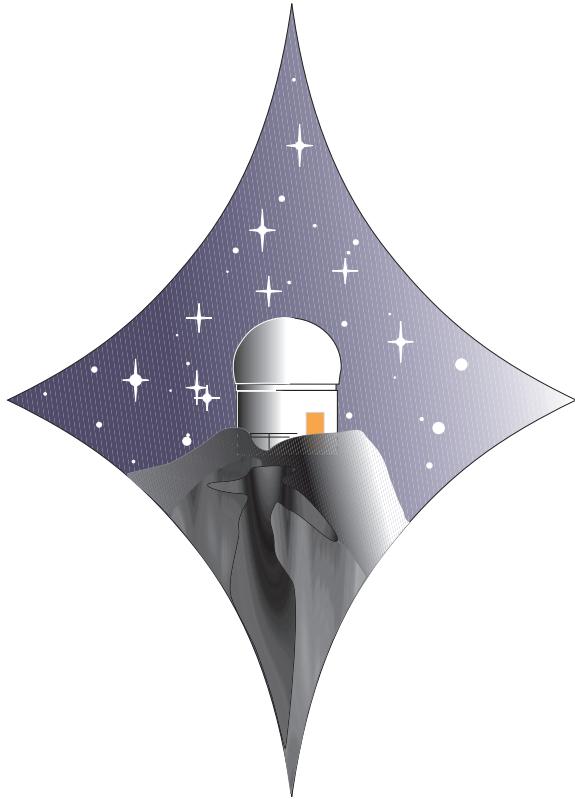


Assignment - Hydrodynamics



Assignment at the University Observatory Munich
Ludwig-Maximilians-University Munich

Submitted by
Eric Muires and Johannes Stoiber

Supervised by
Dr. Ildar Khabibullin, PD Dr. Klaus Dolag, Dr. Tadziu Hoffmann

Munich, 12th of March 2023

Contents

1 Tutorials	1
T0 - Introduction: connecting, compiling, running, reading data	1
T1 - Free-fall Collapse	2
T2 - Generating Field Lines: Voids and Superclusters in Cosmological Simulations	3
T3 - Density Estimates: Particle Distributions and Kernels	4
T4 - The Riemann Problem	5
T5 - A Collapsing Gas Cloud: Gravity Meets Hydrodynamics	6
T6 - Propagating Sound Waves: Sound Speed and Interference	7
T7 - Supernova Explosion: Simulating a Blast Wave	8
T8 - Flow through a Nozzle: Accelerating Flows and Boundaries	9
T9: Heat Transport: Equilibration and Historic Cooling of Amphitheaters	10
T10 - Viscosity: Flow between Plates and Jets	11
T11: The MHD Shocktube Problem	12
T12: Simulating a Disk Galaxy: Reproducing the SK relation	13
2 Extensions	15
E1: ETGs as remnants of Galaxy Mergers and their Kinematics	16
E2: A Collapsing Gas Cloud with Different Initial Velocities	18

The PYTHON Code used for this assignment and the results of each experiment can be found on GitHub: <https://github.com/Johannes-Stoiber/Hydrodynamics-Tutorials>

Chapter 1

Tutorials

T0 - Introduction: connecting, compiling, running, reading data

The goal of this experiment was getting to know how to run numerical experiments. The necessary steps are 1 Connecting, 2 Setup, 3 Experiment. The details of connecting to the USM machines via ssh will be omitted in the following, because we already knew how to do this beforehand. The programming language used for setting up and visualizing the results will be PYTHON, whereas the code that actually performs the experiment is OPENGADGET3¹. This is true for every experiment in this *Hausarbeit*.

Setup

The important files for setting up the experiment are box.param and Config.sh in the OpenGadget3 folder. The first one contains information like time stepping, which snapshot file format is used (format 2), cosmological parameters, box size, parameters for the Tree algorithm (responsible for calculating the gravitational force), SPH parameters (the lagrangian solver of the hydrodynamical equations), and many more. The latter file holds information about the simulation features like which kernel should be used, if viscosity and thermal conductivity are included, or MHD settings, and also many more.

Before the final setup of the initial box can be done (box.ic) the OPENGADGET3 program is compiled using the MAKE command. Before that one has to specify and load which MPI (message passing interface) and compiler should be used. In our case the Intel compiler and the MPI provided by the OpenMPI Project² were called using the commands MODULE LOAD INTEL and MODULE LOAD OPENMPI. The compilation should be performed within the OpenGadget3 folder. During this process the information in the Config.sh file are already included in the program. The resulting executable is called P-GADGET3.

The next script that was run is the setup_grid.py file. It creates the initial conditions file.

Experiment

Now the experiment can actually be performed with the command MPIRUN -NP 1 P-GADGET3 BOX.PARAM. The executable P-Gadget3 takes box.ic, will run over the given time steps specified in box.param and create 11 snapshot files (snap_000 to snap_011). "-np 1" specifies the number of cores used for the computation.

Finally the results can be display using the script SHOW.PY, which reads in the "POS " block of each Snapshot file and creates a little animation presenting these positions at their respective time steps.

¹<https://wwwmpa.mpa-garching.mpg.de/gadget/gadget2-paper.pdf>

²<https://www.open-mpi.org/>

T1 - Free-fall Collapse

The goals of this tutorial were to improve our knowledge of creating initial condition files and plotting the results and to learn about the collapse of a uniform sphere as a representation of a galaxy cluster. This was done in two steps: Simple consideration of the free-fall time and related quantities and the simulation and analysis of the results.

Free-fall Time

The free-fall time is given by $t_{\text{ff}} = (3\pi/32G\rho_0)^{1/2}$. The mean matter density of the universe is about $\rho_{m,0} \sim 1.3 \cdot 10^{-30} \text{ g/cm}^{-3}$, which gives a free-fall time of $t_{\text{ff,gc}} \approx 1.8 \cdot 10^{18} \text{ s} \approx 58 \text{ Gyr}$. This time is longer than the age of the universe which would imply that galaxy clusters are unable to form on timescales observable in our current universe. However, we can see them in the sky, indicating that this consideration misses some effects like the expansion of the universe. In order to perform the simulation of a collapse we need to know the size of the spatial region from which a galaxy cluster collapses. The typical mass of galaxy clusters is $m_{\text{gc}} \sim 10^{14} M_{\odot}$. A rough estimation of the box size is $r \sim (m_{\text{gc}}/\rho_{m,0})^{1/3} \sim (10^{70} \text{ m}^3)^{1/3} \sim 4 \cdot 10^{23} \text{ m} \approx 13.5 \text{ Mpc}$, which is almost the box size of the simulation below (the length units are kpc).

Setup of Initial Conditions and Simulation

After changing the required parameters and configurations given on the lecture website, the most important part of the setup_grid.py program that was changed, is the condition to select only particles within a sphere of radius 6000 kpc.

Also the particle type was changed to 1 which are dark matter particles. Afterwards the same steps as in T0 were performed with a slight modification of the show.py program.

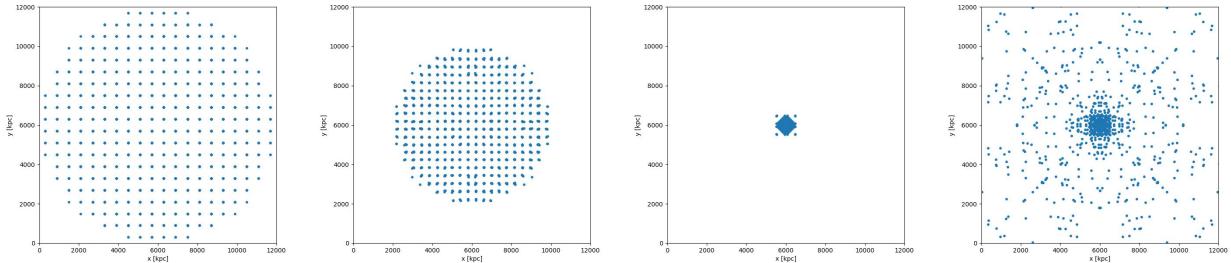


Figure 1.1: First panel: initial grid (snap_001), Second panel: snap_030, Third panel: densest approach of the particles (snap_043), Fourth panel: After the collapse the particles spread out again (snap_050)

The time unit is simply $t_{\text{unit}} = l_{\text{unit}}/v_{\text{unit}} = 3.085678 \cdot 10^{16} \text{ s}$ (length and velocity units as given in the box.param file). As the densest region is seen in snap_043 and the time between snapshot is 0.5, the free-fall time in the simulation is about $t_{\text{ff}} \sim 43 \cdot 0.5 t_{\text{unit}} \approx 21 \text{ Gyr}$. This is again larger than the Hubble time, but lower than the time we calculated before. It is still larger than the age of the universe, because the simulation also doesn't include the expansion of the universe (density was bigger in the beginning of the universe). The free-fall time is independent of the mass, which means in a uniform density field galaxy clusters and galaxies are supposed to form at the same time, but in reality massive objects form later (Li et al. 2008). One reason for this might be the expansion of the universe.

T2 - Generating Field Lines: Voids and Superclusters in Cosmological Simulations

This tutorial was a demonstration of streamlines to indicate particle paths through a dynamic galaxy cluster. It aimed to improve the understanding of streamlines and understand computational methods and their limitations. This was done by loading in a file of particle positions in a 3-D volume and mapping a grid to that volume. Each grid volume would track the average velocity of the particles moving through it to create streamlines. These streamlines can then be used to make judgements on which regions of the cluster are more or less dense, and show filaments leading to dense regions and streamline divergences from void-like structures.

Setup

The initial setup was done by using g3read.py to import the position and velocity data of the particles in a 48000 kpc per side cube. From this, streamlines were created with gridsizes $N_{grid} = 20$ and $N_{grid} = 25$. Each grid volume then computes the average of the particle velocities within the volume using a mesh to create a directional flow for the streamline which is then further plotted.

