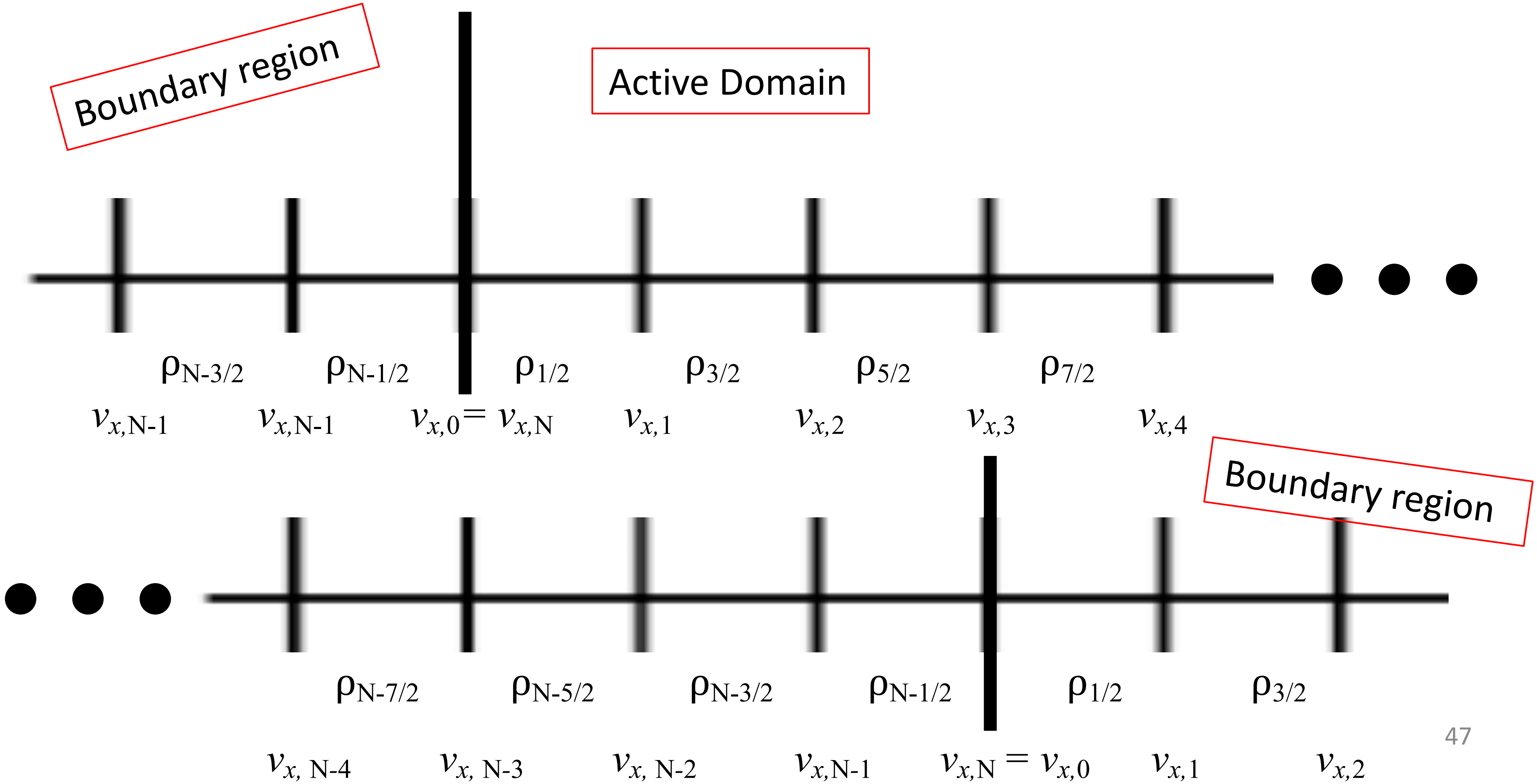
- We almost have enough information to run a simple simulation, but what happens at the edge of the simulation?
- Similar to solving differential equations, boundary conditions are required:
    - Reflective:





Boundaries

- We almost have enough information to run a simple simulation, but what happens at the edge of the simulation?
- Similar to solving differential equations, boundary conditions are required:
  - Periodic:



