Towards AMR Simulations of Galaxy Formation

Nigel Lewis Mitchell

Abstract

Numerical simulations present a fundamental building block of our modern theoretical understanding of the Universe. As such the work in this thesis is primarily concerned with understanding fundamental differences that lie between the different hydrodynamic schemes. In chapter 3 I outline the optimisations I make to the *FLASH* code to enable larger simulations to be run. These include developing and testing a new FFT gravity solver. With these complete, in chapter 4 I present results from a collaborative code comparison project in which we test a series of different hydrodynamics codes against a suite of demanding test problems with astrophysical relevance. As the problems have known solutions, we can quantify their performance and are able to develop a resolution criteria which allows for the two different types to be reliably compared.

In chapter 5 we develop an analytic model for ram pressure stripping of the hot gaseous haloes of galaxies in groups and clusters. We test the model against a suite of hydrodynamic simulations in the SPH *GADGET-2* code. To ensure that the spurious suppression of hydrodynamic instabilities by SPH codes does not bias our results, I compare our findings to those obtained with the *FLASH* AMR code and find excellent agreement.

Chapter 6 presents work in which we unambiguously determine the origin of the difference between the entropy cores formed in AMR and SPH codes. By running mergers of model clusters we are able to systematically explore the various proposed mechanisms and determine that turbulent mixing generates the higher entropy cores within AMR codes but is suppressed in SPH codes.

The startling differences between the two hydrodynamic schemes presented in chapter 6 leads me to investigate their affect upon different sub-grid physical recipes. In chapter 7 I present the implementation of a sub-grid star formation recipe in *FLASH* and find strong differences in the way the two codes model pressure laws. I extend the work in chapter 8 by implementing a kinetic supernova feedback mechanism in *FLASH* and contrasting it with the results from the *GADGET-2* code. I find that AMR codes dissipate energy much more efficiently than in SPH codes.

Towards AMR Simulations of Galaxy Formation

by Nigel Lewis Mitchell

A thesis submitted to the University of Durham
in accordance with the regulations for
admittance to the Degree of Doctor of Philosophy.

Department of Physics
University of Durham
June 2010

Contents

List of Figures

List of Tables

Declaration

I hereby declare that the work presented in this thesis was undertaken between 2005 and 2009 whilst the author was a research student at the Department of Physics at Durham University under the supervision of Professor Richard Bower and Dr. Tom Theuns.

This work is not substantially similar to any that I have submitted for a degree or diploma or other qualification at any other University. I further state that no part of this thesis has already been or is being concurrently submitted for any such degree, diploma or other qualification.

This thesis is the result of my own work and includes nothing which has been done in collaboration except where specifically indicated in the text. Some figures and tables are reproduced from the work of other authors, and are always credited in the associated caption. I list here the parts of this thesis which have been published. The simulations presented in Chapter 5 were presented as part of the following paper:

"Ram Pressure Stripping the Hot Gaseous Halos of Galaxies in Groups and Clusters,"
McCarthy, I. G., Frenk, C. S., Font, A. S., Lacey, C. G., Bower, R. G., Mitchell, N. L., Balogh, M. L., Theuns, T., 2007, **MNRAS**, 383, 593M.

Chapter 4 was published as part of the following paper for which the author was responsible for all of the *FLASH* work as well as joint work on the analysis:

"A Test Suite for Quantitative Comparison of Hydrodynamics Codes in Astrophysics,"
Tasker, E. J., Brunino, R., Mitchell, N. L., Michielsen, D., Hopton, S., Pearce, F. R., Bryan, G. L., Theuns, T., 2008, **MNRAS**, 390, 1267T.

Chapter 6 was published in the following paper for which the author performed all of the *FLASH* simulations and performed a joint analysis with the co-authors:

"On the Origin of Cores in Simulated Galaxy Clusters,"
Mitchell, N. L., McCarthy, I. G., Bower, R. G., Theuns, T., Crain R. A., 2009, **MNRAS**, 395, 180M.

Chapters 7 and 8 are due to be published. The author is responsible for all work presented unless explicitly specified.

Acknowledgements

This thesis stands not just a testament to my efforts over the past four years but also to the great number of people that have helped me. I consider myself extremely fortunate to have had the support of so many talented individuals and although I can attempt to thank them all here, I am in no doubt that my words cannot truly convey how deeply grateful I am to them all.

First and foremost I would like to thank both of my supervisors for the immense amount of time and effort that they have put in over the past four years. Professor Richard Bower's endless source of enthusiasm and all encompassing knowledge of astronomy has helped keep me enthusiastic when the going got tough and ensured that I didn't lose sight of the big picture when I got lost in the details. Dr. Tom Theun's meticulous attention to detail and thorough understanding of numerical techniques has helped keep me on the straight and narrow, taught me to question what I do and to develop my rudimentary skills into a rigorous scientific discipline. I have found the combination of the two to be an exceptional mix which perfectly compliment each other. They have helped me an inordinate amount and I owe them a great deal.

Thanks goes to those that have collaborated with me; Dr. Ricardo Brunino, Dr. Rob Crain, Dr. Frazer Pearce, Dr. Milan Raicevic, Dr. Paul Ricker, and Dr. Elizabeth Tasker. It has been a pleasure working with you all. It has been an especially great privilege to work with my colleague and part mentor Dr. Ian McCarthy. His genius has helped fuel my research whilst his humour always made it fun. Without the tireless efforts of Dr. Lydia Heck the simulations presented here would never have come to be. Upon each of the many occasions I knocked on her door, she dropped what she was doing to help me, regularly working long into the night. I am truly indebted to her.

Finally I would like to offer a great thanks to all of the loving support from my family. To my aunty Fiona who helped ferry me back and forth between Durham and home, to my Grandmother, Grandad and cousin Heidi who have always made me feel extra special and to my brothers; Alex and Daniel who regularly quiz me on everything astronomical and physical in nature. Their enthusiasm and interest in my work has always made it feel so much more worth while. Ultimately none of my successes could ever have been achieved without the loving care and encouragement from my mother. Ever since a young age she brought me countless books to fuel my thirst for knowledge and laid the foundations for a lifetime. Thank you everyone!

Preface

"Astronomy leads us to a unique event, a universe which was created out of nothing, one with the very delicate balance needed to provide exactly the conditions required to permit life, and one which has an underlying (one might say 'supernatural') plan."

Arno Penzias (Nobel prize in physics for discovery of the Cosmic Microwave Background)

Thus it is that ever since Man first looked up into the heavens, fascinated by what he saw, he has striven to try and understand what forces govern and bind everything around us. In this, the study of astronomy is to many a way of gaining a deeper understanding of the way the "maker" thinks. With simple insight, Penzias highlights the underlying reason why we study cosmology - we wish to comprehend that supernatural plan, from its inception at the "Big Bang" to its ultimate and as of yet unknown grand finale.

Galaxy formation provides a remarkable tool with which to try and determine the answer to these questions. They are the building blocks of the Universe, providing the seeds for life and a laboratory for testing the most exotic physics. Even now the energies involved at the Universe's birth are still well beyond even the most ambitious experiments by Man. Clusters of galaxies are the largest bound structures in the Universe, containing a myriad of different competing physical processes. The number of different galaxies and stars they contain of all different types and ages represents an average cosmological sample. Thus they are the closest we can get to looking in on the Universe from the outside and gaining that tentative glimpse of what is to come.

Chapter 1 | *Introduction*

Numerical simulations underpin most modern theoretical advances in astronomy and provide a firm test bed for new and existing models. Despite their importance to the field, it is often unknown what affect the many different numerical implementations have upon the accuracy of the models they represent. When a new interesting effect is observed, is it a potentially key source of new physics or is it a numerical artefact? Throughout the work presented in this thesis I will attempt to approach various discrepant issues between theory and observation and investigate to what extent these conflicts can be resolved by improving the way numerical schemes are used. By contrasting the results from different numerical schemes we can gain an invaluable insight into the intricacies and limitations of these techniques and from this begin to develop superior next generation codes.

I begin in this chapter by outlining the building blocks of the Universe; its general composition (§ 1.1) and some of the many different physical processes that play key roles in galaxy formation. These include the effects of gravity and tidal stripping (§ 1.2) as well as hydrodynamic interactions (§ 1.3), including shocks (§ 1.3.1), turbulent instabilities (§ 1.3.2) and radiative cooling (§ 1.3.3).

Through § 1.5 I will present an outline of the inter-stellar medium and the difficulties presented when modelling it self-consistently in full cosmological simulations. I will describe the process of star formation (§ 1.5.1) and how they generate metals (§ 1.5.2). § 1.5.3 and § 1.5.4 will describe the two types of supernova and how they both heat and enrich the inter-stellar medium.

In § 1.6 I will summarise the basic observable properties of the different galaxy types including the Milky Way and other spirals (§ 1.6.1), ellipticals (§ 1.6.2), lenticulars (§ 1.6.3), dwarfs (§ 1.6.4) along with groups and clusters of galaxies (§ 1.6.5).

In § 1.7 I will then introduce the different numerical techniques that have allowed us to study processes as complex as galaxy formation, such as N-body simulations (§ 1.7.2), semi-analytics (§ 1.7.4) and full hydrodynamic simulations (§ 1.7.5).

I will round off with a summary of the current hierarchical paradigm (§ 1.8) and our view of galaxy formation (§ 1.8.1) including proposed solutions for the apparent

discrepancy between the number of satellite galaxies predicted through dark matter only simulations and those actually observed (§ 1.8.2). I will also describe the problems with the "cooling flow crisis" and the current lack of consensus on how cool cores and non-cool cores are made (§ 1.8.4). I will finish by outlining the structure of this thesis.

1.1 Building Blocks of the Universe

The fundamental building blocks of the Universe can be grouped into three main categories; baryonic matter which constitutes around 4.4% of the total mass-energy of the Universe, dark matter which represents 22% and dark energy which accounts for the remaining 73% of the total mass-energy of the Universe (see, e.g., Bennett et al., 2003; Hinshaw et al., 2009). Despite representing a minor fraction of the total mass-energy in the Universe, baryonic matter is the most readily observable and has been used to infer the existence of the two more major contributors to the total mass-energy content. Both dark and baryonic matter are influenced by gravity with baryonic matter also being subject to the Strong, Weak and Electromagnetic interactions.

Dark matter has yet to be directly detected but a wealth of observational evidence is available to infer its existence. Fritz Zwicky first hypothesised its existence in 1933 when looking at the motions of nebulae within the Coma cluster (see Zwicky, 1933, 1937). By assuming that the cluster was in Virial equilibrium, he was able to relate the square of the kinetic energies of the galaxies within Coma to the gravitational potential energy of the system. He arrived at a much larger mass for the system than was observable in the form of gas and luminous stellar material concluding that there must be another unseen form of matter present, gravitationally binding the system together.

His initial hypothesis was later supported by studies of the rotation curves of stars within individual galaxies. Those at the edges of the galaxy are seen to move much faster than they should if they remain bound to the galaxy only through gravitational attraction to the other baryonic matter according to Newtonian mechanics. Thus there must be far more mass than is detected observationally otherwise stars and gas would have been thrown out of the galaxy a long time ago (see, e.g., Rubin et al., 1980; Rubin, 1983).

In addition the gravitational lensing of background objects by intervening massive galaxy clusters allows for an alternative means with which to measure the actual mass of galaxy clusters. All of these techniques have proven to infer much higher masses than

that present in luminous baryonic matter.

The difficulty directly detecting dark matter in collisional detectors (e.g. UK Boulby mine, Minnesota mine, Sudan and SNOLAB underground detectors) has led to the belief that it possesses an extremely small cross section and is therefore effectively collisionless.

The third component, dark energy is still poorly understood and was invoked to explain the current increase in the expansion rate of the Universe and is usually modelled through the introduction of a cosmological constant or through a scalar field.

1.2 Gravity

Given the immense size of the cosmos it is often surprising how the relatively weak force of gravity acting between two bodies can dominate interactions on the grandest of scales so completely. The Strong nuclear force which binds quarks and gluons together, forming the basic building blocks of baryonic matter is some 10^{39} times stronger than gravity. The Weak interaction which mediates the decay of elementary particles is still $\sim 10^{37}$ times stronger. You would therefore expect gravity to be negligible and indeed it is over sub-atomic scales, however as they both fall off rapidly beyond this, gravity can take precedence on astronomical scales. Even the electromagnetic force which has an infinite range and is some 10^{37} times stronger, fails to contribute to structure formation as unlike gravity it can both attract *and* repel charged bodies and the Universe as a whole appears neutral. Thus no force can quite compete with gravity in terms of its absolute influence on the Universal large scale structure.

Newton's law of gravitation states that the force of attraction between N bodies is given by,

$$m_i \mathbf{a_i} = \sum_{n}^{N-1} -\frac{G m_i m_n}{r_{ni}^2} \hat{\mathbf{r}}_{\mathbf{ni}} \tag{1.1}$$

where the net acceleration vector $\mathbf{a_i}$ for the i^{th} particle is determined by summing the individual contributions between itself and each of the other $N-1$ particles. The masses m_i and m_n represent the mass of the i^{th} particle and the n^{th} particle that we are summing over with connecting position vector, $\hat{\mathbf{r}_{\mathbf{ni}}}$. The above calculation is performed separately for each of the N particles in the simulation. Newton's law of gravitation is found to be a very good approximation to the behaviour of gravity on solar to cosmological scales and only requires Einstein's more complex description in General relativity in the close vicinity of very compact massive objects such as black holes.

1.2.1 Tidal Stripping

As galaxies pass by each other, stars, gas and dark matter can be preferentially stripped off the smaller object through gravitational attraction in a process referred to as tidal stripping.

For a test particle in a bound structure with a fixed gravitational potential, the escape velocity (equation 1.2) is dependent upon the host structure's mass. The more massive an object, the higher the escape velocity and the more strongly bound material is.

$$v_{escape} = \sqrt{\frac{2GM_{galaxy}}{r}} \tag{1.2}$$

In the case of tidal stripping, the perturbing presence of other gravitating bodies can help accelerate particles out of their local gravitational minima within the galaxy, allowing them to either accelerate out into space or to fall into the deeper potentials of the more massive galaxies. This causes the gravitational potential to vary dynamically over time. A tidal radius, r_t, beyond which material in the less massive galaxy is stripped away by the larger galaxy or group of galaxies can be defined as

$$\frac{r_t}{R} = \left(\frac{M_{gal}(r_t)}{M_{grp}(R)(3 - d\ln M_{grp}/d\ln R)} \right)^{1/3} \tag{1.3}$$

where R is the three dimensional group-centric radius of the galaxy, $M_{gal}(r)$ is the total mass of the orbiting galaxy within radius r, and $M_{grp}(R)$ is the total mass of the larger galaxy, group or cluster within radius R (see King, 1966; Binney and Tremaine, 1987).

Structures that are too tightly bound to be stripped can still experience changes in morphology resulting from this gravitational harassment.

1.3 Hydrodynamics

Baryonic matter can occupy an enormous range of physical states, from simple ionised and molecular gases which can condense to form dust particles, to the dense planetary bodies and stars. A large fraction of the baryonic mass in the Universe is in the form of gas, the behaviour of which can be described using the Euler hydrodynamic equations. These are relatively simple to derive analytically. I start by first defining a flux as the amount of a given quantity (mass, momentum, energy) which passes through a given surface over a unit of time.

For a given surface, ∂V, with unit normal, $\hat{\mathbf{n}}$, I can define a mass flux as

$$\rho \mathbf{v} \cdot \hat{\mathbf{n}} dA \tag{1.4}$$

a momentum flux due to transport of material and the acceleration due to a pressure force, P along the normal as

$$[\rho \mathbf{v}\mathbf{v}\hat{\mathbf{n}} + P\hat{\mathbf{n}}]\, dA \tag{1.5}$$

and a total energy flux due to the transport of material and the work done by the pressure force as

$$\rho\,[\varepsilon + P]\,\mathbf{v}\cdot\hat{\mathbf{n}} dA \tag{1.6}$$

For a fixed volume the only change to the conserved quantities of mass, momentum and energy can occur through the flux of material through the surface ∂V. Thus integrating the equations over space and time gives

$$\int_V \rho(x,t_1)\ dV - \int_V \rho(x,t_0)\ dV \;=\; -\int_{t_0}^{t_1} \oint_{\partial V} \rho\mathbf{v}\cdot\hat{\mathbf{n}}\ dA\ dt \tag{1.7}$$

$$\int_V \rho\mathbf{v}(x,t_1)\ dV - \int_V \rho\mathbf{v}(x,t_0)\ dV \;=\; -\int_{t_0}^{t_1} \oint_{\partial V} [\rho\mathbf{v}\mathbf{v} + P]\,\hat{\mathbf{n}}\ dA\ dt \tag{1.8}$$

$$\int_V \rho\varepsilon(x,t_1)\ dV - \int_V \rho\varepsilon(x,t_0)\ dV \;=\; -\int_{t_0}^{t_1} \oint_{\partial V} [\rho\varepsilon + P]\,\mathbf{v}\cdot\hat{\mathbf{n}}\ dA\ dt \tag{1.9}$$

rearranging equation 1.7 allows it to be written as:

$$\int_V [\rho(x,t_1) - \rho(x,t_0)]\ dV + \int_{t_0}^{t_1} \oint_{\partial V} \rho\mathbf{v}\cdot\hat{\mathbf{n}}\ dA\ dt = 0 \tag{1.10}$$

Taking equation 1.10 and dividing through by $(t_1 - t_0)$ gives

$$\int_V \frac{\rho(x,t_1) - \rho(x,t_0)}{(t_1 - t_0)}\ dV + \frac{1}{(t_1 - t_0)} \int_{t_0}^{t_1} \oint_{\partial V} \rho\mathbf{v}\cdot\hat{\mathbf{n}}\ dA\ dt = 0 \tag{1.11}$$

Now taking the limit at which $t_1 \longrightarrow t_0$ and assuming that the derivative $\frac{\partial\rho}{\partial t}$ exists for all x implies,

$$\int_V \frac{\partial\rho}{\partial t}\ dV + \oint_{\partial V} \rho\mathbf{v}\cdot\hat{\mathbf{n}}\ dA = 0 \tag{1.12}$$

Applying Gauss's theorem to convert the surface integrand to a volume integrand gives:

$$\int_V \frac{\partial\rho}{\partial t}\ dV + \int_V \nabla\cdot\rho\mathbf{v}\ dV = 0 \tag{1.13}$$

By merging the two integrands I come to

$$\int_V \left(\frac{\partial\rho}{\partial t} + \nabla\cdot\rho\mathbf{v}\right)\ dV = 0 \tag{1.14}$$

As this has to be true for all (even small) volumes, I can conclude that the integrand has to be zero, giving me the differential form of the mass transport equation.

$$\frac{\partial\rho}{\partial t} + \nabla\cdot\rho\mathbf{v} = 0 \tag{1.15}$$

Following the same approach equations 1.8 and 1.9 can be re-written in similar form:

$$\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v}\mathbf{v}) + \nabla P = \rho \mathbf{g} \tag{1.16}$$

$$\frac{\partial \rho E}{\partial t} + \nabla \cdot [(\rho E + P)\,\mathbf{v}] = \rho \mathbf{v} \cdot \mathbf{g} \tag{1.17}$$

These Euler hydrodynamic equations allow us to determine the way in which compressible fluids evolve over time from some initial state and form a vital tool for modern astrophysical simulations.

1.3.1 Hydrodynamic Shocks

Shocks are formed when a pressure discontinuity arises. In gases this normally occurs when material is travelling faster than the local sound speed in the gas, c_s. For a gas, the adiabatic sound speed is defined as,

$$c_s = \sqrt{\frac{\gamma P}{\rho}} \tag{1.18}$$

where γ is the adiabatic index, P the pressure and ρ is the density. As information in a fluid can only be transmitted at the sound speed, a shock wave that travels above the sound speed does not influence material in front of it. Thus the medium does not react to its presence until they spatially coincide. This can lead to a discontinuity in both the pressure and density. A shock wave acts to return the system to a new pressure equilibrium, driving a wave of higher pressure towards the lower pressure region.

The shock wave that forms can be broken down into three components; the shock front, a contact discontinuity and a rarefaction (see figure 1.1). At the shock front the density jumps as material just before the shock is instantaneously compressed by the leading edge of the shock wave. This results in a sharp jump in the density, pressure and velocity. The shock front corresponds to the current interface between gas in the pre and post-shock states.

At the contact discontinuity the density changes abruptly but the pressure remains constant. From the back of the contact discontinuity passes a rarefaction sound wave which travels in the opposite direction to the shock wave. As we pass along the rarefaction, the velocity of the medium, the pressure and density smoothly return to the values they possessed on the higher pressure side before the shock wave was generated. The rapid change in density and pressure at the shock front results in rapid thermalisation of kinetic energy.

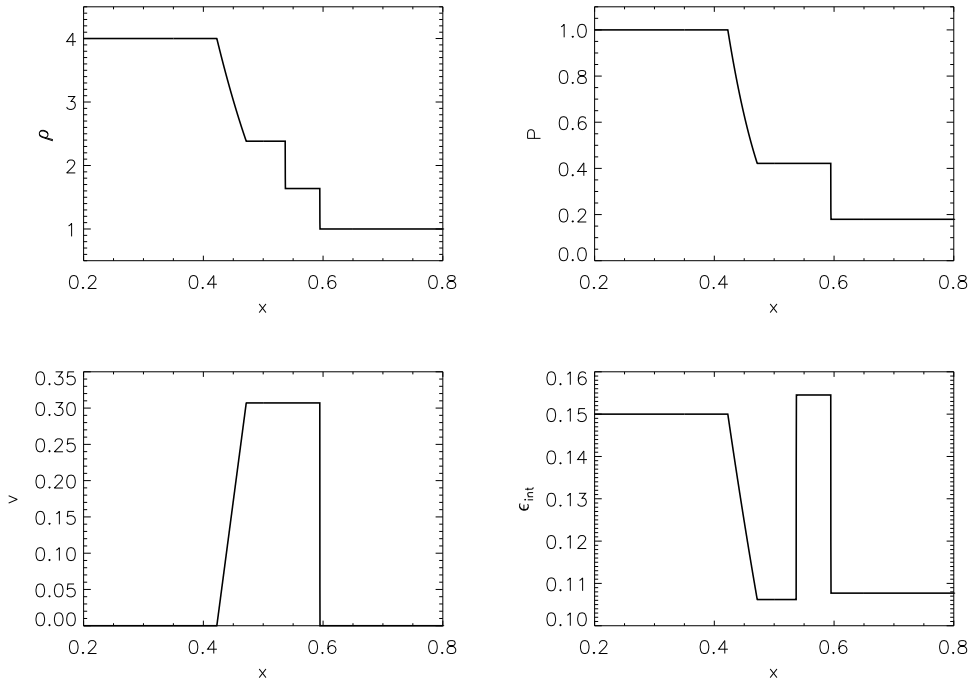

Figure 1.1: Cross section of a shock wave travelling from left to right. Plotted are the density (top left), pressure (top right), velocity (bottom left) and internal energy (bottom right) of the gas. The shock front at $x = 0.6$ is responsible for thermalising large amounts of kinetic energy leading to the large jump in ε_{int}. Whilst the pressure and fluid velocity remain constant across the contact discontinuity at $x = 0.525$, the density sharply increases. The rarefaction between $x = 0.4 - 0.45$ corresponds to a sound wave travelling to the left away from the shock front. It leads to a gradual return of the density, velocity and pressure to the initial left hand gas state.

Shocks are often characterised using a Mach number which gives their velocity relative to the sound speed. Those travelling much faster than the sound speed are termed "supersonic" whereas weak shocks that are just above the sound speed with a Mach number of around one are termed "transonic."

1.3.2 Turbulent Instabilities and Ram Pressure Stripping

1.3.2.1 Kelvin-Helmholtz Instabilities

When the interface between two different but continuous fluids is subject to a velocity shear, Kelvin-Helmholtz instabilities can arise. These normally appear as small waves rising out away from the interface and can lead to mixing of material in the near vicinity of the interface. As time progresses, the amplitude of these instabilities increases and forms swirling vortices like those shown in figure 1.2 until the interface between the fluids becomes smeared out and material in the two effectively mixed. The formation of Kelvin-Helmholtz instabilities can be suppressed if there exists additional surface tension or if the media are particularly viscous.

The timescale over which Kelvin-Helmholtz instabilities form is given by (see, e.g. Nulsen, 1982; Price, 2008),

$$\tau_{KH} \equiv \frac{2\pi}{\omega} \tag{1.19}$$

where

$$\omega = k\frac{(\rho\rho')^{1/2}v_{rel}}{(\rho+\rho')} \tag{1.20}$$

with ρ and ρ' the densities of lighter and denser mediums, k is the wave number of the instability and v_{rel} is the relative shear velocity of the two fluids. When calculating the time required to completely disrupt an object it is worth noting that the wave responsible for its destruction is typically of order the same size. Therefore for a fluid stream of length, r', the wavelength responsible for completely disrupting it would be $k \sim 2\pi/r'$,

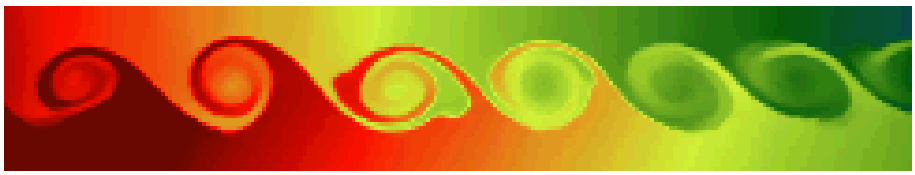

Figure 1.2: Image of a Kelvin-Helmholtz instability forming between two different continuous fluids with a velocity shear at the interface. The waves rising away from the interface give rise to mixing of the fluids in a fractal turbulent cascade

thus equations 1.19 and 1.20 reduce to give,

$$\tau_{KH} \approx \frac{r'}{v_{rel}} \left(\frac{\rho'}{\rho} \right)^{1/2} \tag{1.21}$$

1.3.2.2 Rayleigh-Taylor Instabilities

Another form of hydrodynamic instability is the Rayleigh-Taylor instability which is formed whenever two fluids of different densities force upon each other. Figure 1.3 reproduced from Li and Li (2004) shows the time evolution of a Rayleigh-Taylor instability as a denser fluid (blue) falls under gravity through the lighter medium (yellow). As the lighter fluid is displaced by the leading end of the dense fluid, it is drawn into the void left behind. As the two compete to minimise their potentials, the lighter fluid is pulled down behind the dense leading knot by convective currents and pierces into it, disrupting and mixing the two media. Due to the velocity shear along the fluid interfaces, many Kelvin-Helmholtz instabilities are also seen forming. As the dense knot passes through the less dense medium it can also become subject to ram pressure stripping. Due to its motion relative to the lighter fluid, the dense gas can be preferentially stripped off. Ram pressure stripping can be reduced in media where strong magnetic fields are present, the gas is highly viscous or in the case of gravitationally bound structures - the gravitational field strength is high. However in smaller less tightly bound structures such as dwarf galaxies, such ram pressure stripping may be an important physical effect.

Both Kelvin-Helmholtz and Rayleigh-Taylor instabilities are important for driving mixing of material in a cascading turbulent flow. Despite being a potentially important mechanism in galaxy formation both for the redistribution of high mass elements from star formation and for the redistribution of entropy, the nature of turbulence is still relatively poorly understood. It is not yet known upon which scales the turbulent cascade terminates - whether at macroscopic scales or if it continues down to atomic scales. The work of Kolmogorov (1941) theorised that there is a scale invariant level upon which any turbulent cascade dissipates energy. If this is true then turbulence should produce a predictable Kolmogorov power spectrum in the velocity and density fluctuations and only a finite level of resolution should be necessary to fully resolve the turbulent cascade.

1.3.3 Metal Dependent Cooling

As gas collapses into gravitational potential wells, it compresses and shocks. Before it can continue to collapse it must dissipate this excess energy and as a result the cooling

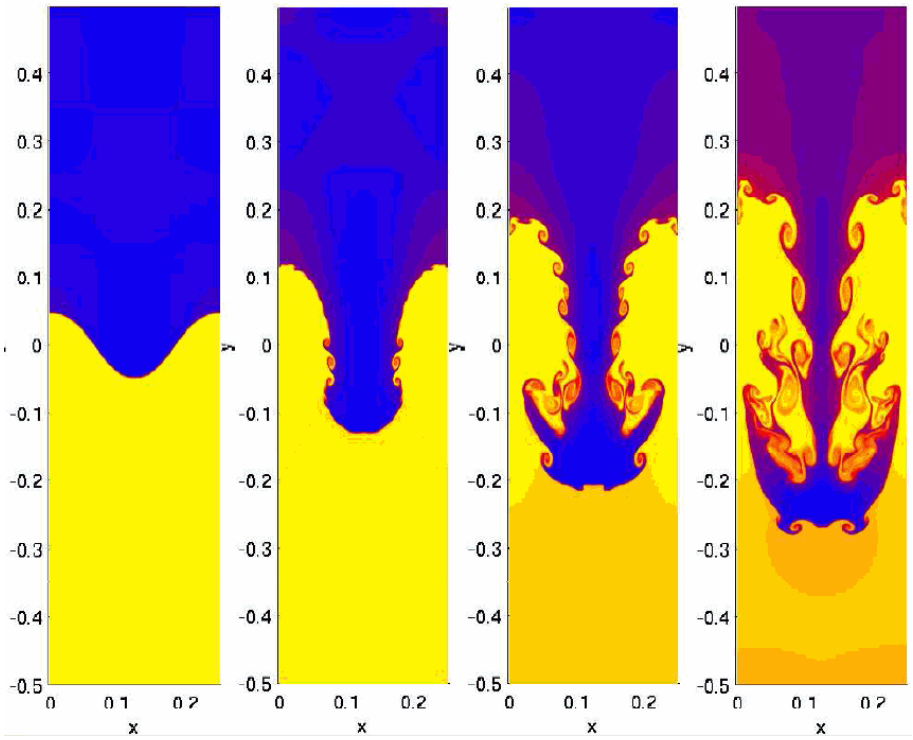

Figure 1.3: Rayleigh-Taylor instability forming as a denser fluid (blue) falls under gravity through a less dense fluid shown at increasing times from left to right. As the lighter fluid is displaced by the denser one, both attempt to minimise their potential energies with the lighter fluid rushing into the void left behind by the denser material. As the lighter fluid rushes in it is drawn up behind the leading dense knot. As it penetrates the denser fluid, it disrupts the knot and mixes the two fluids. Due to the velocity shears, Kelvin-Helmholtz instabilities can also be seen forming at the interfaces of the two fluids. Image courtesy of Li and Li (2004), Los Alamos National Laboratory.

rate directly affects structure formation and the rate of star formation. Radiative cooling can occur via three pathways; free-free, free-bound and bound-bound transitions.

Free-free transitions involve the emission of a photon of electromagnetic radiation as an electron is accelerated through the electrostatic field of an ion. This Bremsstrahlung emission occurs in highly ionised plasmas at temperatures above $\sim 10^7$K in regions such as the hot gas in the haloes of galaxy clusters - the intra-cluster medium (ICM) or around Active Galactic Nuclei (AGN).

Below temperatures of $\sim 3\times10^7$K heavier nuclei begin to recombine with free electrons resulting in free-bound emission. Bound-bound transitions occur as electrons decay from one quantised energy state to a lower bound energy state. As the energy states within an atom, ion or molecule are quantised, the photons emitted form a spectra which is unique to that particular species. These spectra have allowed the chemical composition of stars and light emitting gas to be determined, providing a wealth of information about astrophysical environments. The presence of higher mass elements than helium, collectively referred to as "metals," increases the cooling rate since they present more excitation states with which to allow free-bound and bound-bound transitions. The presence of neutral gas between an emitting object and the observer can result in an absorption line or feature as the neutral gas is excited, reducing the transmission of energy at wavelengths equivalent to the quantised energy levels within the absorbing gas.

Calculating the cooling due to the different free-free, free-bound and bound-bound interactions for a multitude of species with different relative abundances, densities and temperatures can be a demanding task. Consequently tables of the cooling function $\Lambda(T)$ are often calculated by dedicated numerical schemes and then tabulated (see for example the CLOUDY code of Ferland et al., 1998). Numerical simulations can then determine the energy loss by finding the corresponding value of $\Lambda(T)$ as a function of density, metallicity and temperature. An example of the cooling function as a function of temperature is plotted in figure 1.4 for $n_H = 0.1$ cm^{-3} and includes the ionising effects of a Haardt and Madau (2001) meta-galactic UV/X-ray background. The contributions to the cooling function are plotted separately for hydrogen & helium (black-dashed), carbon (red), nitrogen (blue), oxygen (purple), neon (green), magnesium (orange), combined silicon, sulphur and calcium (cyan) and iron (yellow) relative to the total rate for solar abundances (black). The contribution due to the higher mass elements becomes more important as the temperature drops below $\sim 10^7$K and they begin to recombine. Above this the cooling function is dominated by Bremsstrahlung as all species are fully ionised.

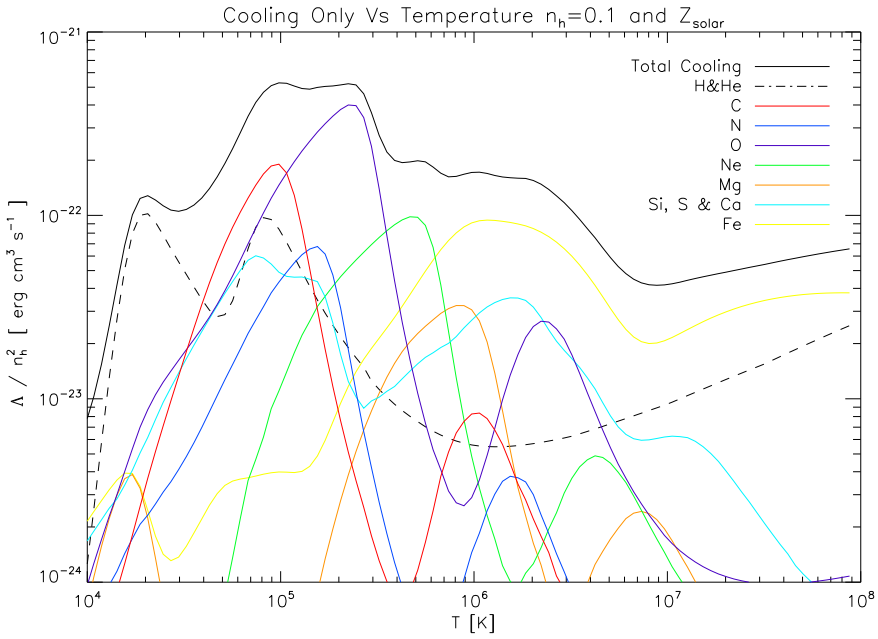

Figure 1.4: Cooling function $\Lambda(T)$ per unit number density squared, as a function of temperature. The contributions from hydrogen & helium (black-dashed), carbon (red), nitrogen (blue), oxygen (purple), neon (green), magnesium (orange), combined silicon, sulphur and calcium (cyan) and iron (yellow) are plotted separately with the combined total in black. The results are shown for solar abundances with $n_H = 0.1\,\mathrm{cm}^{-3}$ and include the ionising effects of a meta-galactic UV/X-ray background shown here for z=0.

Inverse-Compton scattering also plays an important role at very high temperatures as fast moving, high energy electrons impart energy when they collide with low energy photons in the Cosmic Microwave Background (see § 1.8). At temperatures below 10^7K the higher mass elements such as iron, neon and oxygen recombine to dominate the cooling function until we reach very low temperatures at which point hydrogen dominates.

The cooling function $\Lambda(T)$ is highly sensitive to the density and typically scales as the number density squared, n_H^2. As the density increases sharply as we tend towards the core of most astrophysical structures such as the cores of galaxy clusters, the cooling rate accelerates rapidly leading to a runaway cooling effect.

