

# Smoothed particle hydrodynamics: A brief overview

Dr. James Wurster

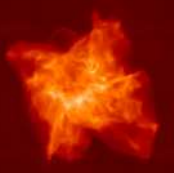
January 17, 2021



University of  
St Andrews



# *Smoothed Particle Hydrodynamics*



Time:  $1.9 \times 10^{-3}$  Myr

Non-ideal MHD,  $\mu_0=3$

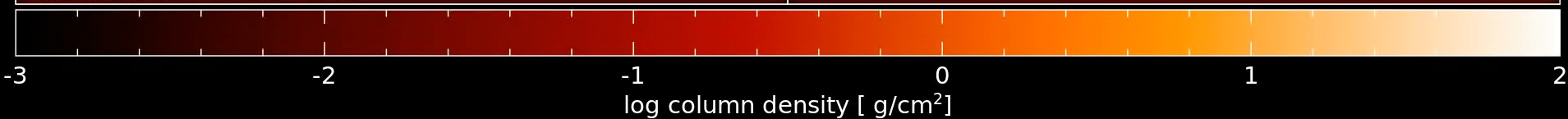
Hydro



0.50 pc

0.50 pc

Wurster, Bate & Price (2019)



# Historical Overview



1977AJ.....82..1013L

THE ASTRONOMICAL JOURNAL

VOLUME 82, NUMBER 12

DECEMBER 1977

## A numerical approach to the testing of the fission hypothesis

L. B. Lucy<sup>1)</sup>

*Institute of Astronomy, Cambridge, United Kingdom  
European Southern Observatory, Geneva, Switzerland  
(Received 12 August 1977; revised 16 September 1977)*

A finite-size particle scheme for the numerical solution of two- and three-dimensional gas dynamical problems of astronomical interest is described and tested. The scheme is then applied to the fission problem for optically thick protostars. Results are given, showing the evolution of one such protostar from an initial state as a single, rotating star to a final state as a triple system whose components contain 60% of the original mass. The decisiveness of this numerical test of the fission hypothesis and its relevance to observed binaries are briefly discussed.

### INTRODUCTION

THE hypothesis that fission is the mechanism by which close binaries are formed has regained favor in recent years. Those responsible for its revival (Lynden-Bell 1964, 1965; James 1964; Stoeckly 1965; Roxburgh 1966; Bodenheimer and Ostriker 1970; Lebovitz 1972, 1974) have rebutted earlier theoretical objections (see also Ostriker 1970) and have discussed the hypothesis in the context of our current understanding of pre-main-sequence evolution. The early history of the fission hypothesis and the related investigations into the figures of equilibrium of rotating liquids has been summarized by Chandrasekhar (1969).

Although fission is now commonly considered to be the most likely explanation for the existence of close binaries, the hypothesis cannot be regarded as proved until the evolution of a rotating protostar has been followed from an initial state as a single star to a final state as a detached binary system. This is a formidable problem, however, since it requires the ability to compute the three-dimensional motion of a self-gravitating, compressible gas. Fortunately, some simplifying circumstances make it less than forbidding. First, the high frequency of close binaries over a wide mass range surely implies that no special characteristics of the properties of stellar matter are essential to binary formation; consequently, these properties need not be treated accurately.

A second and crucial simplification concerns the spatial resolution of the calculation. Because the initial departure from axial symmetry is due to the onset of dynamical overstability for a mode of low order, we might reasonably hope that the subsequent evolution can be adequately followed with a low-resolution description of the protostar's structure. If this is indeed so, the problem can be tackled with present-day computers.

On the assumption, therefore, that a decisive test of

the fission hypothesis might be provided by a three-dimensional gas dynamical calculation of low spatial resolution, the bulk of this paper is devoted to describing (Sec. II) and testing (Sec. III) a numerical scheme for carrying out such calculations. This scheme is then used (Sec. IV) to follow the contraction of a rotating protostar and results illustrating the fission mechanism are obtained.

### I. ASSUMPTIONS AND EQUATIONS

In this section, after stating our assumptions, we derive the basic equations in the form used when applying the numerical technique of Sec. II.

(a) *Assumptions.* A rotating, axisymmetric, optically thick protostar of homogeneous composition will be the starting point of the calculation, and this protostar's evolution will be followed up to and beyond the point of instability to a nonaxisymmetric perturbation. To ensure that contraction does not halt prior to this point, energy generation by nuclear burning will be omitted. Accordingly, the basic equations are those describing the motion of a self-gravitating, compressible gas with entropy changes occurring only as a result of radiative conduction.

In accordance with the argument that the detailed properties of stellar matter cannot be of decisive importance, we assume that the matter is a fully ionized perfect gas and that radiation pressure may be neglected; the ratio of specific heats  $\gamma$  and the mean molecular weight  $\mu$  are then constants. In addition, we assume that the opacity  $\kappa$  is independent of state variables.

(b) *Units.* In the interest of computational accuracy, it is useful to express dimensions in terms of a time-dependent length scale  $R(t)$  chosen so as to largely eliminate the protostar's contraction. We also adopt  $M_*$ , the protostar's mass, as the unit of mass,  $\tau_* = (R^3/GM)^{1/2}$  as the unit of time, and  $T_* = (\mu m_H/k)(GM/R)$  as the unit of temperature. In terms of these basic units, we now take  $R/\tau_*$  to be the unit of velocity,  $1/\tau_*$  to be

*Mon. Not. R. astr. Soc.* (1977) 181, 375–389

## Smoothed particle hydrodynamics: theory and application to non-spherical stars

R. A. Gingold and J. J. Monaghan\* *Institute of Astronomy,  
Madingley Road, Cambridge, CB3 0HA*

Received 1977 May 5, in original form February 2

**Summary.** A new hydrodynamic code applicable to a space of an arbitrary number of dimensions is discussed and applied to a variety of polytropic stellar models. The principal feature of the method is the use of statistical techniques to recover analytical expressions for the physical variables from a known distribution of fluid elements. The equations of motion take the form of Newtonian equations for particles. Starting with a non-axisymmetric distribution of approximately 80 particles in three dimensions, the method is found to reproduce the structure of uniformly rotating and magnetic polytropes to within a few per cent. The method may be easily extended to deal with more complicated physical models.

### 1 Introduction

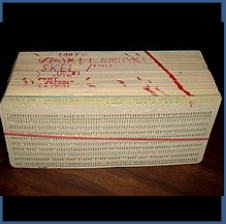
Many of the most interesting problems in astrophysics involve systems with large departures from spherical symmetry. This may occur either because the initial state lacks spherical symmetry, as in the case of a protostar forming from a dense interstellar cloud, or because non-spherical forces arising from rotation or magnetic fields, as in the case of the fission of a rotating star, play an important part in the dynamics. Frequently these sources of non-spherical symmetry will be found combined.

Because of the complexity of these systems numerical methods are required to follow their evolution. However, the standard finite difference representations of the continuum equations are of limited use, because of the very large number of grid points required to treat each coordinate on an equal footing. If, for example, 20 points along the radial direction give adequate accuracy for a spherical polytrope, we may require  $(20)^3$  such points to give the same accuracy for a highly distorted polytrope. This difficulty is mirrored in the evaluation of multiple integrals.

For the astrophysical problems a numerical method which allows reasonable accuracy for a small number of points is required. Ideally it should also be simple to program and robust. An early attempt to provide such an alternative to the standard finite difference method was made by Pasta & Ulam (1959). They replaced the continuous fluid by a fictitious set of

\*Permanent address: Mathematics Department, Monash University, Clayton, Victoria 3168, Australia.

<sup>1)</sup>Permanent address: Department of Astronomy, Columbia University, New York, NY 10027.



# Historical Overview

1977AJ.....82..1013L

THE ASTRONOMICAL JOURNAL

VOLUME 82, NUMBER 12

DECEMBER 1977

## A numerical approach to the testing of the fission hypothesis

L. B. Lucy<sup>1)</sup>

*Institute of Astronomy, Cambridge, United Kingdom  
European Southern Observatory, Geneva, Switzerland  
(Received 12 August 1977; revised 16 September 1977)*

A finite-size particle scheme for the numerical solution of two- and three-dimensional gas dynamical problems of astronomical interest is described and tested. The scheme is then applied to the fission problem for optically thick protostars. Results are given, showing the evolution of one such protostar from an initial state as a single, rotating star to a final state as a triple system whose components contain 60% of the original mass. The decisiveness of this numerical test of the fission hypothesis and its relevance to observed binaries are briefly discussed.

### INTRODUCTION

THE hypothesis that fission is the mechanism by which the fission hypothesis might be provided by a three-dimensional gas dynamical calculation of low spatial resolution, the bulk of this paper is devoted to describing

## Useful reviews:

- Monaghan (1992) ARAA 30: 543--574
- Monaghan (2005) Rep. Prog. Phys. 68: 1703--1759
- Springel (2010) ARAA 48:391--430
- Price (2012) J Comp Phys. 759, 231

## A PowerPoint presentation focusing on magnetic fields in SPH:

- <https://jameswurster.bitbucket.io/files/spmhd.pdf>

Departure from axial symmetry is due to the onset of dynamical overstability for a mode of low order, we might reasonably hope that the subsequent evolution can be adequately followed with a low-resolution description of the protostar's structure. If this is indeed so, the problem can be tackled with present-day computers.

On the assumption, therefore, that a decisive test of

weight  $\mu$  are then constants. In addition, we assume that the opacity  $\kappa$  is independent of state variables.

(b) *Units.* In the interest of computational accuracy, it is useful to express dimensions in terms of a time-dependent length scale  $R(t)$  chosen so as to largely eliminate the protostar's contraction. We also adopt  $M$ , the protostar's mass, as the unit of mass,  $\tau_* = (R^3/GM)^{1/2}$  as the unit of time, and  $T_* = (\mu m_H/k)(GM/R)$  as the unit of temperature. In terms of these basic units, we now take  $R/\tau_*$  to be the unit of velocity,  $1/\tau_*$  to be

*Mon. Not. R. astr. Soc.* (1977) 181, 375–389

## Smoothed particle hydrodynamics: theory and application to non-spherical stars

R. A. Gingold and J. J. Monaghan\* *Institute of Astronomy,  
Madingley Road, Cambridge, CB3 0HA*

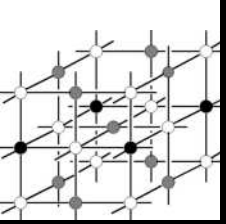
Received 1977 May 5, in original form February 2

tion give adequate accuracy for a spherical polytrope, we may require  $(20)^3$  such points to give the same accuracy for a highly distorted polytrope. This difficulty is mirrored in the evaluation of multiple integrals.

For the astrophysical problems a numerical method which allows reasonable accuracy for a small number of points is required. Ideally it should also be simple to program and robust. An early attempt to provide such an alternative to the standard finite difference method was made by Pasta & Ulam (1959). They replaced the continuous fluid by a fictitious set of

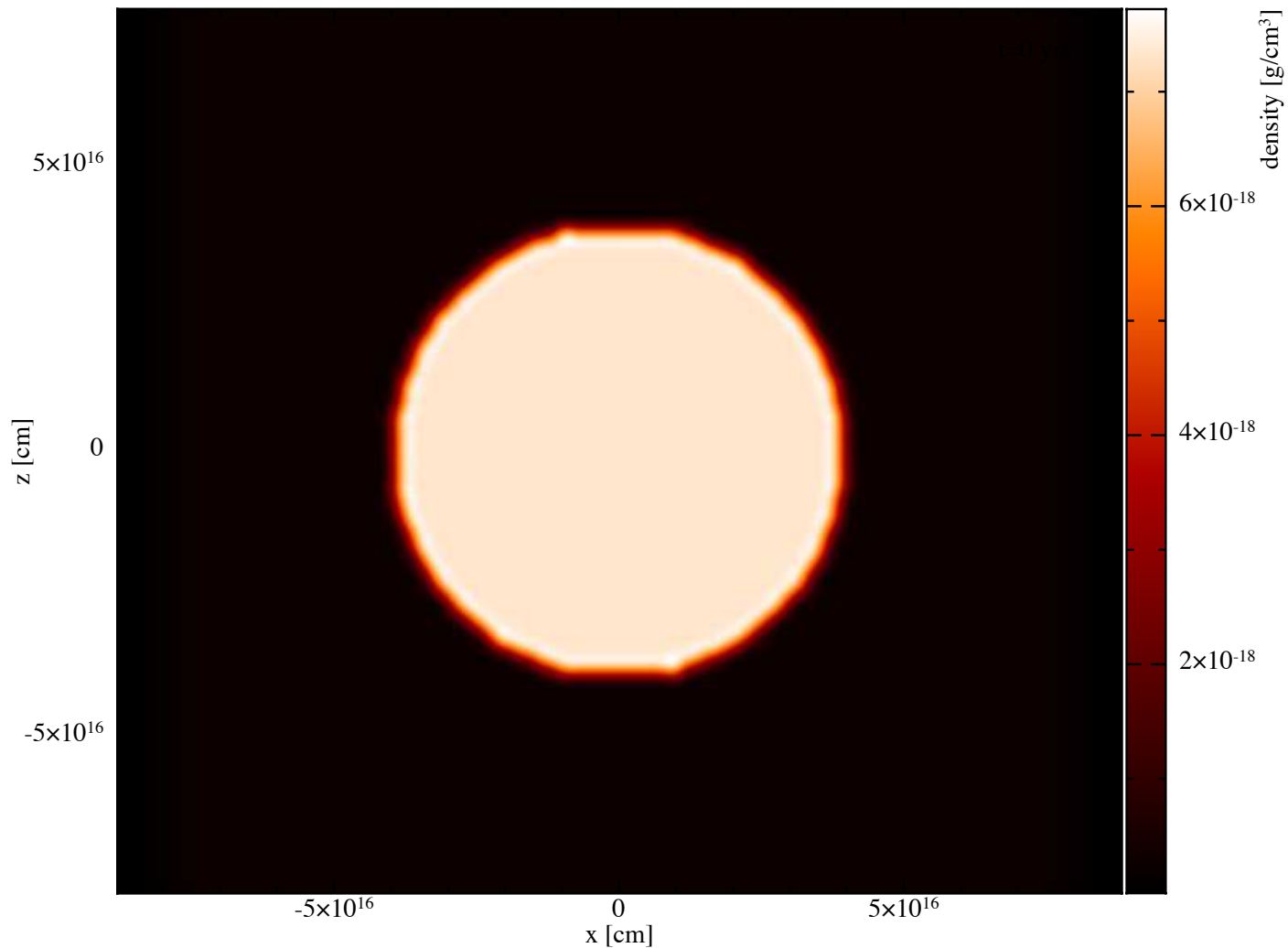
\*Permanent address: Mathematics Department, Monash University, Clayton, Victoria 3168, Australia.

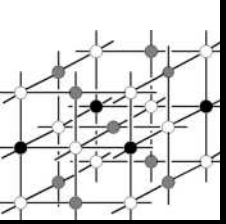
<sup>1)</sup> Permanent address: Department of Astronomy, Columbia University, New York, NY 10027.