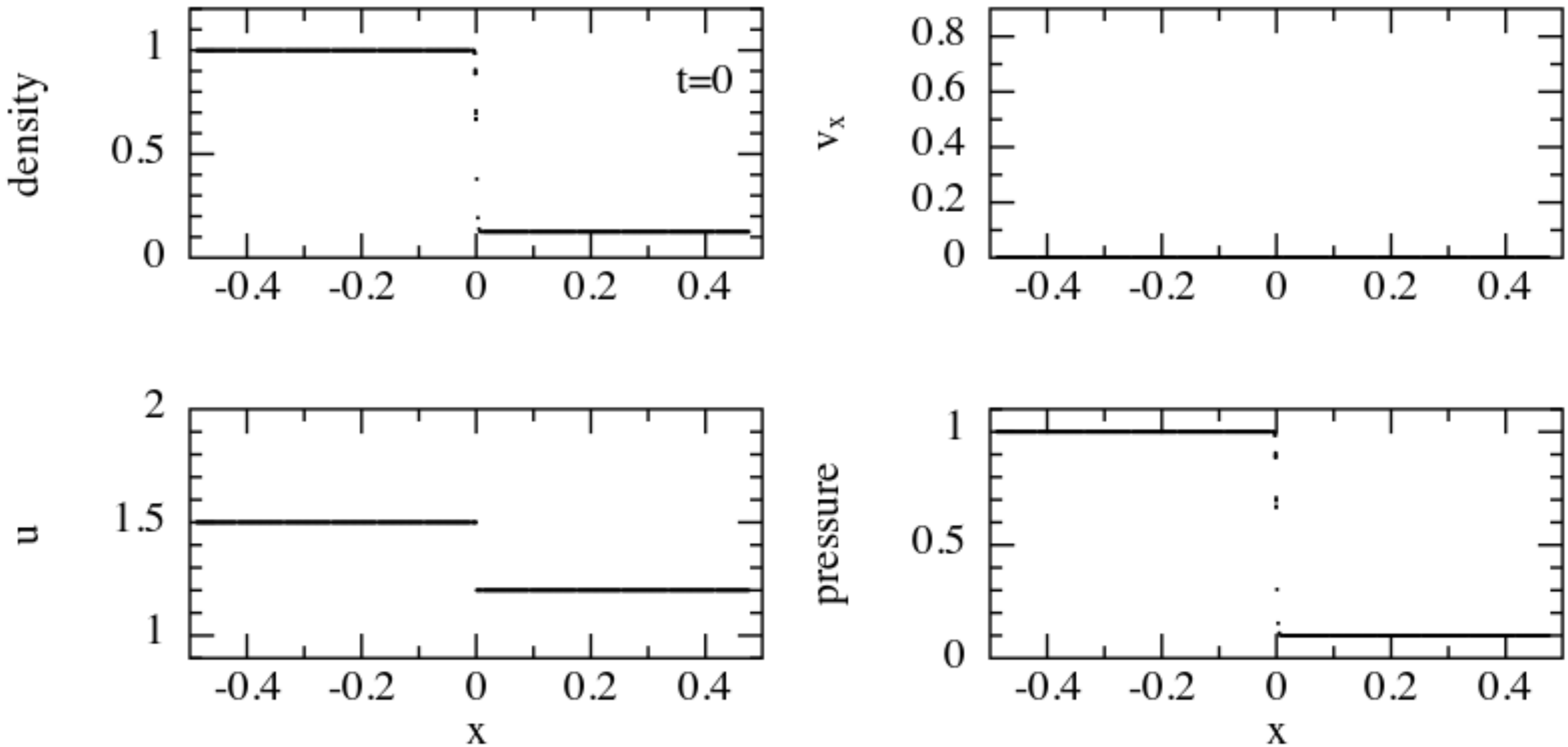
Complex Calculation: Required Components

- To solve any system numerically, we require

A method to divide the region (e.g. grids)
A method to describe the evolution of the region
(i.e. the set of fluid dynamics equations)
A method to describe the edge of the region (i.e. boundary conditions)
The initial properties of the system (i.e. initial conditions)