1.4 The Jeans Criterion and Gravitational Instability

The Jeans criterion defines the point at which the hydrodynamic pressure of the fluid is no longer able to oppose gravitational collapse. In a system that is in hydrostatic equilibrium,

$$\frac{dP}{dr} = -\frac{G\rho M_{enclosed}}{r^2} \tag{1.22}$$

small deviations in the pressure or density can be damped down providing that they are small enough, smoothing them out and maintaining equilibrium. However when the magnitude of the perturbation is sufficiently large enough, it becomes amplified causing the cloud to fragment and collapse. This happens when the free fall time (equation 1.23) is less than the time taken for sound waves (equation 1.18) within the medium to act against the perturbation by increasing the pressure gradient.

$$t_{ff} = \sqrt{\frac{1}{G\rho}} \tag{1.23}$$

Thus by combining the two equations it is possible to determine the Jeans length - the minimum scale over which density perturbations can grow; λ_J.

$$\lambda_J = c_s\sqrt{\frac{1}{G\rho}} \tag{1.24}$$

At this point the thermal pressure within the medium resulting from release of gravitational energy during collapse is no longer sufficient to oppose the cloud's own self-gravity and it collapses.

1.5 The Multi-Phase Interstellar Medium (ISM)

The interstellar medium (ISM) consists of a tenuous mixture of gas and dust which lies between the stars. It is predominantly gas ($\sim 99\%$) with the remaining 1% representing dust - grains of condensed gas and molecules which form complex fractal structures (Zeilik and Gregory, 1998).

As light from other galaxies outside of our own - the Milky Way, must pass through the intervening ISM, understanding the way in which it processes light is fundamental in ensuring that we do not misinterpret observations. As light travels from some distant emitting source, the intervening gas and dust preferentially absorbs and scatters light at the higher energy end of the continuum. As a result light that was originally emitted in the Ultra-Violet frequency band is absorbed and re-emitted at longer infra-red wavelengths.

This leads to a reddening of light due to dust extinction which must be accounted for when studying distant sources (see, e.g. Calzetti, 2001; Calzetti et al., 1994).

Absorption and emission spectra from the ISM can tell us a great deal about the local environment. The ISM provides the fuel for forming new stars and the rate at which it is accreted/blown out of galaxies, groups and clusters will directly affect their star formation rates. As stars eject metals formed during and at the end of their lives, the metal content of the ISM can tell us a lot about the previous star formation history. The return of metals from early generations of stars also leads to more rapid cooling, accelerating future star formation.

Modelling the ISM has proved however to be exceptionally difficult. The ISM is effectively a multi-phase medium that can be split into a warm/hot phase ($10^4 - 10^6$K) and a cold dense molecular cloud phase ($T \sim 10 - 100$K). The cooling of gas and its collapse into large dense molecular clouds signifies the beginning of star formation. These dense clouds shield the interior from the radiation from nearby stars and stellar winds that are driven as particles are pushed away under radiative pressure.

Molecular clouds can span to be a few parsecs across, containing between $10^4 - 10^6 M_\odot$ of metals and molecular hydrogen. The densities within these stellar nurseries typically range from $10^2 - 10^3$ n_H cm^{-3} but within the cores of the most massive giant molecular clouds, densities can reach 10^6 n_H cm^{-3} (see Ferrière, 2001). This vast dynamic range requires considerable resolution and precision to self-consistently track all phases at once.

The cores of molecular clouds become ever denser through accretion of mass until they become Jeans unstable. The exact mass at which molecular clouds fragment and form proto-stars is uncertain as physical effects other than the Jeans instability influence fragmentation and collapse. For instance there is much observational evidence for the existence of turbulence within the ISM. Observations of variations in the intensity of HI spectral features have revealed a scale invariant Kolmogorov power spectrum (see § 1.3.2) ranging from molecular cloud to galactic scales (Dutta et al., 2008, 2009a,b; Roy et al., 2008).

The effects of this turbulence upon the ISM are currently a point of contention between the academic community. Turbulence could increase the thermal pressure, resisting collapse and thereby increasing the Jeans mass. This would lead to an increase in the star formation efficiency as denser molecular clouds form (see, e.g. Elmegreen and Efremov, 1997; Elmegreen, 2002). In particular, Dutta et al. (2009b) has observed a correlation between the level of turbulence and star formation rate implying that a more turbu-

lent medium does indeed result in greater star formation. Alternatively turbulence could lead to increased fragmentation as it enhances density perturbations resulting in smaller effective Jeans masses (see Offner et al., 2008; Klessen, 2001; McKee and Ostriker, 2007).

The ISM is also highly magnetised with Heiles and Troland (2005) finding that Arecibo 21 centimetre absorption line data of the Milky Way ISM indicates median magnetic field strengths of $6.0 \pm 1.8\mu G$. Within the cores of galaxy clusters, magnetic field strengths of the intra-cluster medium (ICM) can escalate to several $10\mu G$ (McKee and Ostriker, 2007).

Our current theoretical understanding of the impact that such strong magnetic fields has upon the collapse of molecular clouds to form proto-stars is still limited. A series of theoretical studies have shown that due to the stretching and folding of magnetic field lines by turbulence, the energy stored in the magnetic field can be greatly amplified (see, e.g. Dubois et al., 2009; Zhang et al., 2009). Authors such as Stone and Zweibel (2009) and Hosking and Whitworth (2004) claim that gas is channelled along magnetic field lines, smoothing the flow and preventing it from fragmenting down to form stars. This can lead to magnetic fields regulating the level of turbulence. Conversely this stabilisation of the flow could prevent the formation of a single central over-density, instead allowing for the formation of multiple smaller stellar cores and accelerating the fragmentation process (see, e.g. Boss, 2009; Li et al., 2009; Boss, 2002). Ultimately the complex interplay of these different processes determines the initial mass function of stars (IMF) - the number of stars that are produced of a given mass (for example see Hennebelle and Chabrier, 2008). Hence these issues can directly influence many other observable galaxy properties.

To a large extent the current lack of understanding of the ISM is due to the inherent difficulty in modelling all of these physical interactions self-consistently over such a large dynamic range. Magneto-hydrodynamic simulations are extremely demanding and the exceptionally high resolution which is needed to resolve the different phases of the ISM means that most galaxy formation simulations lack the power to resolve ISM physics. To overcome this many codes implement sub-grid physical recipes which use simple analytic prescriptions to describe processes which occur below the resolution limit. However it is not yet clear whether identical sub-grid prescriptions behave identically in different hydrodynamic codes. Agertz et al. (2007) clearly highlights the differences in the ability of different hydrodynamic schemes to resolve the formation of fluid instabilities. In chapter 6 I give recent evidence that one branch of hydrodynamic codes, SPH, artificially suppresses turbulence leading to a difference in the temperatures and densities of gas within the cores

of simulated galaxy clusters. In light of this difference, in chapter 7 I investigate the effect that different hydrodynamic implementations have upon the properties of a model disk galaxy using identical sub-grid implementations.

1.5.1 Formation of Stars

When molecular clouds fragment, the release of massive amounts of gravitational energy combined with the extreme densities and pressures initiates hydrogen fusion via the Proton-Proton reaction chain (equations 1.25). First single hydrogen nuclei are fused to produce a Deuteron atom nucleus along with a positron and electron neutrino. The chain continues with the immediate annihilation of the positron with an electron to generate at least two photons. The combination of the Deuteron and an additional proton produces the helium nucleus, ${}^{3}_{2}He$, along with another photon.

$$\begin{aligned} {}^{1}_{1}H + {}^{1}_{1}H &\longrightarrow {}^{2}_{1}D + e^{+} + \nu_e + 0.42MeV \\ e^{-} + e^{+} &\longrightarrow 2\gamma + 1.02MeV \\ {}^{2}_{1}D + {}^{1}_{1}H &\longrightarrow {}^{3}_{2}He + \gamma + 5.49MeV \end{aligned} \tag{1.25}$$

For stars that are equally or less massive than the Sun, with core temperatures in the range of 10 to 14 mega Kelvin, the final reaction phase proceeds as

$${}^{3}_{2}He + {}^{3}_{2}He \longrightarrow {}^{4}_{2}He + 2{}^{1}_{1}H + 12.86MeV \tag{1.26}$$

There are additional reaction paths available to those that can reach higher temperatures. For instance, at temperatures in the range 14 to 23 mega Kelvin, the PPII branch becomes dominant with the energy production in the second stage being dependent upon whether ${}^{7}_{3}Li$ is in a ground or excited state (Zeilik and Gregory, 1998).

$$\begin{aligned} {}^{3}_{2}He + {}^{4}_{2}He &\longrightarrow {}^{7}_{4}Be + \gamma \\ {}^{7}_{4}Be + e^{-} &\longrightarrow {}^{7}_{3}Li + \nu_e + 0.861MeV/0.383MeV \\ {}^{7}_{3}Be + {}^{1}_{1}H &\longrightarrow 2{}^{4}_{2}He \end{aligned} \tag{1.27}$$

When the star has used up the hydrogen in the core for burning, fusion stops. This causes a radical drop in the energy output. Without thermal pressure to resist further collapse it once again contracts under its own gravity. This has a two fold effect, it increases the heat in and around the core, igniting a hydrogen burning shell around it. This then raises the core to a temperature sufficient to ignite the next step in the generation of higher mass elements - helium burning.

The ignition of helium burning boosts the star's energy output dramatically, causing the outer envelope to puff up out to much larger radii. This can cause disruption of the outer layers, shedding mass as a stellar wind. The increase in the star's surface area causes the surface temperature to drop even though the total energy output of the star has increased. This leads to the emission shifting to lower frequencies at the redder end of the spectrum, thus the star becomes a red giant. Through this mechanism young stars appear much bluer than the older redder stellar populations.

1.5.2 Creation of Metals Within Stars

When the helium fuel supply in the core is exhausted, the star repeats the process of collapse leading to the ignition of new reaction chains within the core; helium to carbon, carbon to neon, oxygen and finally silicon. Each stage increasing the number of shells burning around the core, giving the star an onion like construction. This continues up until silicon burns to form nickel, $^{56}_{28}Ni$ which decays to form cobalt, $^{56}_{27}Co$ which then decays to form iron, $^{56}_{26}Fe$. Iron, $^{56}_{26}Fe$, represents the most strongly bound of all nuclei with higher level fusion processes becoming endothermic and thus energetically unfavourable. In practice the half-life of $^{56}_{28}Ni$ is significantly longer than the remaining life-time of the massive star. Fusion thus stops once nickel is reached. Only the most massive stars can reach this stage, stars the size of the Sun stop at helium whilst smaller stars fizzle out after hydrogen burning stops.

As more massive stars generate higher temperatures and pressures through their gravitational contractions, they burn through their fuel sources faster than smaller stars even though there is more of it. This means that although the Sun can expect a lifetime of around 10 billion years (a significant fraction of the current age of the Universe) a star ten times more massive lives for a factor of a thousand times less.

Regardless of whether the star stops at hydrogen burning or continues right up to nickel, it must eventually stop. At this point depending upon the mass of the star, it either collapses gradually to form a white dwarf or undergoes cataclysmic core collapse supernova.

The threshold mass for core collapse supernova is determined by the Chandrasekhar mass limit (Chandrasekhar, 1931, 1935). This specifies the maximum mass of a star which supported by electron degeneracy pressure can support itself against gravity. As the Pauli exclusion principal requires that no two fermions can occupy the same quantum state at the same time, when electrons are forced close enough together that their quantum states

begin to coincide, one must be forced into a higher energy state. This requires an input of energy which acts like a resistive pressure to the further compression of matter. This acts in addition to the standard electrostatic repulsion of electrons.

Although gravitational collapse during the formation of a star can initially increase the density, squeezing out the empty space between atoms, eventually it reaches a point at which the electrons are compressed close enough that their quantum states begin to overlap. At this point a structure that is less massive than the Chandrasekhar mass limit can support itself indefinitely from gravitational collapse. The mass limit depends upon the relative proportion of electrons to nucleons (both protons and neutrons). In low mass stars that stop at Carbon fusion, the ratio is $\sim 1/2$ leading to a Chandrasekhar mass of $1.44\,M_\odot$ (Ferris, 1991; Mazzali et al., 2007).

For stars less massive than $8\,M_\odot$, the last death throws of the star will shed the outer layers, leaving it at a mass low enough for the core to support itself through electron degeneracy pressure. The remnant, a white dwarf, will continue to glow through the gradual emission of trapped thermal energy until the star eventually cools down and fades out of the observable spectrum.

1.5.3 Core Collapse Supernova

Stars greater than $8\,M_\odot$ complete the fusion cycles up to nickel within 30 million years and then terminate. As energy generation within the star's interior dramatically shuts off, a series of events that unravel in less than a second seal the star's fate and convert a cataclysmic implosion into an explosion that can be observed billions of light years away.

Prior to collapse, the fusion of silicon at the edge of the dense nickel core continues to increase its mass, oblivious to the lack of energy generation that now takes place within the core. Instead of fusion driven thermal pressure, electron degeneracy pressure is now the mechanism resisting core collapse. When the core's mass exceeds the Chandrasekhar mass limit, fuelled by the rapid burning of silicon to nickel, it can no longer support itself against gravity and it collapses down. The core at this point has a mass in the region of 1.2 to $1.5\,M_\odot$.

As the core collapses, gravitational potential energy is released, heating the core, resulting in the dissociation of a small fraction of the Iron nuclei. Although this increases the number of free nucleons and thereby the pressure they create, the energy required to overcome the binding energy comes from the electrons. In addition, the presence of free protons and electrons at such high densities drives "electron capture" whereby an electron

and proton combine to form a neutron and a neutrino. The neutrino escapes from the dying star taking with it some energy whilst the neutron increases the ratio of nucleons to electrons, reducing the electron degeneracy pressure and Chandrasekhar mass. As the pressure support comes predominantly from the electron degeneracy pressure, this simply acts to reduce overall pressure support and accelerate the implosion.

When the core reaches a density of 400 billion grams per cubic centimetre it becomes opaque to neutrinos, trapping them inside and preventing further energy escape. At this point the Chandrasekhar mass takes on a new meaning, from being the maximum mass which can support itself against gravity to the total mass which can collapse as a single homologous unit. The homologous collapse enforces the condition that the infall velocity is proportional to the radius, which as the density decreases with radius ensures that the sound speed (equation 1.18) decreases with increasing radius. This configuration allows sound waves travelling outwards away from the core to come to a stand still at the "sonic point," where the infall velocity equals the sound speed in the medium. At the sonic point sound waves begin to build up.

When the core is compressed to a density of 2.7×10^{14} g cm^{-3} nuclear pressure steps in and suddenly halts the collapse. Although the core now stops rapid contraction, its momentum as well as that of the material above it forces it beyond nuclear densities. Eventually the core bounces back and the stored energy is released as a sound wave that propagates outward to the sonic point. Once it arrives at the sonic point it combines with the sound waves generated by material outside of the core striking the interface, creating a massive pressure discontinuity. This pressure differential slows material falling into the core, creating a velocity discontinuity. It is this velocity discontinuity which constitutes a shock. Unlike sound waves, shocks can irreversibly change the medium they travel through, altering the density, pressure and entropy. But their most significant feature is that whilst sound waves are limited by definition to the sound speed in the medium, shock waves are not. This allows the shock to travel out past the sonic point and through the outer layers of the star.

After passing through 100 to 200 kilometres of stellar material the shock wave stalls due to the dissipation of energy as it dissociates nuclei, particularly those of Iron around the core. This presented a problem for astronomers until James R. Wilson of the Lawrence Livermore laboratory realised that the shock is later resurrected by the release of neutrinos from the core. During the collapse, neutrinos are imparted approximately half of the gravitational energy released during the collapse. As the core releases them, roughly one

in a thousand collides with nuclei in the outer regions of the star, leading to dissociation and an increase in pressure. It is this that refreshes the shock wave and disrupts the remains of the dying star, casting off the majority of the mass into the depths of space. Any remaining material that is still gravitationally bound, falls back onto the remnant. What is left behind is visible as a gaseous nebula enriched with the products of the star's lifetime of fusion reactions, within which lies either a neutron star or black hole depending upon the star's initial mass (Heger et al., 2003).

1.5.4 Type Ia Supernova

Although stars less massive than $8M_{\odot}$ cannot undergo core-collapse supernova, some are still able to go supernova through the accretion of material from a binary partner. The binary consists of two main sequence stars, the primary star more massive than its secondary partner. The more massive primary quickly evolves off the main sequence, into a red giant until it sheds off its outer layers and reverts to a white dwarf, its collapse prevented purely by electron degeneracy pressure. The secondary star eventually follows, evolving off the main sequence to become a red giant. The loss of mass by both stars through stellar winds helps remove angular momentum from the system, drawing the two bodies closer together until material from the red giant begins to fill the Roche Lobe. Accretion of mass from the Roche Lobe by the white dwarf steadily increases its mass, adding a layer of hydrogen to its otherwise carbon and oxygen dominated surface. Whereas increased burning in a conventional star leads to stellar expansion of the outer layers, permitting the increased energy to be radiated, white dwarfs are too massive and tightly bound to expand. As electron degeneracy pressure is independent of temperature, the white dwarf is unable to regulate fusion processes that occur within its core, leading to a runaway reaction. As its Carbon core ignites, the shock wave spreads to the surface where due to gravitational confinement, it burns rapidly across the face of the star in a massive deflagration wave. When the wave reaches the opposite side of the star to that which it broke through, it collides with itself, projecting material both into space and back into the core (Jordan et al., 2007). This causes complete disruption of the star, releasing as much energy in a few months as the Sun emits in a billion years (Ferris, 1991).

As all white dwarfs have a peak mass defined by the Chandrasekhar mass limit and are constrained to ignite carbon deflagration at the same mass, they all create supernova with similar luminosities. In addition, radioactive decay of nickel (^{56}Ni) produced in the supernova shock, to give cobalt (^{56}Co) and then iron (^{56}Fe) causes their emission to have

a characteristic gradual decline over time, allowing for easier light curve calibration. This has allowed studies of type Ia supernova light curves to be used as "standard candles" in distance and recession velocity determinations. Through the work of authors like Riess et al. (1998) and Perlmutter et al. (1999) the pioneering work of Edwin Hubble proving the expansion of the Universe, has been enhanced to determine Omega matter (the mass density contrast) and Lambda (the cosmological energy constant), providing evidence for the existence of dark energy.

1.5.5 How Supernova Shock Waves Enrich the ISM

Once the shock wave escapes from the supernova's surface, casting off the outer envelope, it expands adiabatically. This continues until radiative losses become important at which point it switches to a momentum conserving "snow plough" phase, decelerating as it sweeps up material. Where several supernova bubbles converge, they can form giant superbubbles between 100 to 200 parsecs across. Providing this supernova driven wind travels faster than the galaxy's escape velocity (equation 1.2), material may be able to leave the galaxy. The ejection of gas available for star formation can restrict the star formation rate and for more massive galaxies, this material can rain back in on the galaxy, stimulating late bursts of star formation. The efficiency with which supernova are able to regulate the flow of gas available for star formation is currently a hot topic in astronomy and may provide an ideal means by which to control so called cooling flows and the over-production of stars which occurs without it (White and Frenk, 1991).

Shock waves from supernova are also an important site for the nucleosynthesis of the heaviest elements in the Universe (i.e. all those beyond Iron). Although shortly after the Big Bang, there was a brief period of time whereby the densities and pressures were optimum for fusion of hydrogen to produce deuterium, helium and trace amounts of lithium, rapid expansion would have terminated nucleosynthesis before higher mass elements were produced. Therefore the observed abundances of high mass elements must have come from supernova. The primordial abundances predicted by the Big Bang cosmological model are in exceptional agreement with those observed, something which is considered one of the great triumphs of modern astronomy.

The presence of these light primordial elements but initial absence of any other metals means that the cooling of gas to form the first stellar populations (population III) would have been relatively slow. This in turn would have allowed for the formation of massive stars, far greater than those left today. These would have ended their short lives

in spectacular core collapse supernova, enriching the ISM from which the next generation of stars would be formed (Population II). This helps explain why although the oldest observable stars today are extremely metal poor they still contain some metal content above that expected from Big Bang nucleosynthesis. More recent stellar populations, including the latest population I, are much smaller than their ancestors, arising from the more efficient metal cooling allowing smaller mass stars to form quicker. In addition, the presence of metals can catalyse the fusion of hydrogen through the CNO cycle (equations 1.28), making younger stars burn through their fuel faster. This mechanism which uses carbon, nitrogen and oxygen, is responsible for the extremely high luminosities of the main-sequence young and most massive blue supergiants.

$$\begin{aligned}
&{}^{1}_{1}H + {}^{12}_{6}C \longrightarrow {}^{13}_{7}N + \gamma \\
&{}^{13}_{7}N \longrightarrow {}^{13}_{6}C + e^{+} + \nu_e \\
&{}^{1}_{1}H + {}^{13}_{6}C \longrightarrow {}^{14}_{7}N + \gamma \\
&{}^{1}_{1}H + {}^{14}_{7}N \longrightarrow {}^{15}_{8}O + \gamma \\
&{}^{15}_{8}O \longrightarrow {}^{15}_{7}N + e^{+} + \nu_e \\
&{}^{1}_{1}H + {}^{15}_{7}N \longrightarrow {}^{12}_{6}C + {}^{4}_{2}He
\end{aligned} \tag{1.28}$$

Observationally supernova offer a wealth of information on star forming environments and other energetic phenomena through the metal absorption and emission peaks that appear in spectra.

1.6 Observed Properties of Galaxies

Although each galaxy is as individual as a snow flake, they can be drawn into generalised categories using Hubble's tuning fork classification scheme (figure 1.5). Hubble believed that all galaxies started out as spheroidal galaxies and that as they evolved, they became more elliptical, unravelling to display dazzling spiral arms. Because of this, those at the left of the Hubble sequence are referred to as "early-type galaxies" whilst those to the right are "late type galaxies". Since observations showed that spiral galaxies existed both with and without a bar in the central region, Hubble classed this as two separate states that the galaxy could potentially evolve into, giving his evolutionary tree a tuning fork appearance.

Galaxies typically consist of between tens of millions (10^7) to trillions of stars (10^{12}) spread across galaxies that are between 100 to 100,000 parsecs across. Most galaxies are

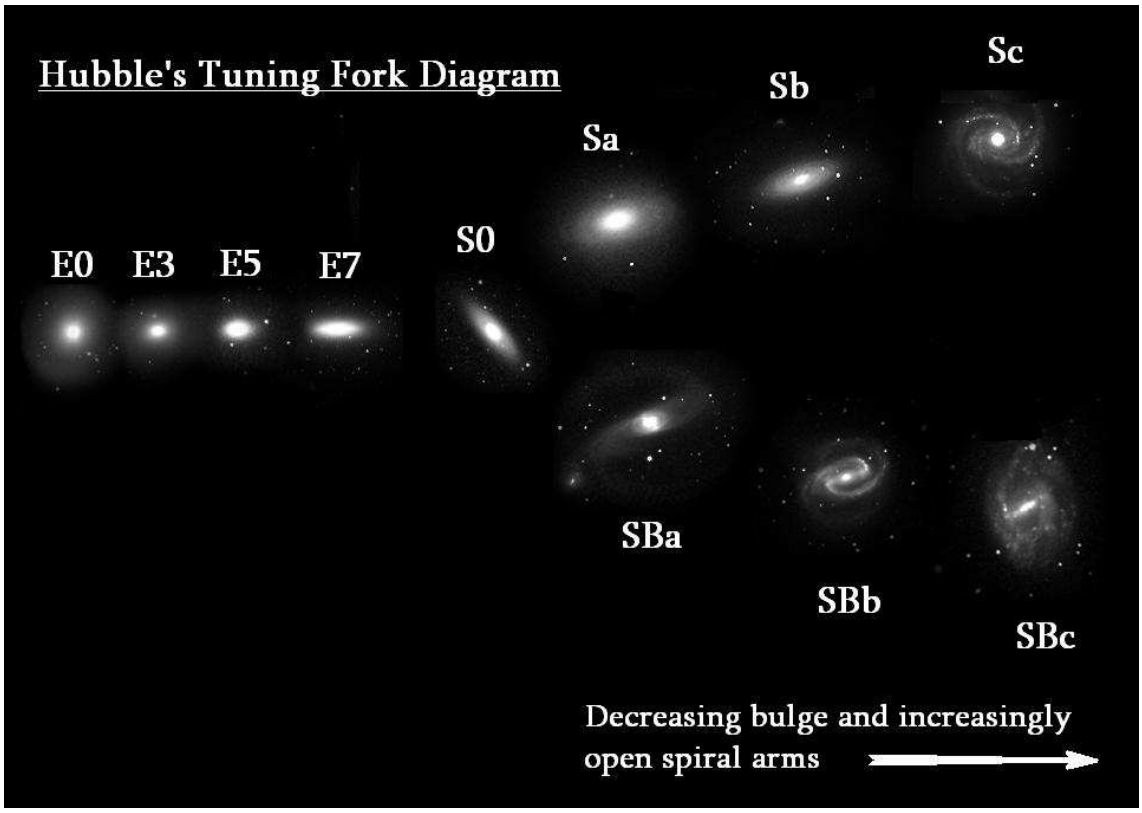

Figure 1.5: Hubble's famous tuning fork diagram displaying the varying elliptical galaxy classes on the left with the two spiral classes on the right. The spirals are subdivided depending upon whether they posses a bar in the centre, how tightly the spiral arms are wound and how massive the central bulge is. Hubble incorrectly believed that galaxies evolved along the tuning fork from left to right. In addition to those shown, there is a separate class for irregular galaxies which fail to fit into the classes shown. Galaxy images courtesy of the Sloan Digital Sky Survey.

then separated by millions of parsecs. It is estimated that there are more than 125 billion galaxies in the observable Universe (Rubin, 1983) with around 77% of those observed found to be spirals, 20% elliptical and 3% irregulars (Zeilik and Gregory, 1998). These values however are subject to observational biases, as spirals represent the most luminous galaxies and are observable out to great distances.

Massive galaxy redshift surveys such as the 2MASS galaxy redshift catalogue (Skrutskie et al., 2006), the Sloan Digital Sky Survey (Abazajian et al., 2005), the 2-Degree Field Galaxy Redshift Survey (Colless, 1999) and CfA redshift survey (Falco et al., 1999) have confirmed earlier work by authors such as Abell (1965) that galaxies cluster on large

scales into discernible web like structures interconnected through a series of filaments up to around 20 mega-parsecs long (see, e.g. Doroshkevich et al., 2004). Such large scale structure has formed a key observable with which to develop and test cosmological theory.

As the evolutionary timescales over which galaxies form is vast, the only way to piece together galaxy formation is by using large surveys to obtain sufficiently large statistical samples of the galaxy populations during the different epochs.

1.6.1 The Milky Way & Properties of Spiral Galaxies

Orbiting our Sun, lying around 8.5 kpc from the galactic centre we can observe the great disk of our own galaxy - the Milky Way, as described by the ancient Greek, Claude Ptolemy (90-168 AD), "a zone as white as milk" which runs as a band across a clear night's sky. Shapley was able to determine the location of the galactic centre based upon the roughly spherical distribution of globular clusters - compact groups of around $10^5 - 10^7$ stars. He assumed correctly that these would naturally orbit around the galactic centre (see, e.g., Shapley, 1939).

We now know that the Milky Way is a spiral type galaxy and consists of a thin disk of stars and gas with radius $\sim 25 - 30$ kpc and effective thickness $\sim 400 - 600$ pc. An older stellar population occupies a spherical stellar bulge of radius $\sim 2 - 3$ kpc around the galactic centre and an extended gaseous halo stretches out to more than 30 kpc (Ferrière, 2001). Stars within the disk have been found to follow roughly circular orbits, which at the Sun's radius gives a velocity of $v_\odot \approx 220\,\mathrm{kms}^{-1}$ (Kerr and Lynden-Bell, 1986) implying a rotational period of around 240 million years. The total mass of the Milky Way has been estimated at $7 \times 10^{11}\,\mathrm{M}_\odot$. The spiral arms themselves are sites for continual star formation and as a result often appear very blue.

Direct observations of the orbits of stars around the galactic centre have led to evidence for the existence of a super-massive black hole of order $4 \times 10^6\,\mathrm{M}_\odot$ (Ghez et al., 2008, 2005, 1998) located within a dusty torus. Hubble space telescope maps of Paschen-α emission indicate strong magnetic fields from the presence of fine linear gas filaments channelled around the field lines and diffuse emission tangential to the torus suggests an outflow from the region (Wang et al., 2010). The galactic centre is also found to be a region of strong star formation with many gaseous "planetary nebulae" being formed by gas thrown off in strong stellar winds.

There now exists strong evidence to suggest that most massive galaxies have super-massive black holes at their centres with a strong correlation between the black hole mass

and the bulge luminosity (Kormendy and Richstone, 1995; Marconi and Hunt, 2003), mass (Häring and Rix, 2004) and velocity dispersion (Gebhardt et al., 2000; Ferrarese and Merritt, 2000). Spirals can be broken into two categories; those with and without a central bar. The majority are found to posses a bar structure which blends smoothly into the spiral arms and is believed to funnel gas into the core, fuelling growth of the black hole at the centre. In some cases this can result in the formation of jets of highly relativistic particles expelled from the poles, tangential to the galactic disk as large amounts of gravitational potential energy is released by accreting gas (Zeilik and Gregory, 1998). Galaxies that possess such Active Galactic Nuclei (AGN) are often amongst the most luminous persistent sources in the Universe and they are believed to be the origin of the extremely luminous, very distant Quasi-stellar radio sources (quasars). At least nineteen quasars have been confirmed at redshifts greater than $z = 6$, emitting the same energy as around ten trillion suns - more than a thousand times the total energy output of the Milky Way (Willott et al., 2010, 2009, 2007).

Morphologically, spiral galaxies are found to undergo an increase in the size of the galactic bulge in terms of its radius and mass as they evolve from early type to late type galaxies. This is also accompanied by a reduction in the surface-brightness and luminosity of the bulge. The disk however tends to maintain a more consistent profile as we pass between the different galaxy types suggesting that the bulge may form separately to the disk (Oohama et al., 2009). In this way the processes which form the bulge may resemble those that form ellipticals more than those of the disk.

1.6.2 Elliptical Galaxies

The key properties of elliptical galaxies are their lack of gas and significant star formation (see, e.g., Binney and Tremaine, 1987). Ellipticals are classified according to their ellipticity, with E0 being almost spherical whilst E7 are highly elongated. Referred to as early-types based upon Hubble's hypothesised evolutionary sequence they include the largest observed galaxies. With their smooth density profiles and no disk component they contain very old and red stellar populations. Most of these stars are found to already be in place by early times between redshifts two and one, with around $\sim 80\%$ of all stellar mass in place by redshift $z = 1$ (see, e.g., López-Corredoira, 2010; Cimatti et al., 2008; Arnouts et al., 2007; Abraham et al., 2007; Daddi et al., 2005)

Their surface brightness is found to die off smoothly as a function of radius until they blend into the sky background. This often makes determining their true sizes precariously

difficult but measurements of the radius within which half their luminosity is contained gives values in the region of 20kpc for giants such as M87 to 0.2kpc for the dwarf M32 galaxy (Binney and Tremaine, 1987).

Unusually most large highly luminous ellipticals show no strong signs of rotation which is normally present in most other self-gravitating masses and in disk galaxies whilst smaller less luminous ellipticals often do show some signs of rotation (see, e.g. Lee et al., 2010). Bulges in disk galaxies are often referred to collectively as "spheroids" although their shapes may not be spherical.

Like disk galaxies, ellipticals can also be subject to periods of AGN feedback which can drive morphological changes as gas is heated and expelled from the core, disrupting the galaxy's density profile (see, e.g., Diehl and Statler, 2008).

1.6.3 Lenticular Galaxies

Lenticulars or S0 galaxies according to the Hubble sequence, share many of the properties of spiral galaxies with the presence of a rotating disk, bulge and stellar halo. Though like elliptical galaxies they are extremely gas poor. The lack of gas leads to very low star formation rates and as a result an absence of spiral arms. As they are found in abundance in cluster environments where they can represent up to half of the galaxies present, it suggests that they may be spiral galaxies which have fallen in and had their gas removed through ram pressure stripping or tidal processes within the cluster (van Gorkom, 2004).

1.6.4 Dwarf Galaxies

Despite the prominence of very large spiral and elliptical galaxies, the majority of galaxies actually fit into the dwarf galaxy category; galaxies about one hundredth the size of the Milky Way and containing only a few billion stars. Some ultra-compact dwarf galaxies have been found that are only ~ 100 pc across.

Although recent observations have made considerable headway in detecting the satellite dwarf galaxies around the Milky Way, they are few in number and present a considerable challenge with current observational equipment (see, e.g., Mateo, 1998; Willman et al., 2005a,b; Zucker et al., 2006b,a; Belokurov et al., 2010).

Dwarf galaxies are found to be extremely gas and metal poor, $-2.3 \lesssim [Fe/H] \lesssim -1.5$ (see, e.g. Belokurov et al., 2010; Irwin et al., 2007) and possess no recent population I stars brought about through the lack of gas with which to fuel star formation.

Observations of the dynamics of the star clusters within dwarf galaxies such as the Small and Large Magellanic Clouds have indicated that there must exist modest amounts of non-baryonic dark matter although less than would typically be found in most larger galaxies. Velocity dispersions show that the galaxies possess a high degree of rotation; $17 \pm 2\,\mathrm{kms}^{-1}$ for NGC 147 and $15 \pm 5\,\mathrm{kms}^{-1}$ for NGC 185 and also indicate a slight flattening of the profile at larger radii at what could potentially hint at the remnants of a stellar disk (Geha et al., 2010).

1.6.5 Galaxy Groups and Clusters

Just as stars cluster into globular clusters so too do galaxies into groups and clusters. The largest clusters contain many thousands of galaxies and have total masses including dark matter of between $10^{14} - 10^{15}\,\mathrm{M}_{\odot}$. They are the largest bound structures in the Universe and can even influence the local Hubble flow with a dipole being detected in the WMAP data indicating a global motion of the local group in which the Milky Way resides, towards the "Great Attractor." This colossal agglomeration of galaxy clusters including the Norma supercluster, leads to a $600\,\mathrm{kms}^{-1}$ motion towards the Hydra constellation.

As these structures have such deep gravitational potential wells, they tend to keep hold of their gas with any winds and AGN activity ultimately only being able to expel material from the core for a limited period of time before it falls back in. As such they can be treated in many respects as closed boxes retaining information of prior galactic processes (Voit, 2005).

There is a marked increase in the fraction of elliptical galaxies and a reduction of disk galaxies as we tend towards more densely clustered galaxies. This is combined with an increase in the luminosity of the spheroidal component with increasing density (see, e.g., Dressler, 1980; Andreon, 1998). The increase in S0 and reduction in spiral type galaxies is found to be most pronounced in galaxy clusters where the galaxy velocity dispersion is lower ($\sigma < 750\,\mathrm{kms}^{-1}$) and at lower redshift ($z < 0.5$) implying that the population of S0 galaxies may arise through galaxy-galaxy interactions which are more efficient in these regimes (Just et al., 2010). Isolated spirals in the field can constitute up to 60% of all luminous galaxies but this can drop to less than 10% in dense cluster environments.

In large clusters the most luminous galaxy often displays unique properties. They are often static and sit at the centre of the cluster's gravitational potential well, are several times more luminous than other galaxies within the cluster and have a dim stellar halo that extends up to a megaparsec from their core. These galaxies are termed the Brightest

Cluster Galaxies or BCG (Binney and Tremaine, 1987).

The presence of large extended HI tails trailing behind galaxies that have passed through cluster environments in the Virgo, Centaurus and Coma clusters suggests that galaxies within the cluster may be subject to extensive ram pressure stripping forces, an effect which may strongly determine the morphology of cluster members (Koopmann et al., 2008; Chung et al., 2007; Higdon et al., 2006).