This is done iteratively through each grid-volume, or mesh point, creating an overall effect by tracing the flow of each "test particle" located at the centre of the grid-volume following the average path of particles within the volume.

Results and Analyses

Figure 1.2 shows the two primary results with differing N_{grid} values. An important distinction that can be seen between them is towards the top left at $N_{grid} = 20$ the streamlines separate at a greater value along the x-axis. This is due entirely to the resolution size of the volume. As the resolution improves this becomes increasingly accurate, until a fine enough resolution is reached that many grid-volumes cease to contain any particles.

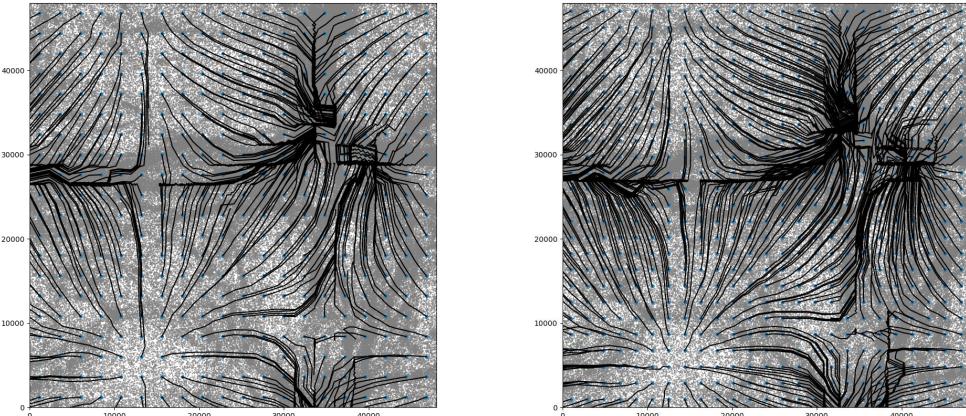


Figure 1.2: First panel: $N_{grid} = 20$, Second panel: $N_{grid} = 25$. Taken from centre of grid cube.

The streamlines show clear grouping and path preferences for matter as it collapses towards denser regions. There are several filaments visible as the lines converge and move towards the densest regions in the volume. The densest region is likely the clustering and converging of streamlines in the top right corner of the grid at this height, with two distinct voids in the bottom right and bottom left from where the streamlines diverge. The coagulation of mass as shown by the streamlines is due to dark matter structures which go on to trap baryonic matter, which then falls into the potential created by the overdense dark matter regions and leaves voids in underdense dark matter regions.

T3 - Density Estimates: Particle Distributions and Kernels

For the density field in the simulation box two of the most important aspects to consider are the initial particles distribution and the kernel used to reconstruct the density field from the particle positions in SPH. Another goal was to learn how to write a BASH script, that runs multiple simulation with varying parameters.

Initial particles distribution

In T00 and T01 we used a cubic lattice distribution, a simple grid. Other possible distributions are a hexagonally close packed (hcp) lattice, a random distribution, a so called gravitational glass distribution and many more (See Fig. 1.3). The gravitational glass distribution is created by starting at random positions, then evolving the positions with reversed gravity into equilibrium (Diehl et al. 2012). This way the distribution is homogeneous, unlike in the random distribution, where the minimal and maximal distance to the nearest neighbors of one particle are very different, and is isotropic, unlike in the hcp distribution, where we can clearly see that some directions are preferred. Therefore the glass distribution is the best choice for cosmological simulations (Diehl et al. 2012).

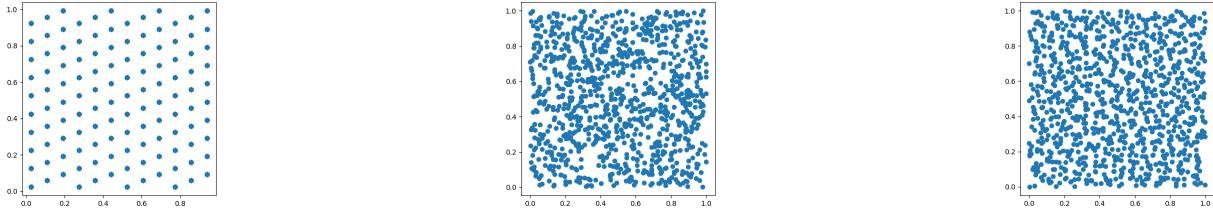


Figure 1.3: First panel: hcp distribution, Second panel: random distribution, Third panel: glass distribution

Kernel and Number of Nearest Neighbors

The reconstruction of the density from particle positions is done using the *kernel method*:

$$\langle \rho_i \rangle = \sum_j m_j W(\mathbf{x}_i - \mathbf{x}_j, h) \quad (1.1)$$

where $W(x)$ is the kernel, which could be a cubic or quintic spline or a so called *Wendland kernel* (Summary can be found in Dehnen and Aly 2012) and h is the *smoothing length* which is often defined as $h^3 = 3N_{\text{ngb}}/4\pi\rho$, where N_{ngb} is the number of nearest neighbors. Varying the kernel function and N_{ngb} gives different density estimates. Fig. 1.9 shows how the logarithm of the ratio of the initial density ($\rho_{\text{ini}} = \sum_i m_i / \text{BoxSize}$) and the simulation density ($\rho_{\text{sim}} = \sum_i \rho_i / N$). The first value for the quintic spline and the first two for the Wendland C6 kernel in Fig. 1.9 are incorrect, because the simulation crashed for these values. The initial distribution was the given file `glass_10x10x10`.

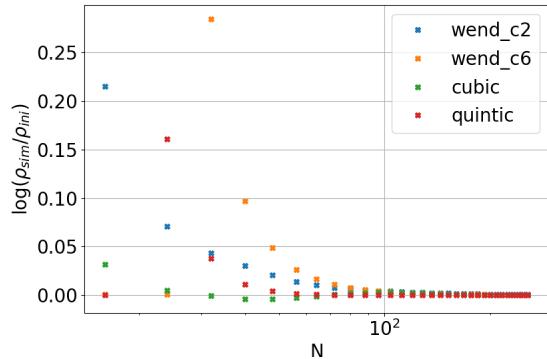


Figure 1.4: Logarithm of the ratio of the simulation density and the initial density for different numbers of nearest neighbors and different kernels (Wendland C2, Wendland C6, cubic spline and quintic spline).

T4 - The Riemann Problem

In this exercise non-uniform initial conditions were produced and examined how the Riemann problem is solved for a special case, a shock tube, is solved by a simulation. The Riemann problem is a simple discontinuity in initial conditions of overdense and an underdense regions which causes a pressure wave propagation to travel through the gas.

Initial Conditions

The initial setup was generated using the glass box from T3. In the underdense regions the boxes were placed adjacent and not overlapping with one another, in the overdense regions the glass box was duplicated in half-width steps such that they overlap half of their width with the adjacent box causing a region that is eight times as dense as its neighboring region. This is demonstrated below in Fig.1.5 showing a long slab which has the dimensions of 1x1x20 glass boxes.

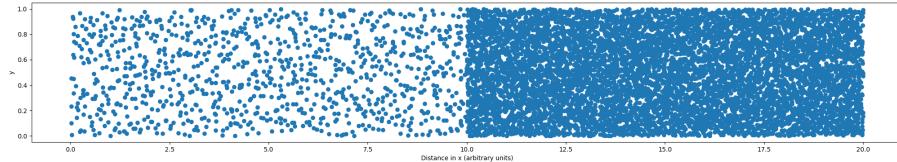


Figure 1.5: Position plot of particles in the x and y directions, with every tenth particle shown to avoid overcrowding.

This simulation aimed to solve the following equation,

$$\frac{\rho_1}{\rho_5} \frac{1}{\lambda} \frac{(1-P)^2}{\gamma(1+P) - 1 + P} = \frac{2\gamma}{(\gamma-1)^2} \left[1 - \left(\frac{P}{\lambda} \right)^{\frac{\gamma-1}{2\gamma}} \right]^2 \quad (1.2)$$

where subscript 1 denotes the overdense region and subscript 5 denotes the underdense regions, $P = \frac{P_c}{P_5}$ and $\lambda = \frac{P_1}{P_5}$. Where P is the parameter to be solved for.

Shock tube

There were two solutions to the simulation. A semi-analytical solution obtained by plotting the LHS and RHS of the above equation to solve for P , and the simulation results as an evolution of time. The results are show below in Fig.1.6.

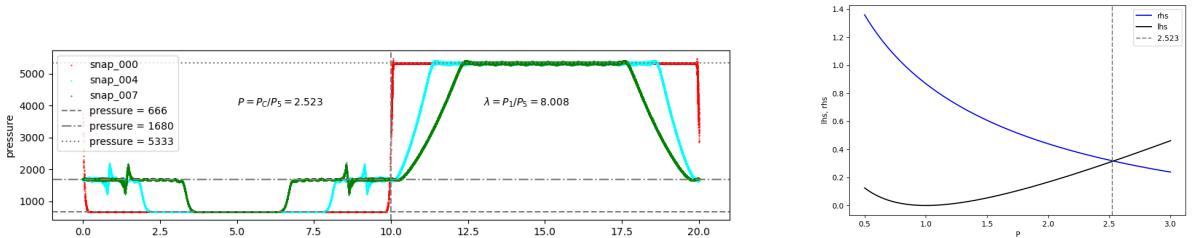


Figure 1.6: First panel: time evolution of parameters of the simulation, Second panel: Semi-analytical solution with $P = 2.523$

The results above show that the overdense region begins spreading and the outgoing diffusion of gas causes strong ripples in internal energy as seen in the plot of ϵ , indicating a highly accelerated and high-velocity wave propagating through the system. The plot on the right shows how a quick and easy plotted solution can provide the value of P_c/P_5 is shown to be 2.523. By taking the ratios of pressures in the simulation from Fig.1.6 the value of P is found in exact agreement at a value of $P = 2.523$.