Initial conditions: Sod Shock

- Initial conditions for the Sod Shock
- Boundary Conditions: fixed



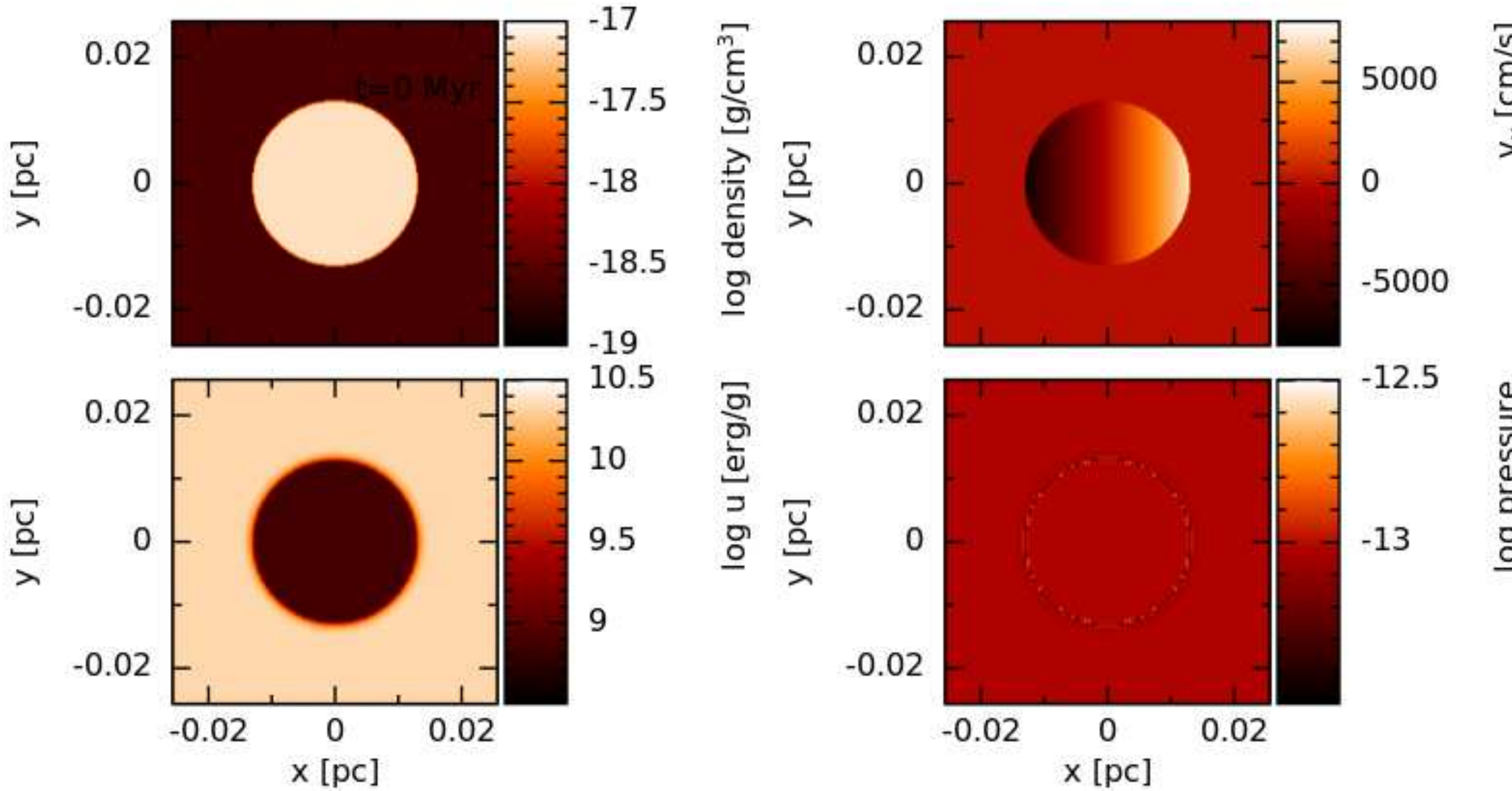
Complex Calculation: Required Components

- To solve any system numerically, we require

A method to divide the region (e.g. grids)
A method to describe the evolution of the region
(i.e. the set of fluid dynamics equations)
A method to describe the edge of the region (i.e. boundary conditions)
The initial properties of the system (i.e. initial conditions)