Many galaxy clusters have been found to display very high X-ray surface brightness within the core regions, a strong indication of hot gas reservoirs cooling through Bremsstrahlung emission. As the cooling time is much shorter than the Hubble time and less than 300 Myr in the cores, this allows gas to cool and flow inwards as the pressure drops. This formation of a sub-sonic mass inflow of cooling gas is referred to as a cooling flow (see, e.g., Fabian and Nulsen, 1977; Fabian, 1994). As the gas cooling rate is a function of density squared, the inflow of mass should further accelerate the cooling process leading to a runaway cooling effect. This would lead to the creation of large cold gas reservoirs with which to form stars however the star formation rates are found to be negligible compared to the mass inflow rates (O'Connell and McNamara, 1989; Johnstone et al., 1987).

Additionally observational studies have shown that clusters can be split into those with "cool" cores and those with "non-cool" cores. Cool core clusters are found to possess much cooler, denser cores and have peaked metal distributions within the core. Non-cool core clusters on the other hand have much hotter, less dense cores with a more uniform metal distribution throughout the cluster (see, e.g. Fukazawa et al., 2000; De Grandi and Molendi, 2001; Matsushita et al., 2002; Schmidt et al., 2002; Churazov et al., 2003; De Grandi et al., 2004; Tamura et al., 2004). These differences are often defined using the concept of entropy. Material that is cool and dense possesses much lower entropy than that with higher temperatures and lower densities. Entropy is in essence a measure of the probability of a state and the potential for disorder that it shows. Higher entropy systems are those that display the most chaotic behaviour. The second law of thermodynamics states that the entropy of any given isolated system will tend to increase. Decreases in the overall entropy of the system are forbidden. The physical origin of the differences between these two cluster types is currently unknown although there are a number of competing theories which I cover in § 1.8.3 and § 1.8.4.

1.7 Computer Modelling

To date there exist a variety of simple problems with direct relevance to galaxy formation for which elegant analytical solutions have been derived, the accretion of gas onto a point mass perturbation (Bertschinger, 1985) being a prime example. However even these ground breaking works lack many of the fundamental physical processes which we observe to be important for galaxy formation. It is generally accepted that true galaxy formation is mediated by a complex interplay of gravitational and non-gravitational heating and cooling processes on an enormous range of scales. The shock heating of gas as it undergoes collapse and the ram pressure stripping of smaller sub-structures as they fall into larger haloes are strongly driven by gravity. Alternatively, star formation and supernova present mechanisms in which nuclear processes can inject energy into the surrounding environment, driving further shock waves through the ISM. Such stellar feedback enriches the ISM with higher mass elements known collectively as 'metals,' which can accelerate its cooling, leading to further collapse and star formation. In this way atomic and molecular processes can also strongly contribute. Analytic works have attempted to extend the work of Bertschinger to include the effects of cooling usually through the use of simple power law cooling curves (see Bertschinger, 1989; Abadi et al., 2000) but solutions which self-consistently describe the interaction of all these forces have yet to be found. This comes about in part by their highly non-linear evolution and complex interdependence.

Modern astronomy has been able to overcome this impediment by implementing each of the necessary physical processes into numerical codes which evolve the system from some initial given state, in a series of smaller steps. This often involves applying the effects of each physical process separately, one after the other in a process known as operator splitting and has the advantage of permitting different physical mechanisms to be decoupled from each other. Using this technique, numerical simulations have come to form a firm test bed with which to compare current theory to observations. Through simulations, astronomers have been able to observe various phenomena that normally play out over the course of millions to billions of years instead over more human timescales, such as the formation of the large scale structure of the universe or the swallowing of stellar material by super-massive black holes. These insightful wonders have opened the eyes of many to the magnificence of the Universe and the joys of physics. As such they are of fundamental importance to modern astronomy and have led to our current theoretical understanding which I outline in § 1.8.

1.7.1 Concept of Order and Consistency

A measure of the consistency of a numerical scheme is often measured in terms of its "order." Assuming that a numerical technique makes no initial errors in following the behaviour of the system and thus it initially agrees with the system it is tracing, the order is a measure of the error made in the subsequent step.

For a step size h and a difference δ^h_{n+k} between the analytic solution and the result obtained using the numerical technique over k steps past the n^{th} step, consistency is obtained if

$$\lim_{h\to 0} \frac{\delta^h_{n+k}}{h} = 0 \tag{1.29}$$

i.e. the system is said to converge upon a single solution that matches the analytical result. Even if the analytical result is not known, convergence upon a fixed solution when using progressively smaller step sizes is usually used as a good test of the performance of a given code. A robust code will converge whilst a code which fails convergence may lack the necessary resolution or may be numerically unstable.

The scheme is said to have order, p, if

$$\delta^h_{n+k} = O(h^{p+1}) \quad as\ h \to 0 \tag{1.30}$$

As a consistent solution is one that converges upon the analytical result as the step size progressively decreases, it must have order greater than zero to converge. An order of zero would imply that the deviation remains constant as the step size decreases and thus the scheme is inconsistent with the analytic solution. Increasing orders result in a more rapidly declining error as the step size tends to zero.

1.7.2 N-Body Simulations

N-body simulations were amongst some of the earliest cosmological simulations to be performed, allowing the motion of a series of self-gravitating particles to be followed over time (see, e.g. Aarseth, 1985; Davis et al., 1985; Efstathiou et al., 1985). As these systems are governed purely by Newton's law of gravity they proved to be relatively simple to code up (see § 1.7.3 for the equations of motion) and initial simulations featured between a thousand and 32^3 particles. Since then the scale of N-body codes has massively accelerated, part driven by the advances in hardware capabilities and partly due to improvements in gravitational algorithms (see chapters 2.5 and 3.3).

As the backbone upon which merger trees and halo mass functions for many semi-analytic models are built, the Millenium N-body simulation has revolutionised the way

astronomers work (Springel et al., 2005b). Offering an unprecedented cosmological sample of many hundreds of thousands of galaxy groups and clusters, it has allowed the evolution of large scale structure according to a ΛCDM cosmogony to be realised. At 2.23 billion light year across the 500 h^{-1} Mpc^3 box contained 2160^3 dark matter particles allowing the detailed formation of dark matter structure to be followed from $z = 127$ to present day. In addition the outputs are often used by simulators to carefully select initial conditions that will result in cosmic structures in line with their experimental needs.

The Aquarius project aims to take the results from the Millenium to even higher resolution (Springel et al., 2008). By using the same cosmological parameters as the Millenium simulation but now enclosing a much smaller volume into which a Milky Way like halo will form, the aim is to be able to provide much greater insight into the rich sub-structures that should be present in our galaxy. In this way we can determine the number of dwarf galaxies that should exist based on the numbers of bound sub-structures. If there exists a discrepancy between the observed number of galaxies of a given mass and the number with that corresponding dark matter halo mass in the N-body simulations then it may highlight the need to include more complex gas physics. Such gas physics can be incorporated at a relatively modest computational cost using semi-analytic models with the N-body simulation's dark matter results providing the necessary framework (see § 1.7.4). Another competing group have launched the Via Lactea INCITE simulation which also simulates a galaxy sized halo with one billion particles but using different initial conditions to the Millenium run (Kuhlen et al., 2008). This offers a unique opportunity to contrast and validate results.

Thanks to the success of these massive N-body simulations, a firm theoretical understanding of the assembly of collisionless dark matter is now established allowing the more complex task of understanding the baryonic component to be approached. As dark matter has yet to be directly detected, baryonic physics remains the best means for understanding the cosmos, however unlike dark matter which interacts purely through gravity, the hydrodynamic interactions are far more complex.

1.7.3 Time Integration and the Leapfrog Method

When evolving a given particle, we could take the initial position relative to the rest of the particles to calculate the force and thus acceleration, $\mathbf{a}$, acting upon it using equation 1.1. For a given step size, Δt, velocity, $\mathbf{v}_n$, and assuming the acceleration is constant we can

calculate its new position as

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \mathbf{v}_n \Delta t \tag{1.31}$$

and then update the velocity to give

$$\mathbf{v}_{n+1} = \mathbf{v}_n + \mathbf{a}_n \Delta t \tag{1.32}$$

At this point the new gravitational acceleration is computed and the process is repeated. As this Euler integration scheme takes no account of the change in acceleration as the particles move during the timestep, it is only accurate to first order. To increase its accuracy in space and time it is necessary to offset the velocities by half a timestep from the positions and accelerations. As this uses positions and velocities that are interleaved in time, they effectively "leapfrog" over each other as they are updated;

$$\begin{aligned} \mathbf{x}_{n+1} &= \mathbf{x}_n + \mathbf{v}_{n+1/2} \Delta t \\ \mathbf{v}_{n+1/2} &= \mathbf{v}_{n-1/2} + \mathbf{a}_n \Delta t \end{aligned} \tag{1.33}$$

The off-setting of positional and velocity components allows the leapfrog method to regain accuracy to second order in space and time (see Press et al., 1992) making it vastly more appealing than the simple Euler scheme.

1.7.4 Semi-Analytic Models

Given the wealth of information from massive N-body simulations in terms of structure formation and the assembly histories of mass, it is possible to construct predictions for the baryonic properties of galaxy populations using semi-analytic modelling techniques. Semi-analytic models, using the merger trees and halo mass functions of dark matter only runs which describe the way in which dark matter builds up into structures, assume that gas traces the dark matter and apply simplified prescriptions to describe the gas physics. In this way both quiescent processes such as star formation, gas cooling and enrichment of the ISM by supernova can be modelled along with ram pressure stripping, merger induced heating, star bursts and periods of AGN activity (see, e.g., Baugh, 2006). Their high modularity allows for easy addition of new physics or the improvement of existing routines. The most powerful feature of semi-analytic models is their ability to provide results for a cosmologically representative sample of the Universe at an extremely modest cost relative to a full hydrodynamic simulation. In fact they can be run many times over in the time needed to run a single direct hydrodynamic simulation. This permits a much

wider parameter space to be systematically explored, allowing semi-analytic models to play a key role in furthering our understanding.

Physical recipes implemented by semi-analytic models often feature a series of adjustable "free" parameters which can be varied to fit observational results when the underlying physics is poorly understood. A good example is that of AGN feedback in which direct simulations are hampered by the extremely high spatial and temporal resolution needed to model a complete jet in situ with an active cosmological cluster. This leaves the manor in which a jet couples to the ICM uncertain. To tackle this, authors such as Croton et al. (2006) and Bower et al. (2006) have developed semi-analytic models that simulate the probable effects of AGN but permit the varying of unknown parameters. The Bower et al. (2008) model ejects a fraction of the central X-ray emitting gas from within the cluster core when AGN feedback becomes active and adds it to the halo gas. To determine when the AGN becomes active they add the variable parameter α_{cool} which specifies a ratio between the free fall and cooling times. This determines the point at which the halo is quasi-hydrostatically cooling, a condition they believe is necessary for the radiatively inefficient accretion phase to form and fuel a jet. Other parameters include the efficiency of star formation and the time scale upon which ejected gas cools to become available again for star formation. These parameters are varied until the model matches observational constraints such as the break in the luminosity function - a basic property of all galaxies and clusters that describes the relative abundances of galaxies with different luminosities. The resulting best fit parameters can then provide us with some insight into the workings of these physical processes and help us tailor full hydrodynamic simulations to use the most appropriate values.

Despite their advantages, semi-analytic models still need to be calibrated through the use of direct hydrodynamic simulations. Although a new analytic prescription may give reasonable agreement with observation, it does not guarantee that it is physically justified. Because of this, semi-analytics and full hydrodynamic simulations are seen as complimentary techniques that advance hand-in-hand. A prime example being the calibration of ram pressure stripping in the Durham Galform model (Font et al., 2008) using suites of full hydrodynamic simulations by McCarthy et al. (2008b, see chapter 5).

1.7.5 Full Hydrodynamic Simulations

Thanks to the success of massive N-body simulations, a firm theoretical understanding of the mass assembly of dark matter is now established allowing the emphasis to switch

towards understanding the more complex baryonic component. As dark matter has yet to be directly detected, baryonic physics remains the best means for understanding the cosmos. Unlike collisionless dark matter which can be dealt with using simpler N-body techniques, the modelling of baryonic physics presents a far greater challenge. In addition to the need to model the evolving hydrodynamic properties of one or more gas phases by solving the hyperbolic differential equations for compressible fluid dynamics (equations 1.15, 1.16 & 1.17), it is necessary to include the effects of other physical mechanisms such as radiative cooling, star formation and supernova feedback.

There exist two distinct main categories for approaching the modelling of the fluid flow; Smooth Particle Hydrodynamics (SPH) which models the continuous fluid as a series of discrete particles of a given mass that advect with the fluid flow, and Eulerian mesh codes which map the properties of the fluid to a discrete mesh and solve for the fluxes through the cell walls. I will discuss these two different implementations in greater detail in chapter 2.

Unlike semi-analytic codes, full hydrodynamic simulations do not require pre-compiled halo merger trees to describe the mass assembly over time. Instead they incorporate (often fewer) physical processes that are all self-consistently evolved together in time. The effects of each one being able to influence the outcome of all others. This allows full hydrodynamic simulations to give us a much greater insight into the coupling of different physical mechanisms which can often be lost in semi-analytic techniques.

As full hydrodynamic simulations are much more demanding in terms of computation and memory consumption, it is often necessary to limit the resolution range and the size of the volume being simulated. As many processes such as star formation and supernova feedback in cosmological simulations take place over scales below the finest level of resolution, it is necessary to model them using sub-grid prescriptions in a similar fashion to semi-analytic models. For instance my sub-grid implementation for star formation based upon the recipe of Schaye and Dalla Vecchia (2008) is calibrated using empirical observation (see chapter 7). Other processes such as radiative cooling (see § 1.3.3) for which accurate cooling rates can be determined for given densities, temperatures and compositions but at great computational expense, often use pre-tabulated look-up tables which speed up the evolution of the cosmological simulation.

Current state-of-the-art full hydrodynamic simulations include the Galaxies-Intergalactic Medium Interaction Calculation project (*GIMIC*, see Crain et al., 2009) which features all of the fundamental gas physics for galaxy formation; metal dependent cooling in the

presence of an ionising UV background, star formation and supernova feedback. By carefully selecting five different regions with mean over-densities of -2, -1, 0, $+1$ and $+2\sigma$ from the *Millenium* simulation (Springel et al., 2005b), all regions of scientific interest are covered from the most massive clusters to the deepest voids. Each spherical region has a radius between 18 and 25 h^{-1} Mpc ensuring that high spatial and mass resolution can be achieved whilst the use of the *Millenium* simulation for initial conditions also ensures that the time dependent boundary conditions are known. As each of the *GIMIC* simulations present a considerable investment in computing time, they only use one set of parameters, concentrating more on the environmental effects of various physical phenomena. They therefore lack the ability to determine the effects of varying features such as the IMF or stellar feedback. To compliment *GIMIC* the Over Whelmingly Large Simulations project (*OWLS*, see Schaye et al., 2010) presents a suite of over fifty large cosmological simulations in which the effects of differing the implemented sub-grid physics is investigated.

The *Horizon-Mare Nostrum* project offers a complimentary approach to that of the *GIMIC* and *OWLS* simulations. Whereas *GIMIC* and *OWLS* are run in an SPH code, the *Horizon-Mare Nostrum* project adopts the AMR framework. As these two codes differ fundamentally in their construction, the way in which sub-grid physics is implemented will too. More importantly, recent works have highlighted fundamental differences in the performance of the actual hydrodynamic implementations. The first cosmological comparison of the formation of a cluster by Frenk et al. (1999) revealed that different cluster core properties were obtained using different hydrodynamic implementations. Agertz et al. (2007) followed on from this by demonstrating that SPH codes artificially suppress the formation of Kelvin-Helmholtz and Rayleigh-Taylor instabilities. AMR codes on the other hand are able to resolve these features. These are important for effects such as ram pressure stripping and the mixing of metals generated through star formation. Understanding these differences between the various hydrodynamic implementations is fundamental to our understanding of results from large cosmological simulations and as such is the key motivation for the work presented in this thesis.

1.8 Theoretical Understanding of the Universe

The current paradigm depicts a young homogeneous Universe around thirteen and a half billion years ago into which small quantum scale fluctuations naturally arise. As inflation in the early Universe sets in, these small fluctuations are blown up to macroscopic scales

from which they begin to grow, accreting mass from their surroundings. As these small structures develop, they attract each other through their self-gravity and merge to form larger structures. As these continue to merge with ever larger structures in a hierarchical mode of formation, close passages by other bodies can impart angular momentum and set them spinning (see, e.g. White and Frenk, 1991).

As dark matter structures begin to collapse, baryonic gas is drawn into their deep potential wells. As this gas falls in, it is squeezed to higher densities at which point it can shock, converting its gravitational and kinetic energy of motion into thermal energy. In order for the gas to continue to collapse into the dark matter potential well, it must radiate away this thermal energy, usually through the emission of Bremsstrahlung radiation.

This view of hierarchical structure formation has been supported by a wealth of observational evidence with the most significant being provided by the detection of the Cosmic Microwave Background (CMB). The CMB arises from the emission of electromagnetic radiation at the last scattering surface when radiation and matter first decoupled. The serendipitous detection of the CMB by Arno Penzias and Robert Woodrow Wilson of Bell Laboratories in 1965 has been one of the greatest achievements of modern cosmology, with its existence predicted by George Gamow, Ralph Alpher and Robert Herman as early as 1948.

The CMB offers an unprecedented insight into the large scale structure of the Universe and the conditions shortly after the Big Bang which created it. Since early detections by the COBE satellite (Smoot et al., 1992), its power spectrum has been accurately mapped across the entire sky by the Wilkinson Microwave Anisotropy Probe (WMAP: see Bennett et al., 2003). The CMB has an almost perfect black body spectrum, consistent with the Big Bang hypothesis.

At an almost uniform temperature of 2.725 Kelvin, the small fluctuations within the CMB arise due to the effects of structure present at the last scattering surface. Although matter and radiation were in almost perfect equilibrium prior to decoupling, small overdensities in the mass distribution led to variations in the gravitational potential. As photons travelled away from the last scattering surface, those caught within deeper potential wells needed to expend more energy to escape, causing them to redshift. This process is referred to as the Sachs-Wolfe effect (Sachs and Wolfe, 1967) and has provided information on the primordial power spectra.

There are also contributions to the CMB power spectrum which occur between the last scattering surface and the observer. In a matter dominated Universe, the change in

a cluster's potential in the time taken for a CMB photon to enter and leave the cluster is negligible. Thus the energy spent leaving the cluster is equivalent to that gained entering it with the net result being no noticeable wavelength change. If however the Universe is dominated by radiation or dark energy, then the expansion of space in between the photon's time of entry and exit would reduce the late potential. Hence the energy needed to leave would be less than that gained passing into the cluster. This would lead to a net blue-shifting of CMB photons as they pass through galaxy clusters, referred to as the integrated Sachs-Wolfe effect. Additional non-linear growth in non-virialised structures can also lead to very small shifts in the CMB power spectra termed the Rees-Sciama effect. The Sunyaev-Zel'dovich effect also allows for the boosting of CMB photons via inverse-Compton scattering off of very energetic electrons within the cluster, exacerbating the effect.

The integrated effect of these processes over the distance between the last scattering surface and the observer has been successfully observed, a significant accomplishment presenting firm support for the Big Bang cosmology (Granett et al., 2008).

1.8.1 Formation of Galaxies

Despite the elegance of Hubble's initial proposed evolutionary sequence, recent observations along with results from large cosmological N-body dark matter and full hydrodynamic simulations, have refuted it. Spiral galaxies are now believed to form as cold gas flows down through large scale filaments into a dark matter halo. Close encounters by neighbouring bodies exert gravitational torques on the proto-galaxy, causing it to gain angular momentum and spin. This causes the material to collapse down and flatten into a disk (Binney and Tremaine, 1987). When the galaxy halo stabilises, allowing a stable shock to propagate outwards, the gas flowing down the filaments is shocked heated to the virial temperature. The gas must then radiate this energy before collapsing onto the disk, providing a slow and steady fuel source for ongoing star formation.

The formation of spiral arms is still a poorly understood mechanism. They are observed to be differentially rotating, with material towards the galactic core taking less time to complete a full period revolution. Consequently the arms should wind up within a very small amount of time compared to the age of the galaxy, leaving a more elliptical galaxy as the end product. One possible solution is that the spiral arms are the aggregation of multiple local density fluctuations which result in a temporary spiral arm being formed. As dense regions continually build up and then fragment, forming bright new stars, they

could be sheared out to produce a spiral arm. This could happen many times throughout the lifetime of the galaxy, constantly creating new spiral arms to replace those that fade when young stars die out. Alternatively spiral arms could be the result of a violent disturbance from a recent merger, stimulating bursts of star formation as the gas reservoirs increase and gas is crushed to higher densities. Despite these events being relatively rare, they could play a role in the most massive and luminous spiral galaxies (see Binney and Tremaine, 1987). More permanent spiral arms could be maintained through a supernova driven detonation wave that spreads around the disk. As stars die, supernova shocks squeeze gas in neighbouring regions to densities high enough to promote new star formation, these then go supernova and the wave processes around the galaxy. The presence of a bar in spirals is most likely due to the temporary effects of recent tidal interactions, generating a gravity wave that spreads out from the centre of the spiral galaxy (Bournaud and Combes, 2002).

Elliptical galaxies however are believed to represent a morphological change due to the disruption of galaxies through mergers or tidal interactions. The lack of gas but the presence of a stellar disk in S0 type galaxies may be explained through ram pressure stripping which removes gas as spirals fall into dense cluster haloes (see,e.g., Niemi et al., 2010; Kormendy et al., 2009; Ciotti, 2009; Khalatyan et al., 2008; Rickes et al., 2008). This would help explain the above average abundances of spheroids in group and cluster environments as isolated field galaxies will be disturbed less, providing them with the necessary time to flatten and develop spiral arms.

The very red stellar populations in massive elliptical galaxies can be created by regulating the star formation rate and the accretion of gas onto the galaxy. Processes such as AGN feedback and strong stellar winds Croton et al. (2006); Bower et al. (2006, 2008) can heat cold gas and expel the fuel for star formation from the galaxy. This limits the amount of new stars formed, leaving the current populations to age and turn off the main sequence (Khalatyan et al., 2008). This inclusion of a feedback cycle to regulate star formation has been a key advancement in recent cosmological models. It was first necessitated to counter the phenomenon of "cosmic downsizing" in which most stellar mass was already in place by early times at $z \sim 1$, contrary to the hierarchical predictions of a more gradual buildup of mass (Cowie et al., 1996). As the *FLASH* code does not come supplied with any star formation or supernova feedback routines, I implement the models of Schaye and Dalla Vecchia (2008) and Dalla Vecchia and Schaye (2008), comparing their performance to those in the *GADGET-2* SPH code in chapter 7.

1.8.2 Dwarf Galaxies

As many cosmological dark matter only simulations predict many hundreds of thousands of sub-structures to be orbiting in and around the halo of central bright galaxies (see Springel et al., 2005b, 2008; Kuhlen et al., 2008) we naturally expect that there should be many smaller dwarf galaxies present. This dominant population of dwarf galaxies has been extremely hard to locate, with only a few being detected after concerted efforts (Mateo, 1998). It may be that most of these dwarf galaxies are simply far too faint to be detected. However this "missing satellite problem" has resulted in many mechanisms being proposed to reduce their expected abundances. These range from the suppression of small scale power in the primordial mass power spectrum and warm dark matter whereby free streaming leaves perturbations on small scales unconfined, to the suppression of accretion onto low mass haloes after re-ionization.

The lack of spiral arms in observed dwarf galaxies suggests that they may have undergone processes which removed gas from the galaxy and are now mere husks of their original selves. Ram pressure stripping of their gas as they pass through the hot gaseous haloes of larger galaxies may be the dominant mechanism responsible, leaving behind the stellar population. Without the addition of new young stars, those left will quiescently age, becoming red giants and then later small white dwarfs with very low combined luminosities.

In this way the delicate spiral arms could be shredded and tidal interactions could disrupt the stellar component of the disk as it passes through the halo. The large extended HI tails observed trailing behind galaxies that have passed through cluster environments in the Virgo, Centaurus and Coma clusters would appear to add credence to this theory (Koopmann et al., 2008; Chung et al., 2007; Higdon et al., 2006).

Early semi-analytic models used the instantaneous gas stripping approximation where the gas was instantly removed from satellites as they entered into the parent halo. As this rapidly truncated star formation it resulted in satellite galaxy populations that were too red. In chapter 5 we perform full hydrodynamic simulations in order to derive a more accurate analytic prescription for the ram pressure stripping of the hot gaseous halos of galaxies in groups and clusters. When implemented into semi-analytic models this allows for a more gradual removal of the galaxy gas and greatly improves agreement between theory and observation (see Font et al., 2008). The majority of the simulations are performed in the SPH code *GADGET-2*. Although this allows for large suites of

full hydrodynamic simulations to be run in a feasible period of time, it is known that SPH codes tend to spuriously suppress the formation of hydrodynamic instabilities which exacerbate the stripping of gas. These instabilities are fully resolved in Eulerian codes (see, e.g., Agertz et al., 2007). Despite the simulations being tailored to deal with satellite galaxies that are sufficiently massive for the Kelvin-Helmholtz formation timescale to be large compared to the free-fall time, I confirm the validity of the results using the Eulerian mesh code *FLASH*. I find that the *FLASH* simulations show excellent agreement with the *GADGET-2* simulations. However I note that stripping of gas will most likely be more severe for smaller dwarf galaxies in the AMR codes when the Kelvin-Helmholtz timescale is much shorter.

1.8.3 The Cooling Flow Problem in Galaxy Clusters

One of the great conflicts between theory and observation arises when studying cooling flows in clusters of galaxies. As radiative hydrodynamic simulations show that large cooling flows should be established early on but are not observed, nor are the high star formation rates that should accompany them, it has led to the need to implement some form of thermal feedback.

Although star formation can be regulated in lower mass galaxies through supernova feedback, driving winds out of the galaxy and removing material for further star formation, it is insufficient to offset the cooling present in massive galaxy clusters for which the gravitational potential is too deep.

One currently favoured means is that of Active Galactic Nuclei (AGN). AGN may be able to drive large bubbles in the ICM which will expand and buoyantly rise, redistributing the energy and metals throughout the cluster in a way that could offset cooling (see Dalla Vecchia et al., 2004; Roediger et al., 2007). However there still exist many questions about AGN heating mechanisms such as whether mixing of hot material in the jet is efficient enough for a single dominant AGN to offset cooling or whether the jets remain highly collimated and leave the cluster via narrow channels without much mixing?

1.8.4 Formation of Cool and Non-Cool Cores

Although the origin of cool core clusters can easily be understood in terms of a cooling flow, the production of a non-cool core is less well defined. Many attempts have been made at understanding how higher entropy non-cool core clusters are made using different

numerical techniques but large discrepancies often arise between different groups using different codes.

The large cosmological AMR (*ENZO* code) simulations of Burns et al. (2008) indicate that mergers at high redshift play an important role in the establishment of present-day entropy cores whilst those of Nagai et al. (2007) and Kravtsov et al. (2005) (*ART* AMR code) are in disagreement, yielding large cooling flows at late times. In contrast, SPH simulations seem to consistently produce cool cores (Kay et al., 2004; Borgani et al., 2006). Poole et al. (2006) in particular uses a suite of model SPH merger simulations to systematically test the affects of different merger scenarios on the resulting cluster core. He finds that cool cores are far too resilient to the effects of mergers with a fresh cooling flow establishing itself quickly, returning it to a cool core state.

These differences are most likely due to the different cooling, star formation and stellar feedback recipes implemented. However as the Santa Barbara code comparison of Frenk et al. (1999) highlighted differences in the cluster core entropy profiles obtained when using different hydrodynamic schemes, it is highly likely that fundamental differences in these hydrodynamic schemes will also affect the outcome.

The apparent complexity of the issue is further compounded by our current lack of understanding of other phenomena related to cluster evolution including whether the ICM is turbulent, allowing material to mix efficiently (Schuecker et al., 2004; Churazov et al., 2004; Rebusco et al., 2005, 2006), or viscous (Fabian et al., 2005, 2003b,a).

A turbulent ICM would allow metals formed in the core to be redistributed more effectively by merger driven turbulent mixing, matching observations for non-cool core metallicity profiles. Alternatively it would also imply that the observed HI tails seen around clusters, believed to be the relics of stripped satellite galaxies would be more susceptible to disruption. A more viscous ICM would favour their continued existence however there is a limit to its viscosity as a very viscous ICM would cause the orbits of satellite galaxies within cluster environments to decay rapidly, opposite to what is seen.

This current lack of understanding of both physical phenomena and numerical effects makes it extremely difficult to ascertain whether important physical processes are at play when an interesting result is obtained, or whether it is simply a numerical artefact. The work presented in this thesis is an attempt at understanding simulations of galaxy formation and the underlying properties of the different hydrodynamic implementations so that we can begin to answer some of these questions.

1.9 Structure of Thesis

With a wealth of observational data available, contrasting it with suites of simulations has allowed for large leaps forward in our theoretical understanding to be made. As such numerical simulations now underpin much of modern theory. As authors such as Frenk et al. (1999) and Agertz et al. (2007) clearly show that substantial discrepancies exist between the different hydrodynamic implementations, it is essential that we attempt to understand them. By exploring the way in which these codes differ and understanding why we can ultimately begin to quantify what affects these may have upon our current theoretical understanding of the Universe.

I begin by describing the two fundamental hydrodynamic implementations along with a description of their gravity solvers in chapter 2. As some of the work presented in this thesis presents a significant computational challenge, it was necessary to optimise several of the *FLASH* code modules with particular emphasis upon the gravity solver which dominates the run time. In chapter 3 I discuss the working of the current gravity solver and the new Fourier technique which we implement. Tests of the new algorithm's performance along with my other optimisations are also presented.

In chapter 4 I begin my investigation with a comparison of different SPH and AMR codes using a suite of demanding problems with astrophysical relevance. As analytic solutions exist to these problems it makes it possible to quantitatively compare them. The comparison allows us to both determine the level of agreement between different codes that use the same hydrodynamic implementation as well as to contrast the performance of the two different hydrodynamic formulations.

In order to improve the description of the ram pressure stripping of the hot gaseous haloes of galaxies in groups and clusters we derive a simple analytic formula which can be calibrated using hydrodynamic simulations and implemented into semi-analytic models. In chapter 5 I present the derivation of our analytic formula and summarise the suite of SPH simulations which we use to validate it. As Agertz et al. (2007) has shown that SPH codes spuriously suppress the formation of hydrodynamic instabilities which could accelerate the stripping process, I perform identical runs in the *FLASH* AMR code to guarantee that our results are robust.

The results shown thus far indicate good agreement between the two hydrodynamic implementations, providing sufficient resolution is used. However the Santa Barbara code comparison of Frenk et al. (1999) clearly shows that the core entropy profiles are sub-

stantially higher in galaxy clusters formed using AMR codes. SPH codes on the other hand give rise to denser, cool cores. In order to resolve this discrepancy, I use a suite of mergers between simplified model clusters to systematically explore all of the potential mechanisms which could lead to this deviation. As these simulations lack the complexity of true cosmological simulations, we are able to conclusively show that turbulent mixing in the AMR codes, which is suppressed in SPH, is responsible for the higher entropy cores. Although these simulations lack the additional physics present in cosmological simulations; star formation, supernova feedback and cooling, the results still have significant consequences for current and future simulations of galaxy formation. The efficient mixing of hot and cold gas through merger induced turbulent motions could be sufficient to help offset cooling flows and form the observed populations of non-cool core clusters. In SPH codes, their formation requires a combination of very large amounts of feedback and the use of preheating at high redshift.

The next generation of state-of-the-art cosmological simulations is currently underway. These include all the essential ingredients for galaxy formation; metal dependent cooling, star formation and supernova feedback, thus it is necessary to determine what affect the underlying hydrodynamic schemes have upon the sub-grid recipes. As the default distribution of *FLASH* lacks these vital physical mechanisms I choose to implement the same sub-grid recipes for star formation and supernova feedback as used in the *OWLS* and *GIMIC* SPH simulations, into the *FLASH* AMR code. Using model disk galaxies I am able to contrast the performance of the star formation recipes, the polytropic pressure laws that accompany them and the supernova feedback efficiency in the *GADGET-2* SPH and *FLASH* AMR codes. In chapter 7 I reveal interesting differences in the effects of the polytropic equation of state in the two different codes and how it affects the resulting morphology of the galaxy. Although I can adjust the pressure law to gain agreement in the star formation rate and galaxy morphology, it becomes clear that *FLASH* requires more pressure support against gravitational collapse than in *GADGET-2*. In chapter 8 I also find that the feedback efficiency is much lower in the AMR code, with energy in the wind being dissipated far too quickly to make a substantial impact upon the star formation rate. With current cosmological simulations nearing completion, these effects show that there may be strong variation between their results. With this in mind, the work undertaken in this thesis will prove invaluable for aiding in their interpretation and comparison.

Chapter 2

SPH and AMR, the Theorist's Tool Box

2.1 Introduction

The properties of a given continuous fluid can be represented numerically in one of two main fashions. Mesh codes aim to model the fluid by mapping its spatially averaged quantities to a set of discrete points on a mesh. Most mesh codes fix the mesh spatially and then adjust the value of the mesh mapped variables by determining the amount of mass, momentum and energy which advects through the cell surfaces according to the Eulerian equations for compressible hydrodynamics (see 1.3).

Particle based codes on the other hand aim to represent the fluid flow by breaking it up into a series of discrete masses represented by particles which advect with the fluid flow. As these particles are not spatially fixed these codes are naturally Galilean invariant with the behaviour of the fluid being independent of the chosen inertial frame.

My doctoral thesis has made extensive use of the *FLASH* code, an ideal example of an Adaptive Mesh Refinement code (AMR) which follows the Eulerian mesh example. In addition to this much of my work has also involved the use of, and comparison to another family of hydrodynamics codes known collectively as Smooth Particle Hydrodynamics (SPH) which use the particulate representation of the fluid. In this chapter I aim to cover the fundamental components of both families of code, providing the reader with an understanding of terms and techniques which will be discussed in later chapters.

I begin in § 2.2 by outlining the principles used to model the fluid flow in Eulerian mesh codes including the use of a Riemann solver (§ 2.2.1). I will include those implementation specific parameters adopted in *FLASH*. I will then contrast it in § 2.3 with the hydrodynamic scheme used in Smooth Particle Hydrodynamics including components specific to the *GADGET-2* SPH code which features prominently in my work.

In § 2.4 and § 2.5 I will outline the numerical approaches that AMR and SPH codes

adopt to solve for the gravitational potential on the mesh.

I will then outline in more detail the *FLASH* code architecture in § 2.6 including the different refinement schemes implemented (§ 2.6.1) and the changes that must be incorporated when working in a cosmological context (§ 2.6.2).

2.2 Eulerian Mesh Hydrodynamics

Eulerian mesh codes model the fluid flow of an ideal gas by solving Euler's three parabolic hydrodynamic equations for compressible gas dynamics. In conservative form these are the mass (2.1), momentum (2.2) and energy (2.3) equations (see § 1.3 for their derivation).