# *Dividing the domain*

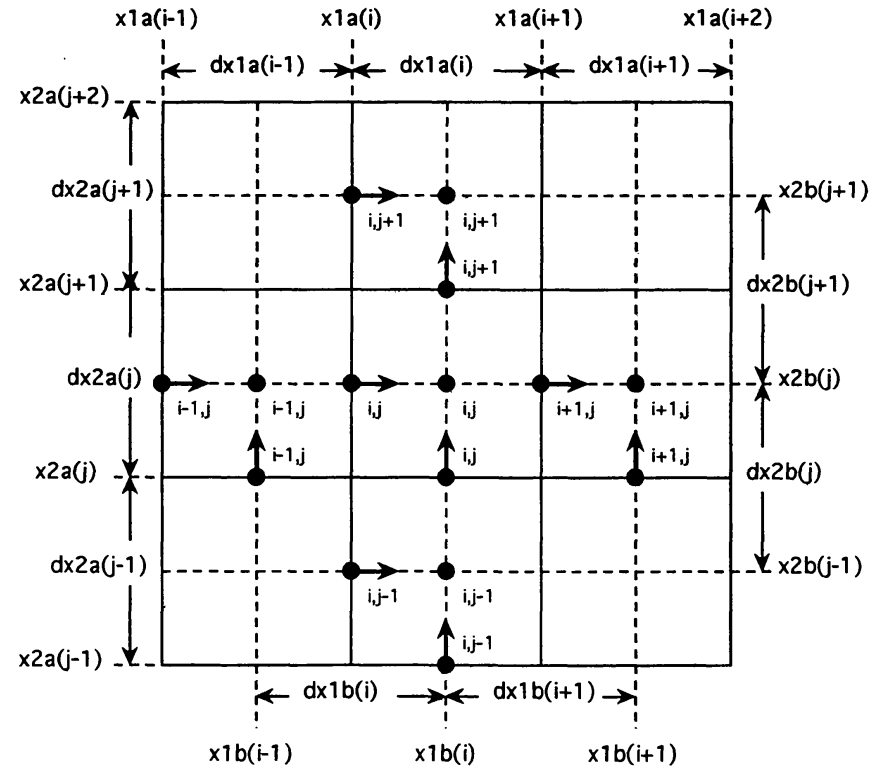
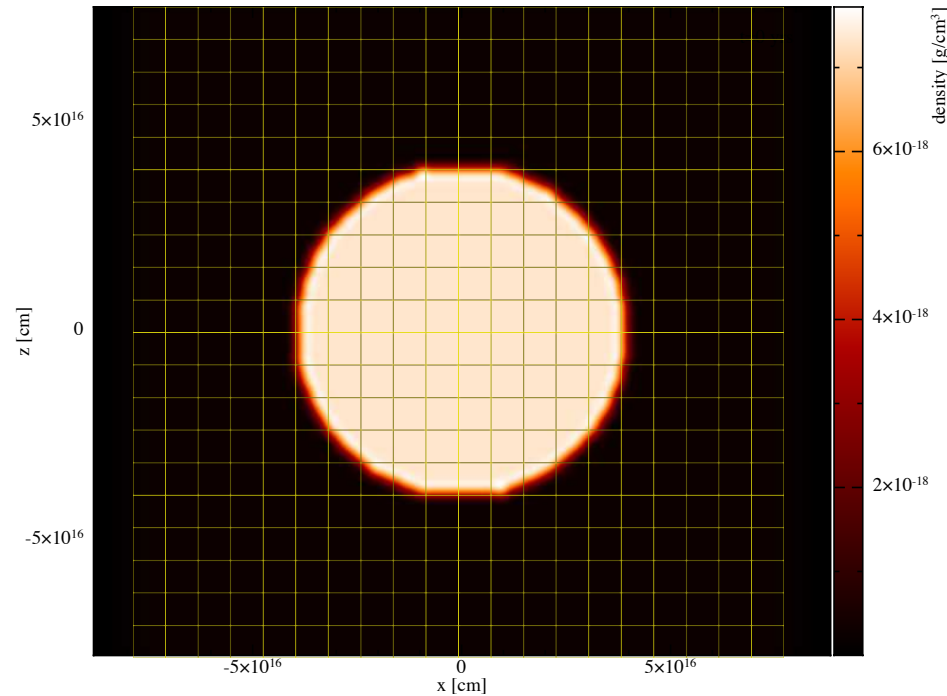
- Given a domain, how do we divide it up?



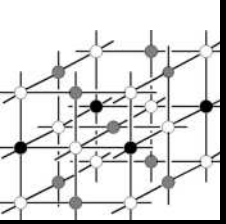


# Dividing the domain: Grid

- Given a domain, how do we divide it up?
  - Eulerian Grid: divide domain into fixed volume elements
    - Cells of well-defined position and volume
    - Scalars are calculated at cell-centre; vectors are calculated at cell-interface

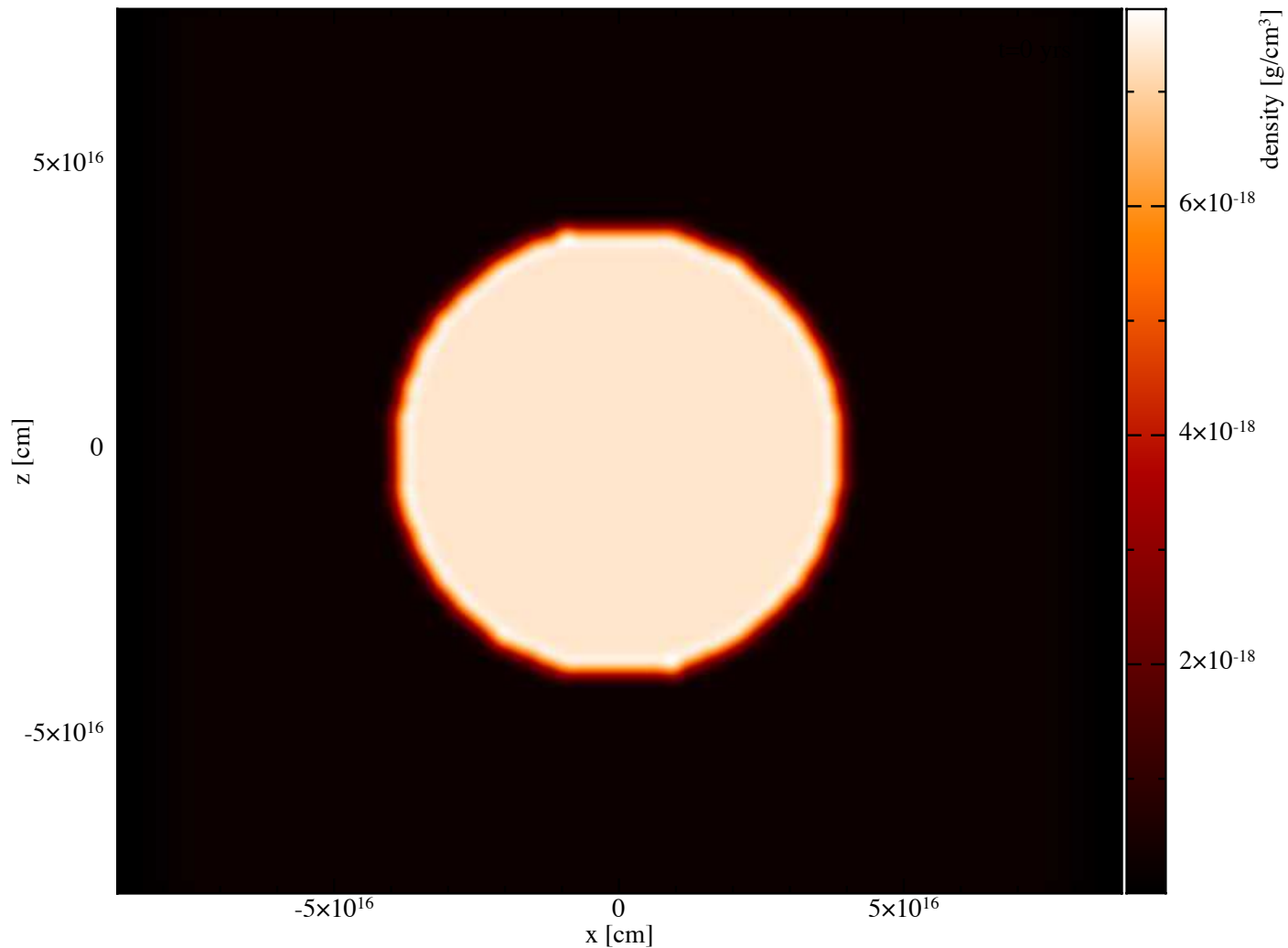


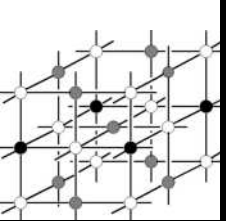




# *Dividing the domain*

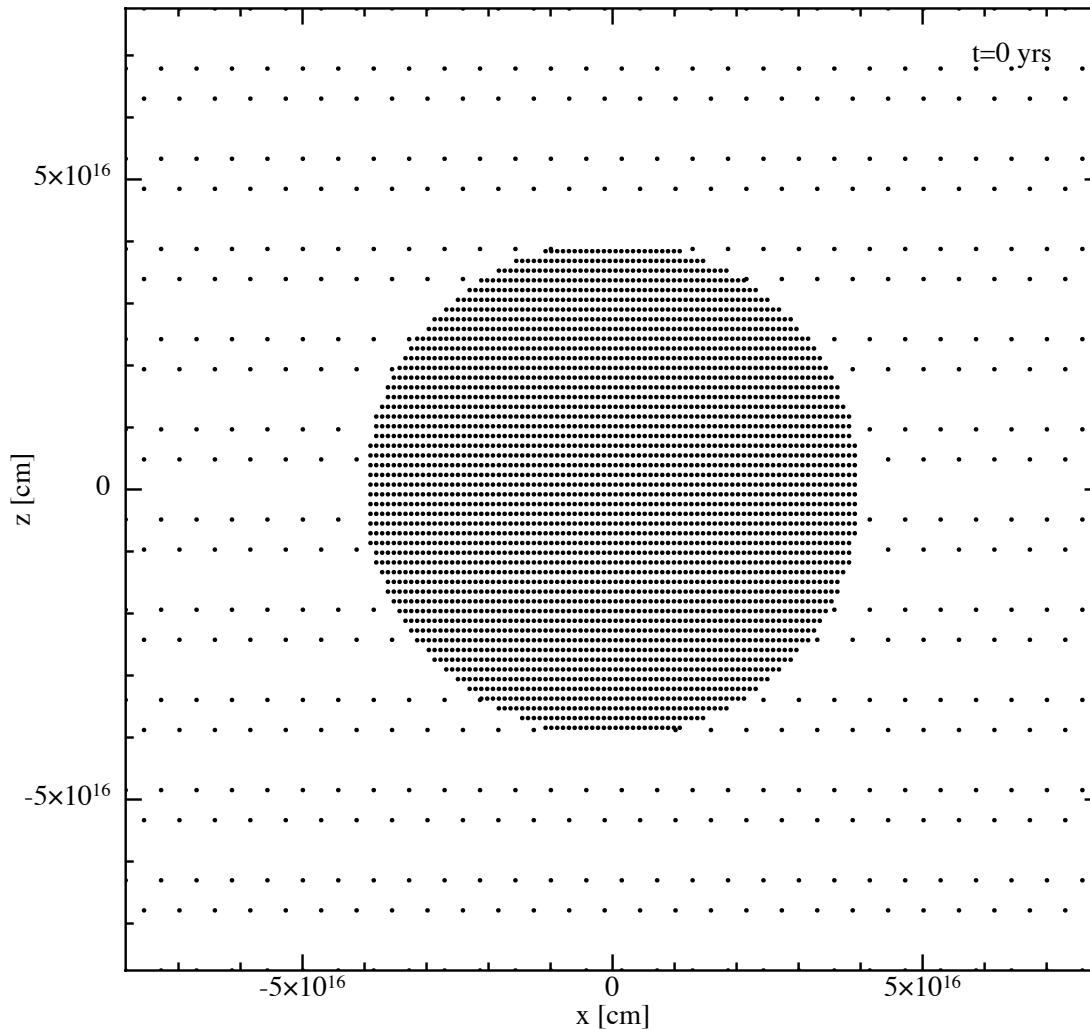
- Given a domain, how do we divide it up?





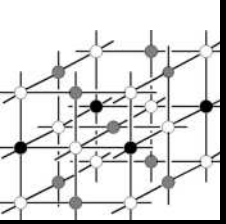
# *Dividing the domain: SPH*

- Given a domain, how do we divide it up?
  - Lagrangian particles: Divide domain into mass elements



- Each particle has *the same fixed mass*
- Dense regions are represented by more particles than less-dense regions
- Characteristics are calculated at the particles' locations

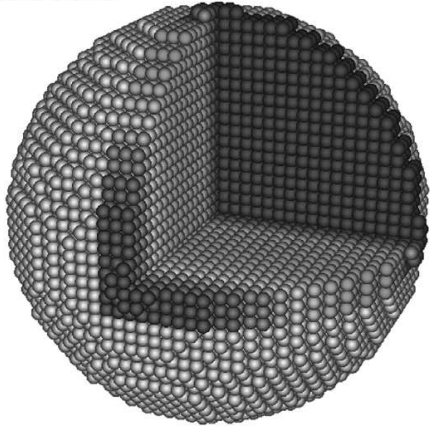




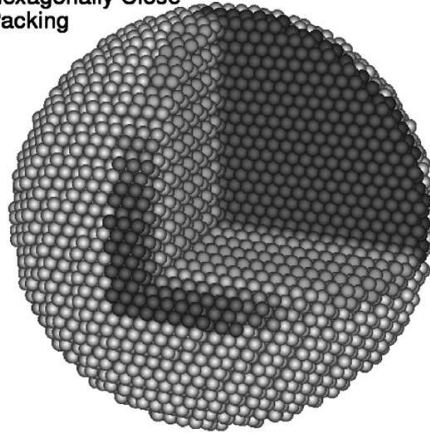
# *Dividing the domain: SPH*

- How do we distribute the initial particles for uniform density (top)? For centrally condensed (bottom)? Does it matter (Morris, 1996)?