T5 - A Collapsing Gas Cloud: Gravity Meets Hydrodynamics

More complicated non-uniform initial conditions were produced and the so called Evrard-collapse (isothermal, $\rho(r) \propto r^{-1}$) investigated.

Initial Conditions

In order to produce a spherical distribution with a initial density profile $\rho(r) \propto r^{-1}$, first a block of 100000 particles in a glass distribution was produced by stacking the glass distribution, that was provided in T3, 10 times in each dimension (enlarge_box.py).

Now the $g(r) \propto r^{-1}$ profile needs to be created. The current density profile is $f(r) \propto 1$. Therefore we need to use a transformation (starting with the number of particles within a spherical shell of width dr):

$$4\pi r^2 f(r) dr = 4\pi r'^2 \rho(r') dr' \iff r^2 \cdot 1 dr = r'^2 \cdot r'^{-1} dr' = r' dr' \iff \\ 1/3r^3 = 1/2r'^2 \iff r' = (3/2)^{1/3} r^{3/2} \propto r \cdot \sqrt{r}$$

We can simply multiply the coordinates of each particle with the square root of its radius. Afterwards a sphere can be cut out similar to T1 (setup_evrard.py).

Evrard collapse

Now the simulation can be performed. Snapshots of the collapse are shown in Fig. 1.7.

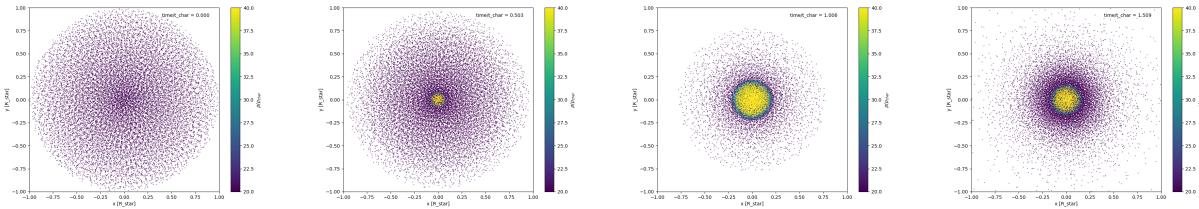


Figure 1.7: Snapshots of Evrards collapse; the colorbar shows the density

After an initial collapse the gas cloud settles into an equilibrium. Further insight into the collapse gives the energy evolution (Fig. 1.8, left). The energies are conveniently written to a file called energy.txt by P-Gadget3 for specified time steps.

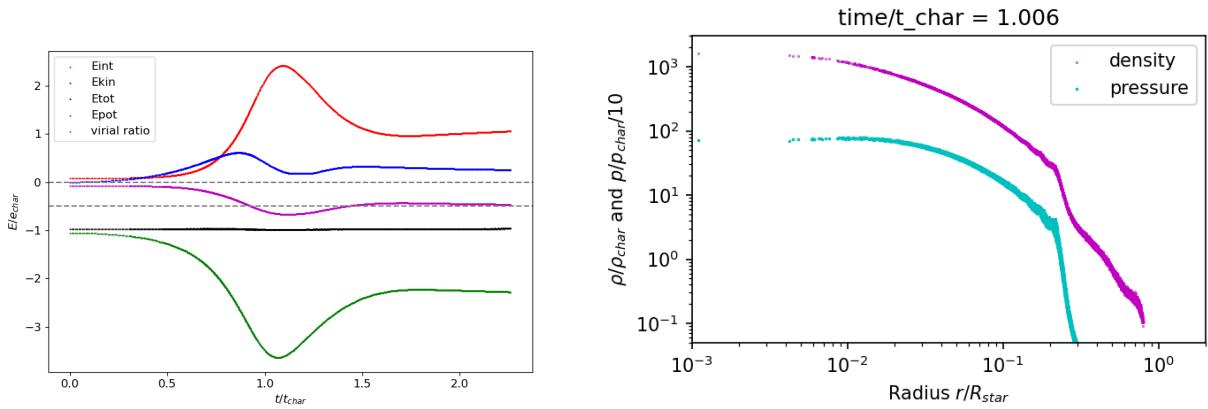


Figure 1.8: Left: Energy evolution of Evrard collapse, right: Density and Pressure profile at $t/t_{char} = 1.06$

After a bounce, that increases internal energy, shortly after $t = t_{char}$, the system virializes as the ratio of internal and gravitational energy approaches -0.5. Finally, after the bounce, a outwards travelling shock wave can be see in Fig. 1.8 (right) at around $r/R_{star} = 0.2$. This analysis follows and was done more precise by Steinmetz and Mueller 1993.

T6 - Propagating Sound Waves: Sound Speed and Interference

This tutorial focused on producing longitudinal sound waves travelling through a gas. The Pressure, Density, and Velocity along the direction of motion, and transverse Velocity, were followed. This was initially set up using the same glass box from Tutorial 3 and extending it in the x direction. A $20 \times 1 \times 1$ rectangular cube was created of instances of these glass boxes, along which the sound wave would travel. For particles in the first two grid boxes, a sinusoidal positive velocity was given to direct them in the $+x$ direction. Similarly for the final two grid boxes they were given an equal velocity distribution travelling in the $-x$ direction. Assuming an adiabatic system the internal energy U was taken as $U = \frac{P}{(\gamma-1)\rho}$. The speed of sound was calculated by following the peak of the Pressure perturbation as it travelled along x until the waves collided. Assuming the speed of sound in the medium is constant, this should give a reasonable answer.

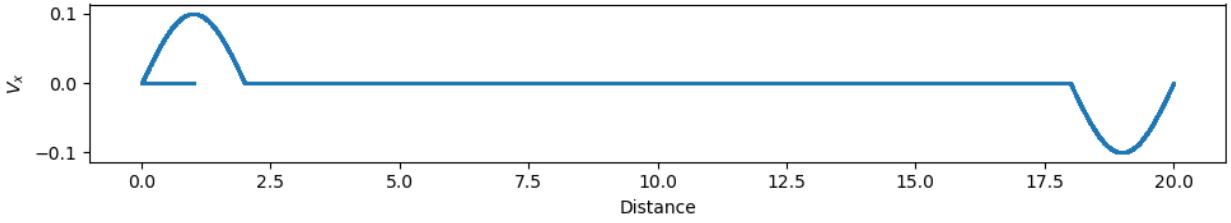


Figure 1.9: Initial velocity distribution of particles along the x direction. Equal and opposite velocity pulses have been generated. Image shown at $t = 0$.

The evolution of this system can be found in a gif on our GitHub repository titled "wave_all.gif". There is shows the full time evolution of the system in Pressure, Density, Velocity in x , and Velocity in a perpendicular direction y .

To determine more parameters of the system we can examine the speed of the wave as it propagates throughout the box. To do this, the peak of the Pressure perturbation is tracked during its progression towards the collision in the centre of the box. This then gives a numerical estimate for the speed of sound in the medium.

This yields a result of a speed of sound $c_s = 0.339$ box lengths per time step. Translating this back into SI creates $c_s = 1.1 \times 10^3$ km/s. Analytically this may be found by:

$$c_s = \frac{\omega}{k} = \sqrt{\frac{dP}{d\rho}} \quad (1.3)$$

where ω is the frequency and k is the wave vector and the differential of Pressure with respect to density in the square root. Then assuming the ideal gas law of $P = 2nk_B T$ the speed of sound can be re-written as:

$$c_s = \sqrt{\frac{2k_B T}{m_p}} \quad (1.4)$$

where k_B is the Boltzmann constant and m_p is the proton mass. The value of c_s we found is comparable to the speed of sound in ICM gas at $c_{s,gas} \approx 10^3$ km/s, which is encouraging.

When changing the wavelength for one of the wave impulses it becomes clear that although the wave itself is different, the overall effects and speed remain the same. In the example found on GitHub under "wave_all_diff_I.gif" the initial conditions are identical with the caveat that the wavelength of the opposing has doubled to encompass four grid boxes from Tutorial 3. The speed at which both waves propagate at are identical, and remains at the speed of sound c_s of the gas.

T7 - Supernova Explosion: Simulating a Blast Wave

The goal of this tutorial was to simulate a blast wave that is triggered by a huge and sudden influx of energy into a small volume, similar to what happens during a supernova. At first the theoretical solution is presented. The initial conditions for such a situation are rather simple (`setup_blast.py`).

Theoretical Solution

As known from the lecture the blast wave can be described by a similarity solution. Only two parameters are known, the added energy and the initial density outside of the shock front: E, ρ_0 . For a known time the only possible combination to get a length scale is

$$\lambda = \left(\frac{Et^2}{\rho_0} \right)^{1/5} \propto t^{2/5} \quad (1.5)$$

Therefore the radius of the shell evolves as $R = \xi_R t^{2/5}$. As the velocity is the time derivative of R it evolves as $u_s = \xi_u t^{-3/5}$. A similar condition can also be derived for the pressure inside the shell $P_1 = \xi_P t^{-6/5}$.

Blast Wave Simulation

The initial conditions for the blast wave where $\rho_0 = 0.125$, $P_0 = 10^{-20}$ and the initiation of 20 particles with $P_{in} = 1$, which is equal to a large deposit of internal energy: $E = P/(\gamma - 1)/\rho$. The resulting shock wave is shown in Fig. 1.10.