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \tag{2.1}$$

$$\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v}\mathbf{v}) + \nabla P = \rho \mathbf{g} \tag{2.2}$$

$$\frac{\partial \rho E}{\partial t} + \nabla \cdot [(\rho E + P)\,\mathbf{v}] = \rho \mathbf{v} \cdot \mathbf{g} \tag{2.3}$$

where ρ is the gas density, $\mathbf{v}$ is the fluid velocity, P is the pressure, E is the sum of the internal and kinetic energy per unit mass given by,

$$E = \varepsilon_{int} + \frac{1}{2}\,|\mathbf{v}|^2 \tag{2.4}$$

$\mathbf{g}$ is the acceleration due to gravity and t represents the system time. The three equations provide the resulting density, velocity and total energy of the gas. The pressure of the fluid can then be obtained from the density and internal energy through the use of a suitable equation of state. The standard equation of state for an ideal gas with ratio of specific heats $\gamma = 5/3$, takes the form

$$P = \rho \varepsilon_{int} (\gamma - 1) \tag{2.5}$$

If the kinetic energy greatly exceeds the internal energy of the gas, then incorrect pressures and temperatures can result, primarily due to truncation errors. These arise from limited numerical precision when using $\varepsilon_{int} = E - \frac{1}{2}|\mathbf{v}|^2$ to obtain the internal energy. To circumvent this the internal energy can be separately evolved when it is less than a given fraction of the kinetic energy (specified by the runtime parameter `eint_switch` $= 10^{-4}$ by default in *FLASH*, the value of which has been ascertained based on numerical tests

with the PPM solver, proven to reduce truncation errors without adversely affecting the dynamics) according to

$$\frac{\partial \rho \varepsilon_{int}}{\partial t} + \nabla \cdot [(\rho \varepsilon_{int} + P)\mathbf{v}] - \mathbf{v} \cdot \nabla P = 0 \tag{2.6}$$

Reactive flows with several different chemical species, essential if we are to model chemical dependent cooling and metal mixing in star forming environments, require the solving of a separate mass advection equation for each chemical or nuclear species being traced,

$$\frac{\partial \rho X_l}{\partial t} + \nabla \cdot (\rho X_l \mathbf{v}) = 0 \tag{2.7}$$

where X_l is the mass fraction of the l^{th} species.

In addition to being able to follow individual chemical species, *FLASH* can also advect "mass scalars" with the fluid. These can be used to trace the movement of material throughout the simulation and are a key tool when following star forming material and the mixing of metals throughout a galaxy. They are advected with the gas density and follow the form of equation (2.8) where ϕ_l is the l^{th} mass scalar. All mass scalars are constrained to lie between 0 and 1 but unlike mass fractions, each mass scalar is independent of the others and therefore they are not re-normalised to sum to one. Using this technique it is possible to independently trace the dispersion of wind and star forming material as is done in chapter 7.

$$\frac{\partial \rho \phi_l}{\partial t} + \nabla \cdot (\rho \phi_l \mathbf{v}) = 0 \tag{2.8}$$

The version of *FLASH* I used throughout this thesis (v2.5) adopts a second-order accurate directionally split Strang-state hydrodynamics algorithm. This allows the more complex Euler equations in three dimensional space to be divided into a series of simpler problems that can be solved separately for the x, y and z-directions.

If all solvers are second-order accurate, then this would result in the overall solution being only first-order accurate. However the splitting scheme of Strang (1968) recovers second-order accuracy by reversing the call of the directionally split solvers every other step. Therefore each pass through the simulation's evolution routine actually performs a double step in time (see figure 2.1).

The Strang methodology has the advantage that the output solution for one of the solvers is the input for the next, thereby ensuring that only a single solution needs to be stored in memory at any one time. Splitting errors can be further reduced by ordering

```
subroutine evolve( )
  hydro (x-sweep)
  hydro (y-sweep)
  hydro (z-sweep)
  source terms
  particle modules
  gravity

  hydro (z-sweep)
  hydro (y-sweep)
  hydro (x-sweep)
  source terms
  particle modules
  gravity
end subroutine evolve
```

Figure 2.1: Flow of the directionally split hydrodynamics solver. By inverting the directions in which the hydrodynamics solver sweeps over cells, 2^{nd} order accuracy in time and space can be regained.

calls with preference upon solving the least "stiff" differential equation first and ending with the most stiff. A stiff differential equation is defined as one which under certain numerical methods used for solving differential equations, can become unstable and lead to lack of convergence. It is mentioned for the reader's interest that the most recent version, *FLASH* 3., also includes an unsplit hydrodynamics solver although it is not tested in this thesis.

The Euler equations are solved using a modified version of the *Piecewise Parabolic Method* (PPM) of Colella and Woodward (1984), descended from the PROMETHEUS code (Fryxell et al., 1989). PPM is a higher-order version of the earlier scheme by Godunov (1959) which treated the fluid flow as a series of discrete slabs, each occupying a single cell on the mesh. The fluid properties at the cell interfaces are used as the input conditions for Riemann's shock tube problem. The Riemann problem describes two uniform states initially separated by a discontinuity. The values of pressure, density and velocity are normally used to describe the Riemann problem which over time will lead to the formation

of a shock and a rarefaction heading in opposite directions away from the initial contact discontinuity (see § 1.3.1). The solution to Riemann's shock tube problem which is solved at each cell interface gives the fluxes through the cell faces. The conserved quantities; mass, momentum and energy are then advected through the cell faces according to these fluxes by solving finite-difference equations. Given that the fluid is treated as a series of constant value slabs at each timestep, this restricts the scheme to being accurate to first-order. PPM improves upon this by reconstructing the fluid values at the cell boundaries using piecewise parabolic interpolants. These require stencils with nine interpolating points - four either side of the interface and one in the cell, hence the need for four guardcells at the edge of each block of data (zones filled with either boundary data or copies of the data from neighbouring blocks, see § 2.6). The use of a Riemann's solver, an algorithm used to solve Riemann's shock tube problem, introduces explicit non-linearity into the difference equations and allows for the calculation of sharp shock fronts and contact discontinuities, a key advantage over SPH codes.

In addition to using parabolic interpolants, PPM also implements monotonicity checking to prevent oscillations in the fluid flow along with intelligent dissipation algorithms, removing the need for artificial viscosity terms. The parabolic interpolations that are used do not necessarily interpolate the central values in the cells as these represent mean properties over the cell. Instead a considerably more complex interpolating function is used to ensure that its average value integrated over the cell is equal to that of the cell mean. This allows the PPM interpolations to recover the actual values of the fluid properties at the interface and not the mean values that are normally stored in the cells. PPM implements contact discontinuity steepening to correct for numerical diffusion of sharp structures as they advect across the mesh and PPM also flattens shock fronts which become thinner than a single cell. This can occur due to the exceptionally low dissipation in the PPM scheme and the natural self-steepening of shocks. This is done in order to prevent artificial oscillations in the fluid flow that can otherwise result behind the shock as it becomes poorly resolved by the mesh.

Despite PPM having the potential to be accurate to third-order, such an improvement is found to yield only a small improvement in accuracy but at a much greater computational cost (Fryxell et al., 2000). The most critical elements of the code however are kept at third or fourth-order accuracy to ensure that the code is both efficient and more accurate than many other codes that are formally accurate to second-order. For an in-depth description of the PPM algorithm I refer the reader to Fryxell et al. (2000).

2.2.1 The Riemann Solver

The most powerful feature of AMR codes is their use of a Riemann solver to model the behaviour of the fluid in the presence of sharp shocks and contact discontinuities. It allows steep density and pressure changes to be accurately tracked using only a few resolution elements, meaning that they can offer exceptionally good resolution whilst operating at modest levels of refinement. They remove the need for artificial viscosity terms which can lead to smoothing of shock fronts over many cells as is common in SPH codes and naturally enable turbulent fluid motions such as Kelvin-Helmholtz and Rayleigh-Taylor instabilities to be tracked.

The Riemann problem itself describes the evolution of two constant states that are initially separated by a discontinuity. As the initial states generated above by the PPM describe two states with parabolic distributions, it is necessary to construct effective states that fit the Riemann problem. Although suitable values would be the zone averaged central values, this would yield an identical setup to the Godunov scheme and thus the higher order-accuracy of the PPM would be lost. Higher accuracy can instead be attained given that the signals within the gas propagate at a finite speed (equation 1.18), thus not all of the material within a cell necessarily influences the boundary. Suitable simulation timesteps are chosen by calculating the shortest time taken for a signal to propagate across a given fraction of the cell based upon the gas sound speed and the cell size. The maximum fraction covered in a timestep dt is determined by the Courant-Friedrichs-Lewy condition and is fixed at 0.8, the default for *FLASH* based upon numerical experiment. This ensures that material cannot traverse more than eighty percent of the cell in a given timestep and thus helps ensure the numerical stability of the Riemann scheme in time.

In the Lagrangian approach shown in figure 2.2, only material enclosed by the two dashed lines corresponding to sound waves travelling away from the interface at speed, c, can intersect with the interface at timestep $n + 1$. Thus, only material enclosed within this space can affect the outcome at the boundary.

Added complications arise when moving to a spatially fixed Eulerian grid as there are now three signal speeds able to affect the fraction of material which influences the boundary over the timestep; $u - c$, u and $u + c$ for a fluid velocity on the mesh, u. Furthermore, whereas in the Lagrangian model, the sounds waves were constrained to move away from the boundary, now these three waves can lie either side of the interface. For a supersonic flow to the right as is shown in figure 2.3, all three characteristic waves

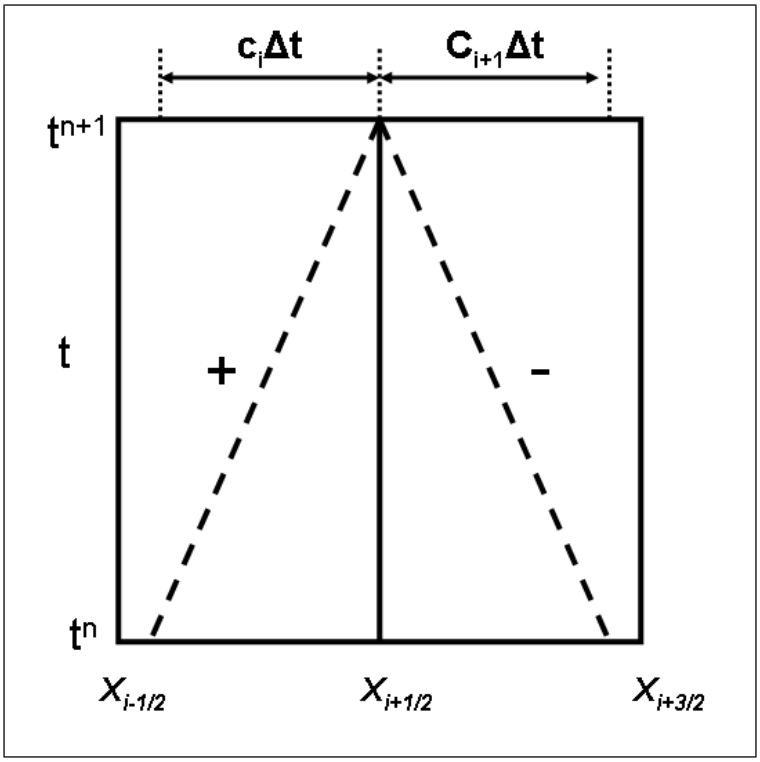

Figure 2.2: Lagrangian view of the characteristic waves that influence the boundary between two cells when constructing the Riemann problem. The dashed lines represent waves that intersect with the boundary at $x_{i+1/2}$, at the next time step, t^{n+1}. Material within the regions $c_i\Delta t$ and $c_{i+1}\Delta t$ can affect the flux of material across the boundary, for c_i and c_{i+1} the sound speed of gas in those regions.

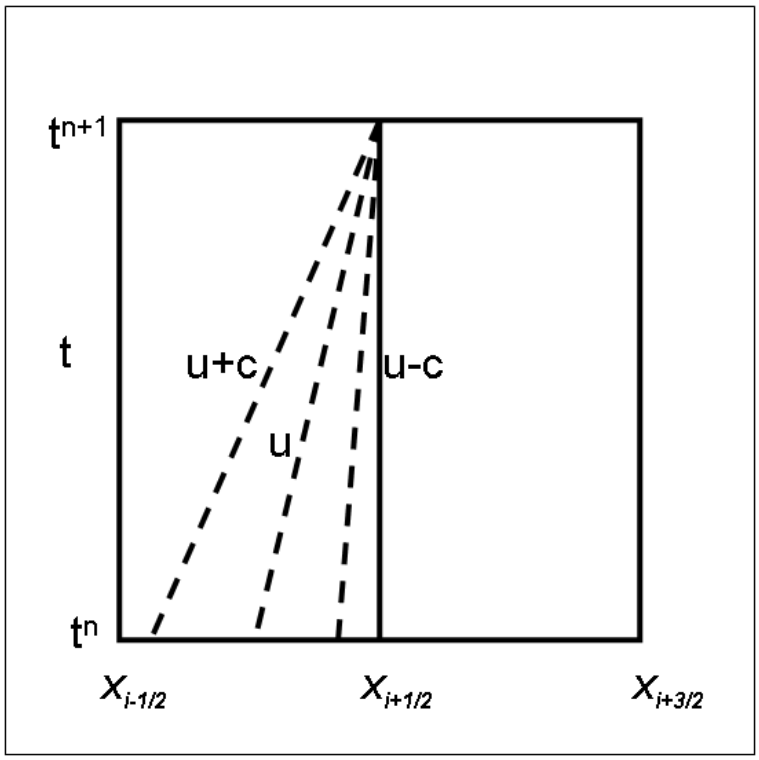

Figure 2.3: Eulerian view of the characteristic waves that influence the boundary between two cells when constructing the Riemann problem for a supersonic flow to the right. The cell boundary is located at $x_{i+1/2}$. The three dashed lines represent the 'entropy wave' with the fluid velocity, u, and the two sound waves $u \pm c$. The supersonic speed of the material passing from the left to the right ensures the $u - c$ sound wave passing towards the interface does not have time to contact material on the right of the cell boundary. Therefore material on the boundary's right is unable to influence interactions at the boundary and all three characteristic waves lie to the left of the cell boundary.

lie to the left of the boundary at $x_{i+1/2}$. Only material from the left cell can influence the flux of mass and energy through the cell face as the supersonic gas speed prevents the $u-c$ sound wave from reaching the material in the right cell before the next time step at t^{n+1}. For a subsonic flow as shown in figure 2.4 there will be one sound wave characteristic on either side of the boundary so that material from both cells can contribute to the interactions at the boundary. In addition the entropy wave associated with the fluid velocity, u, can lie either side of the interface depending upon the direction of the fluid flow.

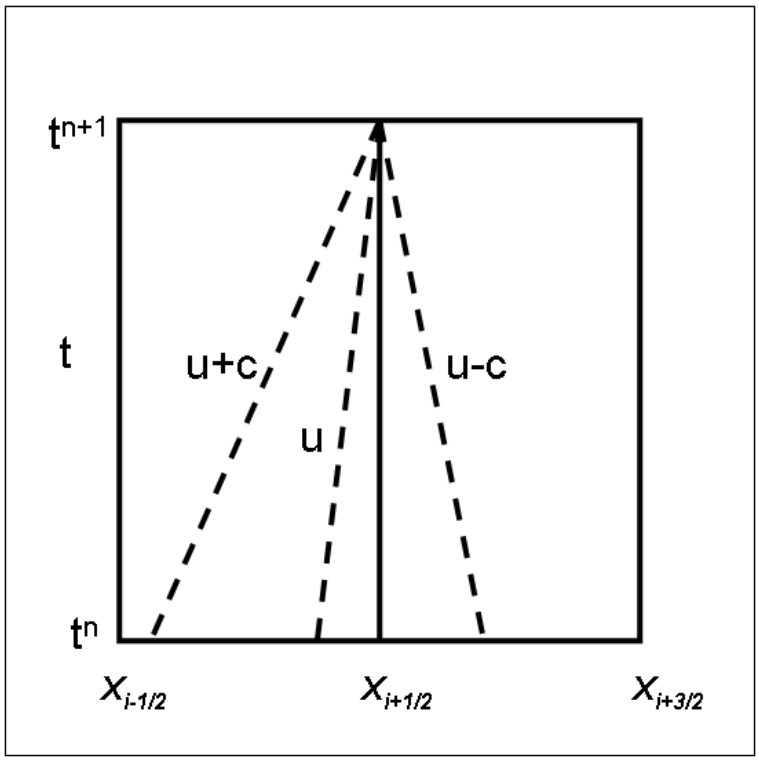

Figure 2.4: Eulerian view of the characteristic waves that influence the boundary between two cells when constructing the Riemann problem for a subsonic flow to the right. The cell boundary is located at $x_{i+1/2}$. The three dashed lines represent the 'entropy wave' with the fluid velocity, u, and the two sound waves $u \pm c$. The subsonic speed of the material passing from the left to the right allows the $u - c$ sound wave passing towards the right to enclose a fraction of the material within the right cell. Thus material from both cells influences the flux of energy and material at the cell boundary. Sound waves reside on both sides of the boundary and the entropy wave can reside either side of the boundary depending upon the direction of the flow of material.

To solve the Riemann problem, an initial guess at which of the above regimes we are operating in is made based upon the average values for the fluid velocity either side of the zone interface and the material enclosed within each sound wave. This is then corrected for by determining the amount of material in each wave which will not reach the interface due to the pressure of the other waves. Riemann's problem is then solved by performing successive Newton-Raphson iterations to obtain the post-shock pressures and fluid velocities at the cell boundaries. These in turn then give the post-shock density either side of the interface. The end result is a conversion of the initial Riemann problem inputs which correspond to spatial averages, into zone averages over the time interval t to $t + \Delta t$. These are then used to determine the flux of material through the cell walls ensuring second order accuracy is maintained in time.

Galaxies in cosmological simulations often have bulk velocities of up to $\sim 1000\text{kms}^{-1}$. Galilean invariance requires that the application of a bulk velocity does not affect interactions in the rest frame of the galaxy. However as the relative velocities on either side of the spatially fixed mesh are used to determine the nature of Riemann's shock tube problem, the addition of a bulk velocity can prove problematic. If a large bulk velocity is applied to the gas in the same direction as a shock, it could strengthen its intensity. Alternatively if the bulk velocity is applied in the opposite direction to the propagation of the shock, it may reduce the shock's strength. This could present a major limitation to Eulerian techniques and therefore in chapters 4 and 6, I present numerical tests in which different bulk velocities are applied to a system and the outcome is observed. Fortunately, although the application of a bulk velocity does indeed alter the resulting galaxy cluster profile, I will show that its affects are relatively small, quantifiable and are still found to converge upon the static solution. Indeed recent work by Robertson et al. (2010) has shown that AMR simulations ultimately converge upon a fixed solution regardless of the bulk velocity providing that sufficient resolution is used, restoring Galilean invariance.

Clearly the use of a Riemann solver increases the complexity of AMR schemes. With the need for these calculations to be performed for every cell face at each timestep, the routine increases the computational overhead required to model the hydrodynamics. However the work presented later in this thesis clearly highlights major differences between the results from simulations run using SPH and AMR methodologies, ensuring that AMR codes maintain an essential part of the theorists tool box, particularly when independently validating the results from other codes with different hydrodynamic implementations.

2.3 Lagrangian Smooth Particle Hydrodynamics

An alternative to the Eulerian mesh approach is to adopt a fully Lagrangian technique such as Smooth Particle Hydrodynamics. In SPH the fluid is modelled by a series of particles each with a fixed discrete mass. Values for other variables can be obtained by interpolating between neighbouring particles within a "smoothing length." For instance the density can be obtained by looping over a set number of nearest neighbouring particles, integrating their masses and dividing through by the volume they occupy. This tends to lead to smoothing of sharp density perturbations such as shock fronts and contact discontinuities over the smoothing length but has been shown to result in accurate reproduction of large scale fluid flow properties (see, e.g., Springel, 2005).

The SPH hydrodynamic equations that describe the evolution of conserved quantities; mass (2.9), momentum (2.10) and energy (2.11) are shown below (Monaghan, 1992):

$$\frac{d\rho_a}{dt} - \sum_b m_b \mathbf{v}_{ab} \cdot \nabla_a W_{ab} = 0 \tag{2.9}$$

$$\frac{d\mathbf{v}_a}{dt} + \sum_b m_b \left(\frac{P_b}{\rho_b^2} + \frac{P_a}{\rho_a^2} \right) \nabla W_{ab} = 0 \tag{2.10}$$

$$\frac{dE_a}{dt} - \frac{1}{2} \sum_b m_b \left(\frac{P_b}{\rho_b^2} + \frac{P_a}{\rho_a^2} \right) \mathbf{v}_{ab} \cdot \nabla_a W_{ab} = 0 \tag{2.11}$$

where m_b corresponds to the mass of particle b, E_a the energy of particle a for a pressure P, density ρ and velocity difference $\mathbf{v}_{ab} = \mathbf{v}_a - \mathbf{v}_b$. Once initial densities and pressures are determined, these equations allow the new particle velocities and energies to be calculated. Once the velocities are updated, the particle positions can be evolved using conventional particle leapfrog techniques (see e.g. § 1.7.3) and then the process is repeated for each timestep.

The heart of SPH techniques lie in the interpolation method which can be used to express any variable in terms of its values at a series of disordered points - the particle positions. The integral interpolant of any function $A(\mathbf{r})$ is defined as

$$A_I(\mathbf{r}) = \int A(\mathbf{r}') W(\mathbf{r} - \mathbf{r}', h) \mathbf{dr}' \tag{2.12}$$

where the integration is over the entire space and W is a function referred to as the "Kernel." The Kernel has two main properties, firstly the volume enclosed by it is normalised to one and in the limit of the smoothing length, h, tending to zero, the Kernel behaves as a Dirac delta function, i.e. $\lim_{h \to 0} W(\mathbf{r} - \mathbf{r}', h) = \delta(\mathbf{r} - \mathbf{r}')$. For numerical work this

integral is approximated by the summation over all particles,

$$A_S(\mathbf{r}) = \sum_b m_b \frac{A_b}{\rho_b} W(\mathbf{r} - \mathbf{r}_b, h) \tag{2.13}$$

Here each particle with index, b, position, $\mathbf{r}_b$, density, ρ_b, velocity, $\mathbf{v}_b$ and mass, m_b, can determine the value of any property A at its position to be given by A_b. In *GADGET-2*, a spline Kernel is adopted of the form (Springel, 2005),

$$W(\mathbf{r}, h) = \frac{8}{\pi h^3} \left\{ \begin{array}{ll} 1 - 6\left(\frac{r}{h}\right)^2 + 6\left(\frac{r}{h}\right)^3, & 0 \leq \frac{r}{h} \leq \frac{1}{2} \\ 2\left(1 - \frac{r}{h}\right)^3, & \frac{1}{2} \leq \frac{r}{h} \leq 1 \\ 0, & \frac{r}{h} > 1 \end{array} \right\} \tag{2.14}$$

As the Kernel is constructed such that only particles within the smoothing length, h, can contribute, it is only necessary to sum over the nearest neighbouring particles in equation (2.13) when computing variables.

The Kernel is constructed so as to be differentiable, thereby ensuring that interpolating functions between the particle positions are also differentiable. Thus to obtain the derivative of A we use

$$\nabla A(\mathbf{r}) = \sum_b m_b \frac{A_b}{\rho_b} \nabla W(\mathbf{r} - \mathbf{r}_b, h) \tag{2.15}$$

or to ensure higher accuracy

$$\rho \nabla A = \nabla(\rho A) - A \nabla \rho \tag{2.16}$$

These provide the fundamental tools necessary to arrive at the SPH hydrodynamic equations. By using the mathematical relation between direct and partial derivatives (Monaghan, 1997),

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{v} \cdot \nabla \tag{2.17}$$

I can rewrite the Eulerian continuity equation (2.1) in total differential form,

$$\frac{d\rho}{dt} - \mathbf{v} \cdot \nabla \rho + \nabla \cdot (\rho \mathbf{v}) = 0 \tag{2.18}$$

which using equation (2.16) can be simplified to give

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{v} \tag{2.19}$$

Therefore using equation (2.15), I can determine the SPH continuity equation to be

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{v}_{ab} = -\sum_b m_b \mathbf{v}_{ab} \cdot \nabla W(\mathbf{r}_a - \mathbf{r}_b, h) = -\sum_b m_b \mathbf{v}_{ab} \cdot \nabla W_{ab} \tag{2.20}$$

The momentum equation can be derived by taking similar steps to obtain Euler's momentum equation in the form,

$$\frac{d\mathbf{v}}{dt} = -\frac{\nabla P}{\rho} \tag{2.21}$$

As I can write the gradient of the pressure as

$$\frac{\nabla P}{\rho} = \nabla\left(\frac{P}{\rho}\right) + \frac{P}{\rho^2}\nabla\rho \tag{2.22}$$

then the change in velocity and thus momentum can be written as

$$\frac{d\mathbf{v}_a}{dt} = -\sum_b m_b \left(\frac{P_b}{\rho_b^2} + \frac{P_a}{\rho_a^2}\right)\nabla_a W_{ab} \tag{2.23}$$

The SPH momentum equation is a realisation of Newton's third law whereby in a medium where the pressure is constant between sets of particles, each set of particles exerts equal and opposite forces upon each other. Hence there exists no net force and no acceleration.

As the equation for the rate of change of thermal energy per unit mass, E_{int}, (Monaghan, 1992)

$$\frac{dE_{int}}{dt} = -\left(\frac{P}{\rho}\right)\nabla\cdot\mathbf{v} \tag{2.24}$$

can be written for particle a in the form,

$$\frac{dE_{int}}{dt} = \left(\frac{P_a}{\rho_a^2}\right)\sum_b m_b \mathbf{v}_{ab}\cdot\nabla_a W_{ab} \tag{2.25}$$

Then by noting that

$$\frac{dE_{int}}{dt} = -\nabla\cdot\left(\frac{P\mathbf{v}}{\rho}\right) + \mathbf{v}\cdot\nabla\left(\frac{P}{\rho}\right) \tag{2.26}$$

the thermal energy equation for particle a can be re-written in the form

$$\frac{dE_{int}}{dt} = \sum_b m_b \left(\frac{P_b}{\rho_b^2}\right)\mathbf{v}_{ab}\cdot\nabla W_{ab} \tag{2.27}$$

By taking the average of equations 2.25 and 2.27 it is possible to rewrite the energy equation in a similar format to the momentum equation 2.10 including the contributions from both particle a and the other particles,

$$\frac{dE_a}{dt} - \frac{1}{2}\sum_b m_b \left(\frac{P_b}{\rho_b^2} + \frac{P_a}{\rho_a^2}\right)\mathbf{v}_{ab}\cdot\nabla_a W_{ab} = 0 \tag{2.28}$$

Though derived in similar but not identical fashion to the Euler hydrodynamic equations, in theory they should provide an identical prescription of the fluid flow. One key area in which Eulerian mesh codes differ from SPH codes is that SPH codes lack the ability to implement shock physics through the use of a Riemann solver. To mimic the behaviour

of shocks an artificial viscosity is implemented in place. This additional term is added to the momentum and energy equations,

$$\frac{d\mathbf{v}_a}{dt} = -\sum_b m_b \left(\frac{P_b}{\rho_b^2} + \frac{P_a}{\rho_a^2} + \Pi_{ab} \right) \nabla W_{ab} \tag{2.29}$$

$$\frac{dE_a}{dt} = \frac{1}{2} \sum_b m_b \left(\frac{P_b}{\rho_b^2} + \frac{P_a}{\rho_a^2} + \Pi_{ab} \right) \mathbf{v}_{ab} \cdot \nabla_a W_{ab} \tag{2.30}$$

where Π_{ab} is given by

$$\Pi_{ab} \left\{ \begin{array}{ll} \frac{-\alpha \bar{c}_{ab} \mu_{ab} + \beta \mu_{ab}^2}{\bar{\rho}_{ab}} & \mathbf{v}_{ab} \cdot \mathbf{r}_{ab} < 0 \\ 0 & \mathbf{v}_{ab} \cdot \mathbf{r}_{ab} > 0 \end{array} \right\} \tag{2.31}$$

with $\bar{c}_{ab}$ being the mean sound speed at the positions of particles a and b, with

$$\mu_{ab} = \frac{h \mathbf{v}_{ab} \cdot \mathbf{r}_{ab}}{\mathbf{r}_{ab}^2 + \eta^2} \tag{2.32}$$

This technique ensures that the artificial viscosity is zero except for when particles are approaching each other in space. When present, the viscosity acts as an excess resistive pressure, thermalising the kinetic energy of the colliding gas and thus mimicking the behaviour of a shock. As long as energy is thermalised in the vicinity of a shock then the large scale flow properties will remain unaffected. The advantages of such an approach include that it is fully Lagrangian, it vanishes for the rotation of rigid bodies and conserves total angular momentum. The values for α and β are set based upon numerical tests and typically take values around two and one respectively. The effect of varying them is explored in chapter 6 when dealing with entropy generation in pure hydrodynamic simulations of cluster mergers and is found to have little affect on the resulting entropy generation in shocks.

As the resolution is effectively determined by the magnitude of the smoothing length and the number of particles it encloses, it is normally altered to enclose a given number of nearest neighbours or in the case of variable mass particles, a given amount of mass. The convention adopted in *GADGET-2* is to fix the Kernel volume so that it encloses a constant amount of mass for the estimated density, thus

$$\frac{4\pi\rho_b}{3} h^3 = N_{sph} \bar{m} \tag{2.33}$$

where N_{sph} is the typical number of nearest smoothing neighbours and $\bar{m}$ is the mean particle mass being smoothed over. This formalism ensures that the smoothing length decreases and thus the effective resolution increases as the density increases. This is an

extremely desirable outcome for astrophysical simulations which aim to follow cosmic structure formation, allowing resolution to be concentrated in regions with the most mass. However it does limit the SPH technique to only being able to refine based upon the density, thus regions of low density that would benefit from the use of high resolution to model shocks such as in black hole powered jets or supernova ejecta are better modelled with AMR techniques.

Due to its robustness and reliability when implementing new physics modules, SPH has proven to be an extremely versatile tool for modern astronomers. The use of an artificial viscosity to induce the entropy generation in shocks, in place of the more computational demanding Riemann solver has also made SPH codes much faster and more memory efficient. This has allowed them to be used for some of the largest, most ambitious simulations to date (e.g. OWLS & GIMIC simulations, see chapter 1). However as the origin of the hydrodynamic equations is slightly different between the AMR and SPH implementations we should nevertheless test their performance using a series of simple test problems for which known solutions exist. To this end the work I present in chapter 4 aims to approach this.

2.4 Gravity Solvers in AMR Codes

Gravity dominates the large scale structure of the Universe and is fundamentally important in almost all astrophysical processes. From the merging of galaxy clusters to the collapse of material to form stars, the energy input from gravitational interactions is enormous. Because of this, accurate modelling of gravitational forces is of paramount importance.

AMR codes solve for the gravitational potential on the mesh using Poisson's equation (3.3) which relates it to the density. It is a simple step to then take the potential and difference it across cells to produce the gravitational acceleration that gas within a cell should experience.

$$\bigtriangledown^2\phi = 4\pi G\rho \tag{2.34}$$

I defer a detailed description of the algorithms used to solve Poisson's equation in *FLASH* to chapter 3. There I discuss the multigrid method and the benefits yielded from combining it with fast Fourier techniques.

The *FLASH* code permits the tracking of collisionless particles with discrete masses,

including their gravitational effects, by mapping their masses via a cloud-in-cell approach to a separate variable on the adaptive mesh, `pden`. The cloud-in-cell assigns the otherwise point mass particle a volume equivalent to that of a mesh cell and weights the density mapped to each cell according to how much of the particle cloud resides within it (see equation 2.35). This allows the particle mass to be mapped to several neighbouring cells. This is an improvement upon simply counting the number of particles in a cell which can lead to a much noisier density field with sharp density contrasts between adjacent cells.

$$\rho_{i,j,k} = \frac{m_{particle}}{\Delta x \Delta y \Delta z} \frac{(P_x - x_{i,j,k})}{\Delta x} \frac{(P_y - y_{i,j,k})}{\Delta y} \frac{(P_z - z_{i,j,k})}{\Delta z} \tag{2.35}$$

where for a cell with indices i, j, k, the density $\rho_{i,j,k}$ is a function of the cell's position $(x_{i,j,k}, y_{i,j,k}, z_{i,j,k})$ and the particle position (P_x, P_y, P_z). Δx, Δy, Δz correspond to the cell size in the x, y and z-direction respectively and $m_{particle}$ is the particle mass.

Once the particle contribution is mapped to the mesh, it is then added to the gas density which is used to determine the gravitational potential. As particles are stored on the processor holding the block which they reside in and a copy is not kept by all processors, it is possible for the mass of a particle to be mapped into guardcells which represent internal cells on neighbouring blocks. As this information needs to be communicated to neighbouring blocks and can entail cross processor communication, the `MapParticlesToMesh` routine presents a substantial strain on hardware.

The use of particles to represent a smooth dark matter distribution is common place in many types of code, however problems with their use in AMR codes may arise when mapping discrete particle properties such as their mass to an overly refined mesh. What may be intended as a smooth dark matter distribution could end up appearing overly concentrated in a few cells instead of smoothly distributed throughout the mesh. To overcome this, Paul Ricker (University of Illinois at Urbana-Champaign) developed a `SmoothMapParticlesToMesh` module. This performs a restriction operation on the leaf block once the particle masses have been mapped to it and checks to see if the parent block on the level below has any cells that are empty. If there are empty cells then the parent restricts the mapped particle density to the next level below and again checks the cell densities. This is repeated until all of the cells within a block contain mass mapped from the particles. At this point the particle density is prolonged back up the mesh, thereby filling empty cells with a representative amount of mass from the particle distribution. This technique accurately represents the initially discrete particles as a smooth distribution with the same properties. It was adopted in the work presented in chapters 6

and 7 as a check to ensure that the particle distribution is adequately sampled relative to the mesh size. It was not used for all runs as adverse sampling effects where found to be negligible and the prolongation and restriction operations require communication up and down the tree which can entail timely cross processor communication.

2.4.1 Solving for the Evolution of Particles

The time dependent evolution of particles in *FLASH* follows a second order accurate variable timestep leap frog method. Particles are advanced from timestep n to $n+1$ with step size Δt using the following difference equations:

$$\begin{aligned} \mathbf{x}_i^1 &= \mathbf{x}_i^0 + \mathbf{v}_i^0 \Delta t^0 \\ \mathbf{v}_i^{1/2} &= \mathbf{v}_i^0 + \tfrac{1}{2}\mathbf{a}_i^0 \Delta t^0 \\ \mathbf{v}_i^{n+1/2} &= \mathbf{v}_i^{n-1/2} + C_n \mathbf{a}_i^n + D_n \mathbf{a}_i^{n-1} \\ \mathbf{x}_i^{n+1} &= \mathbf{x}_i^n + \mathbf{v}_i^{n+1/2} \Delta t^n \end{aligned} \tag{2.36}$$

with coefficients C_n and D_n;

$$\begin{aligned} C_n &= \tfrac{1}{2}\Delta t^n + \tfrac{1}{3}\Delta t^{n-1} + \tfrac{1}{6}\left(\tfrac{(\Delta t^n)^2}{\Delta t^{n-1}}\right) \\ D_n &= \tfrac{1}{6}\left(\Delta t^{n-1} - \tfrac{(\Delta t^n)^2}{\Delta t^{n-1}}\right) \end{aligned} \tag{2.37}$$

In addition to active particles (those with mass), tracer particles which advect with the fluid flow can be incorporated. These are advected using the forward Euler scheme (equation 2.38) with the velocity, $\mathbf{v}$, being obtained by interpolation from the velocity of the fluid on the mesh.

$$\mathbf{x}_i^{n+1} = \mathbf{x}_i^n + \mathbf{v}_i^n \Delta t^n \tag{2.38}$$

2.5 Gravity Solvers in SPH Codes

SPH codes can tackle the issue of calculating the gravitational interaction in a variety of ways. One of the most common and powerful is that of the "tree". The net acceleration experienced by the i^{th} particle due to the gravitational interactions between itself and each of the $N-1$ other particles can be calculated through the direct summation:

$$\dot{\mathbf{v}}_\mathbf{i} = \sum_n^{N-1} -\frac{G m_n}{r_{ni}^2}\mathbf{r}_{ni} \tag{2.39}$$

This summation is performed separately for each of the N particles. Although simple to code up, it offers poor scaling with $N-1$ direct summations being required for each

particle in the simulation. Time can be saved by considering that the attraction between each pair of particles is equal and opposite, thus when calculating the force exerted by particle n on particle i, the value can be stored in memory and used when calculating the effect of particle i on n, reducing the required calculations by a factor of two. However in order for large N-body structure formation simulations to be feasible a superior alternative must be sought.