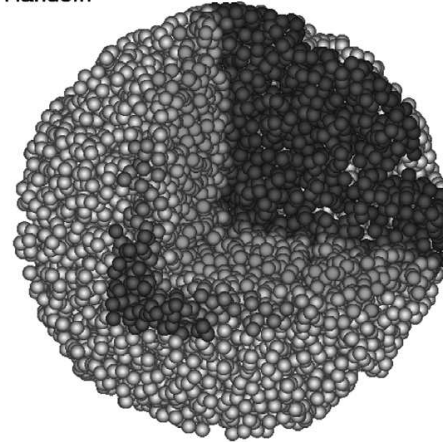
Cubic Lattice



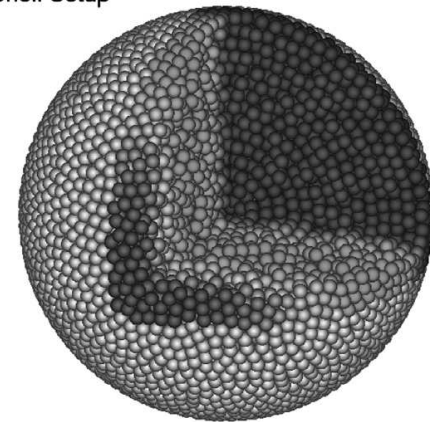
Hexagonally Close Packing



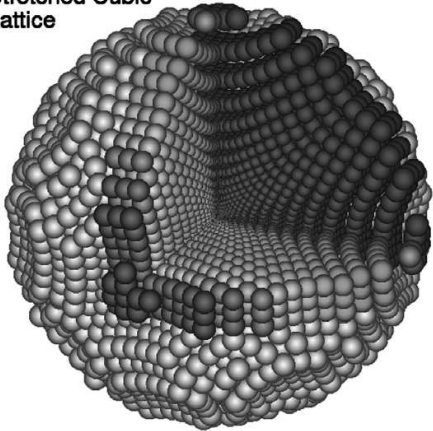
Random



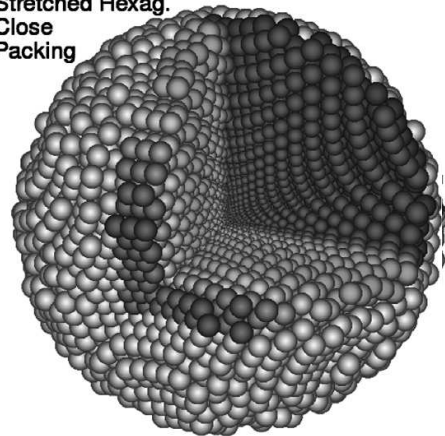
Shell Setup



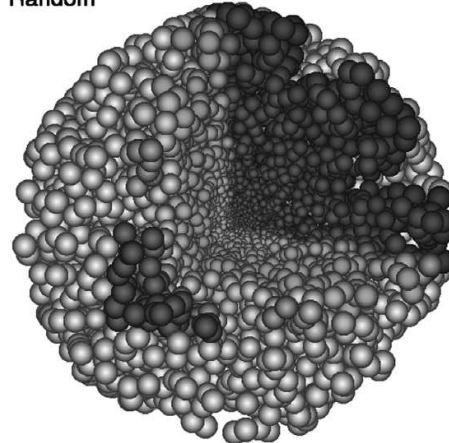
Stretched Cubic Lattice



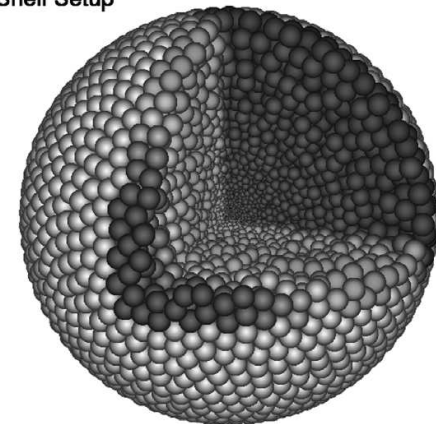
Stretched Hexag. Close Packing



Random



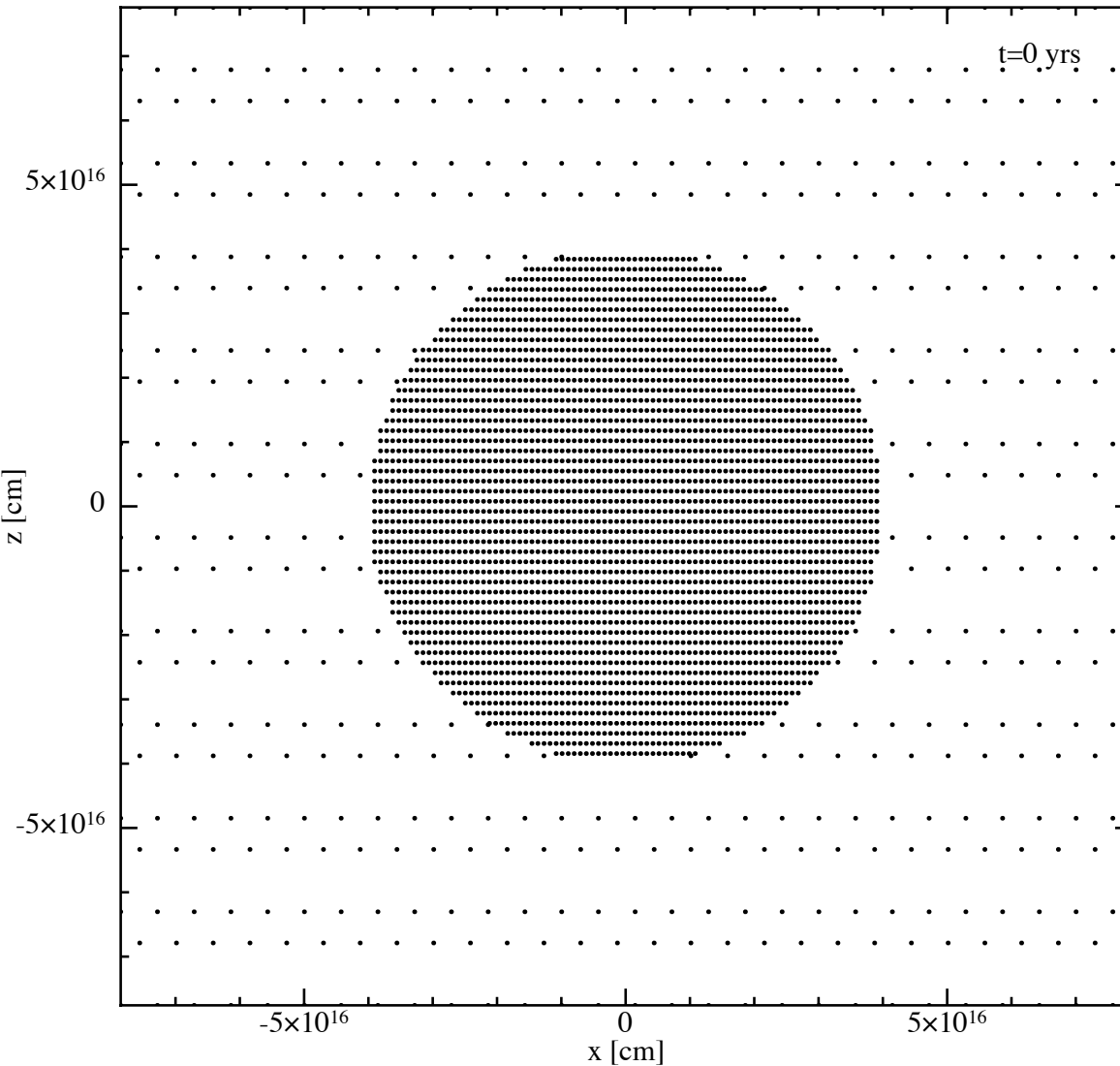
Shell Setup





# Density

➤ How is density calculated?



➤ Density is a fundamental astrophysical property and crucial to astrophysics.

➤ In a grid, density is simply

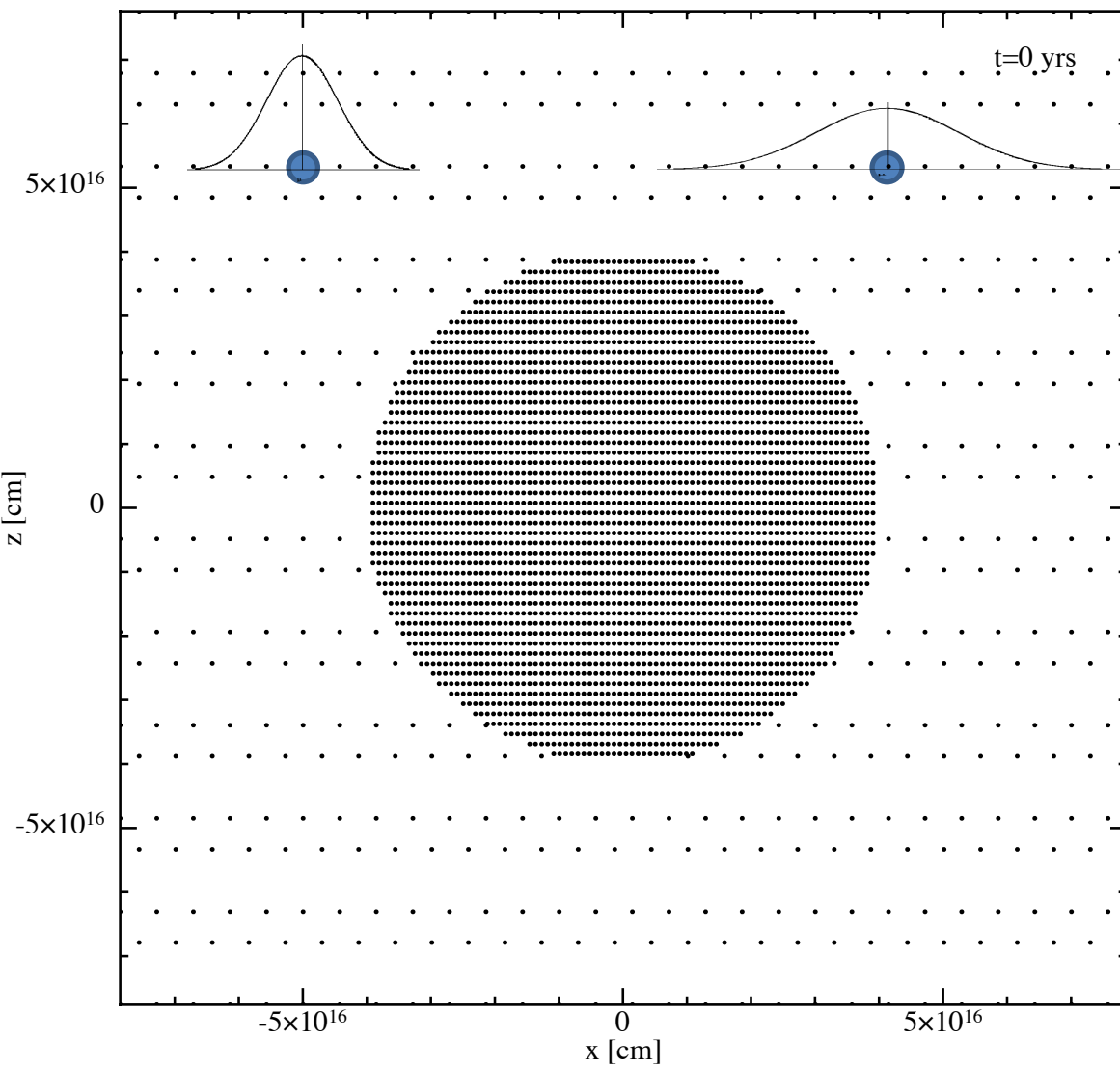
$$\rho(r) = \frac{\text{total mass in cell}}{\text{volume of cell}}$$

➤ How is density calculated on a set of particles?



# Density

➤ How is density calculated?



➤ Density summation

$$\rho(\mathbf{r}) = \sum_{b=1}^{N_{neigh}} m_b W(\mathbf{r} - \mathbf{r}_b, h),$$

➤ where

➤  $N$  is number of neighbours

➤  $m_b$  is particle mass

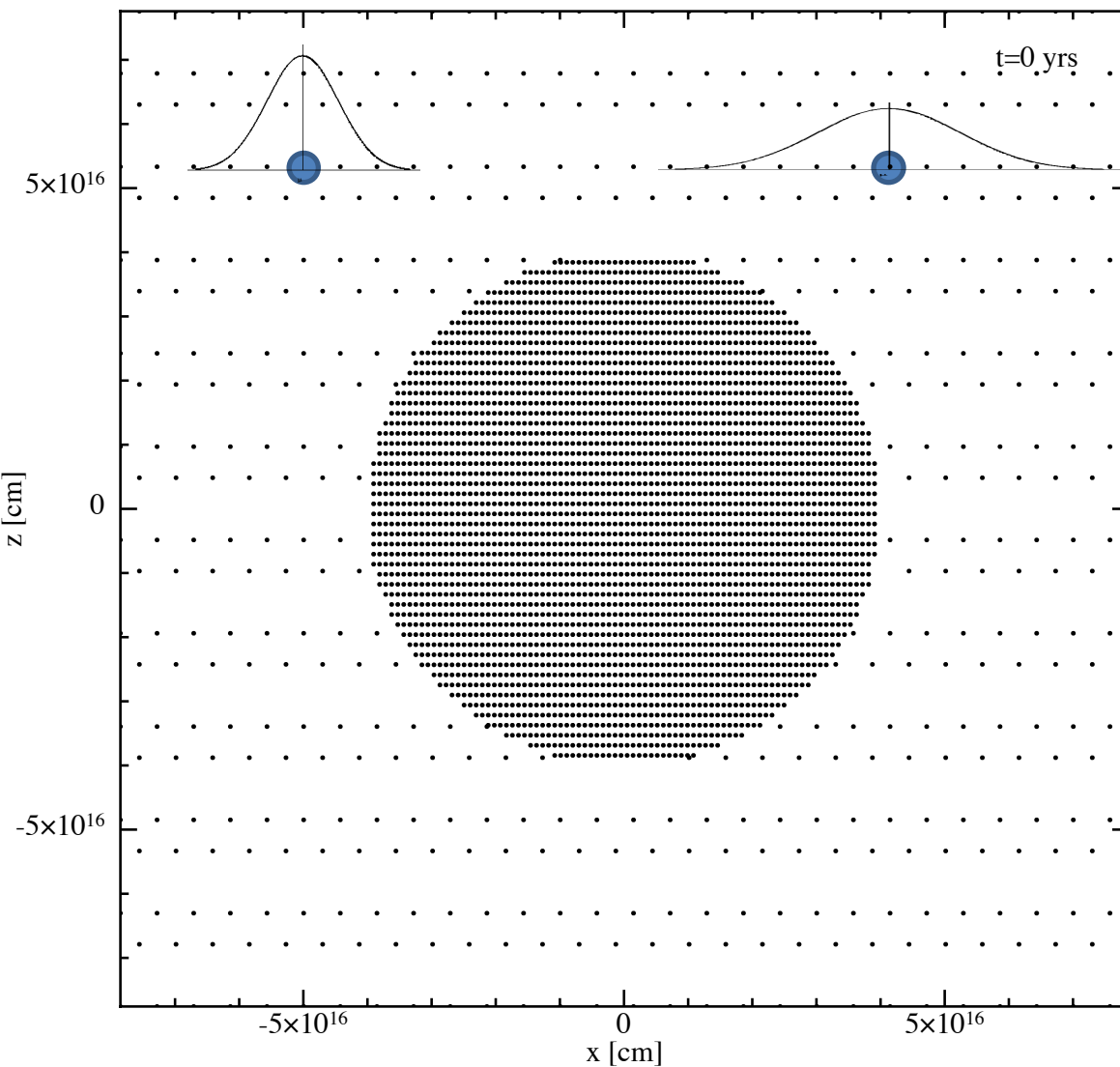
➤  $W$  is smoothing kernel

➤ The kernel is effectively a weighting function to determine how much particle  $b$  should contribute to the value at  $\mathbf{r}$



# Density

➤ How is density calculated?