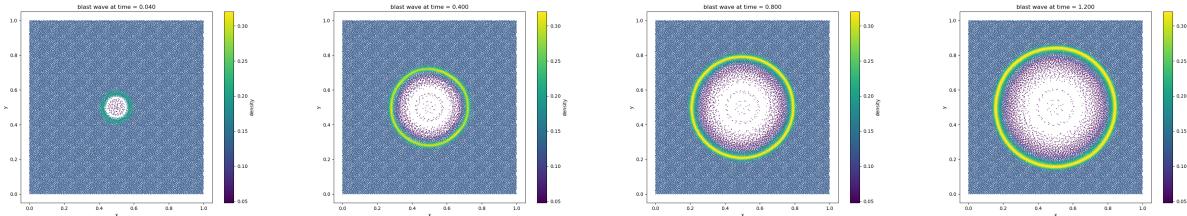


Figure 1.10: Snapshots of blast wave; the colorbar shows the density

The simulation can be verified by plotting the radial profiles of pressure, density and velocity and comparing the features in them to the above given theoretical solutions (Fig. 1.11). The gray dashed lines represent the theory, which line up with the peak in each profile. The peak represents the shock front. The horizontal gray line in the density plot marks the initial density. The coefficients $\xi_{R,u,P}$ can be determined by finding the corresponding quantity at $t = 1$. They are $\xi_R = 0.3160358$, $\xi_u = 0.10180965$ and $\xi_P = 0.00114911$.

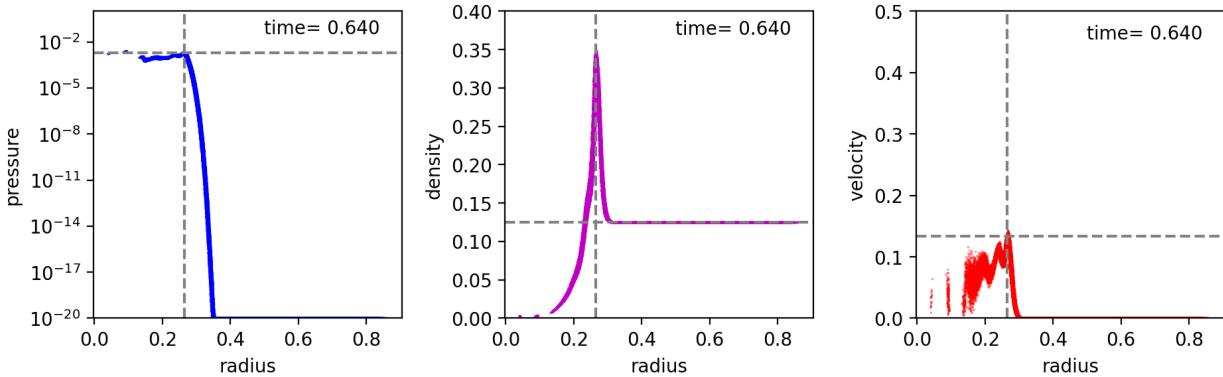


Figure 1.11: Radial profiles of pressure, density and velocity; the gray lines represent the scaling relations

T8 - Flow through a Nozzle: Accelerating Flows and Boundaries

This simulation focused on creating a nozzle and moving a fixed barrier beyond the nozzle to create a pressure force to pull the reservoir of gas beyond the nozzle and examine its properties as the fixed barrier advances further from the nozzle.

How to set up Boundaries and Initial Conditions

The initial system was set up with boundary conditions turned on, and using gas particles with $\text{id} = 0$, rigid and barriers were created. The leftmost barrier is given an initial velocity which will remain unchanged for the duration of the simulation. The nozzle was shaped using a sinusoid. The barriers in x were several layers of particles thick to prevent any accidental leakages from simulation approximations. A figure of the setup is seen in Fig. 1.12.

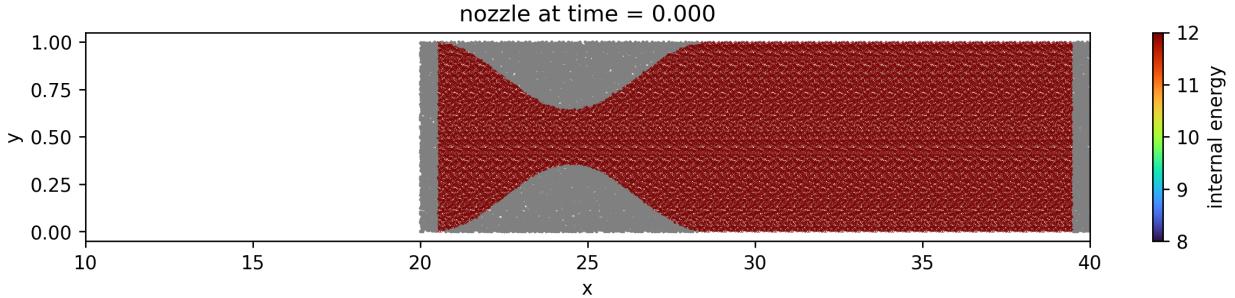


Figure 1.12: Initial Conditions for the flow through a nozzle; stationary boundary particles are shown in gray.

Flow though a Nozzle

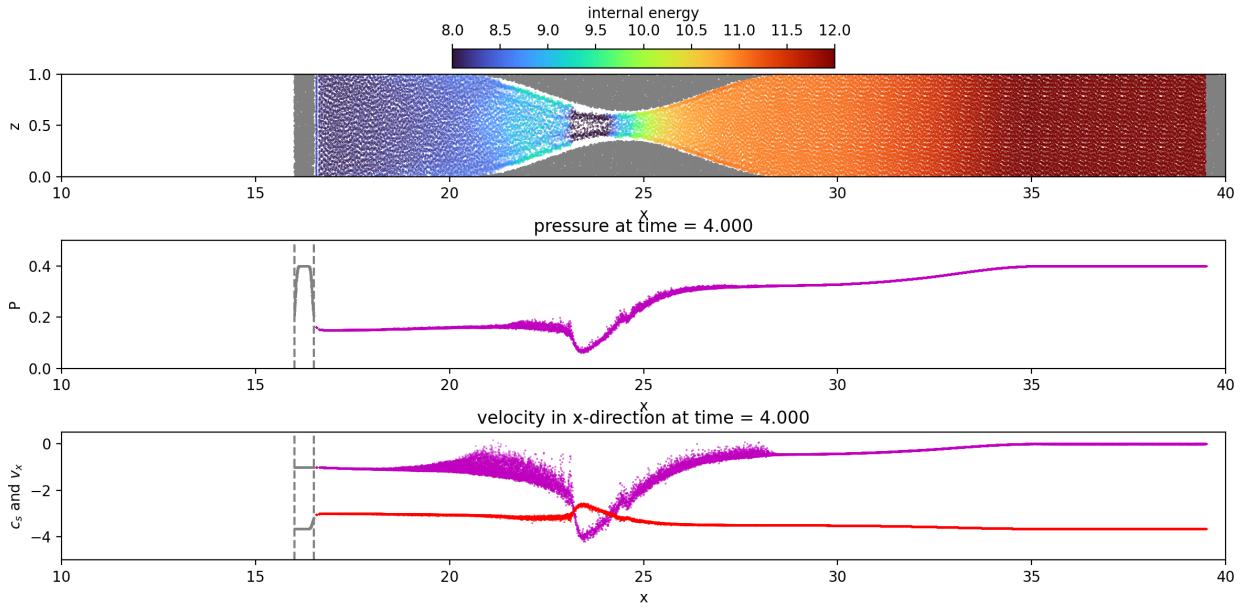


Figure 1.13: Flow through a nozzle at time $t = 4.0$; top panel: coordinates of gas and boundary particles, boundary particles are gray while gas particles are colored according to their internal energy; middle panel: Pressure, bottom panel: gas velocity (magenta) and sound speed (red)

As seen in Fig. 1.13 the gas begins to flow through the system and once past the nozzle the internal energy, and therefore temperature, decreases as it expands into its new volume. This is an expected outcome of expanding gas, even past its acceleration through the nozzle. The pressure wave travelling through the right reservoir slowly drags gas out, decreasing pressure and cooling the reservoir.

T9: Heat Transport: Equilibration and Historic Cooling of Amphitheaters

This experiment introduces thermal conduction into the simulation. The equilibration of two slabs of different temperature in contact and a nozzle that cools the area behind it were examined. The Config.sh and parameter file need to be updated, e.g. CONDUCTION in the former.

Equilibration

The first part was set up in a way, that the slab resembles a solid body (all particles are boundary particles (Sec. T8) and a five times higher temperature on the right side. The simulation was run with periodic boundary conditions and resulted in the evolution shown in Fig. 1.14.

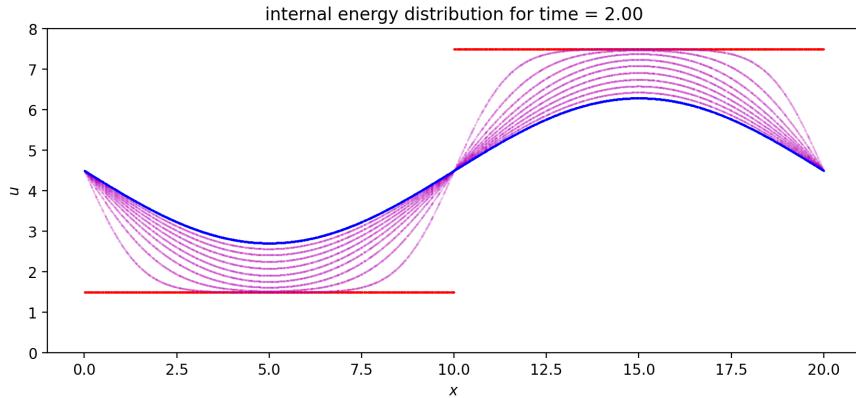


Figure 1.14: Internal energy distribution: initial conditions (red), last snapshot (blue), in between (magenta)

Heat flows from the high temperature region to the low temperature region through the connections between them (middle and periodic boundary). At the center of each region, temperature stays equal to the initial value. The total entropy calculated by $A_{\text{tot}} = \sum_i (\gamma - 1) u_i / \rho_i^{\gamma-1}$ (Jubelgas et al. 2004) is constant throughout the evolution, because the density is constant (solid body), which results in the entropy being only proportional to internal energy, which is conserved.