The hierarchical multipole expansion technique, commonly referred to as a "tree" algorithm involves splitting the overall particle distribution into increasingly larger sub-groups. For each sub-group the local gravitational forces can be calculated by looping over neighbouring particles within the sub-group using equation (2.39). The large scale gravitational forces however are calculated by performing the summation over whole sub-groups, with each group representing their collective masses, each with a single centre of mass. This forms a hierarchical tree with the large scale gravitational forces being calculated for a few massive sub-groups from which the results can be extended down the tree to increasingly smaller scales and sub-groups until we reach individual particles.

Although this ultimately induces small inaccuracies into the gravitational calculation, the error can be controlled by ensuring that particles within each sub-group are sufficiently far enough apart before splitting the sub-group up further. Placing particles which are close to each other into different sub-groups exacerbates the errors incurred.

The tree opening criteria (equation 2.40) defines the point at which the force error induced by using a larger sub-grouping of particles becomes unacceptable and a greater number of integrations over the smaller sub-groups it contains is necessary. The tree opening criteria is chosen such that

$$\frac{GM}{r^2}\left(\frac{l}{r}\right)^2 \leq \alpha|\dot{\mathbf{v}}| \tag{2.40}$$

where M is the mass of the sub-group with an extension of l at a distance, r, from the particle for which the acceleration is being calculated. $\dot{\mathbf{v}}$ represents the particle's acceleration at the last timestep and α is a tolerance parameter. If the left hand term of equation 2.40 is greater than the right hand term then the tree is opened up to include integrations over the smaller sub-groups within the current grouping. This helps minimise the force error introduced in each particle–sub-group interaction by comparing a rough estimate of the truncation error with the size of the total expected force.

The repeated grouping of particles has the advantage of keeping the gravitational integrations local over several particles instead of many thousands. The large scale gravi-

tational terms can then be applied down the tree, from the large scale super-groups to the smallest sub-groups, greatly accelerating the overall process. This results in a technique that scales more closely to $O(Nlog(N))$ instead of the N^2 calculation required for the direct summation over all particles.

In addition to the use of a tree, the particle distribution can be mapped to a mesh, similar to that done in the AMR codes. The gravitational force is then solved for using either iterative schemes or more commonly, fast Fourier transforms to obtain the potential (see chapter 3). Hybrid schemes which use the mesh to accelerate the calculation of large scale gravitational terms but then incorporate the use of a tree to recover small scale gravitational accuracy benefit from the advantages of both approaches.

As interactions in very dense regions such as the cores of galaxies occur over much smaller spaces of time than those in the sparser outer regions like the halo, vast amounts of computation can be wasted by evolving all particles in lock-step. To overcome this, individual adaptive time-stepping can be adopted, ensuring each particle is only evolved with the necessary timesteps they require. The forces acting on the particles within each sub-group in the gravitational tree can be broken into a long range force component due to the other sub-groups and a short range component due to particles within the sub-group. By performing multiple short range particle kicks to every large scale kick, the frequency at which the tree is rebuilt and the number of multipole calculations can be drastically reduced without sacrificing accuracy. For simplicity and to ensure that the scheme remains synchronised, timesteps are typically kept to a power of two smaller than those on the scale above.

2.6 FLASH - an Adaptive Mesh Refinement Code

FLASH was developed at the Alliances Center for Astrophysical Thermonuclear Flashes under contract to the US Department of Energy as part of the Accelerated Strategic Computing Initiative (*ASCI*). As such the code is highly modular, massively parallel and has been run on some of the largest distributed memory machines in existence including the Blue Gene/P supercomputer with 65,536 cores at the Argonne National Laboratory. The goal of the Flash Center is to significantly advance understanding of thermonuclear flashes which occur in and around white dwarfs and neutron stars.

The code is highly modular in construction with the user being able to include additional physics modules simply by adding the phrase #REQUIRES *'module name'* to the

setup's *Config* file. This has allowed me to implement additional modules for star formation, stellar feedback, metal dependent cooling and a new improved gravity solver which was developed in collaboration with Dr. Tom Theuns (Durham University & University of Antwerp).

The *FLASH* code adopts a Eulerian block structured mesh approach which is able to actively refine and de-refine. The mesh is fixed spatially and the simulation volume fits within the "level one refinement" block. Each block is sub-divided into a series of cells, the number of which are specified at runtime. In 3-Dimensions (3D) *FLASH* adopts an octree with a quadtree being used in 2D. When refining the region of space occupied by a block, additional "child" blocks are created in place with each dimension being half as long as those of the larger "parent" block that they cover. In this way a refinement will increase the number of blocks in a 3D simulation by 8 (or 4 in 2D, see figure 2.5). These child blocks can then further refine up to higher levels in the same manor, spawning child blocks of their own. Blocks at the highest level of refinement for that region of space, with no children of their own are termed "leaf" blocks and the data they contain is considered to be the most up to date. The user is able to stipulate a minimum, "*lrefmin*" and a maximum, "*lrefmax,*" level of refinement.

In 3D, each block is split into *nxb*, *nyb* and *nzb* cells in the x, y and z-direction respectively with an additional four cells either side termed "guardcells". These guardcells contain a copy of the information mapped from neighbouring blocks or if the block edges lie on the simulation boundary, boundary conditions are mapped into them. These guardcells are vital for the exchange of information to neighbouring blocks and for the interpolation which takes place when refining a region and copying parent cell data to new children – a process termed "prolongation." They are also required when calculating pre-shock conditions at the edges of the block in the PPM solver (see § 2.2). The addition of guardcells allows a block to be evolved in parallel with other neighbouring blocks whilst minimising the amount of expensive cross processor communication needed to determine boundary values for a given block face. No two neighbouring blocks are permitted a difference in level of refinement greater than one.

The user stipulates whether the simulation should have outflow - zero gradient boundary conditions which allow shocks to leave the domain, reflective - non-penetrating zero gradient boundaries which reflect the fluid's transverse velocity, periodic boundary conditions which wrap around to the opposite side of the simulation box or user defined conditions. For the simulations within this thesis I adopt periodic boundary conditions

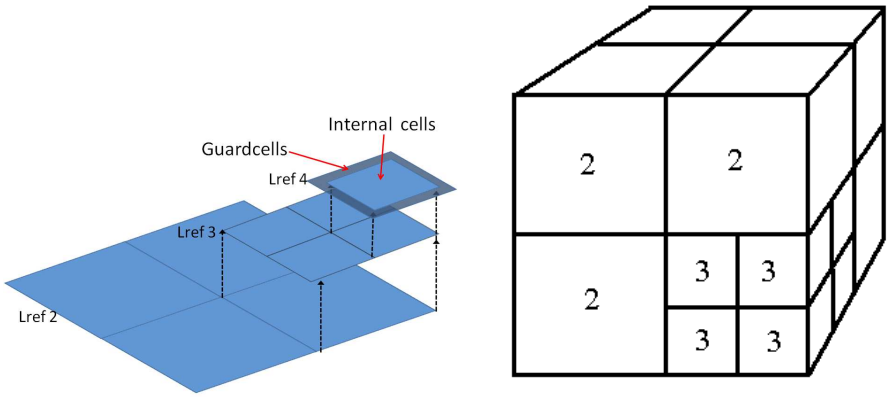

Figure 2.5: Examples of the *FLASH* refinement tree in 2D (left panel) and 3D (right panel) with the refinement 'lref' levels marked. When a block refines it spawns additional blocks in place itself. Each dimension of the new child blocks is half the length of the parent. In 2D this leads to the creation of four new blocks every time we refine a single block (left panel) whilst in 3D this leads to an increase of eight. Those blocks which do not have children at higher levels of refinement are termed "leaf blocks." Data on the highest refinement level is considered the most accurate and up to date. A layer of guardcells are located around each block and contain a copy of neighbouring block data or user defined boundary conditions when at the edge of the simulated volume.

unless stated otherwise.

Compared to a uniform mesh, an adaptive mesh requires additional overhead to support the octree but excels when very large dynamic ranges need to be simulated. In this case regions which are of particular importance such as sharp shock fronts and contact discontinuities can be highly refined by recursively placing down finer blocks. Whereas the memory overhead which would come from highly resolving regions of relative inactivity on a uniform mesh can be reduced by de-refining them. *FLASH* also supports a uniform mesh mode which removes the need for the octree overhead but was found to be of limited use given the high levels of refinement needed for the simulations I present in chapters 4, 5 and 6.

2.6.1 Refinement Schemes

A major advantage of AMR codes is their ability to tailor the refinement based upon any user specified criteria, unlike their SPH counterparts which refine purely on the local density field. A careful choice of refinement scheme is necessary to ensure that we resolve regions with sufficient resolution to accurately solve the hydrodynamic equations whilst not expending resources on over-refining regions when it is not necessary.

The different criteria I used in this thesis include that developed by Löhner (1987). The Löhner criteria works by taking the modified second derivative of a user defined variable and normalising it by the average gradient over a computational cell.

$$E_i = \frac{\mid u_{i+1} - 2u_i + u_{i-1} \mid}{\mid u_{i+1} - u_i \mid + \mid u_i - u_{i-1} \mid + \epsilon[\mid u_{i+1} \mid - 2 \mid u_i \mid + \mid u_{i-1} \mid]} \tag{2.41}$$

Here u_i is the refinement test variable's value in the i^{th} cell. The last term in the denominator is a filter which prevents the refinement of small ripples and can be adjusted by varying the value of ϵ (which is set to 0.01 by default). When extended to three dimensions it has the form:

$$E_{i_1,i_2,i_3} = \left\{ \frac{\sum_{pq} \left(\frac{\partial^2 u}{\partial x_p \partial x_q} \triangle x_p \triangle x_q \right)^2}{\sum_{pq} \left[\left(\left| \frac{\partial u}{\partial x_p} \right|_{i_p+1/2} + \left| \frac{\partial u}{\partial x_p} \right|_{i_p-1/2} \right) \triangle x_p + \epsilon \frac{\partial^2 |u|}{\partial x_p \partial x_q} \triangle x_p \triangle x_q \right]^2} \right\}^{1/2} \tag{2.42}$$

where the sums are over coordinate directions, $p, q = 1, 2, 3$. The resolution is adaptively increased when $E > E_{refine}$ and decreases when $E < E_{derefine}$. The runs presented in this thesis adopt the defaults of $E_{refine} = 0.8$ and $E_{derefine} = 0.2$ unless otherwise stated. This technique has the advantage of being an entirely local calculation to the cells within a block and does not require global communications. It can be applied to any variable or combination of them with no dependence upon systems of units making it an ideal refinement criteria for a variety of different astrophysical phenomena.

In addition to the Löhner criterium, I have also implemented a series of additional refinement criteria tailored specifically to the problems presented in this thesis.

For my runs featured in chapter 4 I implement a refinement criteria based upon the maximum mass that a cell has within a block. This routine calculates the individual masses of each of the cells within a block using the density on the mesh and the known block size and then compares the mass of the most massive cell to a set of criteria. If the mass is above a user defined threshold value ($10^8 M_\odot$ for the gravitational tests shown)

then the region is refined, if the maximum mass drops below a given cutoff ($5 \times 10^7 M_\odot$ in this case) then the region is de-refined. This was used in place of the Löhner criteria which tends to over refine sharp cut-offs at the edge of a given density profile. At these points, the density may be extremely low and there may be very little physical activity, thus resources may be better tailored to those regions where the greatest amount of mass resides. Small fluctuations in these dense regions will yield the greatest deviations to the final King profile and thus resolution is best concentrated there.

In order to fully resolve collapsing regions and prevent artificial fragmentation, the star formation routines in chapter 7 required the code to resolve the Jeans length at all points with a minimum number of cells (up to the maximum permitted level of refinement, *lrefmax*). This required the determination of the Jeans length (equation 1.24) which was then compared to the cell size. If the cells were found to be larger than a given fraction of the Jeans length then their block was refined providing that it would not exceed the maximum imposed level of refinement. If a reduction in the level of refinement would not result in under-resolving the Jeans length then the block was permitted to be de-refined providing that this did not conflict with the requirements of other refinement schemes.

In addition to these refinement schemes, the runs performed in chapters 6 and 7 made use of a minimum density refinement threshold. Below the threshold, refinement was reduced to the minimum, providing that any jumps in refinement between neighbouring blocks did not exceed one. This was imposed in order to remove unwanted refinements in very low density regimes which were outside of the region of interest and had a negligible effect upon core galaxy properties. It also helped control excessive refinements which the Löhner refinement scheme would place down where the galaxy density profile was sharply truncated. Refinements otherwise would have been focussed around the edges of the disk or clustered in regions of relative physical inactivity. I ensured that running with this criteria did not adversely affect simulation results by contrasting them with those run without it. The deviations were found to be negligible for all simulations concerned. Resolution checks in § 6.4.1 show that results from the simulations run with this refinement criteria trace those using a uniform grid of equivalent peak resolution, demonstrating that all physical processes are correctly resolved.

2.6.2 The Cosmology Module

In order to allow the hydrodynamics module to operate without the need for separate modifications when running in a cosmological context, the interpretation of quantities on

the mesh is altered. Coordinates instead correspond to the comoving quantity, $\mathbf{x}$, related to physical distances through $\mathbf{x} = \mathbf{r}/a(t)$ where $a(t)$ is the time dependent cosmological scale factor. The gas velocity, v, is replaced by the comoving peculiar velocity, $\dot{\mathbf{x}}$. The hydrodynamic quantities; density, pressure, temperature and energy are represented by their comoving variants, related to their proper physical quantities (marked with a tilde) as follows,

$$\rho \equiv a^3 \tilde{\rho} \tag{2.43}$$

$$P \equiv a\tilde{P} \tag{2.44}$$

$$T \equiv \frac{\tilde{T}}{a^2} \tag{2.45}$$

$$\varepsilon \equiv \frac{\tilde{\varepsilon}}{a^2} \tag{2.46}$$

This convention allows the equation of state to remain identical to that in a non-expanding universe. The Euler equations for hydrodynamics can thus be written in the form:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \tag{2.47}$$

$$\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v}\mathbf{v}) + \nabla P + 2\frac{\dot{a}}{a}\rho\mathbf{v} + \rho\nabla\phi = 0 \tag{2.48}$$

$$\frac{\partial \rho E}{\partial t} + \nabla \cdot \left[(\rho E + P)\,\mathbf{v}\right] + \frac{\dot{a}}{a}\left[(3\gamma - 1)\rho\varepsilon + 2\rho v^2\right] + \rho\mathbf{v}\cdot\nabla\phi = \mathbf{0} \tag{2.49}$$

$$\frac{\partial \rho\varepsilon}{\partial t} + \nabla \cdot \left[(\rho\varepsilon + P)\,\mathbf{v}\right] - \mathbf{v}\cdot\nabla P + \frac{\dot{a}}{a}(3\gamma - 1)\rho\varepsilon = 0 \tag{2.50}$$

The terms involving $\dot{a}$ are implemented in the `RedshiftHydro` module, separate to the main hydrodynamics routines.

Poisson's equation in comoving form is,

$$\nabla^2\phi = \frac{4\pi G}{a^3}(\rho - \bar{\rho}) \tag{2.51}$$

where $\bar{\rho}$ is the mean comoving matter density. Thus it is necessary for the gravitational routines to explicitly divide the source terms on the right hand side by a^3.

The value for the expansion factor, a, is obtained by solving the Friedmann equation (equation 2.52) given the value for the Hubble constant H_0 at present day (not to be confused with the Hubble parameter $H(t)$) along with the density contrasts for the present

day matter, radiation and cosmological constant. The density contrast being defined as its density relative to the critical density, ρ_{crit}.

$$H^2(t) \equiv \left(\frac{\dot{a}}{a}\right) = H_0^2 \left(\frac{\Omega_m}{a^3} + \frac{\Omega_r}{a^4} + \Omega_\Lambda - \frac{\Omega_c}{a^2}\right) \tag{2.52}$$

$$\rho_{crit} = \frac{3H^2}{8\pi G} \tag{2.53}$$

With $\Omega_c \equiv \Omega_m + \Omega_r + \Omega_\Lambda - 1$ representing the contribution to the overall spatial curvature of the Universe.

2.7 Conclusion

Within this chapter I have described the two dominant approaches to solving the evolution of compressible hydrodynamic fluid flow; the particle approach common to SPH techniques and the Eulerian mesh approach adopted in AMR codes. A clear understanding of these techniques and their differences is essential for the material presented in later chapters. I have outlined the different hydrodynamic equations used in the two different implementations and shown that although derived using different approaches, they are both grounded in sound theoretical concepts. I have shown how the use of a Riemann solver can be used in mesh based codes to determine the post-shock conditions based upon Riemann's problem and shown how SPH tries to mimic the entropy generation in shocks through the use of an artificial viscosity. I have highlighted the potential limitation to the static mesh approach which can lead to Galilean non-invariance through the Riemann solver and I will investigate this issue in more detail in chapters 4 and 6. By comparing the two different means for calculating the gravity in AMR and SPH codes it is clear that SPH typically attains much higher gravitational force resolution than its AMR counterparts but with sufficient resolution they will converge upon the same solution.

Chapter 3

Optimisations: Galaxy Formation in a Flash

3.1 Introduction

Astrophysical simulations involve a multitude of complex inter-dependent interactions that occur over an enormous range of scales. Whilst star formation takes place on scales less than a few parsecs across, the formation of galaxy clusters can span millions. Thus any astrophysical simulation can present a large number of problems to software developers. With most simulations requiring many Gigabytes of memory and trillions of calculations, the demand on hardware is enormous. The advent of parallel supercomputing has made such simulations viable but in order to obtain the best performance from these expensive machines, codes must make use of complex parallel algorithms making the development and testing of these codes often difficult.

Initial work with the *FLASH* code revealed a series of limitations that would render it inappropriate for the more ambitious galaxy formation simulations. Firstly the memory consumption was found to be very high making it impractical to run very large high resolution simulations. This is partly to be expected as AMR codes generally are more memory intensive than SPH codes which do not include the use of "guard cells" (see chapter 2.6) or explicitly track variables such as the temperature. The temperature can instead be derived from the other hydrodynamic variables. However upon close investigation the memory consumption in *FLASH* was found to be far higher than expected based on the number of variables stored in memory, a problem which I address in § 3.2.

Increased memory consumption can degrade performance as distributed computers are divided into a series of "nodes" each of which can possess more than one Central Processing Unit (CPU) and a finite amount of memory. In order to supply larger amounts of memory more nodes must be used, requiring cross processor communication. Although communication performed between CPUs on the same motherboard is relatively inexpen-

sive, communication between different nodes installed on different racks can incur large time penalties. This arises from the small overhead required to establish and begin a communication (often referred to as the "latency") and limitations on the communications bandwidth. Because of this, increasing the amount of memory needed for a simulation, increases the number of nodes it must be divided across and thus the time spent engaging in expensive cross node communication.

Secondly, the gravity solver was found to present a severe bottleneck to any self-gravitating simulation. Previous runs performed by my predecessors attempted to overcome this limitation by restricting their work to phenomena where the change in the gravitational potential over the duration of the simulation was considered negligible (for example Dalla Vecchia et al., 2004). This allows for the adoption of simple analytic potentials, a practice which is still common place amongst *FLASH* users when modelling phenomena such as AGN jets (e.g. Brueggen and Scannapieco, 2009; Roediger et al., 2007) and the ram pressure stripping of disk galaxies (e.g. Roediger and Brüggen, 2007, 2008). However as gravitational collapse is of key significance in the formation of individual galaxies, driving Jeans unstable fragmentation and star formation, a superior solution is required.

In § 3.2 I describe how reductions in the memory consumption have been incorporated into the *FLASH* code without adversely affecting its performance.

In § 3.3 I outline the importance of developing an efficient gravity algorithm and then outline in § 3.4 the drawbacks of simple iterative techniques.

In § 3.5 I describe the operation of the multigrid gravitational solver that comes supplied as one of the default algorithms, along with its limitations. I then present work done in collaboration with Dr. Tom Theuns (University of Durham, University of Antwerp) on the implementation of a Fast Fourier Transform (FFT) technique used to vastly accelerate the gravitational calculation in § 3.6. I perform preliminary tests on the accuracy of the technique in § 3.6.1. In § 3.7 I derive a simple linear interpolation scheme that works with uneven grid spacing and makes use of the known high resolution density field, a tool necessary for the following work in § 3.8. In § 3.8 I carefully analyse the robustness and accuracy of new FFT technique and outline any potential limitations with the scheme. I conclude my findings on the new hybrid FFT-multigrid solver in § 3.8.4.

I then finish by mentioning potential future optimisations that could be made to improve the performance of the *FLASH* code for future astrophysical simulations in § 3.9.

3.2 Memory Reductions

The goal of the Flash Center is to significantly advance understanding of thermonuclear flashes which occur in and around white dwarf and neutron stars. As a direct consequence of this, many modules contain elements pertinent to this original goal such as the ability to simultaneously follow multiple ideal fluids. This necessitates from the need to model the complex equations of state which are believed to govern a neutron fluid at extremely high densities ($\gg 10^6$ g cm^3) and pressures. This requires the use of many separate values for the adiabatic index, γ. The code by default, explicitly tracks two adiabatic indices as part of the method of Colella and Glaz (1985) used to handle general equations of state. The first is necessary to calculate the adiabatic sound speed and is given by

$$\gamma_1 = \frac{P}{\rho}\frac{\partial P}{\partial \rho} \tag{3.1}$$

whilst the second is needed to relate the pressure to the energy

$$\gamma_4 = 1 + \frac{P}{\rho\varepsilon} \tag{3.2}$$

Such a complex equation of state however is not necessary for simulations of the gas present in galaxies, groups and clusters which do not resolve such compact objects. This allows me to remove all references to the two separate adiabatic indices and replace them with a fixed value for the ratio of specific heats appropriate for an ideal gas. This is supplied by the user at run time and by default has the value, $\gamma = 5/3$. This removes the need for two variables to be stored in memory for each cell, cutting both the memory overhead and the storage space required for checkpoint files. By using a single value of gamma it also dramatically simplifies the workings of the Riemann solver. Previously as the value for the ratio of specific heats could vary between cells, it was necessary to combine a series of interpolations and iterations to determine its value at the zone interface. With a single value however there is no variation between cells and thus the Riemann solver can be vastly simplified. As these calculations were performed for each cell, over all directions at every timestep, it has reduced the overhead of the hydrodynamics solver by almost a factor of two. The resulting solver was tested by directly comparing the resulting density, pressure and internal energy profiles from a Sedov blast problem (see 4.4.2 for more details, Sedov, 1959) run with the reduced memory version of *FLASH* and the default release. The results from the two codes were identical to within rounding precision (relative deviations $\leqslant 10^{-16}$).

When closely inspected, it was found that the memory consumption for a single block was much higher than expected based upon the total number of cells and variables stored on the mesh. This results from the static allocation of work arrays which are used during specific routines such as the hydro solver to temporarily store data. The *FLASH* developers initially chose to use static memory allocation for these memory segments as it allows the compiler to perform better memory optimisations. However I opt to dynamically allocate these memory segments upon entry to the necessary subroutines and then de-allocate them upon leaving. This ensures that temporary memory required in routines such as the hydro and gravity solvers does not cumulatively add up, reducing the total simulation memory overhead. Although this prevents the compiler from performing more efficient memory optimisations at compile time, I argue that the reduction in cross processor communication as the code fits onto less nodes greatly outweighs the losses.

An additional memory saving can be made by not storing the temperature. This removes the need to store it for each cell on the mesh, providing a sizable memory saving. Although its removal does imply that any routine requiring its use must calculate it on demand from the other hydrodynamic variables, this occurs at few points throughout the code (e.g. checking the temperature for star formation, see chapter 7) and thus does not detrimentally affect the codes performance. As the temperature is not used when solving the Euler equations in the Riemann solver its absence does not influence the accuracy of the numerical scheme.

3.3 Solving Gravity in a FLASH

In § 1.2 I highlighted the cosmological importance of gravity which drives the collapse of gas to form stars, clusters of stars, galaxies, groups and clusters. As such it is necessary to self-consistently follow gravity in any model that attempts to understand such phenomena. Experience has shown that this is not an easy task however. The force experienced by any particle in space is subject to the attractive force exerted by all other N bodies (see § 1.2) and thus requires an integration that scales as $O(N^2)$. This poor scaling makes tackling large numbers of particles (or cells) increasingly difficult. In addition, the fact that gravity exerts its affect upon all bodies in the simulated volume at all times means that large distributed supercomputer simulations require considerable communication for every timestep. This has led to gravity being a common bottleneck to the run time performance of most astrophysical simulations. To try and overcome this limitation, many

different techniques have been adopted including the SPH "tree" technique as outlined in § 2.5. In mesh based codes a different approach is often implemented in the form of the multigrid technique. The multigrid approach makes use of Poisson's equation (3.3) to solve for the gravitational potential using the density mapped to the mesh. It is a simple step to then take the potential and difference it across cells to produce the gravitational acceleration that material within a cell should experience.

$$\bigtriangledown^2\phi = 4\pi G\rho \tag{3.3}$$

Work with *FLASH* at the ICC revealed that the inbuilt `Poisson solver` module was heavily inefficient and simulations which featured self-gravitation were bottle-necked by the speed of the gravity calculations. As a direct consequence, much of the prominent work done with *FLASH* to date has used simple predetermined gravitational potential fields (recent examples including Roediger and Brüggen, 2007; Roediger et al., 2007; Roediger and Brüggen, 2008; Brueggen and Scannapieco, 2009). Due to the relative limitations of static or analytic potentials when studying more complex structure formation, I have placed a strong emphasis upon developing and validating a more efficient gravity solver for *FLASH*.

3.4 Relaxation Schemes

When applying an iterative scheme to an initial approximation, the result is said to "relax" as it converges almost asymptotically towards the "true" solution. One such iterative scheme (of particular importance to the multigrid technique, see § 3.5) is the Gauss Seidel scheme.

In *FLASH* values are represented by a discrete set of points on a mesh as is shown for the simple one dimensional case in figure 3.1. The gradients at the intermediate points marked $\Delta\phi_1$ and $\Delta\phi_2$ can be obtained by taking the difference in the two adjacent grid point values and dividing by the grid spacing Δx. An approximation to the second derivative of the potential at ϕ_0 can then be obtained by taking the difference between these two adjacent gradients. This allows for Poisson's equation to be discretised,

$$\nabla^2\phi = \frac{\partial^2\phi(0)}{\partial x^2} \cong \frac{\frac{\Delta\phi_2}{\Delta x} - \frac{\Delta\phi_1}{\Delta x}}{\Delta x} = \rho(0) \tag{3.4}$$

thus,

$$\nabla^2\phi \cong \frac{\frac{\phi(+)-\phi(0)}{\Delta x} - \frac{\phi(0)-\phi(-)}{\Delta x}}{\Delta x} = \rho(0) \tag{3.5}$$

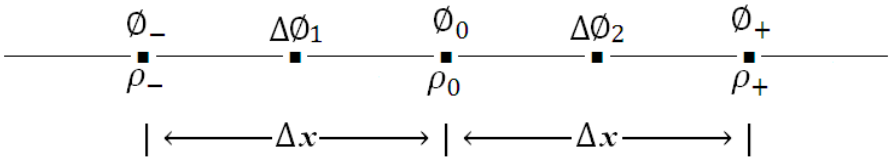

Figure 3.1: Schematic diagram of a FLASH grid section in one dimension. The gradients at the points marked $\Delta\phi_1$ and $\Delta\phi_2$ can be obtained by taking the difference between the two neighbouring grid point values and dividing by the grid spacing Δx. An approximation to the second derivative of the potential at ϕ_0 can then be obtained by taking the difference between these two gradients.

where $\phi(-)$ and $\phi(+)$ are the values of the potential at mesh points immediately before and after the potential at position $\phi(0)$ for which we are evaluating the second differential. This equates to:

$$\frac{\partial^2\phi(0)}{\partial x^2} \cong \frac{\phi(+) + \phi(-) - 2\phi(0)}{(\Delta x)^2} = \rho(0) \tag{3.6}$$

Generalising Poisson's equation to a two dimensional potential and density field, it can be rewritten in the form,

$$\frac{\partial\phi}{\partial t} = \frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial y^2} - \rho \tag{3.7}$$

Thus using discrete differencing, equation (3.7) can be approximated as:

$$\frac{\phi^{n+1}_{j,l} - \phi^n_{j,l}}{\Delta t} = \frac{\phi^n_{j+1,l} + \phi^n_{j-1,l} - 2\phi^n_{j,l}}{(\Delta x)^2} + \frac{\phi^n_{j,l+1} + \phi^n_{j,l-1} - 2\phi^n_{j,l}}{(\Delta x)^2} - \rho_{j,l} \tag{3.8}$$

where j and l are the cell indices on the grid, n represents the step number and Δ represents the cell size (assuming a uniform mesh). This simplifies to

$$\phi^{n+1}_{j,l} - \phi^n_{j,l} = \frac{\Delta t}{(\Delta x)^2}(\phi^n_{j+1,l} + \phi^n_{j-1,l} + \phi^n_{j,l+1} + \phi^n_{j,l-1} - 4\phi^n_{j,l}) - \rho_{j,l}\Delta t \tag{3.9}$$

Forward Centre Time Space (FTCS) differencing (see pg. 827ff. Press et al., 1992) states that the time step in a numerical scheme must not exceed $\frac{\Delta t}{(\Delta x)^2} \leq \frac{1}{2D}$ where D is the number of dimensions, in order for the solution to remain stable and converge. In two dimensions this gives $\frac{\Delta t}{(\Delta x)^2} \leq 1/4$. Therefore setting the iterative step size to its maximum value results in Jacobi's iterative scheme:

$$\phi^{n+1}_{j,l} = \frac{1}{4}(\phi^n_{j+1,l} + \phi^n_{j-1,l} + \phi^n_{j,l+1} + \phi^n_{j,l-1}) - \frac{\rho_{j,l}(\Delta x)^2}{4} \tag{3.10}$$

Equation (3.10) allows us to take an initial approximation to the potential at an initial step, $n = 1$, and update it to that at step $n = 2$ which will more closely approximate the true potential. By running repeated iterations, each time using the solution from the previous step as the input solution to the next, we converge upon the true potential. It may be necessary to perform many iterative steps at each individual timestep in *FLASH* to reach a solution which is sufficiently well converged.

Jacobi's scheme can be improved upon to give the Gauss Seidel scheme (equation 3.11) by using the updated values of ϕ on the right hand side (RHS) of equation (3.10). Thus storing values from previous iterative steps is not necessary and the cell averaging can instead be done in place. Thus the Gauss Seidel scheme often takes a checker-board pattern or "red-black" ordering as the even cells (red tiles on a checker board) are updated and then the odd (the black tiles), repeating this until satisfactory convergence is obtained. This is important for the multigrid technique as it both conserves memory and constitutes a moderate speed up in the performance of the code.

$$\phi_{j,l}^{n+1} = \frac{1}{4}(\phi_{j+1,l}^{n} + \phi_{j-1,l}^{n+1} + \phi_{j,l+1}^{n} + \phi_{j,l-1}^{n+1}) - \frac{\rho_{j,l}(\Delta x)^2}{4} \tag{3.11}$$

Simple to implement, iterative schemes like the Gauss Seidel scheme described tend to remove small scale fluctuations quickly. Unfortunately, as information about neighbouring cells is only passed across one cell at a time, they achieve large scale convergence very slowly. As the mesh size increases, the number of iterations needed to pass information from one side of the mesh to the other increases. This reduces the convergence rate. This scheme is therefore poorly suited to obtaining an accurate potential on a mesh with a uniform level of refinement. However as I will show in § 3.5, the multi-layered properties of AMR codes allows them to accelerate the process by solving for different scale deviations on different refinement levels of the mesh, greatly accelerating the process.

3.5 The Multigrid Solver

The multigrid technique acknowledges the fact that iterative schemes converge very slowly on the large scale (low frequency) terms but are quick to remove small scale (high frequency) deviations between the actual and current potential.

As large scale terms on a fine mesh appear as small scale terms on a coarser mesh, it is possible to solve for these large scale terms on the coarser blocks and then prolong the correction required to higher levels of refinement. The application of iterations at

each higher level can then relatively quickly remove successively smaller scale deviations between the current and actual potential. This allows all spatial scales to converge at similar rates.

The multigrid technique makes use of four fundamental quantities to determine the actual potential. For a given true potential, ϕ, we start with an initial approximation, $\tilde{\phi}(x)$. A measure of the deviation between the actual and current potential is then defined in terms of a "residual",

$$\begin{aligned} R(x) \equiv & \quad \nabla^2\phi(x) - \nabla^2\tilde{\phi}(x) \\ = & \quad \alpha\rho(x) - \nabla^2\tilde{\phi}(x) \end{aligned} \tag{3.12}$$

where the term $\alpha\rho(x)$ is the "source" ($\alpha = 4\pi G$ in the case of gravity), $S(x)$. The residual satisfies the equation

$$\nabla^2 C(x) = R(x) \tag{3.13}$$

whose solution $C(x)$ is the "correction"

$$C(x) \equiv \phi(x) - \tilde{\phi}(x) \tag{3.14}$$

This means the multigrid solver must store an additional four variables in memory. The differential operator $\nabla^2\phi_{ijk}$ is descretised on the mesh at a given level of refinement, l, in the same fashion as in § 3.4:

$$\nabla^2\phi^l_{ijk} = \frac{\phi^l_{i+1,j,k} + \phi^l_{i-1,j,k} - 2\phi^l_{i,j,k}}{\Delta^2_{i,l}} + \frac{\phi^l_{i,j+1,k} + \phi^l_{i,j-1,k} - 2\phi^l_{i,j,k}}{\Delta^2_{j,l}} + \frac{\phi^l_{i,j,k+1} + \phi^l_{i,j,k-1} - 2\phi^l_{i,j,k}}{\Delta^2_{k,l}} \tag{3.15}$$

where *i, j, k,* represent the cell indices and Δ is the linear cell size in the respective direction. At each step the multigrid solver is initialised as follows:

1. Upon first calling the solver we zero the solution, residual and correction across the mesh, but for subsequent calls we use the previous solution as an initial guess.

2. Compute the residual, $R^l_{ijk} = S^l_{ijk} - \nabla^2\phi^l_{ijk}$, on all leaf blocks. Compute the discrete L2 norm of the residual and the source (e.g. Source L2 norm $= |\rho|^2 = \langle\rho|\rho\rangle$).

The multigrid solver then performs the following V-cycle as it traverses down and then back up the refinement tree:

1. Zero the correction on the highest level of refinement, $C_{ijk}^{l_{max}}$.
2. For each level, l, from l_{max} down to $l = 2$:
   (a) Copy the solution ϕ_{ijk}^{l} to a temporary variable τ_{ijk}^{l}
   (b) Zero the correction on the level below, C_{ijk}^{l-1}.
   (c) Apply the relaxation operator several times to the solution on level l: $\nabla^2 C_{ijk}^{l} = R_{ijk}^{l}$
   (d) Add correction at the current level, C_{ijk}^{l} to the potential, ϕ_{ijk}^{l}
   (e) Compute the residual of the correction equation on all blocks on level l. Restrict this residual to the level below; R_{ijk}^{l-1}.
   (f) Compute residual of the source equation (3.12) on all leaf blocks of level $l-1$ and leave result in R_{ijk}^{l-1}.
3. Solve the correction equation on the coarsest level, applying external boundary conditions. Correct the solution on the coarsest level.
4. For each level l from $l = 2$ back up to l_{max}:
   (a) Prolongate the correction from the level below and add result to C_{ijk}^{l}
   (b) Replace R_{ijk}^{l} with the residual of the correction equation.
   (c) Zero second temporary variable, ν_{ijk}^{l} on levels $l-1$ and l.
   (d) Apply the relaxation operator several times $\nabla^2 \nu_{ijk}^{l} = R_{ijk}^{l}$
   (e) Add ν_{ijk}^{l} to the correction C_{ijk}^{l}
   (f) Copy τ_{ijk}^{l} back into ϕ_{ijk}^{l} on all leaf blocks.
   (g) Add the correction C_{ijk}^{l} to the solution ϕ_{ijk}^{l} on all leaf blocks.
5. Compute the residual $R_{ijk}^{l} = S_{ijk}^{l} - \nabla^2 \phi_{ijk}^{l}$ on all leaf blocks. Compute the discrete L2 norm of the residual.