➤ Density summation

$$\rho(\mathbf{r}) = \sum_{b=1}^{N_{neigh}} m_b W(\mathbf{r} - \mathbf{r}_b, h),$$

➤ where

- $N$  is number of neighbours
- $m_b$  is particle mass
- $W$  is smoothing kernel

➤ Simplest kernel is a Gaussian:

$$W(\mathbf{r} - \mathbf{r}', h) = \frac{\sigma}{h^d} \exp \left[ -\frac{(\mathbf{r} - \mathbf{r}')^2}{h^2} \right]$$

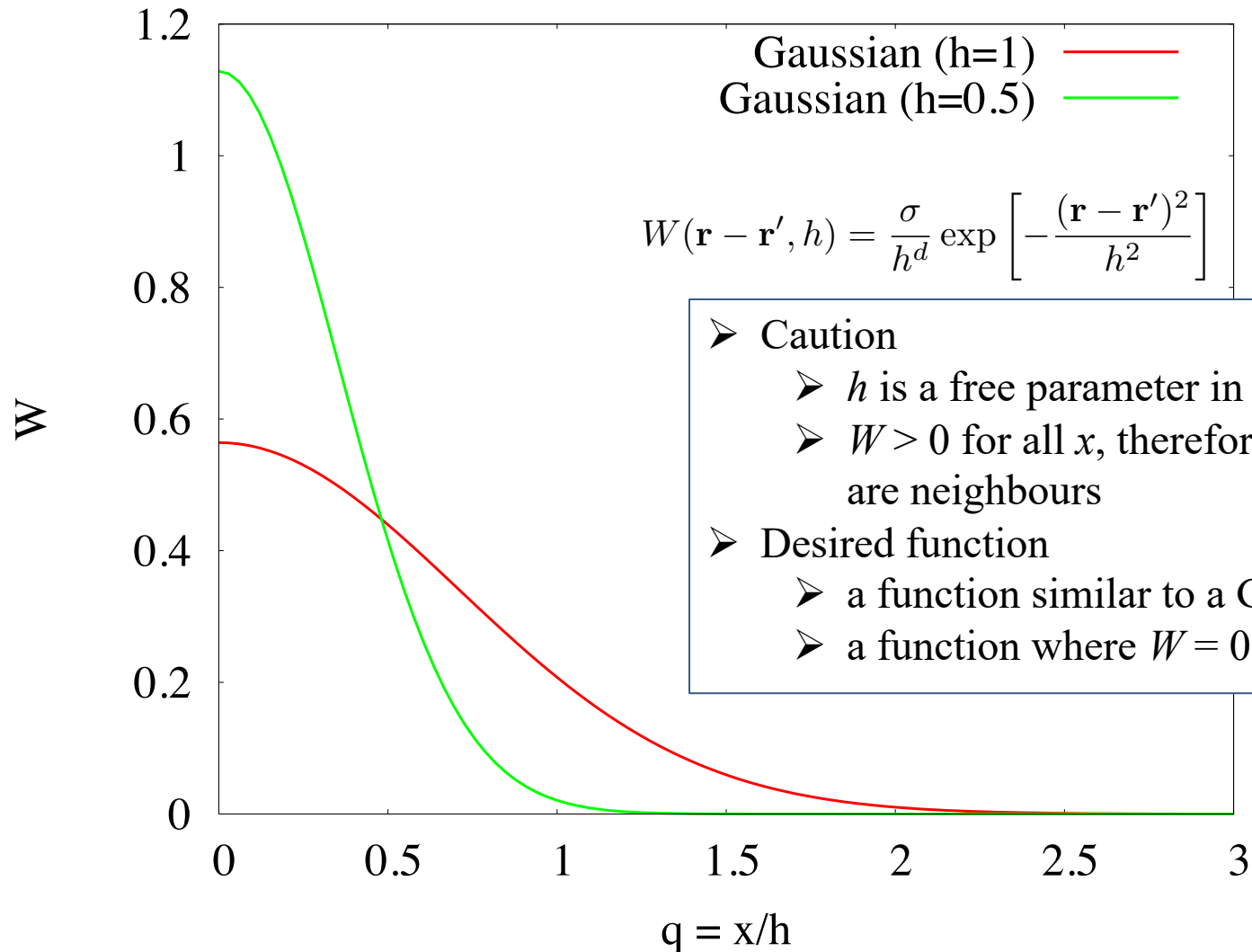
➤ where

- $h$  is a scale parameter



# Density: smoothing kernel

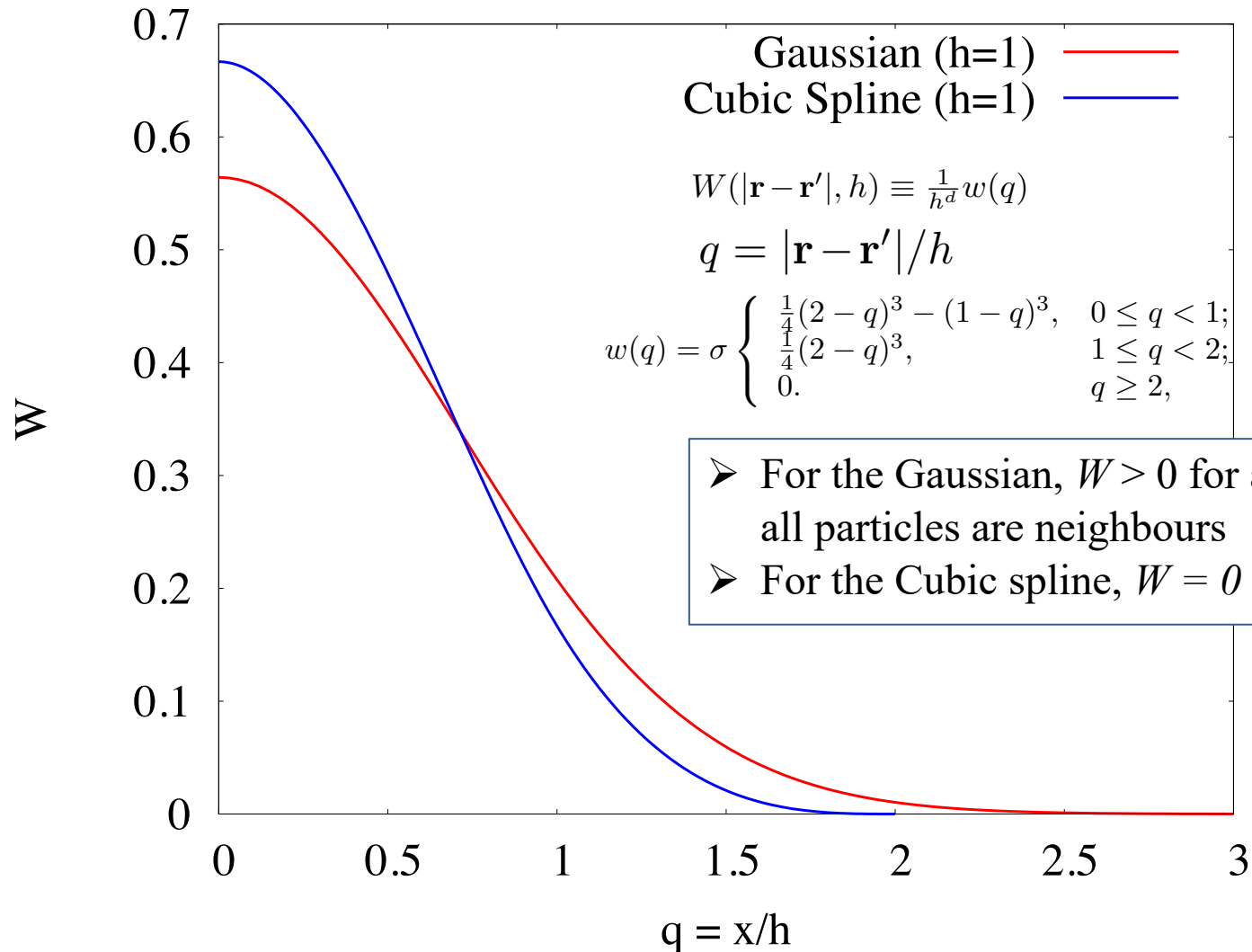
➤ How is density calculated?





# Density: Smoothing kernel

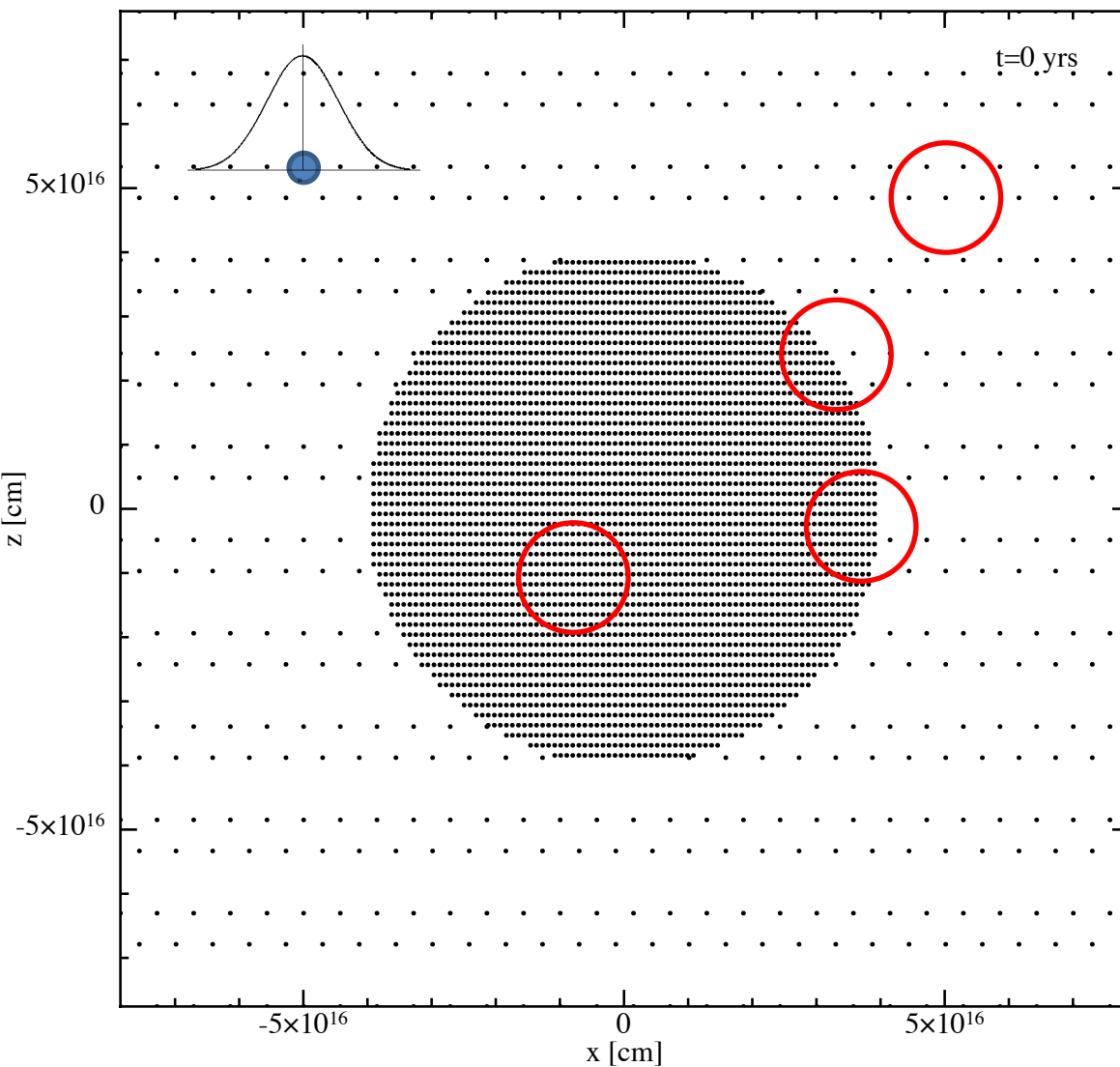
- How is density calculated?





# Density: Determining $h$

➤ How is density calculated?



➤ Density summation

$$\rho(\mathbf{r}) = \sum_{b=1}^{N_{neigh}} m_b W(\mathbf{r} - \mathbf{r}_b, h),$$

➤ Where

➤  $N$  is number of neighbours

➤  $m_b$  is particle mass

➤  $W$  is smoothing kernel

➤ What is  $h$ ?

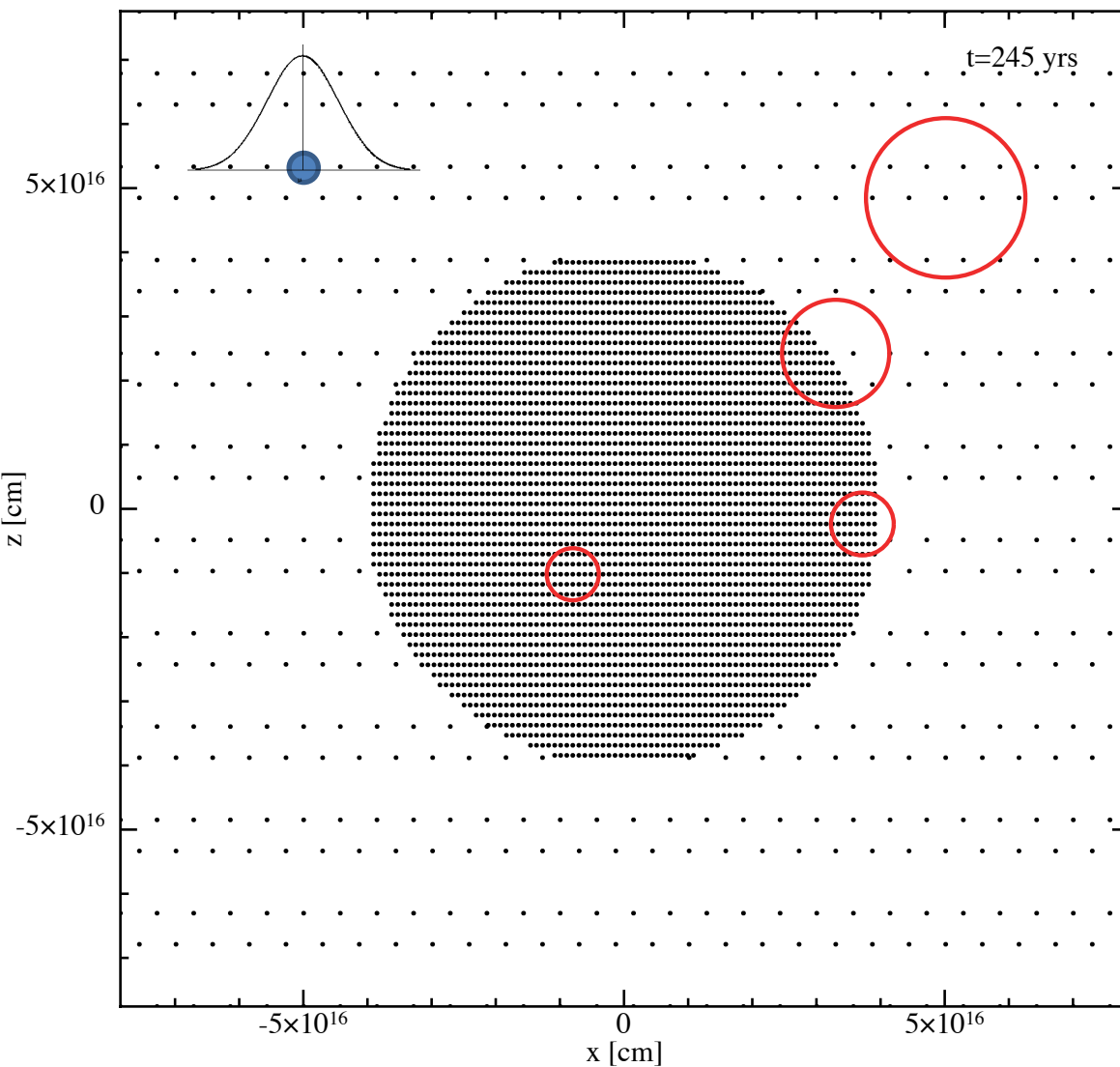
➤ If fixed, then some particles will have many more neighbours than others → bad & inefficient!





# Density: Determining $h$

➤ How is density calculated?