Ancient Cooling Method

Air can be cooled by compressing it and releasing it into a bigger volume. During the expansion the temperature goes down (Same principle as in a refrigerator; Joule–Thomson effect). The setup is done similar to Sec. T8, with a few adaptations (See Fig. 1.15).

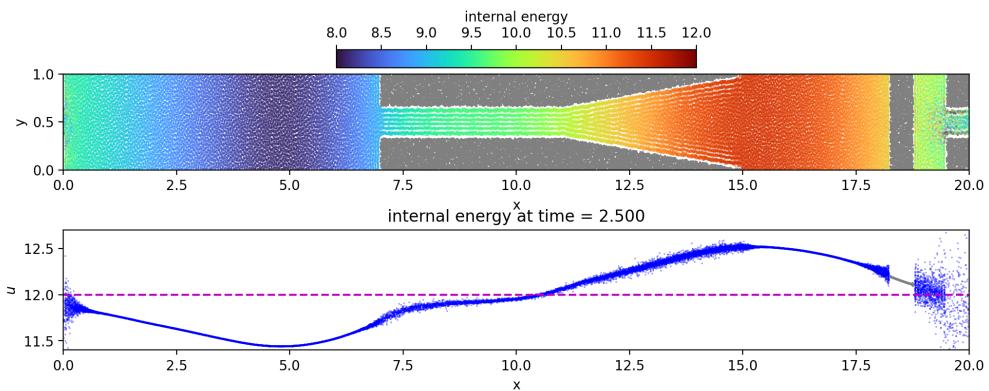


Figure 1.15: Top panel: flow through a nozzle at $t = 2.5$; bottom panel: gas temperature decreases

The gas heats up in front of the nozzle as it is compressed there. Behind the nozzle the temperature decreases, as seen in the bottom panel of Fig. 1.15. Without thermal conduction this will result in a less effective cooling and an internal energy wave propagating in front of the nozzle.

T10 - Viscosity: Flow between Plates and Jets

This tutorial explores viscosity of flows in two ways. The first is by examining the kinetic properties of gas as two plates on either side of the gas move with opposite velocities. The second explores the kinematics of "shooting" a gas out of a tube into a boundary layer of two viscous fluids.

Initial Conditions

Fig. 1.16 shows the initial conditions, on the right the boundary particles at the top of the tube will move downwards with constant velocity forcing the gas out into the other fluid. From the shear of two plates on a fluid we may expect that the viscous fluid will accelerate near the plates in opposite directions to match the plates, and have a velocity gradient that is zero at the centre of the tube. To ensure proper and obvious viscous movement, the different gas components have been given different masses. The gas at the bottom has one particle mass, m_{part} which is calculated simply by the density by volume over number of particles. The others have been then further modified based on this mass. The gas in the top section of the setup on the right has been given a mass of $m_{part}/5$, the gas inside the tube has a mass of $5m_{part}$. For the output, the bottom gas, which demonstrates the interesting behaviour, has been identified with a thicker opacity and will appear darker. This allows us to track them and maintain the colour bar of velocities.

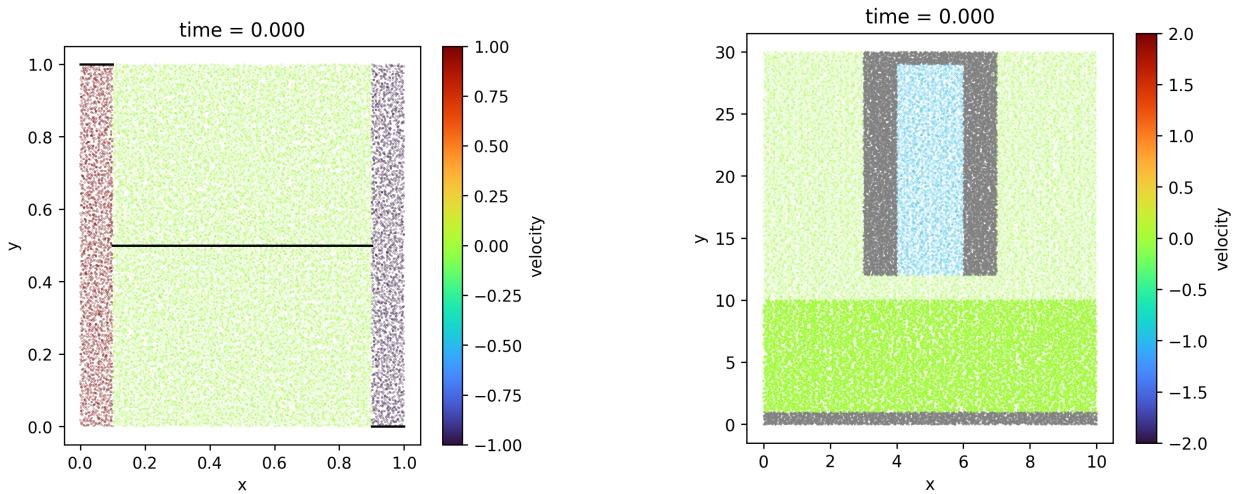


Figure 1.16: Two initial setups, on the left are two plates with opposite velocities with stationary gas, on the right is a tube of gas with the back at a constant downwards velocity and the remaining gas is stationary. Velocity has been added as a line, normalised to the maximum velocity, on the parallel plate experiment.

Results

The simulation results can be found in .gif format on the GitHub under 10-viscosity, named *plates0-20.gif* and *viscous0-250.gif*. As expected, the shearing plates have a velocity gradient that equilibrates to zero at the centre. The tube ejecting a jet of gas creates a mushroom-shaped dome as it enters the lighter bottom medium. This is reminiscent of radio jets, as they eject into the much less dense ICM they create similar mushroom-shaped patterns as they collide and lose energy and begin to dissipate. Eventually, the plate comes out of the tube, fully disrupting the medium below and pressing the accelerated gas up. Interestingly, as the bottom plate leaves the tube and opens the vacuum it created, gas is quickly sucked in through a pressure imbalance and is rapidly accelerated.

This simulation was a demonstration of viscosity and its role in fluid evolution. Viscosity is a type of "fluid friction", it is what causes the acceleration of the gas in the parallel plate setup, as the gas closest to the plates is accelerated, it causes nearby gas to accelerate and so on until the opposite directions meet. Similarly, the viscosity of the system in the accelerated tube of gas is what creates accelerated and heated gas around it as the energy given to it by pushing it out the tube is dissipated to the rest of the container, increasing its kinetic energy (velocity).

T11: The MHD Shocktube Problem

This experiment introduces magnetic fields into the simulation. Its a version of T4, where the magnetic field in y-direction of the high density region is positive while it is negative in the low density region.

Initial Conditions

This exercise tries to recreate the situation as discussed in (Ryu and Jones 1995, setup 5a). The two different slabs have the initial conditions: $(\rho, v_x, v_y, v_z, B_x, B_y, B_z, E) = (1, 0, 0, 0, 0.75, 1, 0, 1)$ and $(0.125, 0, 0, 0, 0.75, -1, 0, 0.1)$. This was achieved similar to T4, using more of the glass distribution boxes to create a slab of dimensions $1 \times 1 \times 100$ and including the respective magnetic fields, using the "BFLD"-block of a snapshot file (Fig. 1.17).

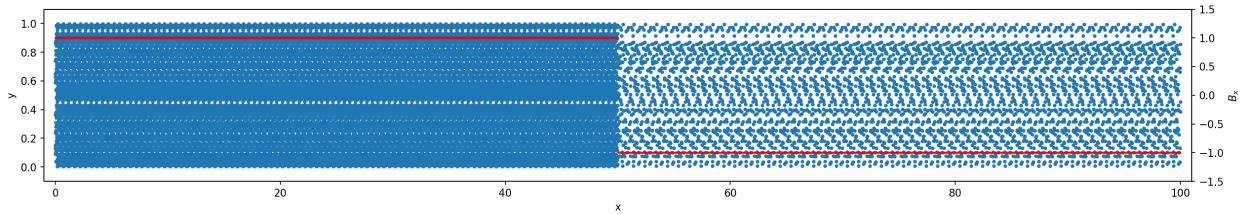


Figure 1.17: Scatter plot of every tenth particles; Magnetic field in x direction (red) w.r.t. the right y-axis

MHD-Shocktube

The numerical solution for the MHD-Shocktube is shown in Fig. 1.18. From left to right: fast rarefaction, slow compound, contact discontinuity, slow shock and fast rarefaction (as in Ryu and Jones 1995, Fig. 5a). The internal energy seems to be setup faulty, as it is too low on the left side and displays a higher spike to the right of the contact discontinuity (dashed gray line). Another artefact is visible at that position: $\text{div}(B)$ departs from zero, which indicates a magnetic charge. This is the case because the initial setup represents a magnetic dipole.