At the end of the V-cycle we compare the ratio of the residual and source norms to a defined threshold value. If the ratio drops below the threshold (default 1×10^{-6}) or we exceed a given number of cycles (default = 100) then we accept the solution, if not then we repeat the V-cycle.

As the multigrid technique uses the previous solution for the next timestep's initial guess, it operates most efficiently when the potential fluctuates very little between steps. In addition, increasing the number of refinement levels in a simulation, increases the number of intermediate steps and therefore iterations within the V-cycle. Thus as the total number of refinement levels increases so too does the run time. Therefore the multigrid solver works optimally for slowly evolving systems with a limited number of refinement levels. However the Universe is characterised as spanning an immense range of scales with shocks and contact discontinuities being a common feature of galaxy forming processes. Very sharp density contrasts can cause the relaxation technique to converge very slowly, even with the increased rate of convergence the multigrid offers for large scale terms. This was evident in the cluster tests presented in chapter 4 (§ 4.5) where the sharp truncation of the King density profile required many V-cycles to accurately converge upon a solution. Small changes in the density profile would lead to either expansion or further collapse, quickly altering the gravitational potential and rendering the solution of little use at the next timestep. A combination of this sharp density change at the edge of the profile and the high number of refinement levels needed to resolve the core (11 in total) produced a crippling timestep of 31 minutes for gravity alone. Clearly this is impractical for simulations that require many tens of thousands of timesteps and thus a superior gravity solver was required to complete these tests. To this end in collaboration with Dr. Tom Theuns, the FFT gravity solver was implemented into *FLASH*.

3.6 Application of an FFT technique

The need to perform multiple V-cycles along with many iterations at each refinement level presents a significant problem to the multigrid technique. However the need to perform many V-cycles can be entirely removed if instead we are able to determine the exact potential on a given level. The resulting potential at this *"pm level"* could then be prolonged to higher levels of refinement and relaxed to compensate for small scale fluctuations that were previously unresolved. The potential on levels below the *pm level* could be obtained through restriction. Overall such a technique would remove the need for many iterations at many levels and result in a dramatic speed up to the gravity calculation.

To achieve the accurate determination of the potential at a level of refinement higher than the base refinement level, we can make use of Fast Fourier Transforms (FFT) and

the application of a Green's function. A Green's function is an integral Kernel used to solve inhomogeneous differential equations subject to boundary conditions. In this section I outline the principles of the Fast Fourier Transform technique and evaluate its performance in § 3.8 using both smooth analytic and random density functions.

Fourier transforms allow for a function that varies in space to be replicated using a series of separate harmonic frequencies which when combined would form a wave with the same spatial distribution as the input function. Thus the Fourier transform of a function gives information on the power in each harmonic frequency. These include both high frequency components, corresponding to small scale fluctuations in the input function and the low frequency components which represent the large scale fluctuations.

For a continuous function $f(x)$, the Fourier transform is given by:

$$\Im(f(x)) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(x) e^{-ikx} dx = g(k) \tag{3.16}$$

where the inverse Fourier transform is:

$$\Im^{-1}(g(k)) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \Im(k) e^{ikx} dk = f(x) \tag{3.17}$$

However when the function $f(x)$ is sampled by taking N points and mapping them to the grid, the Discrete Fourier Transform (DFT) describing it becomes:

$$\Im(f(x)) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) e^{-i2\pi ux/N} = g(u) \tag{3.18}$$

with an inverse DFT of:

$$\Im^{-1}(g(u)) = \sum_{u=0}^{N-1} \Im(u) e^{i2\pi ux/N} = f(x) \tag{3.19}$$

Thus by this convention, $k \equiv (2\pi u/N)$. Note that the change in sign of the exponent in the DFT compared to its inverse is only a convention, as is the normalisation factor of $1/N$ which could instead be written in terms of a factor of $\frac{1}{\sqrt{N}}$ in front of both the forward and inverse DFT. In numerical terms however the single factor of $1/N$ is more efficient than multiple fractional powers after both the DFT and its inverse.

I begin by considering the discrete difference approximation to Poisson's equation,

$$\frac{\partial^2 \phi(x)}{\partial x^2} \cong \frac{\phi(x+\Delta) + \phi(x-\Delta) - 2\phi(x)}{\Delta^2} = -\rho(x) \tag{3.20}$$

Now as the Fourier transform of the potential $\phi(x)$ can be written:

$$\phi(x) = \sum_k \phi(k) e^{ikx} \tag{3.21}$$

then this allows the Fourier transform of equation (3.20) to be written in the form

$$\frac{1}{\Delta^2}\sum_k \phi(k)[e^{ik(x+\Delta)} + e^{ik(x-\Delta)} - 2e^{ikx}] = -\sum_k \rho(k)e^{ikx} \tag{3.22}$$

which simplifies to give

$$\frac{1}{\Delta^2}\phi(k)[e^{ik\Delta} + e^{-ik\Delta} - 2] = -\rho(k) \tag{3.23}$$

Thus we can now relate the potential in Fourier *"k"* space to the density in k space;

$$\phi(k) = -\frac{\Delta^2}{(2\cos(k\Delta) - 2)}\rho(k) \tag{3.24}$$

This allows us to define a term known as the Green's function ($G(k)$, given for the 1D case in equation 3.25) which directly relates the Fourier transformed density field to the gravitational potential in Fourier space. Thus by taking the Fourier transform of the density on the mesh and convolving it with the Green's function, the resulting product can be inverse Fourier transformed to give the exact gravitational potential on the mesh. The gravitational acceleration can then be obtained by differencing the potential on the mesh.

$$G(k) = \frac{-\Delta^2}{2\cos(k\Delta) - 2} \tag{3.25}$$

Initially the use of large Fourier transforms would not offer any advantages over the multigrid technique as they normally involve taking a matrix vector product which scales as $O(N^2)$. However Fast Fourier Transforms (FFT) operate by breaking down the DFT of size $N = N_1N_2$ into many smaller sized DFTs of size N_1 and N_2 with $O(N)$ multiplications by complex roots of unity. This enables them to scale as $O(Nlog(N))$ making them highly scalable and computationally appealing since they lack the need for many V-cycles inherent in the standard multigrid technique.

Since FFTs work by splitting the transform into two pieces of size $\frac{N}{2}$ at each step, they are often optimised to work with powers of two. Although they can be modified to perform with arbitrary powers at the expense of lower performance I choose to restrict my simulations to use blocks were the number of cells is equal to a power of two.

3.6.1 Testing the Green's Function and FFT

A simple test to confirm that the Green's function allows us to obtain the analytically expected potential can be performed by considering an initial input density profile. For instance using a simple sine wave density profile of the form

$$\rho(x) = A\sin(2\pi k_0 x) \tag{3.26}$$

the corresponding analytic potential from Poisson's equation (dropping the factor of $4\pi G$ for convenience) can be obtained by integrating the density field twice to obtain:

$$\phi(x) = \frac{-A}{4\pi^2 k_0^2}\sin(2\pi k_0 x) = \frac{-1}{4\pi^2 k_0^2}\rho(x) \tag{3.27}$$

As the Taylor series expansion of $\cos(x) = 1 - \frac{x^2}{2} + \frac{x^4}{4!} - \frac{x^6}{6!} + ...$, then by truncating it to include only the significant first two terms and substituting into the Green's function derived in equation (3.25), allows us to express the Green's function as

$$G(k) \cong \frac{-\Delta^2}{2(1 - \frac{(2\pi k_0 \Delta)^2}{2}) - 2} \tag{3.28}$$

which simplifies to give

$$G(k) \cong \frac{-1}{4\pi^2 k_0^2} \tag{3.29}$$

As the inverse Fourier transform of a Fourier transformed function, $f(x)$, returns the original function, i.e.

$$\Im^{-1}(\ \Im(f(x))\ ) = f(x) \tag{3.30}$$

then this implies that as the Green's function in this case is a constant

$$\phi(x) = \Im^{-1}(\ G(k)\Im(\ A\sin(2\pi k_0 x))\ ) = \frac{-1}{4\pi^2 k_0^2} A\sin(2\pi k_0 x) = \frac{-1}{4\pi^2 k_0^2}\rho(x) \tag{3.31}$$

Thus we are able to reconcile the analytically expected result with that obtained when the density field in Fourier space is multiplied by the Green's function and then inverse Fourier transformed. This confirms that the potential can be accurately determined through the use of FFT techniques and a suitable Green's function.

To ensure that the technique holds up when using discrete numerical operations including the use of Fast Fourier techniques, routines were written in IDL to perform the conversion between the density and potential fields using the Green's function and IDL's inbuilt FFT routine.

Initially I tested the ability of IDL's FFT to perform a series of Fourier transforms on a given set of analytic functions with known solutions. Thorough testing using Sine, Cosine, Dirac Delta, boxcar and comb functions yielded the desired results exactly as would be expected, ensuring the numerical FFT works as expected. Furthermore, when a given function was Fourier transformed and then converted back using an inverse Fourier transform, the difference between the initial and final function was less than $\sim 10^{-15}$. As double precision numbers are able to store up to 16 significant figures then this is in agreement with expected cumulative numerical rounding errors. I can therefore confirm

that IDL's FFT is able to accurately Fourier transform and inverse Fourier transform a given input function without substantial degradation of the input function.

Next, when I Fourier transformed a test high resolution density field, multiplied the result with the Green's function and then performed an inverse Fourier transform, the result agreed well with analytical predictions. Initial results yielded root mean squared (RMS) deviations of order 10^{-14} which decreased as the resolution increased. These low level deviations are relatively minor and most likely arise due to the finite number of frequencies available to the FFT with which to model some of the more complex functions on the grid such as the boxcar function.

From these simple tests I can confirm that at a given level of refinement the use of the Green's function and FFT allows for the potential to be obtained to a high degree of accuracy. I will now expand this investigation to test the accuracy of the technique when applied to a multigrid system in which there can be many levels of refinement and for which the potential cannot necessarily be obtained on the highest level owing to memory restrictions.

3.7 Linear Interpolation Scheme with Uneven Grid Spacing and Density Correction Factor

The adaptive mesh in *FLASH* requires the potential to be determined at several different levels of resolution. However memory limits often restrict the maximum size of the FFTW that can be performed. To overcome this it is necessary to prolong and restrict the potential to different levels of refinement. The restriction and prolonging of cells introduces an offset in the cell centres at different levels (see figure 3.2). Hence each point on the low resolution grid is out of phase with the higher resolution grid by either 1/2 or 3/2 cell spacings on the fine mesh. Therefore an interpolation scheme which can deal with non-uniform spacing is required.

In the following sections I investigate the effect that different interpolation schemes have upon the potential. As an intermediary between the simple linear and more complex quadratic interpolation schemes, I derive below an interpolation scheme which combines a linear interpolation of the potential with the benefit of the multigrid's high resolution density field. If such a scheme provides adequate accuracy but at a lower computational cost than the more expensive quadratic scheme used in PPM, it may be worth implementing it into the gravity module.

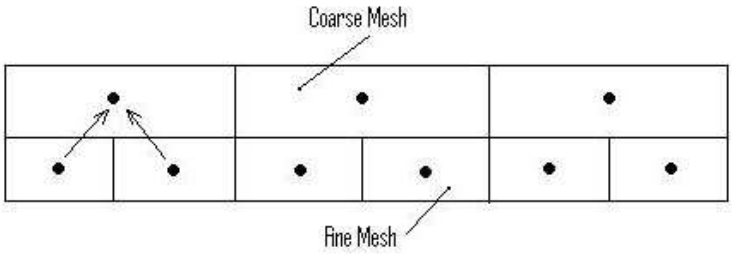

Figure 3.2: Schematic diagram of a FLASH grid section ($1D$) displaying both the high and low resolution grids. A change in refinement brought about by a restriction or prolongation operation introduces a shift in the position of the new cell centres relative to the previous ones. Two fine cells lie between each pair of adjacent coarser grid cells. Interpolating functions need to be able to accommodate this non-uniform spaced mesh.

Considering equation (3.5) with uneven grid spacing as shown in figure 3.3, we can derive the relation

$$\frac{\partial^2\phi}{\partial x^2} \cong \frac{\frac{\phi(+)-\phi(0)}{(\Delta x-\alpha)} - \frac{\phi(0)-\phi(-)}{\alpha}}{\frac{\Delta x}{2}} \tag{3.32}$$

which simplifies to give

$$\phi(0) = \frac{\alpha}{\Delta x}\phi(+) + \frac{(\Delta x-\alpha)}{\Delta x}\phi(-) - \frac{\alpha(\Delta x-\alpha)}{2}\rho(0) \tag{3.33}$$

The first two terms on the RHS correspond to a linear interpolation between the two adjacent points whilst the third term is a correction factor applied using the high resolution density field. The phase difference between the centres of the high and low resolution cells is compensated for by the factors $\frac{\alpha}{\Delta x}$ and $\frac{(\Delta x-\alpha)}{\Delta x}$.

3.8 Testing the Multigrid Technique

Now that I have confirmed the potential can be obtained upon a given mesh using a suitable Green's function and FFT, it is necessary to extend this work to ensure that restriction and prolongation operations in an AMR code can reliably extend the potential to other refinement levels. In order to simulate the processes undertaken in the multigrid method, a high resolution $1D$ grid consisting of 128 cells was generated, with a series of different test functions applied across it.

I begin by generating a high resolution sinusoidal density field (equation 3.26), ρ_{high}, across the 128 point fine mesh. ρ_{high} was then restricted by averaging every pair of values to obtain $\rho_{restricted}$. $\rho_{restricted}$ was then Fourier transformed, multiplied by the

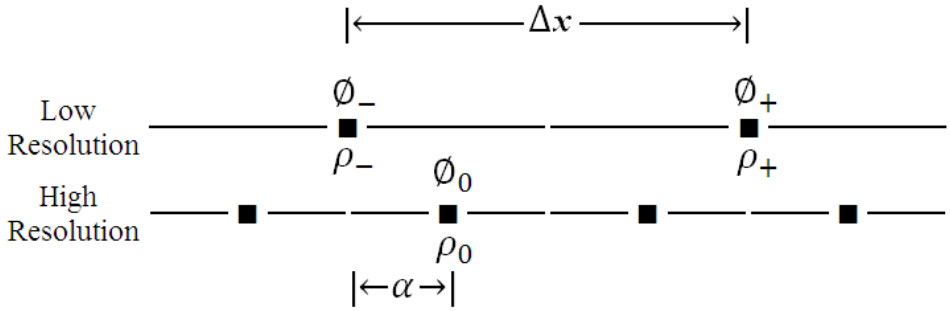

Figure 3.3: Schematic diagram of a FLASH grid section incorporating uneven spacing which arises when restricting or prolonging data between levels of refinement. The values of ϕ_- and ϕ_+ (and their respective densities) reside upon the low resolution mesh whilst the value of ϕ_0 which we are trying to find lies on the higher resolution mesh. The cell centres are marked with black squares. Δx corresponds to the low resolution cell size while α represents the offset between the high and low resolution cells brought about when high resolution cells are averaged to produced the low resolution grid (see figure 3.2).

Green's function and inverse Fourier transformed to yield the low resolution potential ϕ_{low}. ϕ_{low} was then interpolated up to give $\phi_{prolonged}$ and twenty Gauss Seidel iterations were performed on the resulting function to yield $\phi_{relaxed}$. $\phi_{prolonged}$ and $\phi_{relaxed}$ can also be translated back, using either the FFT or the ∇^2 operator to give the density field that would be obtained from those given functions.

As § 3.6.1 confirmed that the use of a Green's function and FFT produces an accurate potential, I adopt the direct conversion of the high resolution density field into the potential, ϕ_{high}, as proxy to the true analytic solution. Ideally any robust numerical scheme which restricts or prolongs the results should still trace these high resolution potentials.

As there are a series of variables which may affect the performance of the multigrid technique, I explore each of them in turn. I begin by testing the accuracy of the scheme using a simple, smooth analytic function for which an analytic potential is known. As there are two ways of representing the data on the mesh, the actual value that the density function holds at a given mesh point and the mean value across the cell, I will simultaneously investigate the results for both schemes in § 3.8.1. I will then investigate the effect of changing the interpolation scheme in § 3.8.2 and finally I will test the new hybrid multigrid scheme with randomly generated density profiles in § 3.8.3.

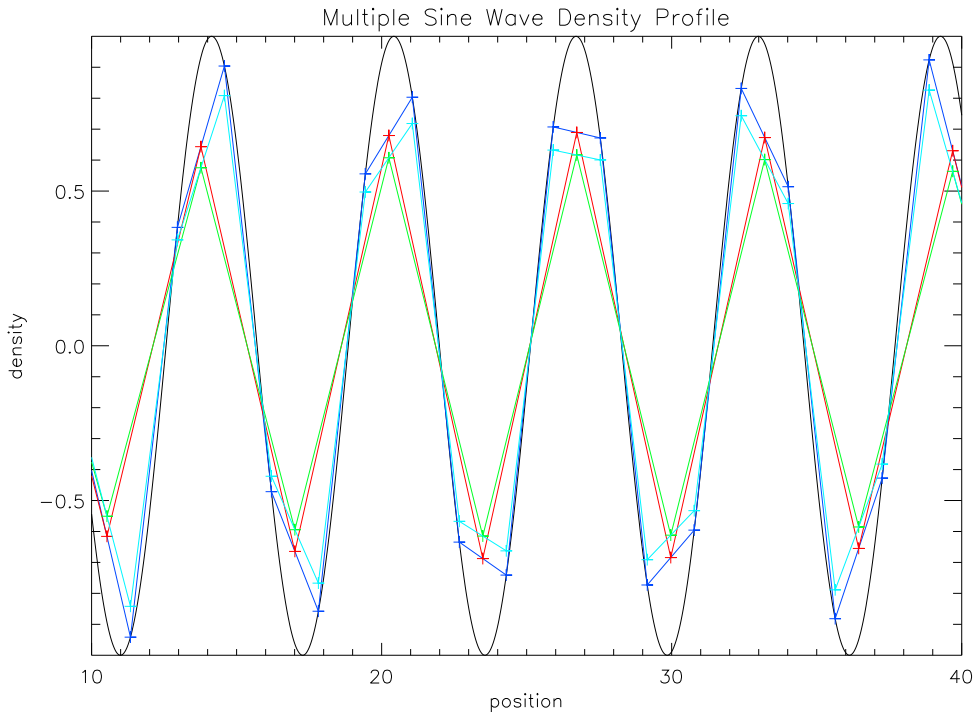

Figure 3.4: Plot of the sinusoidal density field. The analytic density field is shown in black. Density fields using the mean analytic values (light blue) and the actual values (dark blue) are shown. The equivalent restricted density fields are shown, ρ_{low}^{mean} in green and ρ_{low}^{actual} in red. As expected the different conventions for assigning values lead to slightly different amplitudes for the generated function with the mean values being slightly lower than the actual values. Although this changes the magnitude of the function, it does not change the phase or general profile, so the resulting functions should be comparable. I therefore expect the two to maintain the relative amplitude difference when converted into potentials using the FFT technique.

3.8.1 Mean Vs. Actual Values On the Grid

As the variables in a *FLASH* cell represent the mean value across the cell, whilst my analysis so far has assumed the FFT interprets them as actual values, it is constructive to investigate the difference this makes. As can be seen from the plot of the amplitude of the density field against position in figure 3.4, the choice between actual and mean density values can make a modest difference in the magnitude of the values passed to the FFT. As expected the high resolution "mean" densities, ρ_{high}^{mean}, shown in light blue are naturally lower than the "actual" density values, ρ_{high}^{actual} shown in dark blue. The analytic density

field is plotted in black for comparison. Despite the slight differences in their amplitudes, the two high resolution density profiles have the same phase and general profile and thus I would expect them to behave the same.

When restricting the high resolution density fields, the ρ_{low}^{mean} (green) and ρ_{low}^{actual} (red) values are found to accurately interpolate between the pairs of their respective high resolution density profiles, thus the relative difference in profile amplitudes is maintained. This allows a high resolution density or potential field to be restricted to a lower level of resolution and yet still represent the original spatial function it was derived from. When Fourier transforming to obtain the potentials, both low resolution density functions should return low resolution potentials with the same phase and relative amplitudes. This is clearly observed to be the case for figure 3.5 in which the low resolution potentials; ϕ_{low}^{mean} (green) and ϕ_{low}^{actual} (red) are plotted as a function of position. The corresponding high resolution potentials as obtained from direct Fourier conversion of the high resolution density fields; ϕ_{high}^{mean} (light blue) and ϕ_{high}^{actual} (dark blue) are plotted for comparison and are in excellent agreement with the analytic potential as would be expected. The two low resolution potentials should interpolate their corresponding high resolution potentials as in the density field case but this apparently does not occur. Both display a much greater amplitude than their expected ϕ_{high} equivalents. This would suggest that the FFT has recreated their wave forms with values of k_0 that are smaller than they should be analytically or alternatively that the Fourier transform has interpreted the input single sine wave function as a composition of numerous different waves of differing spatial frequencies. This may not be a problem when dealing with more complex signals which require multiple waveforms to reconstruct them and thus I will investigate this effect in more detail for random density fields in § 3.8.3.

Figure 3.6 shows the low resolution potentials in figure 3.5 after being interpolated up to the higher level of refinement using the simple linear interpolation scheme with density correction factor (equation 3.33). The resulting potential using the mean values, $\phi_{prolonged}^{mean}$, is shown in green and that using actual values, $\phi_{prolonged}^{actual}$, is shown in red. Clearly there still exists a large unexpected deviation between the amplitude of the expected and obtained potentials with an RMS deviation between ϕ_{high}^{mean} and $\phi_{prolonged}^{mean}$ of 0.61. Although the potentials do not accurately reflect their expected values, both mean and actual value implementations possess the same problem. Thus it is unlikely to arise from the interpolation scheme although I investigate the effect different interpolation schemes can have in § 3.8.2.

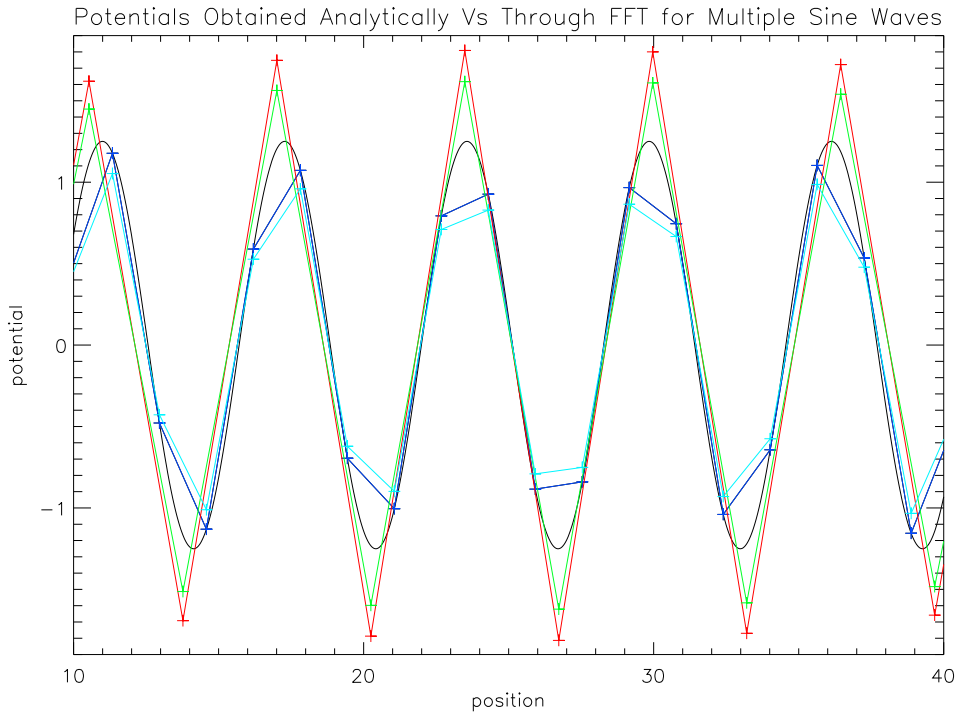

Figure 3.5: Plot of the potentials obtained through the Fourier technique. The high resolution potentials obtained directly from the high resolution density fields using the Fourier technique; ϕ_{high}^{mean} (light blue) and ϕ_{high}^{actual} (dark blue) are shown for comparison and show excellent agreement with the analytic profile shown in black. The two high resolution potentials transform exactly as would be expected, maintaining their relative phase and amplitudes as before in figure 3.4. The low resolution potentials, ϕ_{low}^{mean} (green) and ϕ_{low}^{actual} show the same relative amplitude and phase to each other as expected, however their amplitudes are much higher than what would be expected based on the high resolution potentials. This would appear to indicate that the Fourier transform has incorrectly interpreted the pure sine wave input function as a more complex series of multiple waves, resulting in a different potential output to that which we would expect.

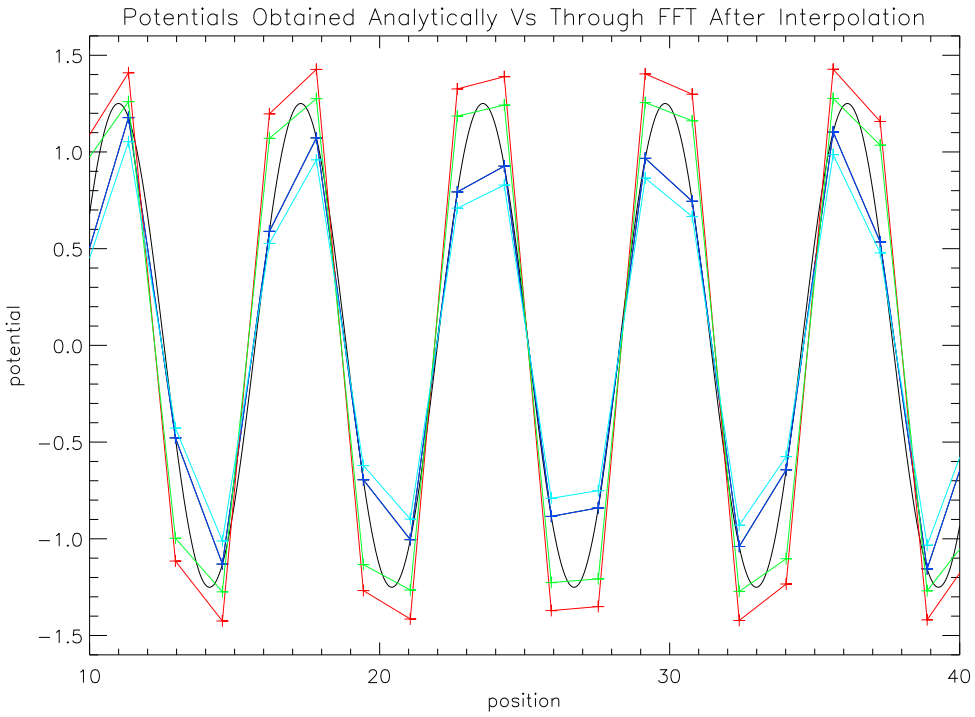

Figure 3.6: Plot of the prolonged potential fields. The analytic profile is shown in black. The un-prolonged high resolution potentials as obtained directly from the high resolution density fields using the Fourier technique; ϕ_{high}^{mean} (light blue) and ϕ_{high}^{actual} (dark blue), are shown for comparison. The prolonged potentials as obtained from interpolating the low resolution potentials; $\phi_{prolonged}^{mean}$ (green) and $\phi_{prolonged}^{actual}$ (red) are plotted. The interpolation has produced consistent results between the two prolonged potentials which maintain the same relative amplitude differences and phases. However they still fail to agree with the expected high resolution potentials which have much smaller amplitudes and agree with the analytic profile.

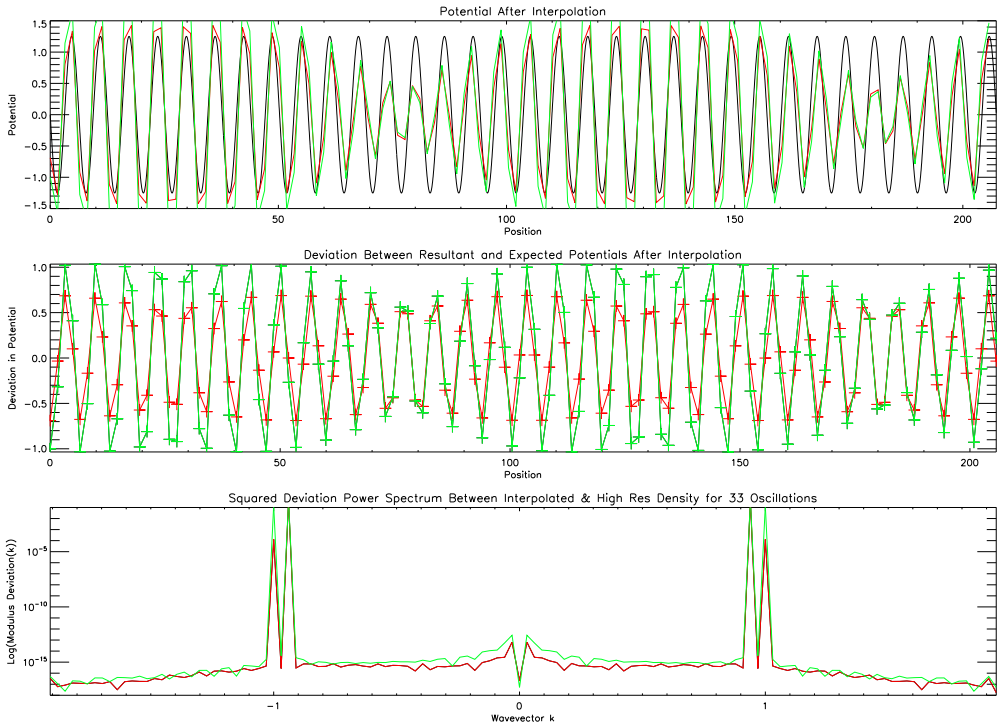

Figure 3.7: The top frame displays the analytic potential as a function of position in black. $\phi^{mean}_{prolonged}$ (green) and $\phi^{actual}_{prolonged}$ (red) prolonged potentials are shown for comparison. The centre frame shows the deviations between the prolonged potentials and their respective high resolution equivalents. The bottom frame corresponds to the power spectrum of the deviations in the potential shown above in the central frame. Large scale fluctuations correspond to low values of k and small scale deviations are represented by large values of k. The power spectrum is symmetrical about the origin. Interestingly despite the input wave function being a pure harmonic sine wave, there are two distinct peaks in the deviation power spectra. These clearly show that the potential is not modelled as a single wave and thus the FFT must have represented the input density fields with several wave functions instead of the one required. This has introduced large scale deviations between ϕ_{high} and $\phi_{prolonged}$. The power scale upon which these deviations occur in both the mean and actual valued potentials is the same implying that the choice of value type does not influence the results.

By taking the Fourier transform of the deviation between each potential and the analytical potential (figure 3.7), it is possible to analyse the phase space distribution of the fluctuations and thereby determine upon which scales the deviations have arisen.

The first frame displays the potential as a function of position across the full mesh with $\phi^{mean}_{prolonged}$ plotted in green, $\phi^{actual}_{prolonged}$ in red and the analytic potential in black. The central frame shows the deviation between $\phi_{prolonged}$ and ϕ_{high} for both the mean values (green) and actual values (red) as a function of position. The bottom frame shows the Fourier transform of the deviation power spectrum (equation 3.34) in logarithm plotted against the wave-vector k.

$$\Im(f(x)) = \frac{1}{\sqrt{2\pi}} \sum_{x=0}^{N-1} f(x) e^{-i2\pi ux/N} = g(u) \tag{3.34}$$

Here $f(x)$ represents the deviation in the potential sampled using N points. $g(u)$ represents the power spectrum with wave vectors $k \equiv (2\pi u/N)$. The central, zero wave vector k_0 represents fluctuations on the largest scale whilst increasingly larger values in *k-space* represent higher frequency terms that correspond to fluctuations on smaller scales. The power spectrum by definition is symmetrical about the origin.

Interestingly despite the input wave function being a pure harmonic sine wave, there are two distinct peaks in the deviation power spectra. These clearly show that the potential is not modelled as a single wave which for some reason has a greater amplitude than expected. Instead the FFT must have represented the input density field with more than the one wave function required. This has introduced large scale deviations between the expected high resolution potentials and the potentials obtained by prolonging low resolution potentials. The power scale upon which these deviations occur in both the mean and actual valued potentials is the same, implying that the choice of value type does not influence the deviation power spectrum. In order to emphasise this large scale deviation I increase the number of periods of the input sine wave represented on the grid whilst maintaining the same effective resolution per cycle. The second panel highlights how the large scale variation in the deviation can be clearly seen for both the mean and actual valued potentials, confirming what is shown in the Fourier power spectrum. Although most of the power in the deviation power spectrum is located at around $k \approx 1$, equivalent to variations over a few cells, there is some power - albeit small in the lowest frequency component.

Since the restriction operator will smooth over small scale structure, it is predictable

that there will be small scale deviations in the recovered potential, $\phi_{prolonged}$, which will show up at high frequencies in the power spectrum. In order to remove these deviations in the potential it is necessary to perform a series of iterations upon the prolonged potentials. Reassuringly, after twenty Gauss Seidel iterations, the prolonged potentials reduce to give RMS deviations of the order of $\sim 10^{-14}$, almost in line with permissible rounding errors. At this point both the mean and actual value based potentials display strong convergence upon their ϕ_{high} equivalents. This provides strong evidence that the multigrid technique works well with such smooth functions regardless of the choice of mean or actual based values. Considering that as you tend towards an infinitely fine grid the mean values will tend towards the actual values, this appears to be reasonable.

The large scale deviation in the prolonged potential in figure 3.7 on the other hand was unanticipated. Ideally I would not expect the restriction operator to interfere with the low frequency, large scale components of the signal, especially as the low resolution density field was shown to accurately interpolate the high resolution density field. In order to determine whether these large scale deviations have been removed from the relaxed potential, $\phi_{relaxed}$, and if so to what extent, I plot the equivalent plot to figure 3.7 using the final relaxed potentials after twenty Gauss Seidel iterations, $\phi_{relaxed}$.

The first window in figure 3.8 displays $\phi_{relaxed}$ as a function of position across the full simulation volume, with the analytic solution shown in black. Only the mean based values are visible (green) since the difference between the mean and actual based potentials is so small. The second window shows the deviation between $\phi_{relaxed}$ and ϕ_{high} using both the mean values (green) and actual values (red) as a function of position. A very clear, almost sinusoidal large scale deviation is still visible for both conventions, although it is vastly smaller than the fluctuations upon the small scales which previously dominated the signal.

The final window shows the Fourier transform of the deviation power spectrum in logarithm plotted against the wave-vector k. The large peaks which corresponded to the small scale fluctuations visible in figure 3.7 have now been removed, however the large scale deviations at low frequency are still relatively undiminished. Further relaxation suggests that of the order of ten thousand iterations would be required to converge on the "true" solution. Given that these large scale deviations currently lie near to the limit of numerical precision I conclude that such an excessive number of iterations is not warranted and that the FFT technique can produce exceptionally accurate results providing that sufficient iterations are undertaken to remove small scale deviations that otherwise present

significant errors. I can also conclude through these numerical experiments that the choice of mean or actual values ultimately yields equivalent results providing that the resolution of the mesh is adequate to sample the input density function properly.

3.8.2 Dependence upon the Choice of Interpolation Scheme

As the prolongation process requires the use of an appropriate interpolation technique I will now investigate the effect of different interpolation schemes upon the accuracy of the prolongation step. I will also investigate whether they have any effect upon the large scale deviations that were detected in § 3.8.1 as this could hold significant consequences for cosmological simulations which aim to trace large scale structure formation.