➤ Density summation

$$\rho(\mathbf{r}) = \sum_{b=1}^{N_{neigh}} m_b W(\mathbf{r} - \mathbf{r}_b, h),$$

➤ Where

➤  $N$  is number of neighbours

➤  $m_b$  is particle mass

➤  $W$  is smoothing kernel

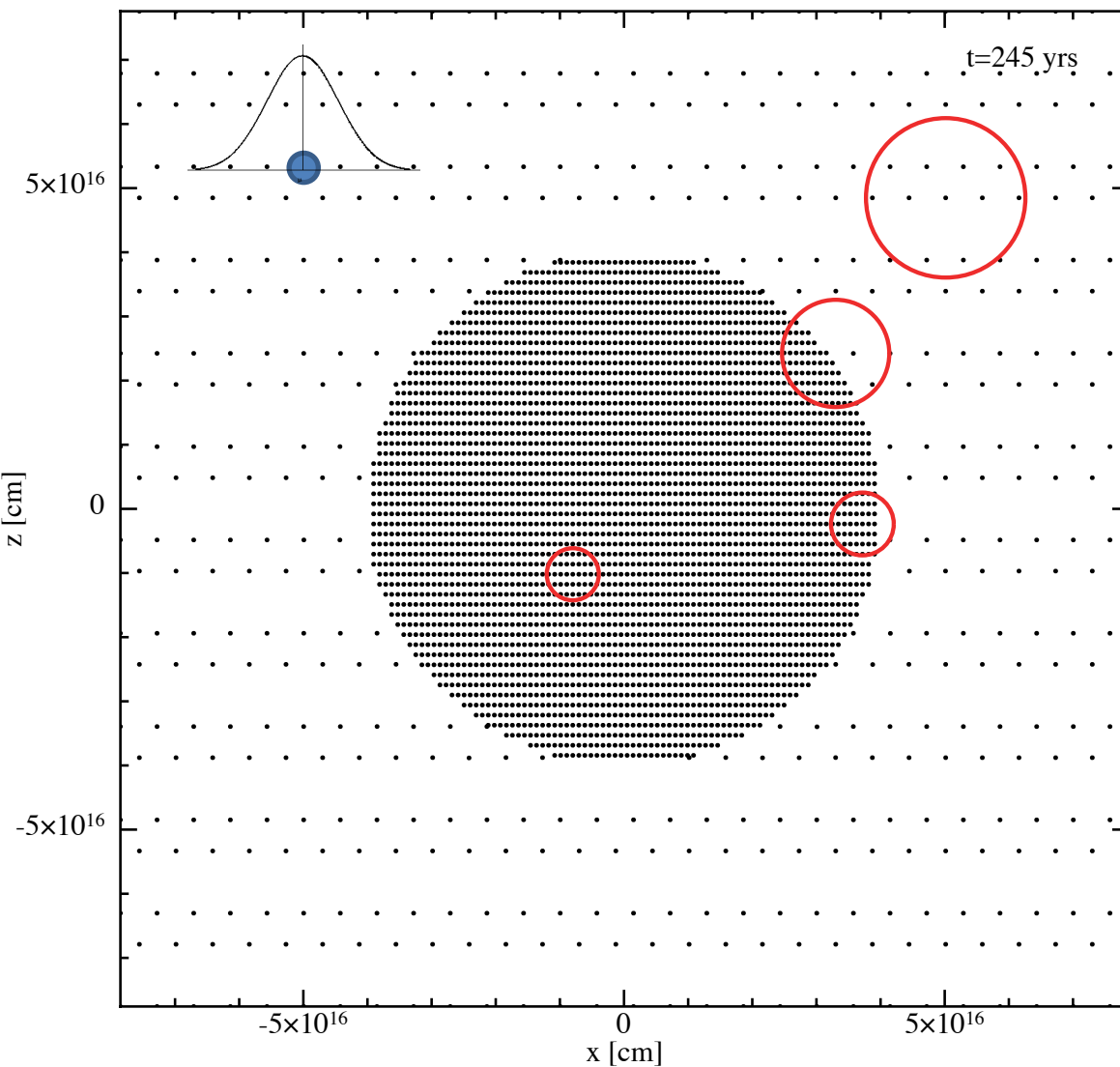
➤ What is  $h$ ?

➤ Variable depending on density



# Density: Determining $h$

➤ How is density calculated?



➤ Density summation

$$\rho_a = \sum_b m_b W_{ab}(\mathbf{r}_a - \mathbf{r}_b, h_a)$$

➤ Smoothing length relation

$$h_a = \eta \left( \frac{m_a}{\rho_a} \right)^{1/3}$$

➤ These equations must be iteratively solved

➤ For a cubic spline in 3D, there will be  $N_{\text{neigh}} \sim 57$

➤ Note:

higher density

↔ smaller smoothing length

↔ higher resolution



# Kernel interpolation

- To calculate a quantity at  $\mathbf{r}$  from its neighbours at  $\mathbf{r}'$ :

$$A(\mathbf{r}) = \int A(\mathbf{r}')\delta(\mathbf{r} - \mathbf{r}')d\mathbf{r}',$$

- Identity

$$= \int A(\mathbf{r}')W(\mathbf{r} - \mathbf{r}', h)d\mathbf{r}' + \mathcal{O}(h^2),$$

- Replace  $\delta$ -function with kernel

$$\approx \int \frac{A(\mathbf{r}')}{\rho(\mathbf{r}')}W(\mathbf{r} - \mathbf{r}', h)\rho(\mathbf{r}')d\mathbf{r}',$$

- multiply  $\rho/\rho$

$$\approx \sum_{b=1}^{N_{\text{neigh}}} m_b \frac{A_b}{\rho_b} W(\mathbf{r} - \mathbf{r}_b, h)$$

- replace integral with summation
- replace volume integral  $\rho dV$  with mass



# First Derivatives

## ➤ First derivative

$$\begin{aligned}\nabla A(\mathbf{r}) &= \frac{\partial}{\partial \mathbf{r}} \int A(\mathbf{r}') W(\mathbf{r} - \mathbf{r}', h) d\mathbf{r}' + \mathcal{O}(h^2), \\ &\approx \sum_{b=1}^{N_{\text{neigh}}} m_b \frac{A_b}{\rho_b} \nabla W(\mathbf{r} - \mathbf{r}_b, h)\end{aligned}$$

## ➤ Warning!

- Derivative is not zero if  $A$  is constant
- A simple way to correct this is:

$$\begin{aligned}\nabla A &= \frac{1}{\Phi} [\nabla (\Phi A) - A \nabla \Phi], \\ \nabla A_a &= \frac{1}{\Phi_a} \left[ \sum_{b=1}^{N_{\text{neigh}}} m_b \frac{\Phi_b A_b}{\rho_b} \nabla W(\mathbf{r}_a - \mathbf{r}_b, h) - \sum_{b=1}^{N_{\text{neigh}}} m_b \frac{\Phi_b A_a}{\rho_b} \nabla W(\mathbf{r}_a - \mathbf{r}_b, h) \right], \\ &= \frac{1}{\Phi_a} \sum_{b=1}^{N_{\text{neigh}}} \frac{m_b \Phi_b}{\rho_b} (A_b - A_a) \nabla W(\mathbf{r}_a - \mathbf{r}_b, h),\end{aligned}$$



# First Derivatives

$$\begin{aligned}\nabla A &= \frac{1}{\Phi} [\nabla (\Phi A) - A \nabla \Phi], \\ \nabla A_a &= \frac{1}{\Phi_a} \left[ \sum_{b=1}^{N_{\text{neigh}}} m_b \frac{\Phi_b A_b}{\rho_b} \nabla W(\mathbf{r}_a - \mathbf{r}_b, h) - \sum_{b=1}^{N_{\text{neigh}}} m_b \frac{\Phi_b A_a}{\rho_b} \nabla W(\mathbf{r}_a - \mathbf{r}_b, h) \right], \\ &= \frac{1}{\Phi_a} \sum_{b=1}^{N_{\text{neigh}}} \frac{m_b \Phi_b}{\rho_b} (A_b - A_a) \nabla W(\mathbf{r}_a - \mathbf{r}_b, h),\end{aligned}$$

- where  $\phi$  is a scalar and  $h$  is assumed to be constant
- The two obvious choices are  $\phi = 1$ :

$$\nabla A_a = \frac{1}{\rho_a} \sum_{b=1}^{N_{\text{neigh}}} m_b (A_b - A_a) \nabla W(\mathbf{r}_a - \mathbf{r}_b, h),$$

- and  $\phi = \rho$ :

$$\nabla A_a = \sum_{b=1}^{N_{\text{neigh}}} \frac{m_b}{\rho_b} (A_b - A_a) \nabla W(\mathbf{r}_a - \mathbf{r}_b, h),$$

- Additional terms are required if  $h$  varies in space & time (see review papers)



# *Continuum Hydrodynamic Equations*

➤ Continuum Equations:

➤ Continuity Equation

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{v}$$

➤ Equation of Motion

$$\frac{d\mathbf{v}}{dt} = -\frac{1}{\rho} \nabla P$$

➤ Energy Equation

$$\frac{du}{dt} = -\frac{P}{\rho} \nabla \cdot \mathbf{v}$$

➤ Equation of state (e.g.)

$$P = (\gamma - 1) \rho u$$

# Discrete Hydrodynamic Equations

## ➤ Discrete Equations:

### ➤ Density Equation

$$\rho_a = \sum_b m_b W_{ab}(h_a); \quad h_a = \eta \left( \frac{m_a}{\rho_a} \right)^{1/3}$$

### ➤ Equation of motion

$$\frac{dv_a^i}{dt} = \sum_b m_b \left[ \frac{P_a}{\Omega_a \rho_a^2} \nabla_a^i W_{ab}(h_a) + \frac{P_b}{\Omega_b \rho_b^2} \nabla_a^i W_{ab}(h_b) \right]$$

### ➤ Energy Equation

$$\frac{du_a}{dt} = \frac{P_a}{\Omega_a \rho_a^2} \sum_b m_b v_{ab}^i \nabla_a^i W_{ab}(h_a)$$

### ➤ Equation of state (e.g.)

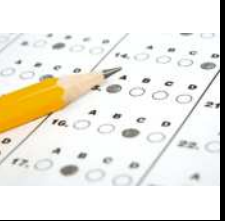
$$P_a = (\gamma - 1) \rho_a u_a$$

### ➤ where

these values are calculated for particle  $a$  using the properties of all the neighbours,  $b$

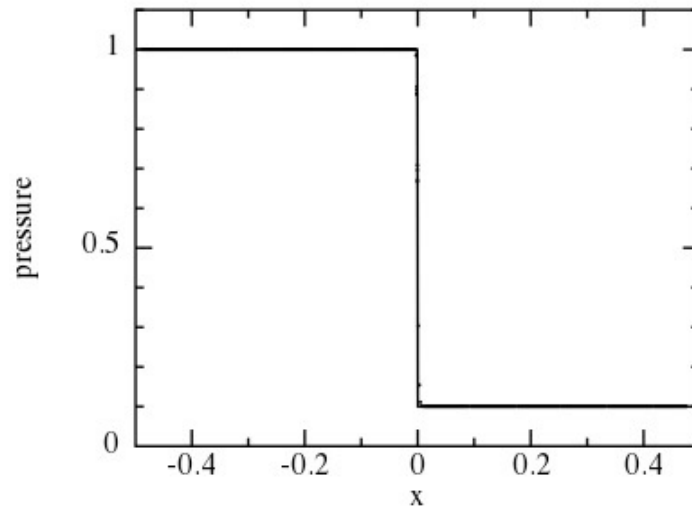
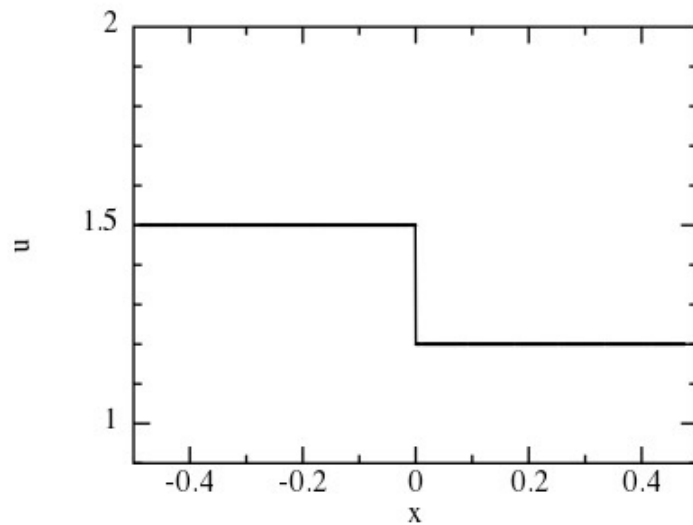
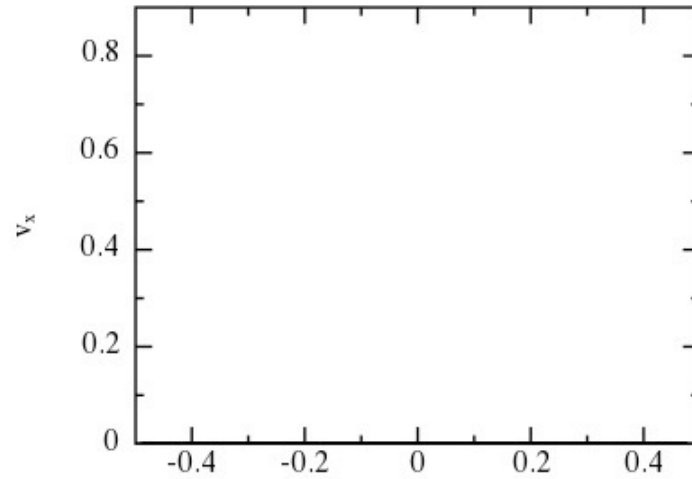
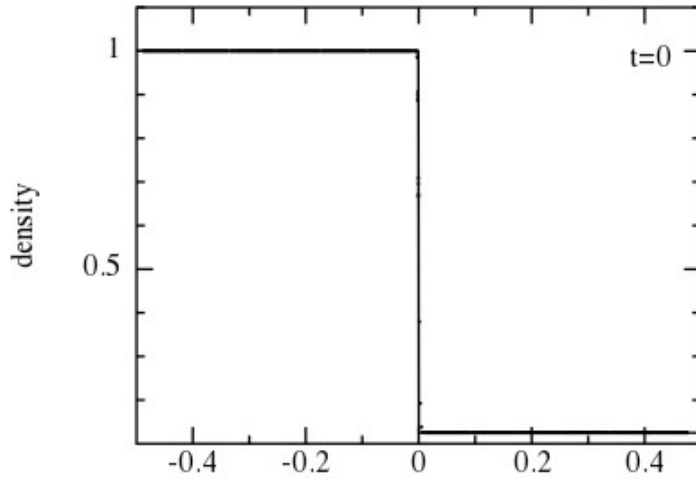
$$W_{ab}(h_a) \equiv W_{ab}(\mathbf{r}_a - \mathbf{r}_b, h_a); \quad v_{ab}^i \equiv v_a^i - v_b^i; \quad \Omega_a \text{ accounts for variable } h$$

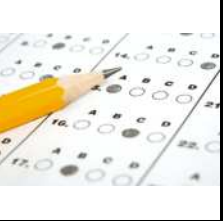




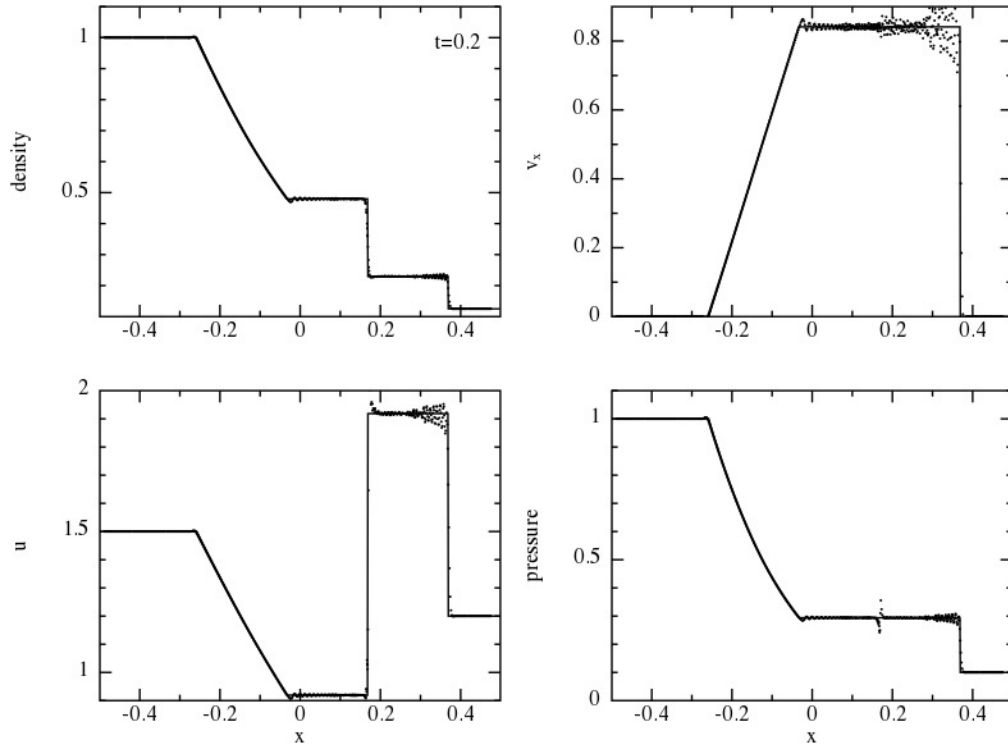
# Sod Shock Tube

➤ As written on the previous slide





# Sod Shock Tube – what went wrong?

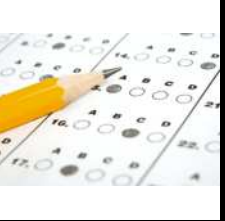


- The previous equations were exact numerical representations of the continuum equations
  - Did not account for numerical effects
- Require numerical terms to compensate for artificial numerical processes:
  - Equation of motion:
    - Artificial viscosity
  - Energy Equation:
    - Artificial conductivity