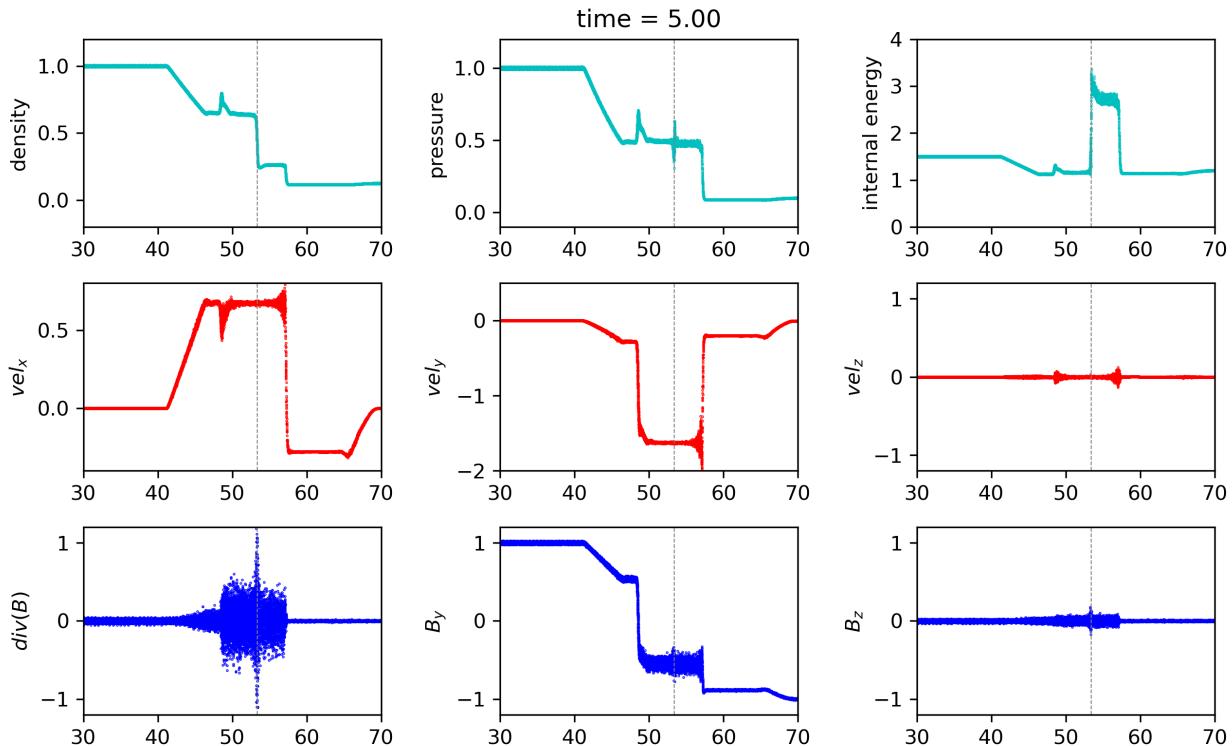


Figure 1.18: Numerical solution to the MHD shocktube; The dashed gray line marks the contact discontinuity

T12: Simulating a Disk Galaxy: Reproducing the SK relation

This tutorial investigates the Schmidt-Kennicutt relation, investigating the star formation rate as a function of density in gas clouds within a galaxy at large. This is done by utilising Open Gadget 3 to simulate a full galaxy with both stellar and gas populations.

Schmidt-Kennicutt Relation

The star formation rate can be defined as a continuous variable in the density of stars as:

$$\frac{d\rho_*}{dt} = \frac{c_*\rho}{t_*} \propto \rho^{3/2} \quad (1.6)$$

where t_* is dynamical time in most cases, which yields $t_* = t_{\text{dyn}} \propto \rho^{-1/2}$. This then reduces to the Schmidt-Kennicutt Relation which states that the increase in the density of the stellar population should with $\rho^{3/2}$ (Katz et al. 1996).

Galaxy Simulation

The first result is the simulation over time from a preformed disk, found in *12-Galaxy* named *galaxy.gif* on the GitHub. This shows an evolution resulting in well formed spiral arms over a relatively short time. As the gas collides and merges, it produces new star forming regions, this can then be explored to demonstrate the Schmidt-Kennicutt relation. This is demonstrated for our results in Fig. 1.19. The relation demonstrates a fit that approximately matches the expected $\propto \rho^{3/2}$ scaling, except the fitted value gives $\propto \rho^n$ where $n = 1.3$. This could be mitigated with several simulations of slightly different initial conditions to give a more accurate mean, however this is extremely computationally expensive and the result is already a good demonstration of the proposed star formation rate.

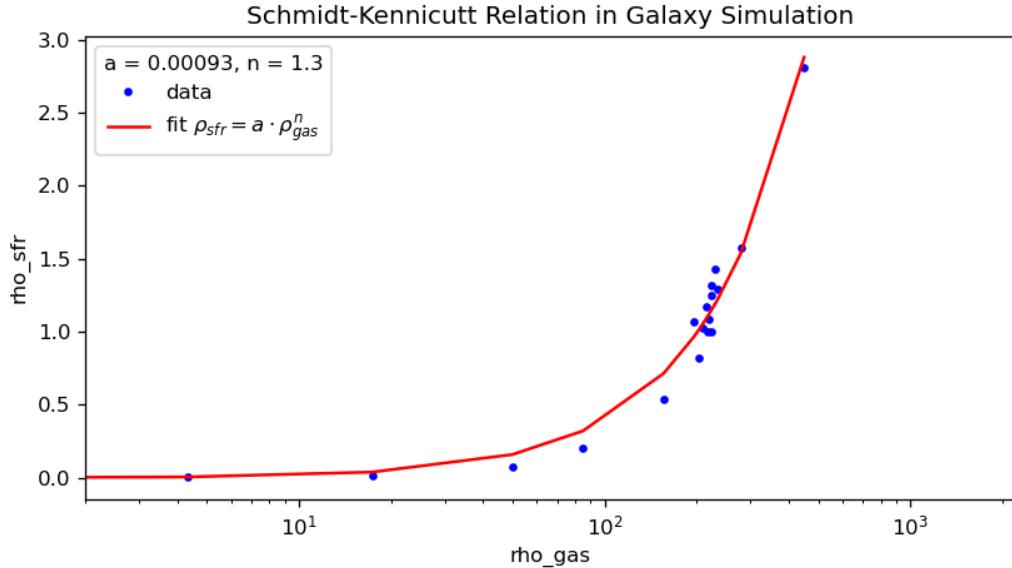
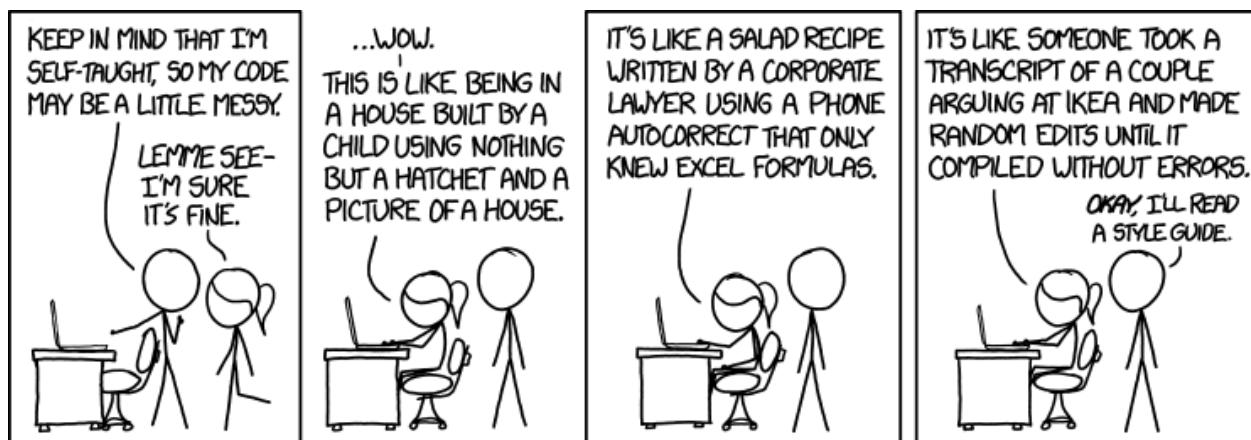


Figure 1.19: Schmidt-Kennicutt relation for simulated galaxy with fitted parameter, $n = 1.3$.

Since this is a disk galaxy, the mass component is dominated by dark matter. This can be found either as a flagged parameter in the simulation or directly from the total mass components of baryonic matter and dark matter. For this disk galaxy, Ω_B was 0.2. This is a product of the initial conditions, and is fairly typical of the universe at large. This is not expected to change during the life of a galaxy, as we can assume Dark Matter is non-interacting with the baryonic matter which undergoes star formation as shown above.

Chapter 2

Extensions



<https://xkcd.com/1513/>

E1: ETGs as remnants of Galaxy Mergers and their Kinematics

Author: Johannes Stoiber

This extension uses the galaxy initial conditions ($M_{\text{stars,bulge}}/M_{\text{stars, total}} = 0.35 \doteq \text{Sa-Type}$ (Laurikainen et al. 2010)) from Sec. T12 but disabling star formation as only the kinematics are investigated here. Different types of spiral-spiral mergers are presented and their remnants analyzed.

Numerical simulations have shown, that early-type galaxies (ETGs) can be remnants of major mergers of spiral galaxies. Depending on the mass ratio and orbital parameters, like the orientation of the initial spins to each other or with respect to the orbital angular momentum, fast rotators or slow rotators are formed (Bois et al. 2011). The physical effects leading to these different statistical distributions of orbits are dynamical friction and violent relaxation combined with the initial conditions.

E1.1 Merger Simulations

Four different mergers were simulated. All of them have a mass ratio of 1:1 with an initial separation of 100 kpc and an initial velocity relative to each other of 40 km/s. The different parameters, like the orientation of the spin of the second progenitor w.r.t. the first φ and the impact parameter b , are listed in Tab 2.1.

name	same	perp	counter	impact
φ [°]	0	90	175	175
b [kpc]	0	0	0	25
λ_R	0.278	0.381	0.223	0.235
$0.31 \cdot \sqrt{\epsilon}$	0.256	0.112	0.249	0.237

Table 2.1: The different initial conditions for the mergers and the resulting λ_R parameter and boundary condition (See Sec. E1.2)

The snapshots right after the first passage of each merger are shown in Fig. 2.1 and their remnants in Fig. 2.2.