I use the same initial high resolution density field as in § 3.8.1 based on the actual values. This was then restricted and Fourier transformed to yield ϕ_{low}. The effect of prolonging ϕ_{low} was investigated with a series of interpolation schemes as shown in Figure 3.9. The prolonged potential as obtained using a simple linear scheme (yellow), a linear scheme with density correction factor (orange, see § 3.7), a linear scheme with weighted density correction factor (red) and IDL's default quadratic interpolation scheme (green), is plotted against position on the grid. The analytic potential is shown in black and ϕ_{high} is plotted in dark blue for comparison.

The linear interpolation scheme with weighted density correction factor is a modified version of equation (3.33) which attempts to compensate for the large scale structure observed in § 3.8.1. It was obtained by determining the difference between the density correction factor applied and that required to correctly interpolate the low resolution potential gained from restricting a single period of the sine wave over the 128 cell grid.

The results as can be seen in figure 3.9 demonstrate that all of the interpolation schemes show poor initial reproduction of ϕ_{high} in line with the previously discussed results in § 3.8.1. This is in part due to the restriction operator which smoothes over small scale power but significantly all of the interpolation schemes give prolonged potentials that all have much greater amplitudes than ϕ_{high}. This gives further support to the finding that the FFT creates ϕ_{low} with an amplitude that is not in line with expectations and is independent of the interpolation scheme chosen. Thus the low resolution potential is initially a poor approximation to the high resolution potential.

The linear interpolation scheme with weighted density correction factor performs marginally better than that of the linear interpolation scheme with an unweighted density correction factor as would be predicted and the quadratic as is adopted by *FLASH*

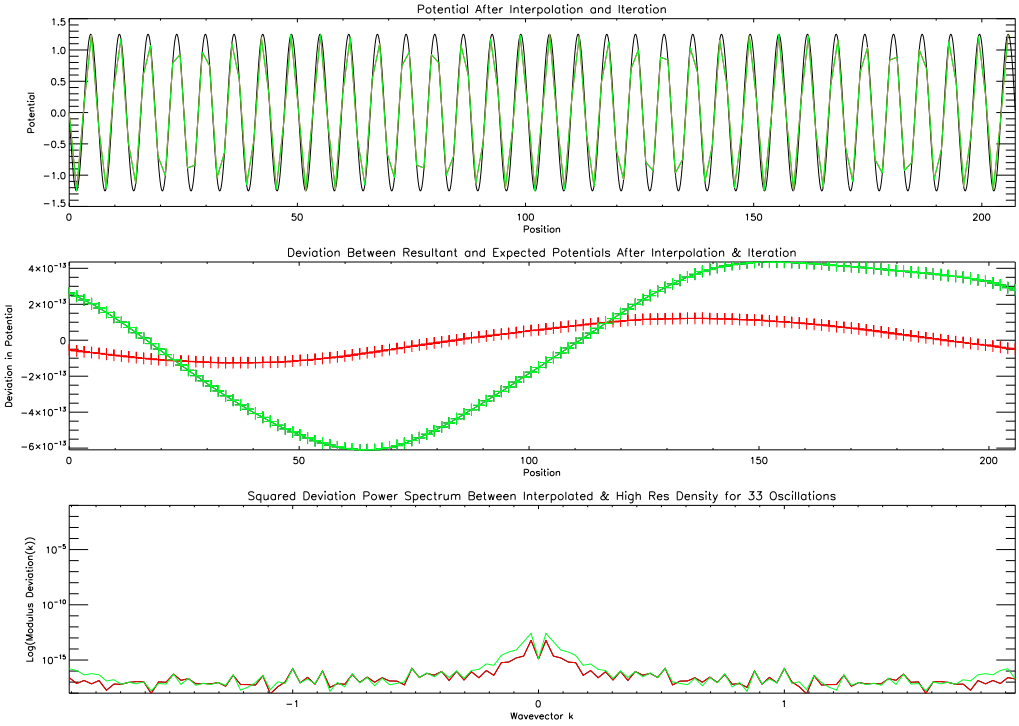

Figure 3.8: The top frame displays the prolonged potentials after relaxation; $\phi^{mean}_{relaxed}$ (green) and $\phi^{actual}_{relaxed}$ (red). The analytic potential is shown in black for comparison. Given the small deviation between the mean and actual based potentials after relaxation, only $\phi^{mean}_{relaxed}$ is visible in the first window. The central frame shows the deviations between the potentials which are derived from low resolution density fields, once they have been prolonged and relaxed, and the corresponding potential obtained by directly converting the high resolution density fields. The final box corresponds to the power spectrum of the deviations in the potential (central frame). Large scale fluctuations correspond to low values of k and small scale deviations are represented by large values of k. The small scale deviations in the potential have been efficiently removed from the potential giving very good agreement with the correct potential. There still exists a large scale deviation since iterative techniques are very slow to remove these although it is near to the limits of numerical precision.

performs at a level between the two. Surprisingly the simple linear interpolation scheme without any density correction factor initially performs the best out of all of the interpolation schemes but I would not expect this to be the case for more complex and non-smooth functions as can frequently occur in astrophysical simulations.

All of the interpolation schemes produce comparable fits after successive relaxation with RMS values within an order of magnitude of acceptable rounding errors. Although this will need confirming for the case with more generalised random density fields, it shows a strong indication that the specific interpolation scheme used does not play a significant role in the generation of fluctuations in the final potential.

3.8.3 Random Density Fields

In order to test the robustness of the multigrid method with more general functions I generated random density fields with a standard deviation of one and a mean of zero. Similar to the results for smooth periodic functions in § 3.8.1 and § 3.8.2, the random fields were found to yield a less than ideal typical RMS between ϕ_{high} and $\phi_{prolonged}$ of ~ 0.0049. Although this RMS is initially better than for the smooth analytic function in § 3.8.1, after twenty Gauss Seidel iterations, the RMS did not reduce down to levels consistent with rounding errors like for the smooth function and instead held at around ~ 0.0042. Although this is tolerable for most cosmological simulations which are often subject to other numerical limitations, it is nevertheless somewhat unexpected and disappointing.

AMR codes typically increase the level of resolution and therefore mesh sampling points over regions of rapid flow change, making the flow appear smoother on small scales. Therefore similar to this effect I chose to smooth the density field using a boxcar function acting over six cell regions. Although this produced a notable 65% reduction to the final *RMS,* it still does not allow us to reconcile the final potential with the expected form to within rounding error.

The first window in figure 3.10 shows the potential as a function of position across the grid, with the second window down giving the respective deviations as a function of position for the four different interpolation schemes. ϕ_{high} is plotted in the first window in black and is used as a proxy to the "true" potential given that no analytic form is available for such a random density field. Plotted are the linear interpolation scheme (yellow), a quadratic (blue), the linear interpolation scheme with density correction factor (red) and its weighted equivalent (green).

As is clearly visible from the lower frame, all of the interpolation schemes posses a

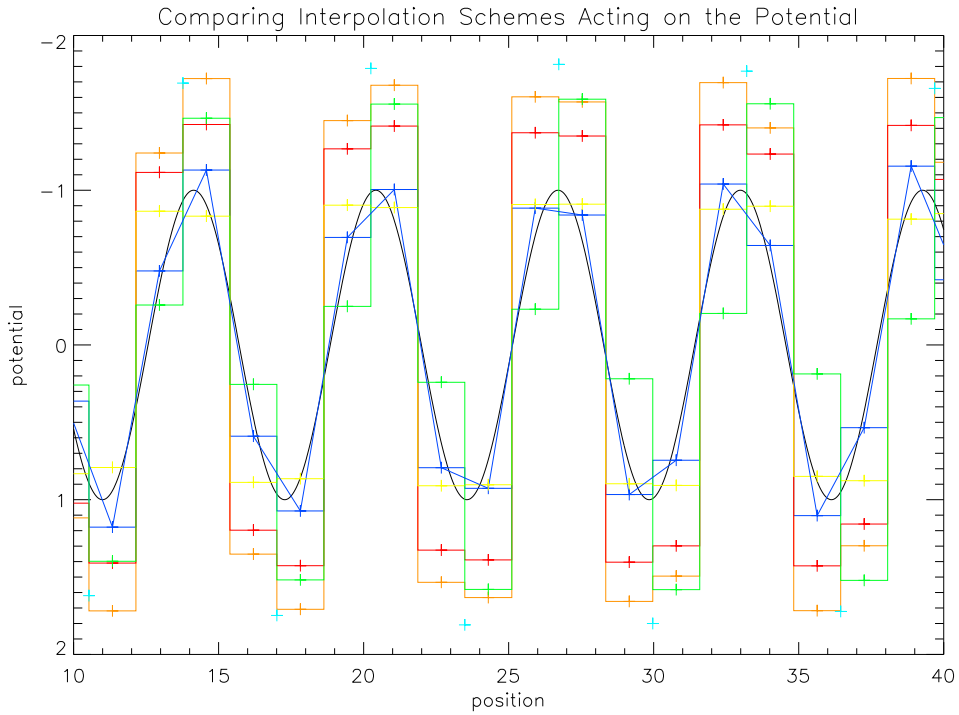

Figure 3.9: Plot of the different interpolation schemes acting upon ϕ_{low} (light blue) with ϕ_{high} (dark blue) shown for comparison. The analytic potential is shown in black. The interpolation schemes used are a simple linear interpolation scheme (yellow), a quadratic (green), a linear interpolation scheme with high resolution density correction factor (orange) and a linear interpolation scheme with weighed high resolution density correction factor (red). The results are shown in a histogram fashion in order to make it easier to differentiate between the performance of each with respect to ϕ_{high}. Although some schemes initially perform better, none of them correct to give the actual potential. Thus it can be concluded that the effect of the interpolation scheme upon the performance of the multigrid scheme is limited and can not be responsible for the large scale deviations visible in figure 3.8.

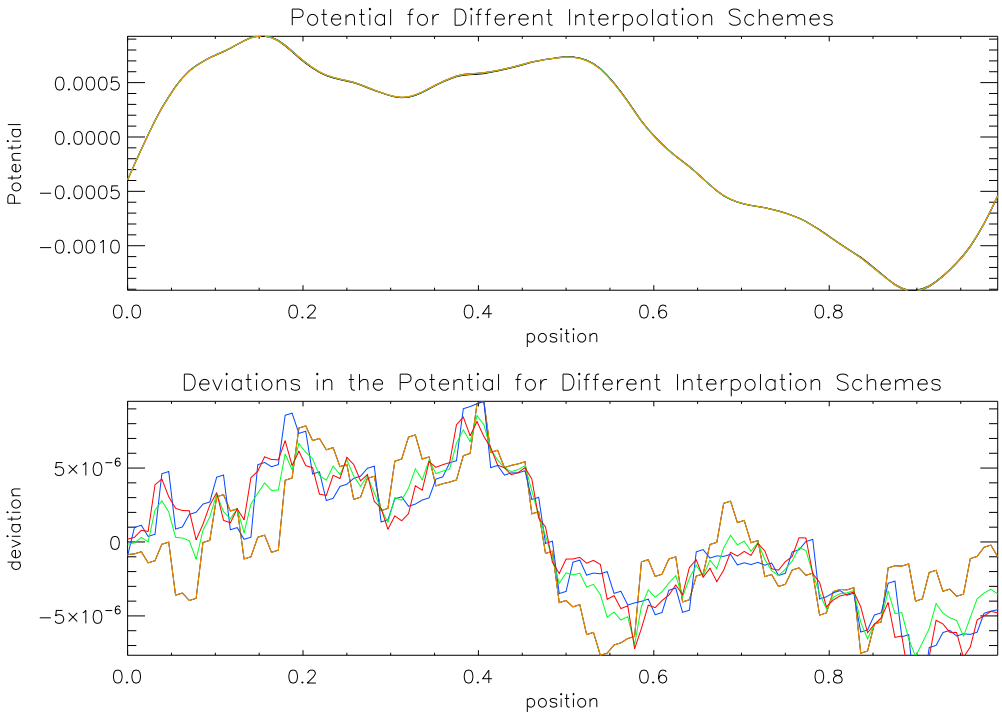

Figure 3.10: The top frame shows a plot of the potentials resulting from different interpolation schemes acting upon a box car smoothed random density field. ϕ_{high} is plotted in black. The resulting deviation between $\phi_{prolonged}$ and ϕ_{high} is shown in the bottom frame. Plotted are the simple linear interpolation scheme (orange), quadratic (blue), linear interpolation scheme with high resolution density correction factor (red) and a linear interpolation scheme with weighted high resolution density correction factor (green). There exists a distinct large scale deviation which is picked up by all of the interpolation schemes. This suggests that the large scale deviation observed for the smooth functions is also present for more random density fields but is more pronounced. Given the way in which the interpolation schemes all show comparable performance this effect can not be attributed to the interpolation method.

similar large scale deviation which traces the general profile of the high resolution potential. The linear interpolation scheme shows marginally more scatter than the other interpolation schemes but is reasonably comparable on a whole with no significant difference being observed between any of the four interpolation schemes. This offers further evidence that the low resolution potential must not be an ideal approximation to ϕ_{high} with some large scale deviation being inherent to the Fourier technique.

Figure 3.11 displays the deviation between the prolonged potential as obtained from three different interpolation schemes and ϕ_{high} using another random density field. The quadratic form is shown in blue, the linear interpolation scheme with density correction factor is shown in red and its weighted equivalent in green. The deviation between $\phi_{prolonged}$ and ϕ_{high} is shown after the initial prolongation as well as after twenty and then one thousand Gauss Seidel iterations. The results after twenty Gauss Seidel iterations show how the small scale fluctuations have been efficiently removed, leaving a smoother deviation profile. As there still exists large scale power in the deviation spectrum, the relative amplitude of the deviation profile after twenty Gauss Seidel iterations is still quite large for all interpolation schemes. After one thousand iterations the deviations reduce down to relatively insignificant levels but still a large scale fluctuation can be seen. Clearly such a vast number of iterations is unfeasible and would prevent us from gaining any advantage over the conventional multigrid solver and so we are left with a quantifiable and small numerical limitation to the multigrid scheme.

It is worth mentioning that in § 3.6.1 I confirm that the Fourier technique is able to accurately reproduce the expected potential at a given level of refinement, these large scale deviations only occurring when we restrict the initial density function. As the original multigrid technique requires the solver to iteratively converge upon the actual potential at the lowest level, it offers no additional accuracy over the Fourier technique described above which will offer the same solution at the base level. Only when we proceed back up through the refinement tree do these large scale terms begin to manifest themselves. Thus I conclude that the multigrid technique possesses inherent large scale deviations which although very small and insignificant for most cosmological simulations, still present a limitation to the accuracy of the gravitational potential that users should be aware of. The new hybrid-FFT gravity solver however does offer a substantial time saving over the conventional multigrid technique with the cluster simulations in chapter 4 benefitting from an impressive factor of 24.5 speed up in run time.

3.8.4 Conclusions on the Hybrid FFT-Multigrid Investigation

After detailed investigation I have shown that the use of an FFT and suitable Green's function can be used to obtain the exact gravitational potential upon a given level of refinement. This scheme is accurate to within numerical rounding precision and can converge upon the potential much faster than any relaxation scheme.

After investigating the application of the new Fourier technique in combination with

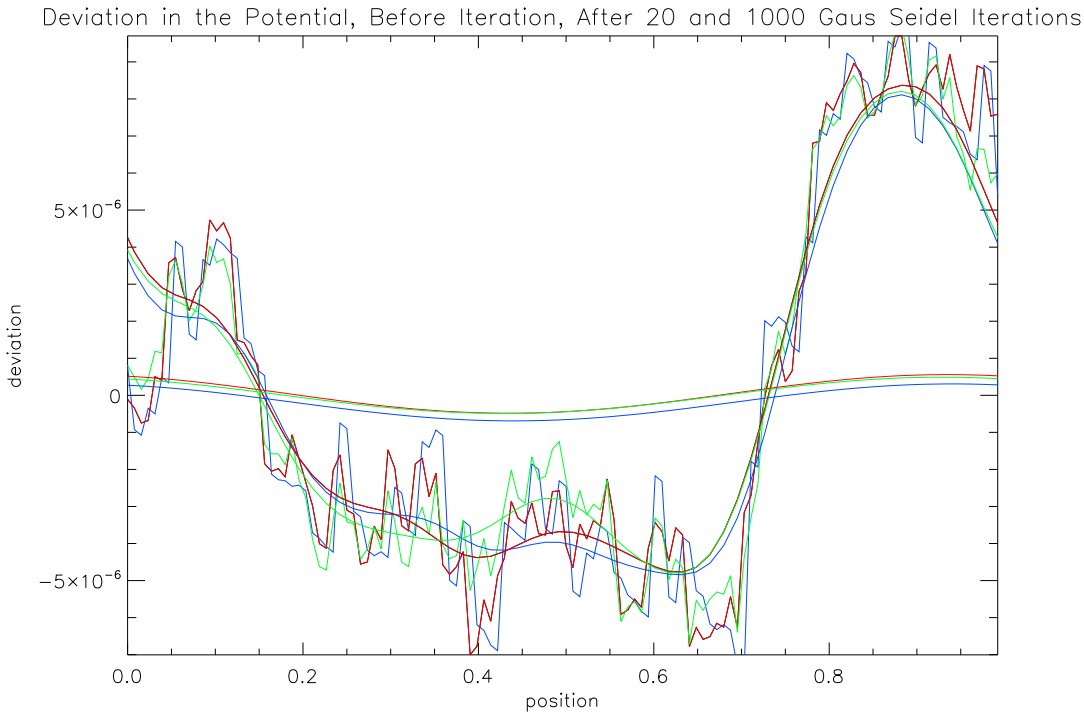

Figure 3.11: The effect of relaxation upon the deviations between the prolonged potentials and the high resolution potential. Plotted are different interpolation schemes; a quadratic (blue), the linear interpolation scheme with high resolution density correction factor (red) and the linear interpolation scheme with weighted high resolution density correction factor (green). The deviation is shown before relaxation, after 20 and 1000 Gauss Seidel iterations (smallest amplitude profile). The initial small scale deviations are quickly compensated for using only a few iterations, leaving a smoother deviation profile, though its amplitude is still fairly comparable. Only after a very large number of iterations does this large scale deviation decrease to leave a tiny deviation.

the multigrid method, I find that there are large scale deviations between the expected and the obtained potential when the Fourier transform is run at a level lower than the peak level of refinement. These deviations are likely to be present in very large simulations that span many levels of refinement since memory restrictions will only allow a finite sized FFT to be performed.

These large scale terms can only be removed with the use of repeated iterative techniques. After a modest twenty Gauss Seidel iterations, the resulting potential relaxes to give a typical RMS deviation in line with permissible rounding errors when dealing with simple periodic functions. This increases to $\sim 0.49\%$ with more general random density functions. Although relatively small, enabling the scheme to work well for a wide range of cosmological simulations (the gravitational convergence with the SPH code *GADGET-2* being presented in chapters 4 and 6) it still highlights a numerical limitation to the accuracy of the scheme which I have now quantified.

These large scale deviations are insensitive to the type of interpolation scheme adopted and the use of mean or actual values as input to the FFT makes no notable difference.

Based upon Fourier power spectra, it would appear that these large scale deviations arise in the Fourier representation of the restricted density function, causing them to be represented by more harmonic frequencies than is necessary, possibly due to limited resolution.

I will follow this work up in chapter 4 by monitoring the behaviour of a model cluster in *FLASH* using the new hybrid FFT-multigrid solver. As the analytic King profile spans a large dynamic range, it will help exacerbate the large scale deviations observed in this chapter, showing clearly how much it will affect the results in a more representative simulation. In addition, the comparison of the results in *FLASH* to those in a selection of other AMR and SPH codes will enable me to confirm whether our *FLASH* code performs favourably with other leading numerical codes.

3.9 Proposed Future Modifications and Optimisations

Work presented as part of this thesis in later chapters clearly highlights particular problem areas that still present technical and resource challenges that will need to be overcome if AMR codes are to continue to remain a key tool to astronomers. As such these are seen as potentially fruitful areas for future work.

Work on the King profile tests presented in § 4.5 highlighted problems with the scaling

of the code with increased refinement. As these tests required a large dynamic range to be implemented, with a total of eleven levels of refinement to adequately resolve behaviour in the dense central regions, this resulted in a vast reduction in the time step at the finest level. This occurs as the Courant-Friedrichs-Levy criteria for stable time stepping implies that as the cell size halves, so too must the time step. This ensures that material cannot traverse multiple cells in a given step which would lead to poor sampling of the fluid flow. As all blocks at all levels of refinement are advanced in lock-step with each other, a bottleneck arises dependent upon the time step in the finest cells. Consequently many blocks on coarser levels are advanced using timesteps that are smaller than what they require for stable evolution. This leads to cells at coarser levels being evolved more frequently than is efficient.

A means of overcoming this has been employed in various SPH codes including *GADGET-2* whereby multi-time stepping is implemented. In this scenario only cells (or particles in the SPH case) at the finest level of resolution are advanced at each step. Cells or particles at levels with larger timesteps are only advanced when those at levels below have covered a period of time equal to the size of their timestep. In *FLASH* this would imply that cells at coarser levels would instead be advanced once for every two steps that cells on the level above take. It has been argued that most of the computational zones will lie in very highly refined regions, thus the time spent evolving those at low levels of refinement will be small. Most cosmological simulations however track properties over a vast range of scales with many blocks being present at all levels of resolution. This has manifested itself in the King profile tests and particularly in my star formation simulations presented in § 7. Here an ideal galaxy simulation would have included the dark matter and gaseous halo spanning out to megaparsec scales. However I was restricted to running in a much smaller box as resolving the Jeans length in dense star forming regions required a peak resolution of tens of parsecs. To resolve both the central galaxy properties and a full dynamic halo would require a great many levels of refinement. Although running simulations of this scale would have been feasible in terms of memory consumption since few cells lie in these highly refined central regions, the entire simulation would be forced to evolve in lock-step with them. This makes running a full box enclosing the halo impractical. Consequently I was forced to compromise and simulate a small sub-region containing just the disk. It is my belief that cosmological simulations in particular would benefit greatly from the addition of multi-time stepping. Many SPH codes and some AMR codes already have multi-timestepping implemented to great success and given the structured

mesh approach of *FLASH* it should be reasonably straight forward to accomplish the same.

The star formation simulations in § 7 also highlighted the drawbacks of imposing a maximum jump in refinement of only one between neighbouring blocks. Despite the densest regions only containing a few blocks, many coarser blocks where required to refine up to them. Figure 3.12 shows a slice through one of the disk galaxies presented in Chapter 7 including star formation and supernova feedback, taken tangential to the plane of the disk. As can be seen, refining the central region requires the addition of many layers of resolution below, with only 34% of the total leaf blocks being present at the peak level of refinement. Thus for every timestep, up to 66% of the evolutionary timestep is spent evolving blocks with a smaller timestep than is necessary.

Thus in conclusion, the majority of the blocks in my astrophysical simulations reside at levels below the peak level of refinement with many levels being required to resolve key physical phenomena such as star formation and fragmentation. As the current implementation only allows a single jump in refinement level between neighbours and the entire simulation evolves in lock-step, a major speed up could be attained using a more flexible refinement scheme combined with multi-timestepping.

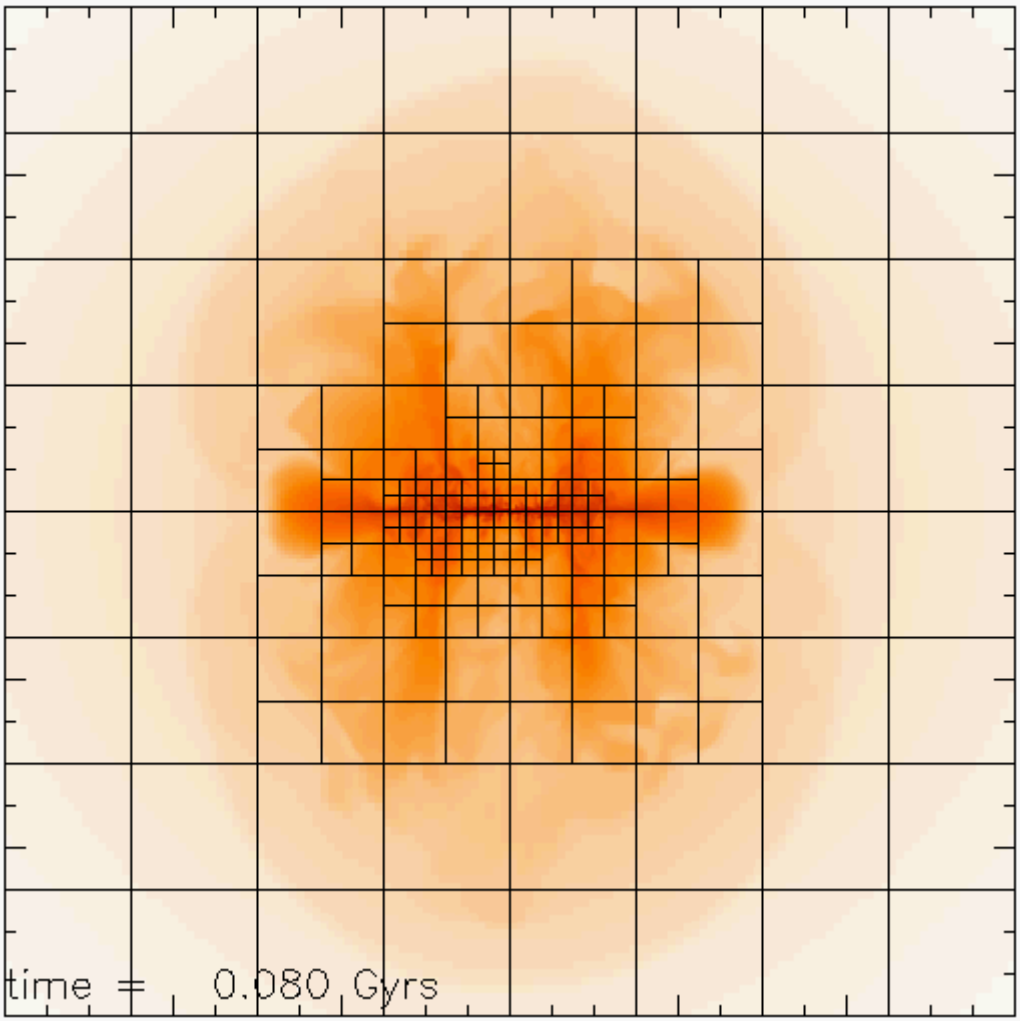

Figure 3.12: Logarithmic density slice through a galactic disk simulation including star formation and feedback. The refinement pattern of the mesh is superimposed over the top so that we can clearly see the spatial distribution of the different sized leaf blocks. Only $\sim$ 34% of the leaf blocks are at the highest level of refinement. The entire simulation is evolved in lock-step using the timestep from the finest refinement level. Thus a large number of blocks at lower levels are evolved with timesteps that are smaller than they need to be. This leads to inefficient evolution of many of the simulation blocks.

Chapter 4

A Test Suite for Quantitative Comparison of Hydrodynamics Codes in Astrophysics

4.1 Foreword

The contents of this section were published as part of a collaborative project in the work by Tasker et al. (2008). The *FLASH* simulations presented in this chapter were performed exclusively by myself. The analysis presented in this chapter was performed as part of a collaboration for which I played a large part, especially with regards the *FLASH* data. The *ENZO* simulations were performed and analysed by Dr. Elizabeth Tasker (University of Florida) whilst the *GADGET-2* and *Hydra* simulations present work undertaken by Dr. Ricardo Brunino and Dr. Stephen Hopton (University of Nottingham).

4.2 Introduction

Over the last decade, numerical simulations have developed into a cornerstone of astrophysical research. From interactions of vast clusters of galaxies to the formation of proto-planetary discs, simulations allow us to evolve systems through time and view them at every angle. They are an irreplaceable test-bed of our physical understanding of the Universe.

A number of hydrodynamics codes have been developed that are widely used in this field and still more are being developed. Fundamentally, they all do the same job; they

solve the equations of motion to calculate the evolution of matter through time. Whether the matter represents a nebula for the birth of a star or a network of galaxy clusters, the basic technique remains the same.

However, the algorithms used to solve these equations vary from code to code and this results in differences in the resulting data. Understanding the origin of these variations is vital to the understanding of the results themselves; is an observed anomaly an interesting piece of new physics or a numerical effect? As observational data takes us deeper into the Universe, it becomes more important to pin down the origin of these numerical artifacts.

Additionally, it is difficult to compare results from simulations run with different codes. With observations, papers clearly state the properties of the instrument such as the diameter of the mirror and the wavelengths it is most sensitive to. While a brief description of the code is always included in theoretical papers, there exists no obvious conversion to other numerical techniques and therefore the results are more difficult for the reader to interpret.

The problem of code comparison is not new and it is a topic that has recently created a great deal of interest. The reason for its current importance is a positive one; improved numerical techniques and increased computer power have resulted in simulations reaching greater resolutions than could have been imagined even a few years ago. However, this high refinement comes at a price; as we start to pick out the detail of these complex fluid flows, the physics we need to consider gets dramatically more complicated. This brings us to the main question code comparison projects are trying to answer; can we use the same tools for this new regime of problems?

A number of papers have come out that tackle this. One of the most famous is the Santa Barbara Comparison Project (Frenk et al., 1999) which compared many of the progenitors of today's codes by running a model of a galaxy cluster forming. Taking a different tack Thacker et al. (2000) compared the performance of a dozen different implementations of a single approach (in this case smoothed particle hydrodynamics, SPH) on standard astrophysical problems that included the Sod shock, examining the range of outcomes that were available from a single technique. They concluded that one of the weaknesses of SPH was the weak theoretical grounding which allows several equally viable formulations to be derived. More recent work includes Agertz et al. (2007), who focus specifically on the formation of fluid instabilities, comparing the formation of Kelvin-Helmholtz and Rayleigh-Taylor instabilities in six of the most utilised codes. Additionally, O'Shea et al. (2005) has completed a direct comparison between two particular codes

(*Enzo* and *Gadget2*, see below) looking at the formation of galaxies in a cosmological context.

All these projects give detailed insights into the differences between the codes, but are unable to provide a quantitative measure of how well a code performs in a particular aspect. This is especially true of the cosmological-based tests of Frenk et al. (1999); O'Shea et al. (2005) where the problem is not sufficiently well-posed for convergence onto a single answer. Agertz et al. (2007) sets out a simpler problem and compares it to analytical predictions, but the system is still sufficiently complex not to have an exact solution. Additionally, no previous comparison has attempted to quantitatively compare different codes to one another; asking whether it is possible to obtain identical results and with what conditions. Without this crucial piece of information, it is impossible to fully assess a piece of work performed by an unfamiliar code or to judge which code might be the most suited to a given problem type. This has resulted in somewhat general comments being made about the differences between numerical techniques which has led to many myths about a code's ability becoming accepted dogma.

The set of tests we present in this chapter are designed to tackle these difficulties with the intention that they might become part of an established test programme all hydrodynamical codes should attempt. We present four problems that specifically address different aspects of the numerical code all of which have expected 'correct' answers to compare to. The first two tests, the Sod shock tube and Sedov blast, are both strong shock tests with analytical solutions. The third and forth tests concern the stability of a galaxy cluster and are primarily tests of the code's gravitational solver. For all four tests we directly compare the codes against the analytic solution and present an estimate of the main sources of any systematic error.

The remainder of this chapter is organised as follows: in section 2 we give a short summary of the main features of each of the four codes we have employed. In section 3 we deal with the Sod shock and Sedov blast tests, setting out the initial and final states and comparing each code against them. We repeat this exercise for a static and translating King sphere in section 4. Finally we discuss our results and summarise our conclusions in section 5.

4.3 Description of the Codes

The two major techniques for modelling gases in astrophysics are smoothed particle hydrodynamics (SPH) and adaptive mesh refinement (AMR). In the first of these, the gas is treated as a series of particles whose motion is dictated by Lagrangian dynamics. In AMR, the gas is modelled by a series of hierarchical meshes and the flow of material between cells is calculated to determine its evolution. There are a variety of codes which utilise both techniques and four of the major ones will be used to run the tests presented in this chapter, two of which use SPH (*Hydra* and *Gadget2*) and two of which use AMR (*Enzo* and *Flash*).

4.3.1 *Enzo*

Enzo is a massively parallel, Eulerian adaptive mesh refinement code, capable of both hydro and N-body calculations (O'Shea et al., 2004; Bryan and Norman, 1997). It has two hydro-algorithms which can be selected by the user; the piecewise parabolic method (PPM) and the Zeus astrophysical code. The PPM solver (Colella and Woodward, 1984) uses Godunov's method but with a higher-order spatial interpolation, making it third-order accurate. It is particularly good at shock capturing and outflows. The Zeus method in *Enzo* is a three-dimensional implementation of the Zeus astrophysical code developed by Stone and Norman (1992)[1]. It is a simple, fast algorithm that allows large problems to be run at high resolution. Rather than Godunov's method, Zeus uses an artificial viscosity term to model shocks, a technique which inevitably causes some dissipation of the shock front. We compare both these hydro-schemes in these tests.

4.3.2 *Gadget2*

Gadget2 is a massively parallel, Lagrangian, cosmological code that is publicly available from the author's website (Springel, 2005). It is an N-body/SPH code that calculates gravitational forces by means of the Tree method (Barnes and Hut, 1986) and is also able to optionally employ a Tree-PM scheme to calculate the long range component of the gravitational interactions. In order to follow the hydrodynamic behaviour of a collisional medium, the code uses the entropy-conserving formulation of SPH described in Springel and Hernquist (2003): the main difference of this approach with respect to the

[1]Note that the hydrodynamics code *ZEUS-3D* has been developed independently of *Enzo (Zeus)* and its performance is not equivalent.

standard Monaghan (1992) formulation of SPH resides in the choice of describing the thermodynamic state of a fluid element in terms of its specific entropy rather than its specific thermal energy. This leads to a tight conservation of both energy and entropy in simulating dissipation-free systems. Additionally, *Gadget2* employs a slightly modified parametrisation of the artificial viscosity (by introducing the so called "signal velocity" as in Monaghan (2001)). The user is allowed to set the strength of this artificial viscosity for the specific problem being considered via an input parameter, ArtBulkVisc. The time stepping scheme adopted by the code is a leap-frog integrator guaranteed to be symplectic if a constant timestep for all particles is employed. In this work we have exploited the possibility of using fully adaptive individual timesteps for all the particles in the simulation, this being a standard practice.

4.3.3 *Hydra*

Hydra is an adaptive particle-particle, particle-mesh code combined with smoothed particle hydrodynamics (Couchman et al., 1995). It has the significant disadvantage that even though massively parallel versions exist (Pearce and Couchman, 1997), the publicly available version is not a parallel implementation and so this code cannot, as released, be used for very large simulations. Akin to *Gadget2*, *Hydra* uses an entropy conserving implementation of SPH, but unlike *Gadget2*, *Hydra* does not have fully adaptive individual timesteps. Although the timestep adapts automatically from one step to the next all the particles move in lockstep.

4.3.4 *Flash*

Flash is a publicly available massively parallel Eulerian AMR code developed by the Alliances Center for Astrophysical Thermonuclear Flashes (Fryxell et al., 2000). Originally intended for the study of X-ray bursts and supernovae, it has since been adapted for many astrophysical conditions and now includes modules for relativistic hydrodynamics, thermal conduction, radiative cooling, magnetohydrodynamics, thermonuclear burning, self-gravity and particle dynamics via a particle-mesh approach. *Flash* uses the oct-tree refinement scheme of the PARAMESH package, with each mesh block containing the same number of internal zones. Neighbouring blocks may only differ by one level of refinement with each level of refinement changing the resolution by a factor of two. The hydrodynamics are based on the PROMETHEUS code (Fryxell et al., 1989). The input states for the Riemann solver are obtained using a directionally split PPM solver (Colella and

Woodward, 1984) and a variable time step leapfrog integrator with second order Strang time splitting is adopted (Strang, 1968). This work uses a modified hybrid FFTW based multigrid solver to solve Poisson's equation and determine the gravitational potential at each timestep. This results in a vast reduction in time spent calculating the self-gravity of the simulation relative to a conventional multigrid solver. *Flash's* refinement and de-refinement criteria can incorporate the adapted Löhner (1987) error estimator. This calculates the modified second derivative of the desired variable, normalised by the average of its gradient over one cell.