Various forms:

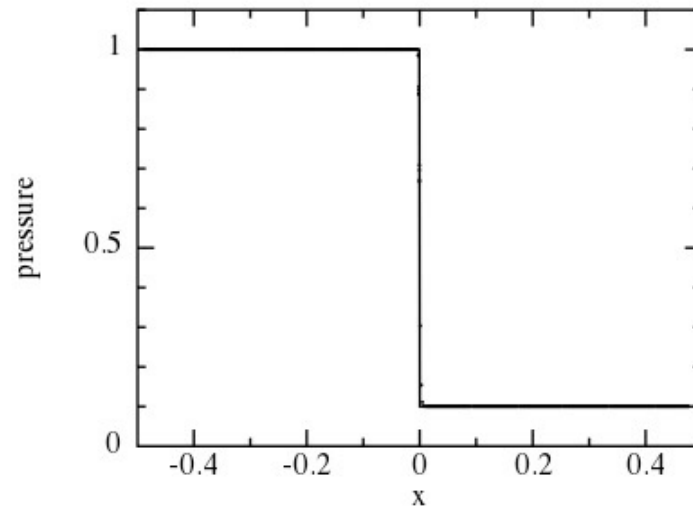
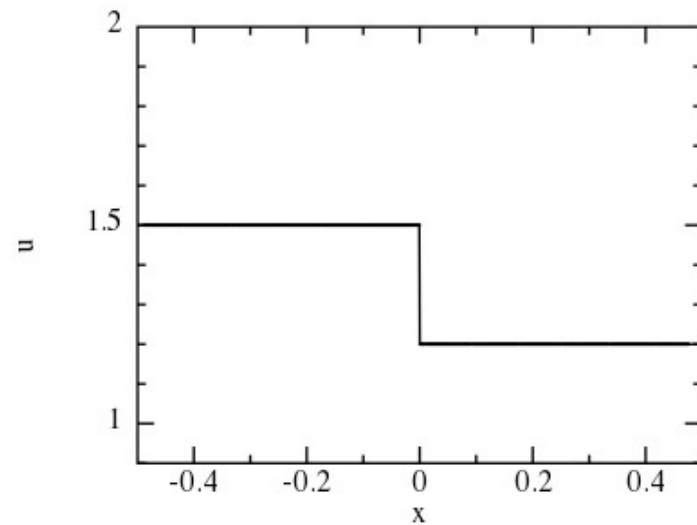
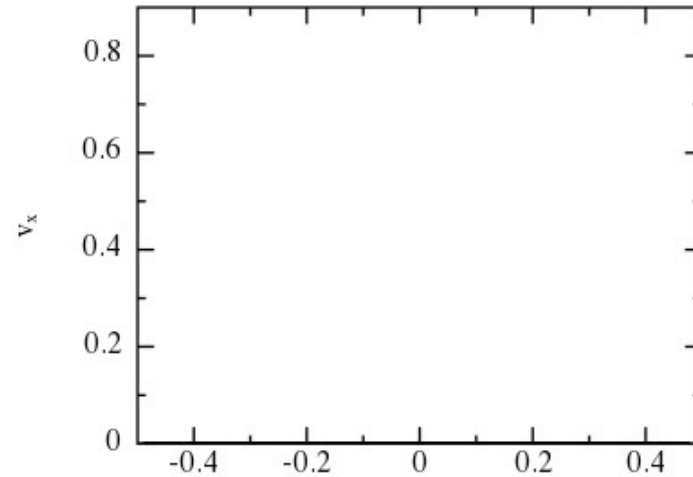
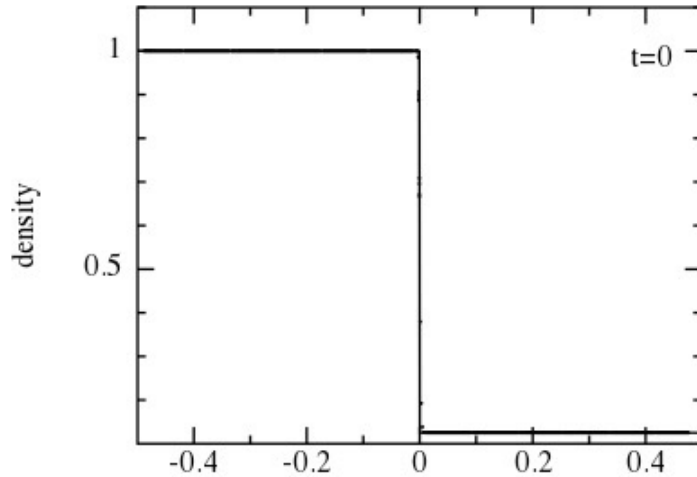
see (e.g.) Monaghan (1992, 2005), Springel (2010), Price (2012)





# *Sod Shock Tube – with artificial terms*

➤ Maximal artificial viscosity and conductivity

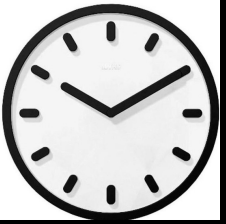




# *Summary to date*

---

- When developing / modifying code:
  - Convert the continuum equations into discrete equations
  - Account for variable smoothing lengths
  - Careful with your choice of discretisation for derivatives
  - Apply artificial terms as required
    - generally only two required for hydrodynamics



# *Timestepping*

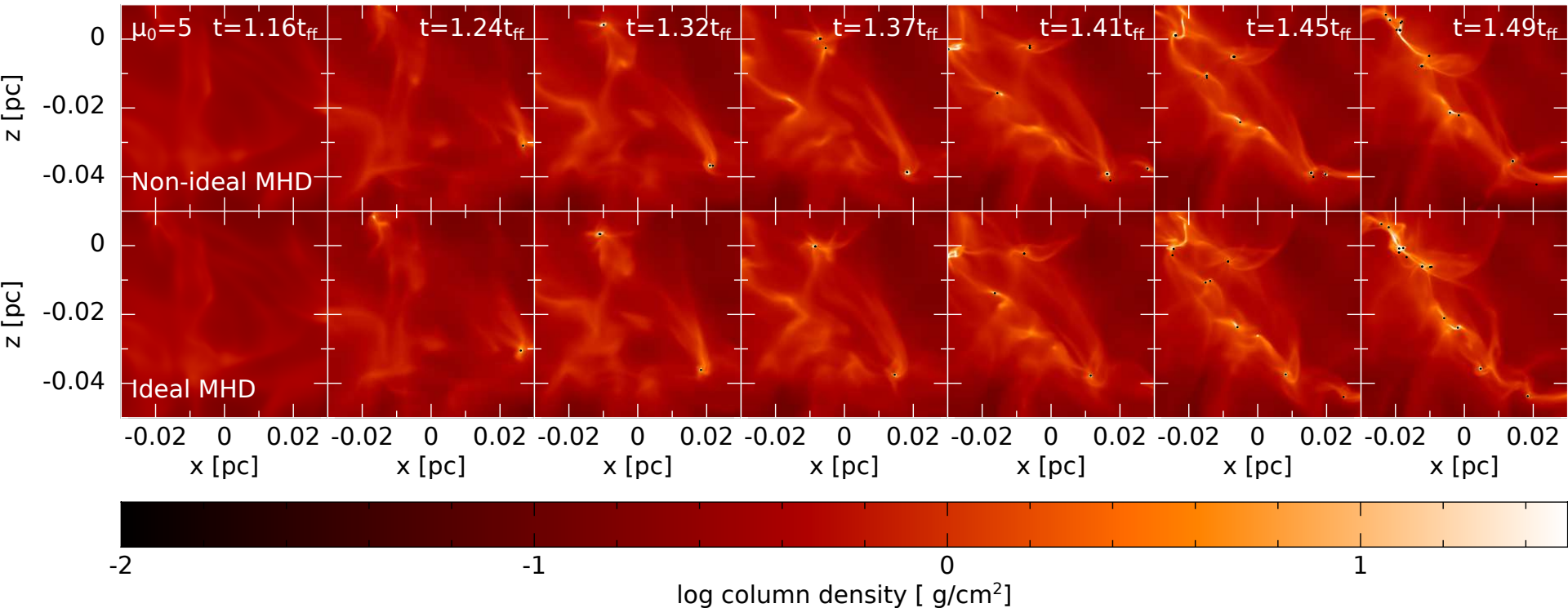
- The timestep must be correctly chosen so that all included processes are resolved in time
- Many physical processes require their own timestep; all are calculated and the smallest is used
- A most general timestep is a CFL-like timestep:

$$dt = C \frac{h}{v_{\text{sig}}}$$

- where  $v_{\text{sig}}$  is the signal velocity, which is related to the sound speed and  $dv$
- $C < 1$  is a coefficient (typically set 0.3)
- For higher densities (i.e. smaller  $h$ ), timestep is shorter
  
- Timestepping options:
  - Global timestepping
    - Every particle is evolved with the same (minimum)  $dt$
    - Excellent energy conservation
    - Simple to implement
  - Individual timestepping
    - Particles are evolved as per their own  $dt$
    - Weaker energy conservation
    - Considerably increases computational efficiency for most projects

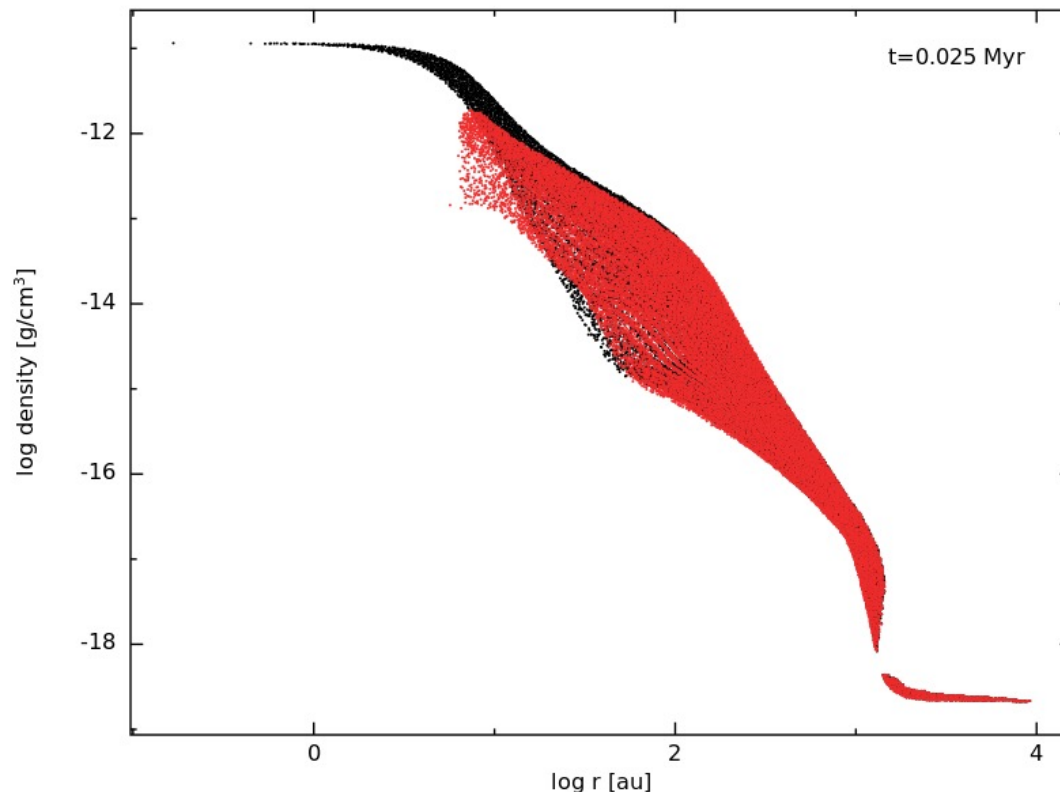
# *Sink particles*

- If gas density becomes too high, then simulation becomes prohibitively expensive to run
- Replace dense regions with ‘sink’ particles (Bate, Bonnell & Price 1995)
  - See star cluster simulation, where each sink represents one star



# *Sink particles*

- If gas density becomes too high, then simulation becomes prohibitively expensive to run
- Replace dense regions with ‘sink’ particles (Bate, Bonnell & Price 1995)
  - Sinks have a fixed (parametrised) radius; in this figure, 6.67au
  - Sinks accrete gas, thus increase their mass
  - Sinks feel *only* gravity: there are no pressure forces and no magnetic forces
  - Sinks act like an inner boundary condition; information is lost near and within the sink radius







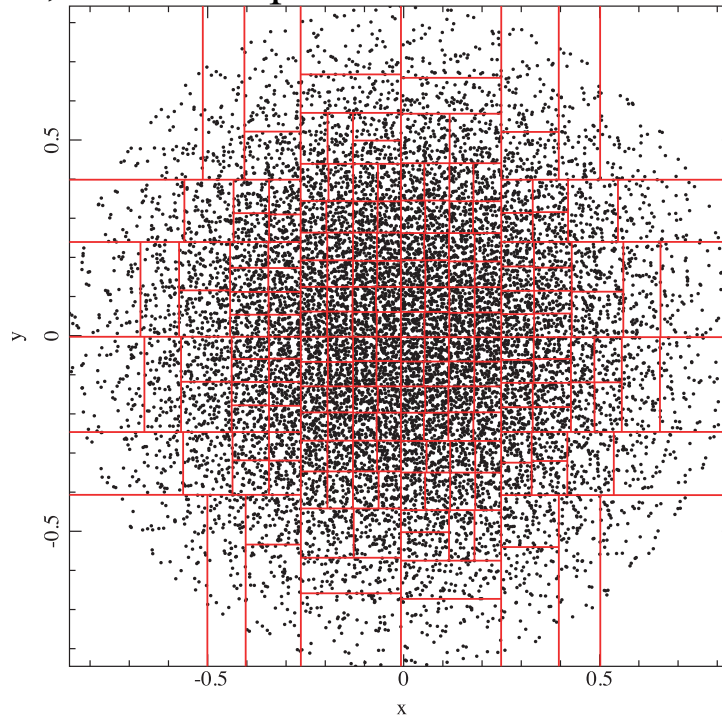
# Gravity

- Momentum equations including gravity:

$$\frac{d\mathbf{v}}{dt} = -\frac{1}{\rho}\nabla P - \nabla\Phi$$

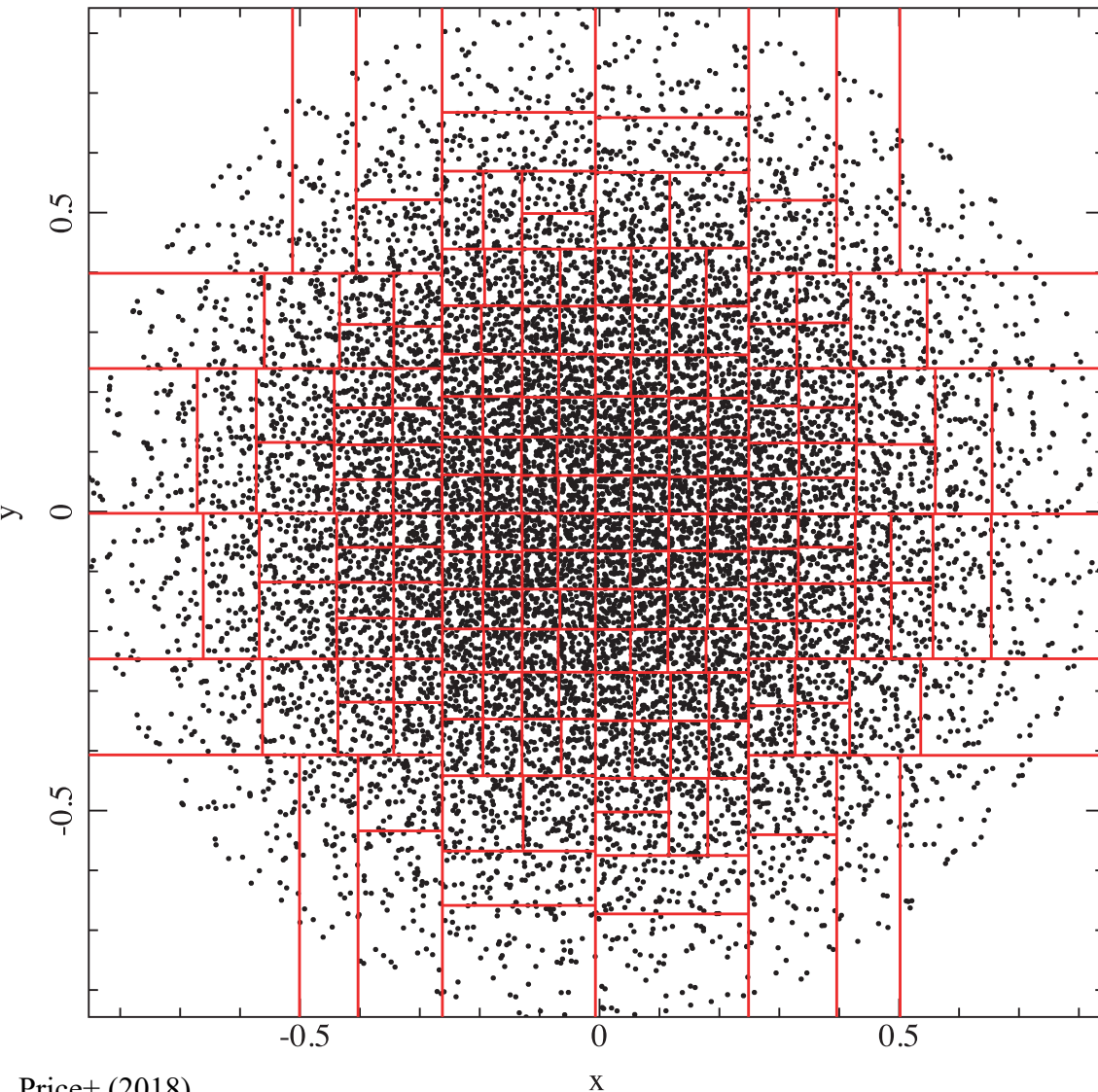
$$\nabla^2\Phi = 4\pi G\rho$$

- Poisson's equation is non-trivial to solve
- Rather, for each particle, we sum the forces from all other particles
- This is done using a tree, where the particles are divided into 'leaf-nodes' of  $\sim 10$  particles:





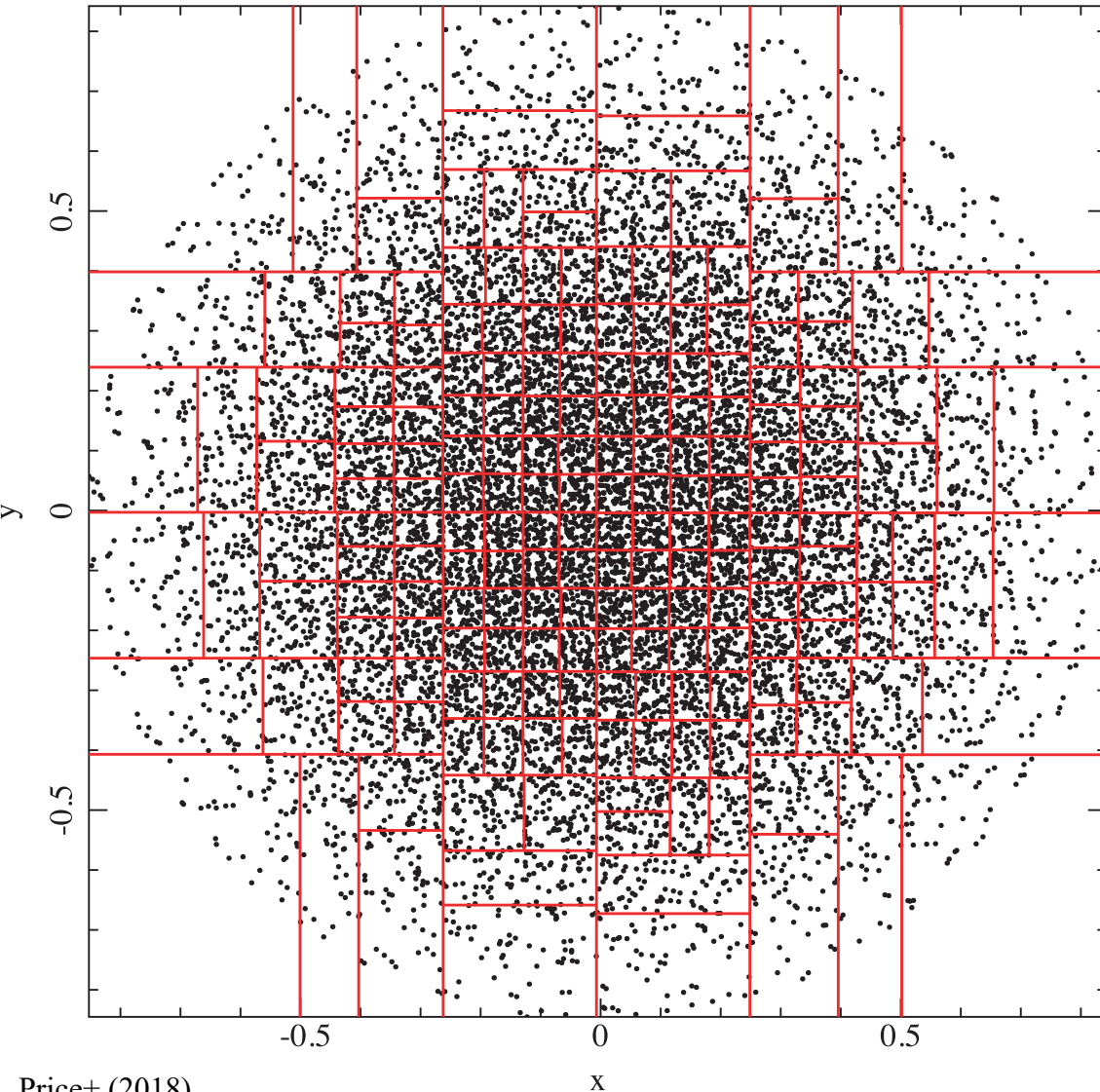
# Gravity: Short-range forces



- For a given particle, we perform an  $N^2$  calculation over all particles within its leaf-node, and all leaf-node within  $2h$
- Not all of these particles are guaranteed to be within  $2h$ 
  - If within  $2h$ , particle contributes a softened gravitational force using a kernel
  - If outside of  $2h$ , particle contributes  $1/r^2$  gravitational force



# Gravity: Long-range forces

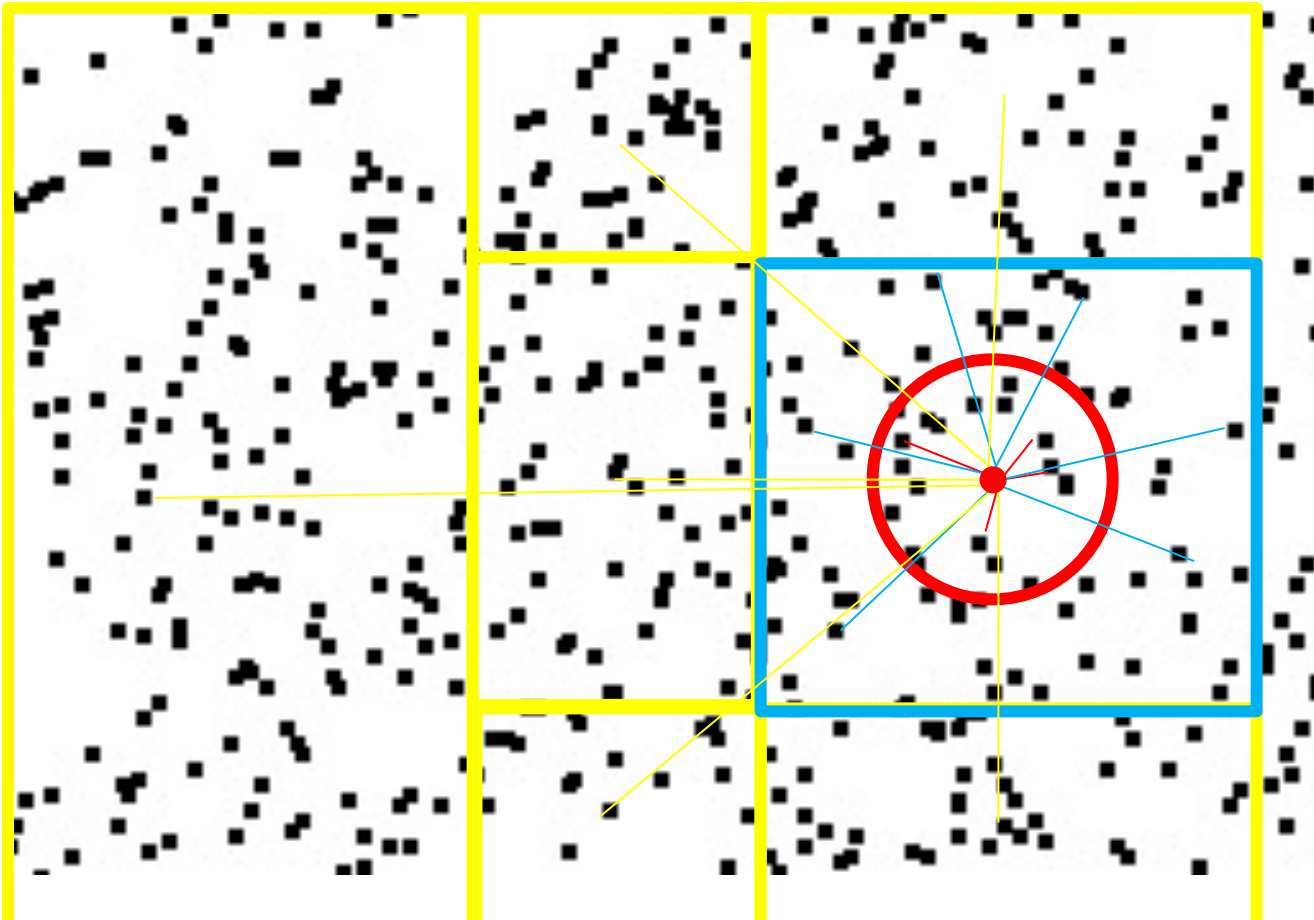


- For particles in remaining leaves, leaves are combined by size and distance from the target particle
- Each combination of leaves should have a similar angular size as viewed from the target particle
- Each combined leaf makes a single contribution to the gravitational force, using total leaf mass and centre of mass location



# Gravity

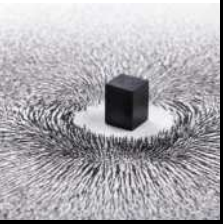
- For efficiency, split gravity into three components:
  - Inside red circle: softened force since neighbour is within  $2h$
  - Inside blue square:  $1/r^2$  force since this represents all leaf-nodes within  $2h$  of the particle
  - Each yellow rectangle is a combined group of leaves; each makes a single contribution





## *Gravity + sinks*

- Sink particles (generally) do not exist in the tree
- Sink-sink & sink-gas gravity are treated as only short range & mid-range
- At every step, this leads to minimum  $N_{\text{sink}}^2 + N_{\text{gas}}N_{\text{sink}}$  gravity calculations
- If sinks have formed a tight binary (or higher order system), then small timesteps are required to resolve the binary orbit. Sink-sink & sink-gas gravity *only* are calculated on these very short timesteps



# *Continuum Magnetohydrodynamic Equations*

➤ Continuum Equations:

➤ Continuity Equation

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{v}$$

➤ Equation of Motion

$$\frac{d\mathbf{v}}{dt} = -\frac{1}{\rho} \nabla \left[ \left( P + \frac{B^2}{2\mu_0} \right) I - \frac{1}{\mu_0} \mathbf{B}\mathbf{B} \right]$$

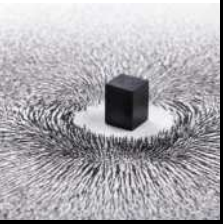
➤ Induction Equation

$$\frac{d\mathbf{B}}{dt} = \nabla \times (\mathbf{v} \times \mathbf{B}) = (\mathbf{B} \cdot \nabla) \mathbf{v} - \mathbf{B} (\nabla \cdot \mathbf{v})$$

➤ Energy Equation

$$\frac{du}{dt} = -\frac{P}{\rho} \nabla \cdot \mathbf{v}$$

➤ Equation of state (e.g.)  $P = (\gamma - 1) \rho u$



# *Continuum Magnetohydrodynamic Equations*

➤ Continuum Equations:

➤ Continuity Equation

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{v}$$

➤ Equation of Motion

$$\frac{d\mathbf{v}}{dt} = -\frac{1}{\rho} \nabla \left[ \left( P + \frac{B^2}{2\mu_0} \right) I - \frac{1}{\mu_0} \mathbf{B}\mathbf{B} \right]$$

➤ Induction Equation

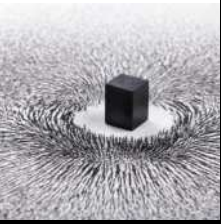
$$\frac{d}{dt} \left( \frac{\mathbf{B}}{\rho} \right) = \left( \frac{\mathbf{B}}{\rho} \cdot \nabla \right) \mathbf{v}$$

➤ Energy Equation

$$\frac{du}{dt} = -\frac{P}{\rho} \nabla \cdot \mathbf{v}$$

➤ Equation of state (e.g.)  $P = (\gamma - 1) \rho u$





# Discrete Magnetohydrodynamic Equations

## ➤ Discrete Equations:

### ➤ Density Equation

$$\rho_a = \sum_b m_b W_{ab}(h_a); \quad h_a = \eta \left( \frac{m_a}{\rho_a} \right)^{1/3}$$

### ➤ Equation of Motion

$$\frac{dv_a^i}{dt} = \sum_b m_b \left[ \frac{S_a^{ij}}{\Omega_a \rho_a^2} \nabla_a^j W_{ab}(h_a) + \frac{S_b^{ij}}{\Omega_b \rho_b^2} \nabla_a^j W_{ab}(h_b) \right]$$

### ➤ Induction Equation

$$\frac{d}{dt} \left( \frac{B_a^i}{\rho_a} \right) = - \frac{1}{\Omega_a \rho_a^2} \sum_b m_b v_{ab}^i B_a^j \nabla_a^j W_{ab}(h_a)$$

### ➤ Energy Equation

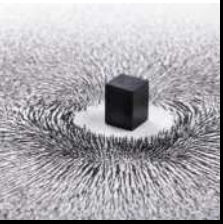
$$\frac{du_a}{dt} = \frac{P_a}{\Omega_a \rho_a^2} \sum_b m_b v_{ab}^j \nabla_a^j W_{ab}(h_a)$$

### ➤ MHD stress tensor

$$S_a^{ij} \equiv - \left( P_a + \frac{1}{2\mu_0} B_a^2 \right) \delta^{ij} + \frac{1}{\mu_0} B_a^i B_a^j$$