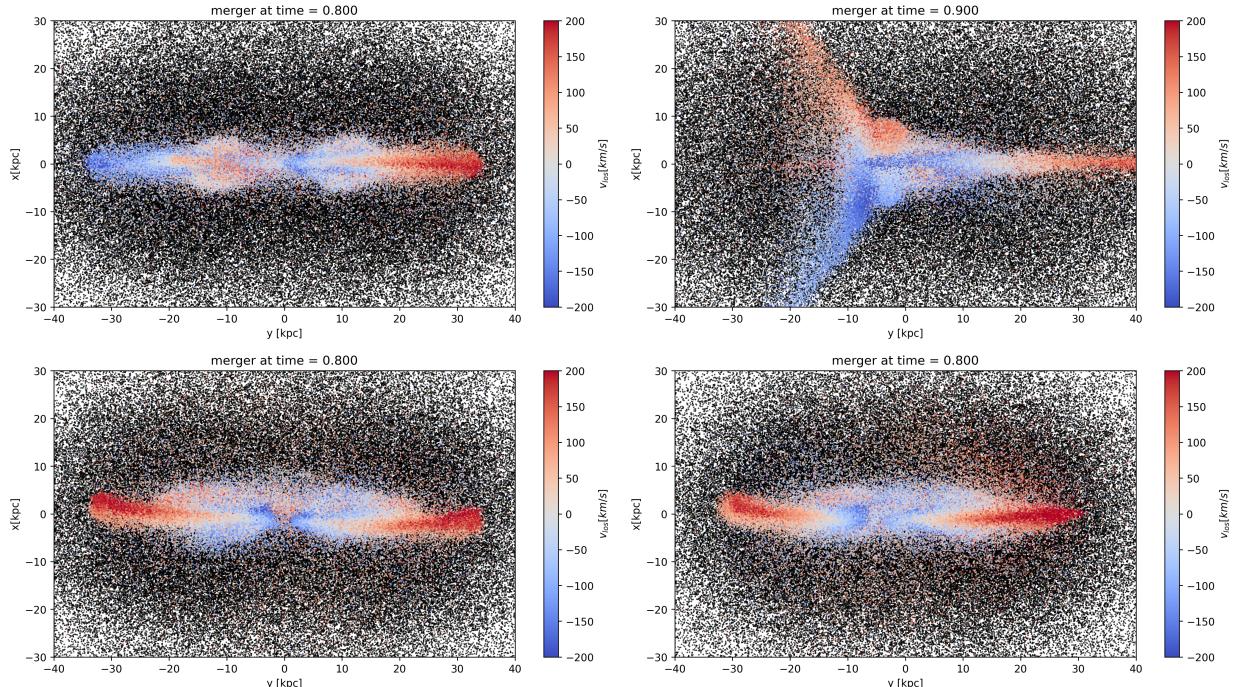


Figure 2.1: Snapshots during the mergers: same at $t = 0.8$, perp at $t = 0.9$, counter at $t = 0.8$, impact at $t = 0.8$; the colorbar shows the line-of-sight velocity or baryons; dark matter particles are shown in black

E1.2 Slow and Fast Rotators

An attempt was made to classify the resulting remnants as slow or fast rotators. The λ_R parameter, introduced by Emsellem et al. 2007, is a reliable proxy for the stellar projected angular momentum (Bois et al. 2011). It is given by

$$\lambda_R = \frac{\sum_i^{N_p} F_i R_i |V_i|}{\sum_i^{N_p} F_i R_i \sqrt{V_i^2 + \sigma_i^2}} \quad (2.1)$$

where R_i is the distance to the center of mass of the i th pixel in a 2d histogram (70x70 bins) within the ellipse shown in Fig. 2.2 and V_i and σ_i are the respective mean line-of-sight velocity and mean line-of-sight velocity dispersion in the bin. The Flux F_i is replaced by the baryonic mass assuming a constant mass-to-light ratio.

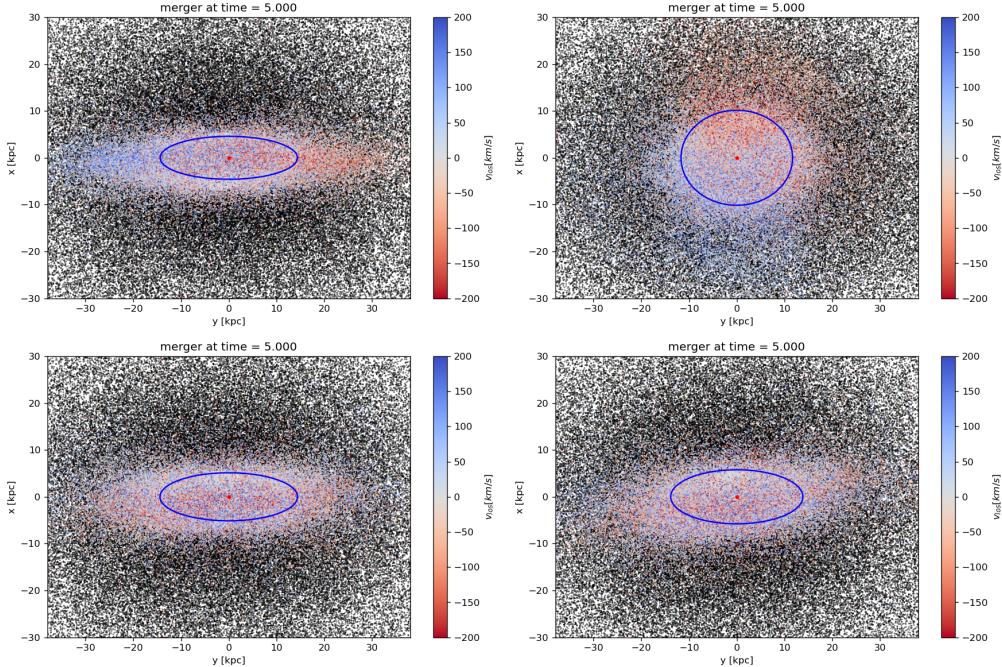


Figure 2.2: Remnants as produced by the merger simulations; from top left to bottom right: same, perp, counter, impact; the ellipses parameters are determined by the 0.05 and 0.95 quantiles of the x- and y-positions; the colorbar shows the line-of-sight velocity of baryons; dark matter particles are shown in black

The resulting λ_R parameter for each merger and the condition for fast rotators found by Emsellem et al. 2011, $\lambda_R > 0.31\sqrt{\epsilon}$, where $\epsilon = 1 - b/a$, are given in Tab 2.1. Their analysis was done at $1R_e$. Assuming this condition is also true for the above ellipses, same and perp are classified as fast, and counter and impact as slow, even though the latter two are close to the boundary. This quantitative result can already be expected by looking at the 2d histograms of the line-of-sight velocity used for calculating λ_R (Fig. 2.3). The counter remnants show no clear velocity pattern, whereas the other two both exhibit a clear rotation.

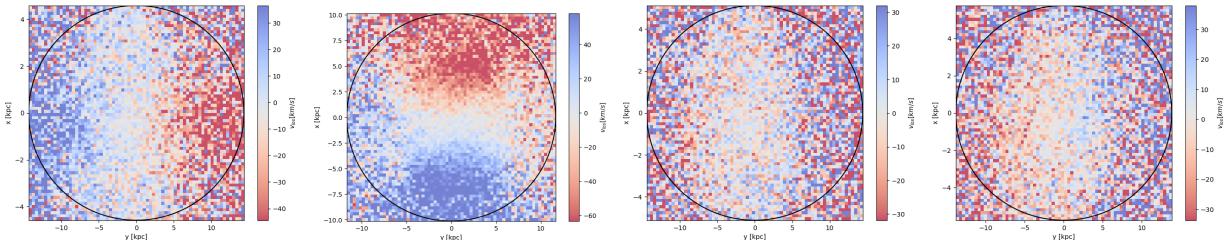


Figure 2.3: 2D histograms of the line-of-sight velocity of the center of the merger remnants; from left to right: same, perp, counter, impact

E2: A Collapsing Gas Cloud with Different Initial Velocities

Author: Eric Muires

This section aims to extend Tutorial 05 on the Evrard collapse of a gas, by changing the initial conditions. This is done in two different ways. First is by adding a small initial rotational velocity to the gas in an attempt to collapse it into a disk. Secondly, a more for-interest imaginary system where the initial rotational velocity is in opposite directions above and below the y-axis.

E2.1 Initial rotational velocity

In this simulation, the setup is identical to the one used in Sec. T5, however to add an additional velocity component the initial setup was given the following a solid-body set of velocities:

$$\mathbf{v} = \sqrt{x^2 + y^2}[-\sin(\theta)\hat{x} + \cos(\theta)\hat{y} + 0\hat{z}]\dot{\theta} \quad (2.2)$$

where $\sqrt{x^2 + y^2}$ is replaced by radius in the xy-plane, $\theta = \arctan(y/x)$, and $\dot{\theta}$ is a constant which has been set to 1.0 in code units. This corresponds to a cylindrical rotation where $\mathbf{v} = v\hat{\theta}$. This leads to a rotation in the xy-plane, with the intention to form a disk collapsed in z. Due to the nature of the arctan argument,

```
numpy.arctan2(y/x)
```

was used to avoid any issues with positive and negative quadrants. A snapshot of the resulting simulation is shown below in Fig. 2.4, visualising the collapsed gas in the xy-plane as well as the xz-plane.

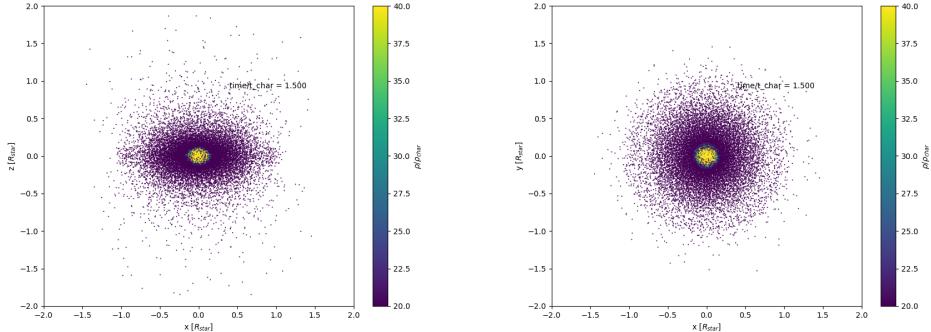


Figure 2.4: Gas density structure at $t = 1.5t_{char}$ showing disk-like structure in z-plane on left, and showing circular symmetry in the xy-plane.