Initial conditions: Astrophysical simulations

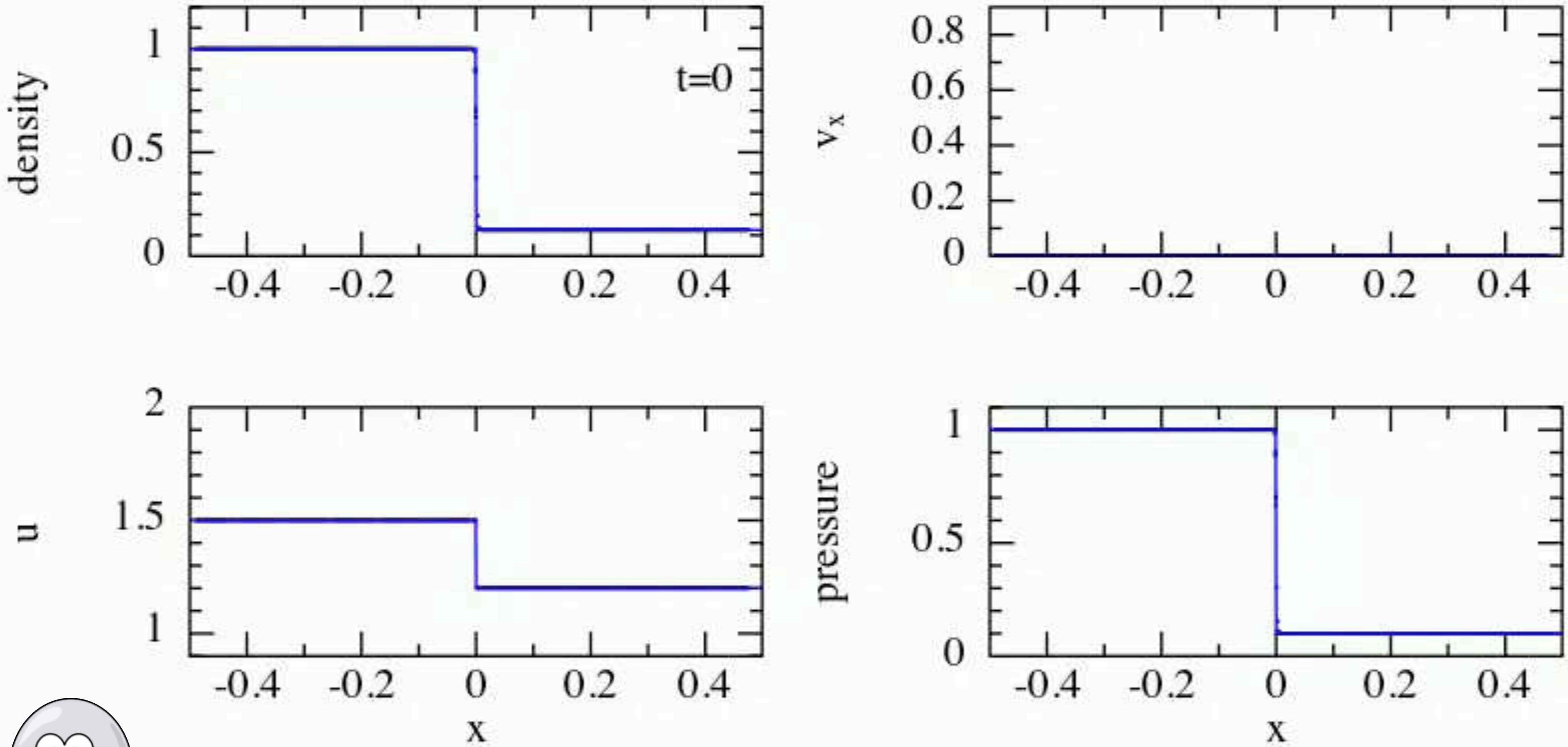
- Initial conditions are incredibly important for any simulation
  - A method to describe the evolution of the region: fluid dynamics equations
  - A method to divide the region: smoothed particle hydrodynamics
  - The initial properties of the system: see below
  - A method to describe the boundaries: sphere-in-box with periodic B.C.s



Complex Calculation: Next steps

- Now that we have the basis of a code, can we now run complex physical calculations?
  - No! 
  - It must first be rigorously tested!
- To test codes, we must run simple test problems where an analytical answer is known
- In numerical hydrodynamics, a common and simple test problem is the Sod Shock Tube (Sod 1978)

Sod Shock Evolution



Sod Shock Evolution

➢ Ringing and instabilities occur at the shock wave and propagate backwards