➤ Note: In all SPMHD equations,  $\mathbf{B}$  has been normalised such that  $B = B/\sqrt{\mu_0}$





# *Magnetic Fields*

- Magnetic fields are numerically difficult
- Require artificial resistivity (see Wurster, Bate, Price & Tricco 2017)
- Require enforcement of  $\nabla \cdot \mathbf{B} = 0$
- Analytically  $\nabla \cdot \mathbf{B} = 0$ , but this is not numerically guaranteed
- To enforce  $\nabla \cdot \mathbf{B} = 0$ , we include ‘divergence cleaning’ (Tricco, Price & Bate 2016)
- In MHD simulations, track  $\frac{h|\nabla \cdot \mathbf{B}|}{|\mathbf{B}|}$  to ensure it remains small (see also Dobbs & Wurster 2021)
- MHD simulations are often slower and more care is generally needed than purely hydrodynamic simulations



# Dust

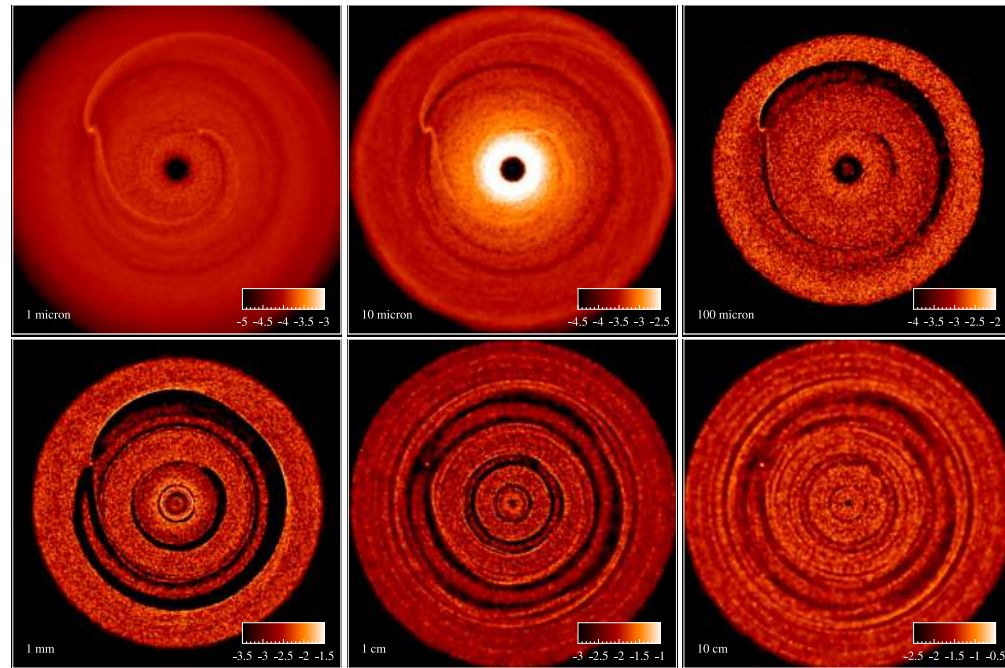
- Dust is a pressure-less fluid that interacts with itself and gas via gravity and drag only
- For large dust grains, two-fluid approach is optimal
  - Dust and gas are each represented using a different set of particles
  - Dust and gas particles can be accreted onto sink particles

$$\frac{\partial \rho_g}{\partial t} + \nabla \cdot (\rho_g \mathbf{v}_g) = 0,$$

$$\frac{\partial \rho_d}{\partial t} + \nabla \cdot (\rho_d \mathbf{v}_d) = 0,$$

$$\rho_g \left( \frac{\partial \mathbf{v}_{tg}}{\partial t} + \mathbf{v}_g \cdot \nabla \mathbf{v}_g \right) = \rho_g \mathbf{f} + K(\mathbf{v}_d - \mathbf{v}_g) - \nabla P_g,$$

$$\rho_d \left( \frac{\partial \mathbf{v}_{td}}{\partial t} + \mathbf{v}_d \cdot \nabla \mathbf{v}_d \right) = \rho_d \mathbf{f} - K(\mathbf{v}_d - \mathbf{v}_g),$$





# Dust

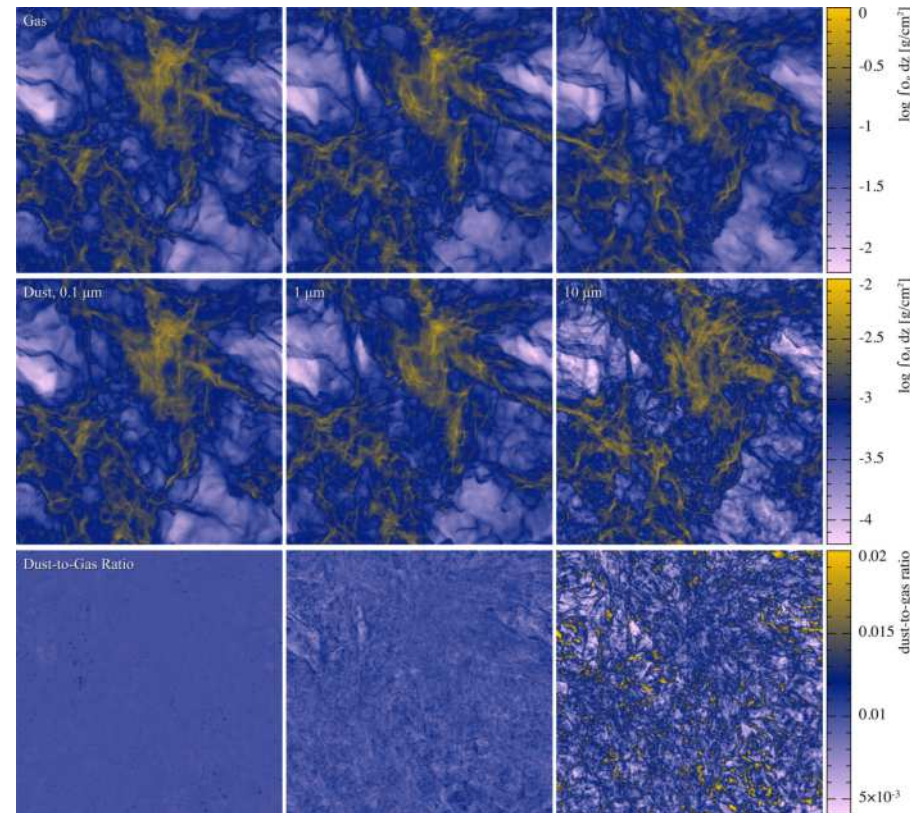
- Dust is a pressure-less fluid that interacts with itself and gas via gravity and drag only
- For small dust grains, one-fluid approach is optimal
  - An evolving fraction of each particle is dust and the remaining fraction is gas
  - These hybrid particles can be accreted onto sink particles

$$\frac{d\rho}{dt} = -\rho(\nabla \cdot \mathbf{v}),$$

$$\frac{d\epsilon}{dt} = -\frac{1}{\rho} \nabla \cdot [\epsilon(1 - \epsilon)\rho\Delta\mathbf{v}],$$

$$\frac{d\mathbf{v}}{dt} = (1 - \epsilon)\mathbf{f}_g + \epsilon\mathbf{f}_d - \frac{1}{\rho} \nabla \cdot [\epsilon(1 - \epsilon)\rho\Delta\mathbf{v}\Delta\mathbf{v}] + \mathbf{f},$$

$$\frac{d\Delta\mathbf{v}}{dt} = -\frac{\Delta\mathbf{v}}{t_s} + (\mathbf{f}_d - \mathbf{f}_g) - (\Delta\mathbf{v} \cdot \nabla)\mathbf{v} + \frac{1}{2} \nabla [(2\epsilon - 1)\Delta\mathbf{v}^2],$$

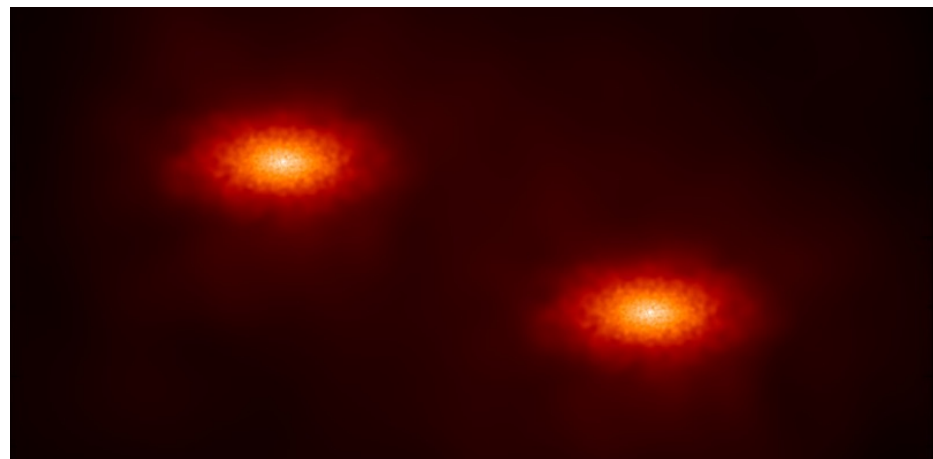
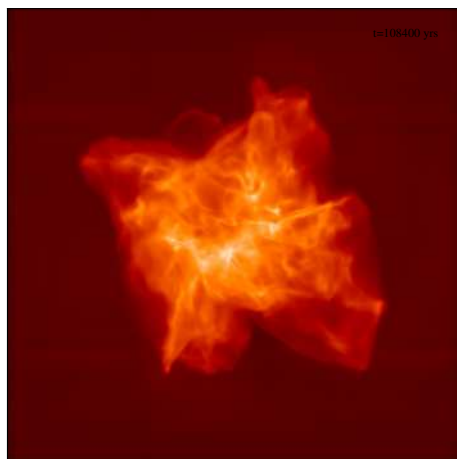
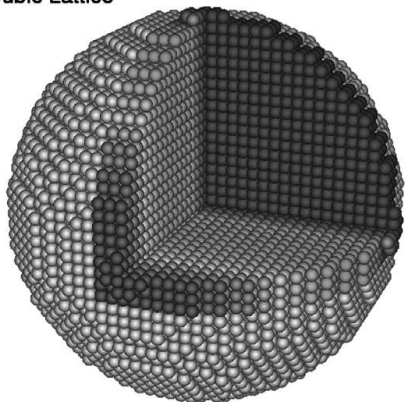




# *Boundary Conditions*

- Boundaries are optional in SPH
  - Particles are free to go where they physics takes them!
  - e.g. collapse of a sphere
  - e.g. evolution of a turbulent hydrodynamic sphere (Bate 2012)
  - e.g. galaxy merger simulation (Wurster & Thacker 2013a,b)

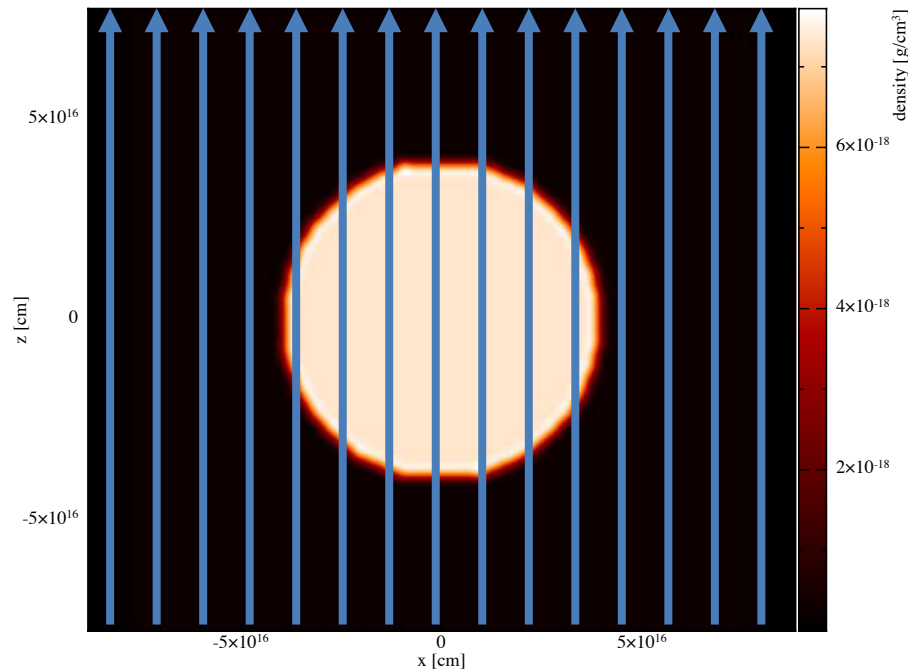
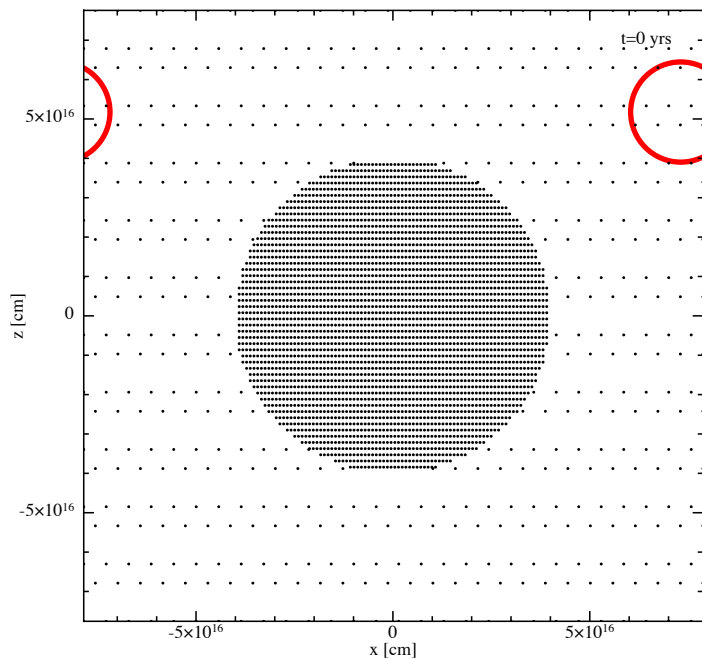
Cubic Lattice





# Boundary Conditions

- Boundaries are *not* optional when evolving magnetic fields!
- Cannot coherently evolve the ‘end’ of a magnetic field line
- Most common magnetic boundary: ‘sphere-in-box’-style with periodic boundaries (e.g. Wurster+ 2016,2017,2018abcd,2019,2020ab,2021)
  - Embed sphere in low density medium (e.g. with density ratio 30:1)
  - Thread magnetic field throughout the entire domain
  - Use periodic boundaries at the edge of the box, where magnetohydrodynamic forces are periodic but gravity is not





# *Summary to date*

- When developing / modifying code:
  - Convert the continuum equations into discrete equations
  - Account for variable smoothing lengths
  - Careful with your choice of discretisation for derivatives
  - Apply artificial terms as required
    - generally only two required for hydrodynamics
    - a third is required for magnetohydrodynamics
  
- Additional physical processes can be modelled via using SPH (e.g.)
  - gravity
  - very dense regions via sink particles
  - magnetic fields (required cleaning, artificial resistivity, & boundaries)
  - dust



# Coding Words of Wisdom

- Always comment your code!
- Never hard-code parameters, even when testing!  
(at the very least, define all parameters at the top of the file for clarity, consistency, and ease of ability to change)

```
return 'N/A'
```

"Always code as if the guy who ends up maintaining your code will be a violent psychopath who knows where you live."

```
suffix = ['B', 'KiB', 'MiB', 'GiB', 'TiB', 'PiB', 'EiB', 'ZiB', 'YiB'][exponent]
converted = float(bytes) / float(1024 ** exponent)
return '%.2f%' % (converted, suffix)
```

~ John Woods



99 little bugs in the code.  
99 little bugs in the code.  
Take one down, patch it around.

127 little bugs in the code...