As seen in Fig. 2.4 the gas cloud collapses into a disk-like shape as it rotates. Clearly, the z-axis is compressed and the xy-axes are elongated. The system suffers from remaining in this state, this is due to the initial conditions of the rotational velocities. They are intentionally set to not be keplerian velocity. This is done to more accurately mimic real gas turbulence, which would not be the exact perfect rotation we desire. This set-up did use a solid-body rotation for the initial setup, which is also not entirely physical. It still, however, provides an insight into how gas may collapse during rotation. The system evolves into the expected shape, and although it does not instantly collapse into the expected disk shape, it remains more-or-less in a disk. This is a success in demonstrating how disks may form even from rather fast rotating gas around a young star. Attempts to increase the initial velocity lead to most of the gas escaping, and attempts to decrease it returned the simulation to the rather uninteresting, and already well explored, result from Sec T5.

From this setup, the next step would be to attempt to add viscosity and for interest, a changing velocity based on the hemisphere in z.

E2.3 Oppositely rotating hemispheres

This section serves no physical reality, as a collapsing gas in nature does not have opposite velocities depending on hemisphere. However, this leads to an interesting solution in theory. With viscosity turned ON, this means that as the top and bottom hemisphere begin their collapse into a disk their velocities begin to cancel near the disk-plane, coming to a standstill. This then returns the problem to a Free-Fall collapse in the disk plane, which should create an interesting pattern as the central star begins to collapse rapidly. This was the idea behind the differential velocities. The setup was the same as the previous simulation, bar the velocity switch at negative z. The parameters were copied from T10 for the parameter file to make the viscosity behave properly.

To create this shear from a differing velocity, the original velocity vector \mathbf{v} is modified by:

$$\mathbf{v}_{shear} = \mathbf{v} \frac{\mathbf{z}}{|\mathbf{z}|} \quad (2.3)$$

where \mathbf{z} is a (N,1) column vector of positions in z. This expression can then be directly multiplied in PYTHON giving a modified array of velocities where every velocity in positions below the z-axis flips sign and rotates oppositely. The velocity in the z axis is preserved at 0. Since this setup is not particularly intuitive, seen in Fig.2.5

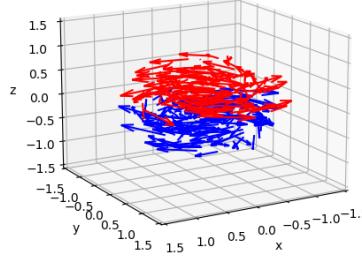


Figure 2.5: Vector plot of velocities showing the initial conditions, red is positive z-axis with an oppositely rotating gas in negative z shown in blue.

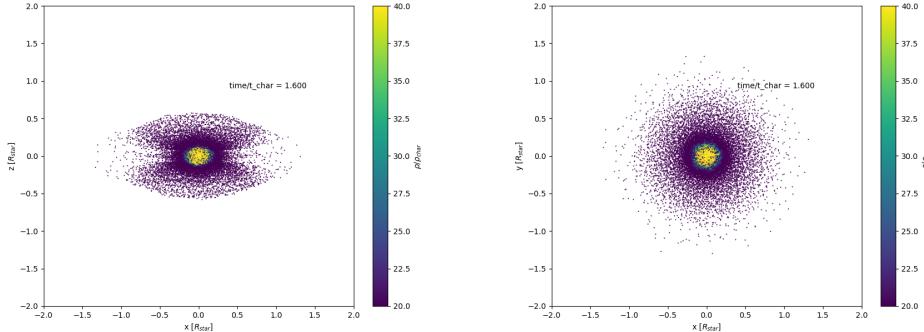


Figure 2.6: Gas density structure at $t = 1.6t_{char}$ showing the intermediate results of the oppositely rotating hemispheres of gas

Fig.2.6 shows the results of the gas as it begins to shear. Initially, two hemispheres begin their collapse onto the z-axis as expected and viscous forces begin acting on the gas on the disk-plane. Contradictory to predicted results where the gas would decelerate due to the viscous movements of oppositely rotating hemispheres, the particles depart from the central disk-plane. This is seen more clearly in the .gif found on GitHub. This can be explained as a result of the viscous forces and initial conditions. Since the gas is able to depart, the viscous force acting on the gas particles appears to accelerate them out of the system, likely a result of over-ambitious initial velocities. This still provides an interesting result as it demonstrates that the simulation can over-correct for collision effects if not treated carefully.

Acknowledgements

In the end we want to thank Dr. Ildar Khabibullin for his insightful lecture on Hydrodynamics, which not only included the physics of fluids but also introduced the some different schemes of N-Body and hydrodynamical simulations. He also helped us with running the above experiments during the tutorials.

We also want to thank PD Dr. Klaus Dolag, who normally gives this lecture and works on maintaining the Code P-Gadget3 that was used in the tutorials. He explained some aspects of how the code works during the tutorials and was there to fix the bigger problems.

Many thanks also to Dr. Tadziu Hoffmann who helped with the tutorials, once stayed late to find the problem that prevented the setup of Extension E1 and explained how the simulations are carried out on ltsp1 with more cores than ltsp14 allowed. He also helped with the setup of Extension E2.

Last but not least we want to thank Tapan Mayukh, our fellow master student, who we discussed with how to setup the initial conditions and how to plot the results using PYTHON every Monday in the last semester.

Bibliography

- M. Bois, E. Emsellem, F. Bournaud, K. Alatalo, L. Blitz, M. Bureau, M. Cappellari, R. L. Davies, T. A. Davis, P. T. de Zeeuw, P.-A. Duc, S. Khochfar, D. Krajnović, H. Kuntschner, P.-Y. Lablanche, R. M. McDermid, R. Morganti, T. Naab, T. Oosterloo, M. Sarzi, N. Scott, P. Serra, A.-M. Weijmans, and L. M. Young. The ATLAS^{3D} project - VI. Simulations of binary galaxy mergers and the link with fast rotators, slow rotators and kinematically distinct cores. *Monthly Notices of the Royal Astronomical Society*, 416(3):1654–1679, Sept. 2011. doi: 10.1111/j.1365-2966.2011.19113.x.
- W. Dehnen and H. Aly. Improving convergence in smoothed particle hydrodynamics simulations without pairing instability. *Monthly Notices of the Royal Astronomical Society*, 425(2):1068–1082, Sept. 2012. doi: 10.1111/j.1365-2966.2012.21439.x.
- S. Diehl, G. Rockefeller, C. L. Fryer, D. Riethmiller, and T. S. Statler. Generating Optimal Initial Conditions for Smoothed Particle Hydrodynamics Simulations. *arXiv e-prints*, art. arXiv:1211.0525, Nov. 2012.
- E. Emsellem, M. Cappellari, D. Krajnović, G. van de Ven, R. Bacon, M. Bureau, R. L. Davies, P. T. de Zeeuw, J. Falcón-Barroso, H. Kuntschner, R. McDermid, R. F. Peletier, and M. Sarzi. The SAURON project - IX. A kinematic classification for early-type galaxies. *Monthly Notices of the Royal Astronomical Society*, 379(2):401–417, Aug. 2007. doi: 10.1111/j.1365-2966.2007.11752.x.
- E. Emsellem, M. Cappellari, D. Krajnović, K. Alatalo, L. Blitz, M. Bois, F. Bournaud, M. Bureau, R. L. Davies, T. A. Davis, P. T. de Zeeuw, S. Khochfar, H. Kuntschner, P.-Y. Lablanche, R. M. McDermid, R. Morganti, T. Naab, T. Oosterloo, M. Sarzi, N. Scott, P. Serra, G. van de Ven, A.-M. Weijmans, and L. M. Young. The ATLAS^{3D} project - III. A census of the stellar angular momentum within the effective radius of early-type galaxies: unveiling the distribution of fast and slow rotators. *Monthly Notices of the Royal Astronomical Society*, 414(2):888–912, June 2011. doi: 10.1111/j.1365-2966.2011.18496.x.
- M. Jubelgas, V. Springel, and K. Dolag. Thermal conduction in cosmological SPH simulations. *Monthly Notices of the Royal Astronomical Society*, 351(2):423–435, June 2004. doi: 10.1111/j.1365-2966.2004.07801.x.
- N. Katz, D. H. Weinberg, and L. Hernquist. Cosmological Simulations with TreeSPH. *The Astrophysical Journal Supplement Series*, 105:19, July 1996. doi: 10.1086/192305.
- E. Laurikainen, H. Salo, R. Buta, J. H. Knapen, and S. Comerón. Photometric scaling relations of lenticular and spiral galaxies. *Monthly Notices of the Royal Astronomical Society*, 405(2):1089–1118, June 2010. doi: 10.1111/j.1365-2966.2010.16521.x.
- Y. Li, H. J. Mo, and L. Gao. On halo formation times and assembly bias. *Monthly Notices of the Royal Astronomical Society*, 389(3):1419–1426, sep 2008. doi: 10.1111/j.1365-2966.2008.13667.x. URL <https://doi.org/10.1111%2Fj.1365-2966.2008.13667.x>.
- D. Ryu and T. W. Jones. Numerical Magnetohydrodynamics in Astrophysics: Algorithm and Tests for One-dimensional Flow. *The Astrophysical Journal*, 442:228, Mar. 1995. doi: 10.1086/17543710.48550/arXiv.astro-ph/9404074.

- M. Steinmetz and E. Mueller. On the capabilities and limits of smoothed particle hydrodynamics. *Astronomy and Astrophysics*, 268(1):391–410, Feb. 1993.