4.4 Shocking Tests

One of the greatest differences between simulations performed now versus those undertaken five years ago is the increasing importance of modelling strong shocks accurately. While it has long been known that the Universe is a violent place, with events such as supernovae, galaxy mergers and AGN generating blasts which rip through the intergalactic medium, simulations did not have the resolution to see such phenomena in detail, so these sharp discontinuities were largely ignored. Now, as we struggle to understand the effects of feedback in galaxy formation, multiphase media are essential physics (Tasker and Bryan, 2008, 2006; Wada and Norman, 2007; Robertson and Kravtsov, 2008). In order to attack such problems codes must be able to capture shocks with some proficiency. These two problems, the Sod shock test and the Sedov blast test, explicitly test the resolution of shock jumps and allow comparison with exact analytical solutions.

4.4.1 Riemann Shock Tube Problem

The shock tube problem (Sod, 1978) has been used extensively to test the ability of hydrodynamics codes to resolve a sharp shock interface (Feng et al., 2004; Shapiro et al., 1996; Ryu et al., 1993). The test set-up is simple, consisting of two fluids of different densities and pressures separated by a membrane that is then removed. The resulting solution has the advantage of showing all three types of fluid discontinuities; a shock wave moving from the high density fluid to the low density one, a rarefaction (sound) wave moving in the opposite direction and a contact discontinuity which marks the current location of the interface between the two fluids. For this test the initial conditions are traditionally chosen such that the pressure does not jump across the contact discontinuity.

We extend the traditional one-dimensional shock tube problem to consider two three-

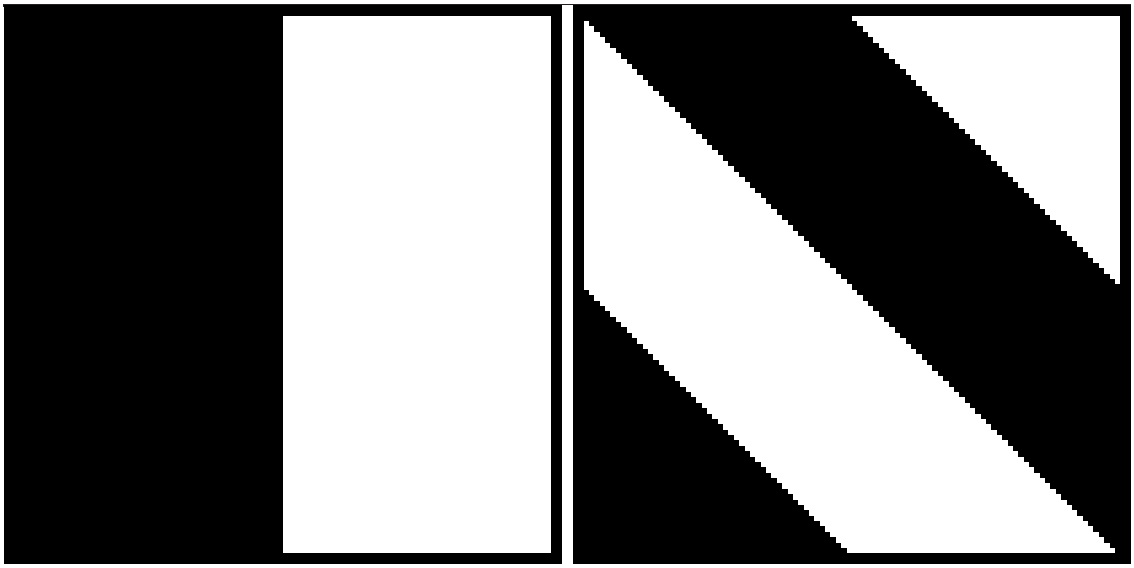

Figure 4.1: Example projections of the initial conditions for a three-dimensional Sod shock test. Black and white regions represent fluids of different densities. The left-hand image shows the shock face oriented along the [1,0,0] plane, while the right-hand image shows it oriented along the [1,1,0] plane. In actuality our second test is oriented in the [1,1,1] plane i.e. oblique to all the axes.

dimensional set-ups; the first of these has the fluid membrane at 90° to the x-axis of the box ([1,0,0] plane), causing the shock to propagate parallel to this axis. In the second test, the membrane is lined up so that its orthogonal unit vector lies at 54.7° to each of the x, y and z axes ([1,1,1] plane). This change in orientation of the shock is designed to highlight any directional dependencies inherent in the code, as illustrated in Figure 4.1. All analysis was performed perpendicular to the original shock plane.

For our particular set-up we chose the initial density and pressure jump either side of the membrane to be from $(\rho_1 = 4, p_1 = 1)$ to $(\rho_2 = 1, p_2 = 0.1795)$ with the fluid initially at rest. The polytropic index was $\gamma = \frac{5}{3}$. Periodic boundary conditions were used and the results were analysed at $t = 0.12$.

4.4.1.1 General Results

Figures 4.2 and 4.3 show the results from all four codes running this test. In the results presented in these figures, both *Enzo* and *Flash* used an 100^3 initial (minimum refinement) grid[2] with two levels of higher refinement each of which decreased the cell size by a

[2]In *Enzo's* case, this is the 'root' grid. *Flash* does not have a root grid, but can specify a minimum cell size that must be maintained everywhere. For ease of typing, we shall refer to the coarsest grid in both cases as the 'initial' grid.

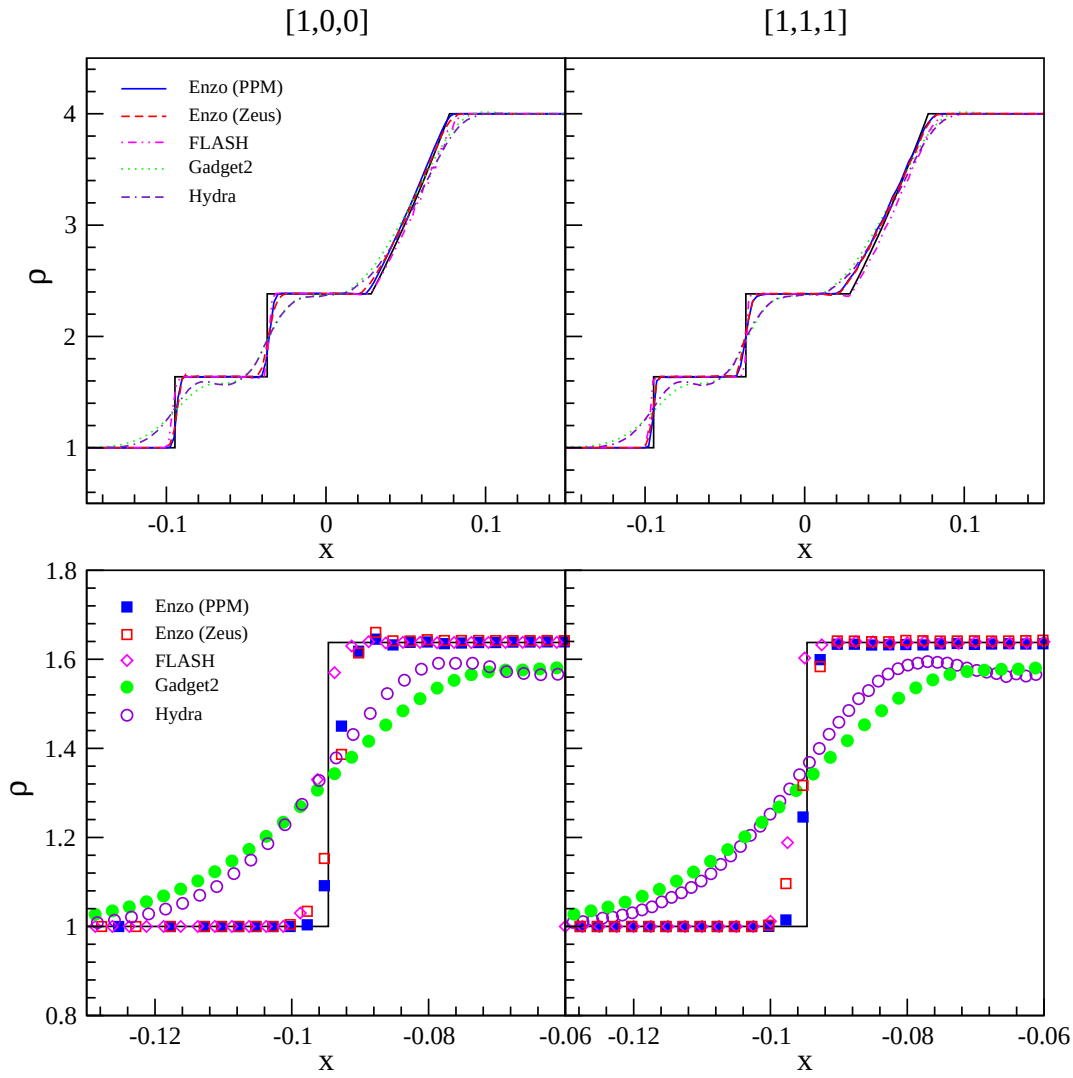

Figure 4.2: Projected density parallel to the shock front from the 3-dimensional shock tube test. Left-hand plots show results from the [1,0,0] shock set up, right-hand from the [1,1,1] set-up. Both are at $t = 0.12$. The bottom plots show a close-up of the shock front itself.

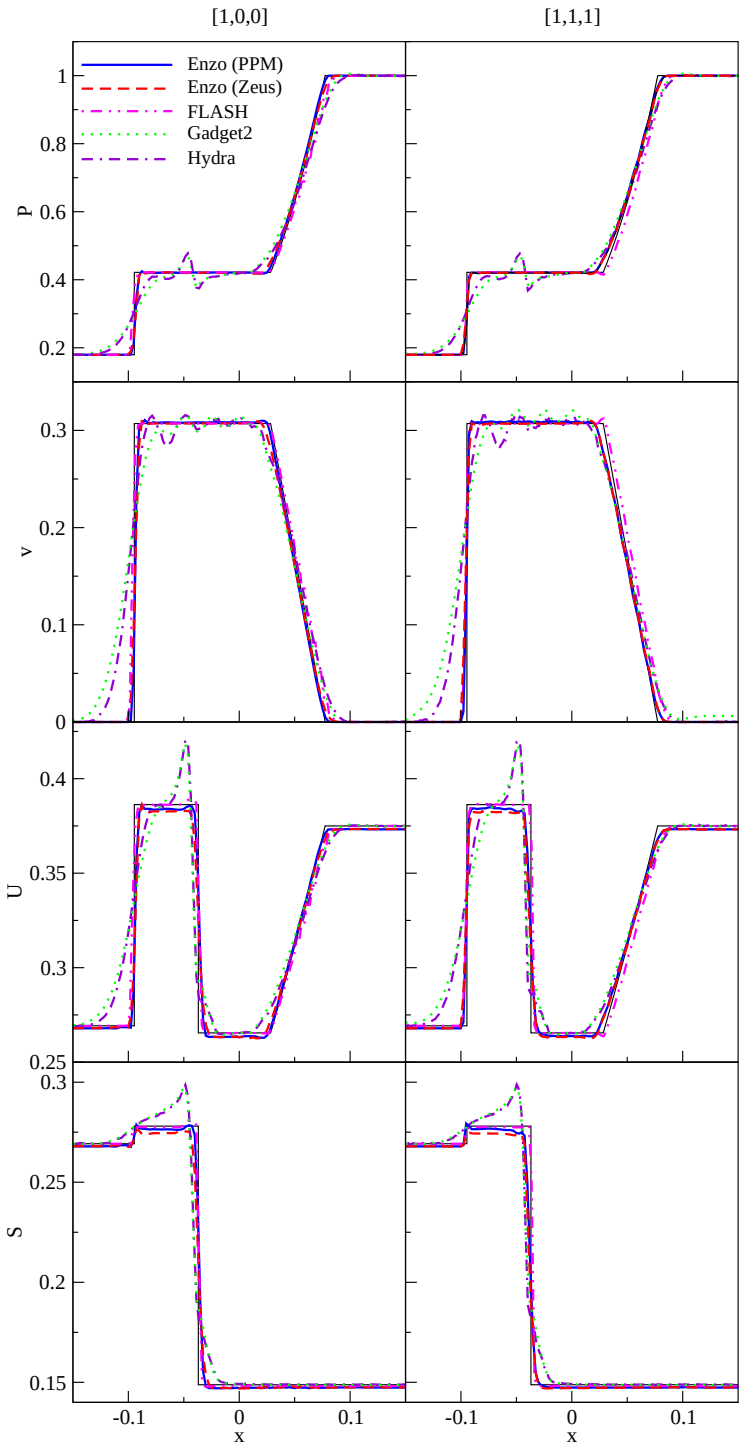

Figure 4.3: Projected data parallel to the shock front from the 3-dimensional shock tube test. The left hand column shows the results from the [1,0,0] shock front alignment, while the right hand column gives those from the [1,1,1] set-up. Top-to-bottom are plotted pressure, internal energy, velocity and entropy. As with figure 4.2 all the panels are from $t = 0.12$.

factor of 2. The smallest cell size in this case was therefore 0.0025 of the unit box. *Enzo* refined anywhere the gradient of the derived quantities exceeded a critical value whereas *Flash* placed refinements according to the Löhner (1987) error estimator described in section 4.3.4 which places sub-grids based on the second derivative of the derived quantities. For the SPH codes, both *Hydra* and *Gadget2* were run with 1 million particles formed from two glasses containing 1.6 million and 400,000 particles. The solid black line in all cases is the analytical solution which requires a shock at $x = -0.095$ and a contact discontinuity in the density at $x = -0.032$ at $t = 0.12$.

What is clear from Figures 4.2 and 4.3 is that all of the codes pass the zeroth level test and successfully reproduce the shock jump conditions, although both the SPH codes suffer from visible ringing and broadening around any discontinuities. *Enzo (Zeus)* does not produce the shock jump condition as accurately as *Enzo (PPM)* and *Flash*, as seen in the plots of energy and entropy in Figure 4.3, where its post-shock values are around 2% lower. In the oblique case, the quadratic viscosity term, QV, in *Enzo (Zeus)* was increased from its default value of 2.0 to 10.0. The effect of this value is discussed more fully in relation to the Sedov blast test in section 4.4.2.3. For the planar case, QV was kept at 2.0. Pleasingly none of the other codes appear to have any visible directional dependence, performing equally well in both the grid aligned and oblique cases we tried. All the codes could equally well resolve the location and smooth rise of the rarefaction wave but both the SPH codes struggle with the contact discontinuity, with a large overshoot not seen by either of the mesh based codes. This is partly due to the initial conditions as for the SPH codes the sudden appearance of a density jump introduces a local source of entropy. In this chapter we are contrasting the results from the different approaches rather than studying the Sod shock problem itself in detail. As it is a standard test case higher resolution results can be found in the individual code's method papers (e.g. Fryxell et al., 2000; Thacker et al., 2000; Springel, 2005).

Closer inspection of the data reveals differences in each code's capacity to handle strong shocks. In the bottom row of Figure 4.2, we show a close-up of the density over the shock-front. Capturing shocks accurately is an area that SPH codes traditionally struggle with more than their Eulerian counterparts due to their inherent nature of smoothing between particles. Indeed, we see in this figure both *Gadget2* and *Hydra* have a smeared out the interface compared to *Enzo* and *Flash's* steep drop in density. Small differences between the AMR codes are also visible here. *Enzo (PPM)* spreads the shock front over three cells, whereas *Enzo (Zeus's)* use of the artificial viscosity term extends this to five.

Flash's PPM scheme gives very similar results to *Enzo (PPM)*, also spreading the shock front over three cells.

Figure 4.3 shows the pressure, internal energy, velocity and computational entropy;

$$s = \frac{T}{\rho^{\gamma-1}} \tag{4.1}$$

over the region of interest for the planar [1,0,0] set-up (left column) and the oblique [1,1,1] set-up (right column).

Both *Hydra* and *Gadget2* show signs of post-shock ringing in the velocity plot. The two Lagrangian codes adopt different implementations of artificial viscosity; with a frequently used choice of the viscosity parameter (ArtBulkVisc = 1) the ringing features in *Gadget2*'s profiles appears to be more pronounced than in *Hydra*'s (not shown in the plot). In order for *Gadget2* to get closer to *Hydra*'s performance, a choice of a significantly higher viscosity parameter has been necessary (namely ArtBulkVisc = 2). The results produced under such a choice are shown in Figure 4.3. The SPH codes also exhibit a large spike in both internal energy and entropy at the location of the contact discontinuity. This is driven by the initial conditions, where two independent particle distributions suddenly appear immediately adjacent to one another.

4.4.1.2 Quantitative Comparison

We saw in the previous section that most of the codes model shock development and sound wave propagation with reasonable success and that they show no directional preference to the orientation of the shock interface. However, differences were apparent between each of the codes, most obviously between the SPH and AMR techniques (unsurprising, since their numerical algorithms fundamentally differ). In this section, we quantitatively compare results from each code and attempt to get as close a match between their results as possible. For simplicity, we confine the SPH codes to using static meshes for this comparison.

Figure 4.4 shows a graphical comparison of the density projection between a *Hydra* run of 1 million particles and *Enzo (PPM)* for different grid sizes. It is clear that a grid size of 20^3 produces significantly poorer results than the *Hydra* data whereas a grid size of 100^3 produces significantly better results particularly in the low density region. Although the situation is confused by the lack of points available with *Enzo*, a grid size of 20^3 can produce no more than 20 distinct values across the length of the volume being modelled, it is clear that this is too few to recover this model. However, even with 50^3 cells in the

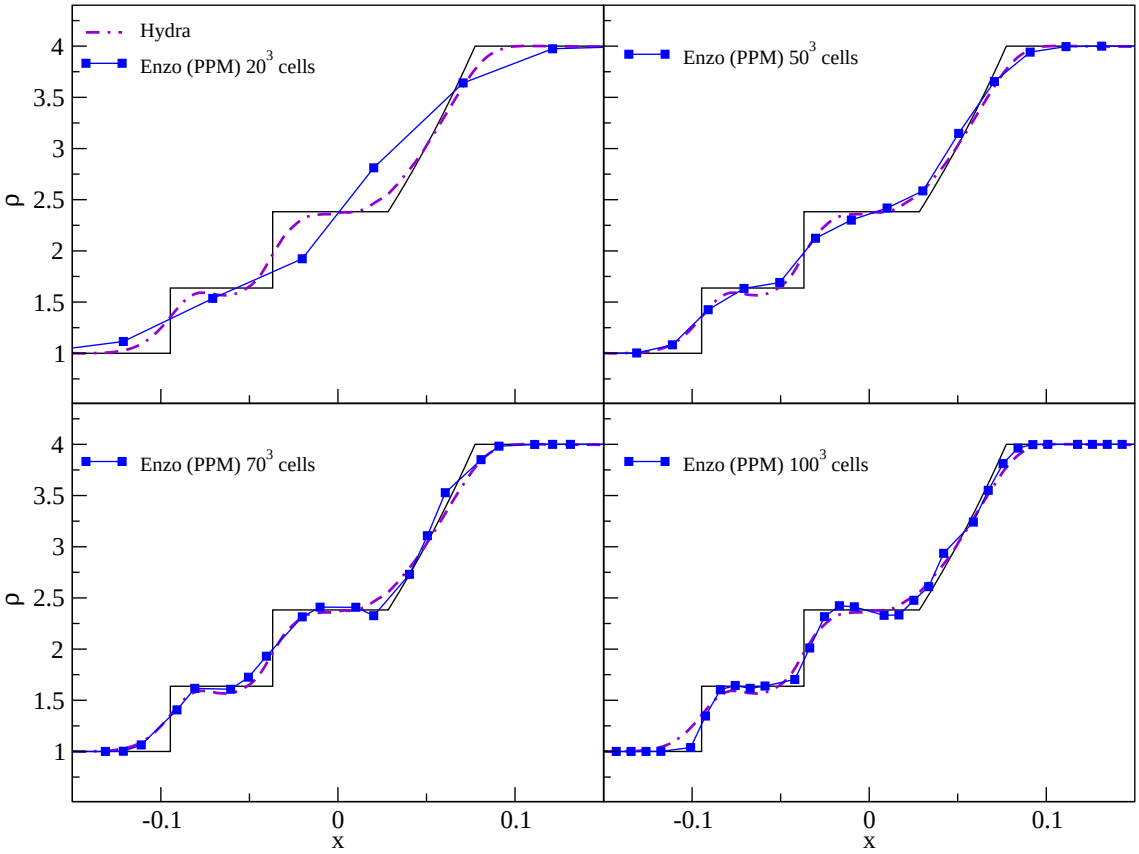

Figure 4.4: Comparison between data from *Enzo (PPM)* at different static grid sizes with *Hydra's* run of 1 million particles. Results plotted show the projection of the density for the [1,0,0] set-up.

box the major features are largely recovered. To make further progress we require a more quantitative way of comparing the results from the codes. To achieve this, we employ a cubic spline to interpolate the data from all runs to the same 178 x points at which we have calculated the analytical values. The residue between each new curve and the analytical solution was then summed and divided by the number of points. We considered the residue both across the whole region of interest from [-0.15, 0.15] and just across the shock-front from [-0.13, -0.06].

The top part of table 4.1 shows the residues from *Enzo* over these ranges for increasing static grid size. As Figure 4.4 showed, the overall fit across the whole region of interest improves rapidly with the number of grid points. The improvement for the AMR run highlights one of the greatest assets of adaptive grid coding; the ability to tag cells for refinement based on slopes or shock-fronts, allowing the extra resolution to be concentrated on these problematic areas. Table 4.1 also shows that *Enzo (Zeus)* typically has poorer results than *Enzo (PPM)* for equal numbers of grid cells, something that came out

Table 4.1: Residue of the analytical solution of the Sod shock density in the planar [1,0,0] set-up for different static grid sizes and SPH resolutions for *Enzo, Hydra* and *Gadget2*.

	Enzo (PPM) residue	*Enzo (Zeus)* residue
Grid size	Interface (Shock)	Interface (Shock)
10^3	0.343 (0.383)	0.238 (0.220)
20^3	0.232 (0.166)	0.173 (0.198)
30^3	0.164 (0.164)	0.165 (0.172)
40^3	0.129 (0.125)	0.166 (0.174)
50^3	0.095 (0.089)	0.119 (0.130)
60^3	0.094 (0.110)	0.114 (0.118)
70^3	0.076 (0.082)	0.105 (0.126)
80^3	0.086 (0.082)	0.081 (0.090)
90^3	0.061 (0.075)	0.088 (0.088)
100^3	0.069 (0.064)	0.085 (0.079)
AMR run (400^3)	0.032 (0.030)	0.035 (0.036))

	Hydra residue	*Gadget2* residue
No. particles	Interface (Shock)	Interface (Shock)
8M ($234^3, 148^3$)	0.0559 (0.065)	
1M ($117^3, 74^3$)	0.0852 (0.104)	0.0817 (0.089)
250k ($74^3, 48^3$)	0.112 (0.121)	0.113 (0.102)

in Figures 4.2 and 4.3, and is a result of energy loss due to the inaccurate treatment of the shock jump conditions by *Enzo (Zeus)*.

The lower part of table 4.1 shows the residues for *Hydra* and *Gadget2* at different resolutions. Both SPH codes show similar results, with similar residues in each at the same resolution. If we compare these values to the nearest *Enzo* result we can get an estimate for the number of particles needed per grid cell in the low density region for similar recovery of the shock profiles. Included after the number of particles for each of these runs is the effective number of particles in the high and low density regions. As is obvious from this table and Figure 4.4 roughly matching accuracy occurs somewhere between these two limits. This is equivalent to when there is approximately one SPH particle per grid cell.

The analysis done in this section does have one obvious flaw; introducing adaptive meshes allows the grid-based codes to achieve a high resolution while employing many less cells overall. Our equivalence of one SPH particle per cell applies to cells in the high resolution region not the total number of cells in the simulation. As such, this comparison works best in situations where the resolution is needed in the highest density areas. In both this problem type and the one described in the next section, the high density areas are not those that require higher resolution, rather the region of interest is that where the density is changing rapidly. Since usually SPH particles are tied to the mass flow of a system, they are not able to increase the resolution over shock fronts, unlike their grid counterparts. However, in many astrophysical problems, such as those we shall meet in section 4.5, this is not necessarily a serious limitation.

4.4.2 The Sedov Blast Wave Test

The Sedov Blast Test (Sedov, 1959) is an intense explosion caused by a quantity of energy deposited in the centre of the simulation box. The result is a strong spherical shock that propagates through the background homogenous medium.

This test is particularly appropriate in astrophysics since it represents well the physics required to deal with supernova explosions. An inability to resolve the resulting shock front will lead to an incorrect estimate of the energy deposited into the galaxy and the volume it affects.

It also poses different problems to both the particle and grid schemes described above. Although the shock front expands as it travels outwards it also sweeps up particles which increase the density contrast of the shell relative to the ambient medium. The particle

codes will therefore find it progressively easier to reach the required resolution. Since the shock front is also spherical, grid codes will have to cope with any artificial grid-alignment effects. It is therefore a test that is both appropriate and taxing.

The analytical calculation of the shock propagation is described in full in Sedov (1959) and Landau and Lifshitz (1959) who show the shock front's radius is given by:

$$r(t) = \left(\frac{E_0}{\alpha\rho_0}\right)^{1/5} t^{2/5} \tag{4.2}$$

where E_0 is the initial energy injected, ρ_0 is the background density and $\alpha = 0.49$ for an ideal gas with $\gamma = 5/3$.

Since there is no cooling or gravity in this test, the quantities above are all unitless. In our calculations, the initial density of the medium is $\rho_0 = 1$, the explosion energy $E_0 = 10^5$. The shock expands into a box of side 10 and we compare our results at $t = 0.1$. Figure 4.5 shows density projections (density integrated along the length of the simulation box) of the simulation box during the Sedov test for times from 0.025 to 0.1.

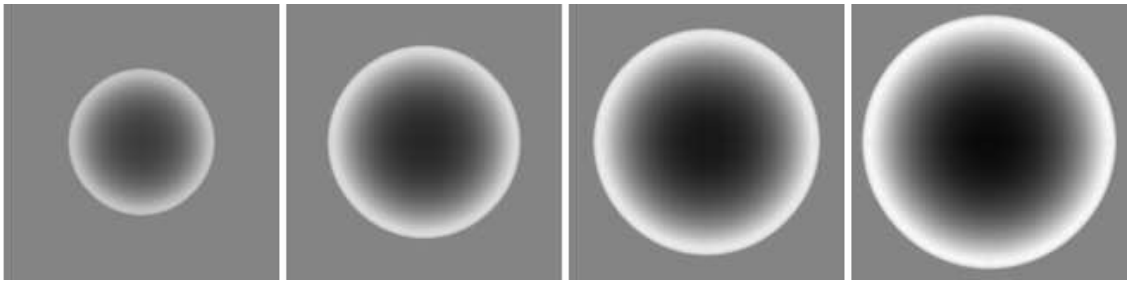

Figure 4.5: Density projections of the Sedov blast test performed using *Enzo (PPM)* at $t = 0.025, 0.05, 0.075$ and 0.1.

4.4.2.1 General Results

Figure 4.6 shows the results from the Sedov blast test at $t = 0.1$ for each code and the analytical solution. For this test, *Enzo* and *Flash* used an initial grid of 100^3 with two levels of refinement, each reducing the cell size by a factor of two. These subgrids were placed where the shock front was detected. This gave a minimum cell size of 0.025 over the shock. For the AMR codes the central energy was injected into a spherical region of radius 3.5 grid cells on the finest levels, corresponding to $r = 0.0875$, a common choice for this test. Both *Hydra* and *Gadget2* used 1 million particles relaxed into a glass, with the energy added to the central 32 particles in a top hat distribution.

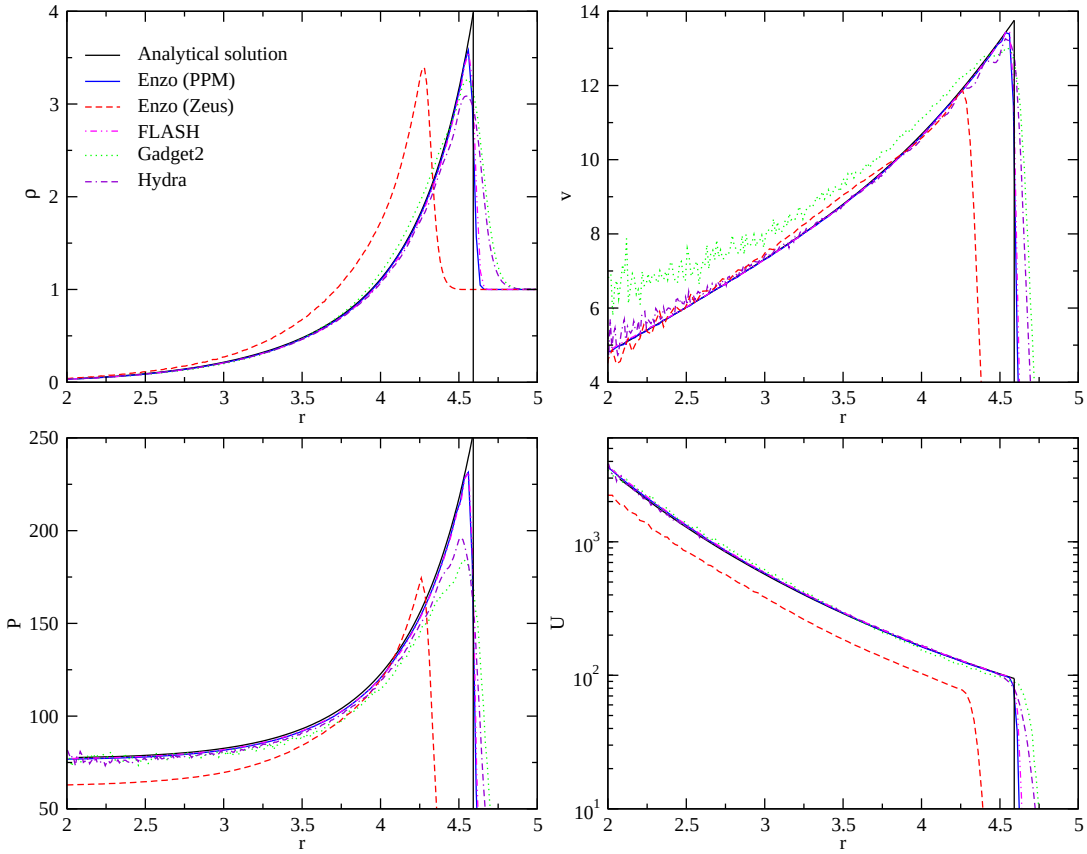

Figure 4.6: Results from the Sedov blast test for each code at $t = 0.1$. Clockwise from top left shows density, velocity, pressure and internal energy. The black solid line marks the analytical solution.

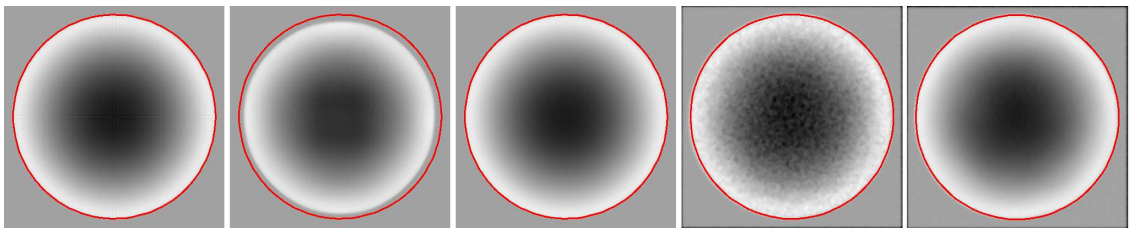

Figure 4.7: Density projections of the Sedov blast test at $t = 0.1$ for all codes, with the expected shock position marked by the red circle. Left to right shows *Enzo (PPM)*, *Enzo (Zeus)*, *Flash*, *Gadget2* and *Hydra*. The density ranges from $[10^{0.6}, 10^{1.23}]$.

We have deliberately chosen a large energy jump for our Sedov blast in order to make this test challenging. A short discussion is added to the end of this section describing some of the pit-falls we encountered while attempting to successfully complete it. What is immediately apparent from Figure 4.6 is that *Enzo (Zeus)* performs significantly worse on this test than all the other codes, with its shock edge lagging substantially behind the analytical position. It shows a significant energy loss in the first few timesteps, loosing 25% of its internal energy during this stage. In the previous section, we saw that *Enzo (Zeus)* spread the shock front out slightly more than *Enzo (PPM)* and *Flash*, which both utilise Godunov schemes, but the difference was minimal and the shock-front was still sharp and well represented. Here, however, we see that *Enzo (Zeus)* underestimates the shock position at $t = 0.1$ by almost 4%. The cause of this significant energy loss is the production of a diamond-shaped, rather than spherical, shock front at very early times. This is discussed further in section 4.4.2.3. Plots of the internal energy and pressure in Figure 4.6 also add to illustrate this effect, showing the effect of *Enzo (Zeus)*'s early energy loss and the resulting low pressure prior to the shock front. Both *Enzo (PPM)* and *Flash* perform this test extremely well, matching the analytical solution to the shock's edge. The two SPH codes successfully recover the location of the peak of the Sedov blast shell but it is smoothed out in the radial direction, producing a lower peak density and a broader shell. For the *Gadget2* run the gas velocity interior to the blast appears to be enhanced in Figure 4.6. This is an artifact of the entropy scatter discussed below. The gas interior to the blast front will rapidly sort itself in entropy, resulting in smooth density and temperature profiles, leaving the velocity profile disordered (and so enhanced).

Figure 4.7 shows density projections of the final state of the simulations at $t = 0.1$. The analytical position of the shock front is shown by the solid red line. As can be seen on Figure 4.7 although the location of the shock front in the *Gadget2* figure is well recovered there seems to be a lot of "noise" producing a grainy appearance to the shaded image. Although energy and entropy are well conserved by *Gadget2* for extreme shocks like this one the *Gadget2* viscosity implementation introduces a lot of entropy scatter. The high entropy particles are too hot for their surroundings and drive small bubbles, producing the structure seen in Figure 4.7. This feature of *Gadget2* does not appear for more normal shock jumps such as that studied in the previous section or for less energetic Sedov blasts. We have tried several approaches to resolve this problem which are mentioned in section 4.4.2.3 below for those interested.

4.4.2.2 Quantitative Comparison

As the previous section showed, four of our codes appear to do a reasonable job of following the location of the Sedov blast front. Due to its inability to conserve energy initially, *Enzo (Zeus)* does not perform this test well and only achieves its current results by adapting code parameters (see "code-specific issues" below). In order to quantitatively compare the codes we have measured the position of the peak of the density, the maximum density obtained and the width of the shock front at half this maximum density as a function of time. The results of this exercise are given in Figure 4.8. We see that *Enzo (PPM), Flash, Hydra* and *Gadget2* are all capable of accurately following the location of the shock front. As is visible on Figures 4.6 and 4.8 the two SPH codes do not obtain the high maximum densities recovered by the AMR codes. *Gadget2*'s maximum peak position is lower than *Enzo (PPM)* and *Flash* by 8% and *Hydra* is 13% less. *Enzo (Zeus)*'s lag in the shock front propagation makes its position equivalent to a time of $t = 0.085$, at which time it is between the PPM-based schemes and the SPH techniques with a top density value 6% below *Enzo (PPM)* and *Flash*.

The lower SPH peak density is not a surprise as the inherent smoothing of the SPH method broadens the shock front and lowers the maximum recoverable density. The AMR codes have the ability to insert additional resolution elements where quantities are rapidly changing rather than where lots of mass has piled up and this acts to improve their fit to the analytic solution in the region of the blast. Despite this disadvantage, the SPH codes correctly reproduce the shock front position and, at around 10%, the differences in their

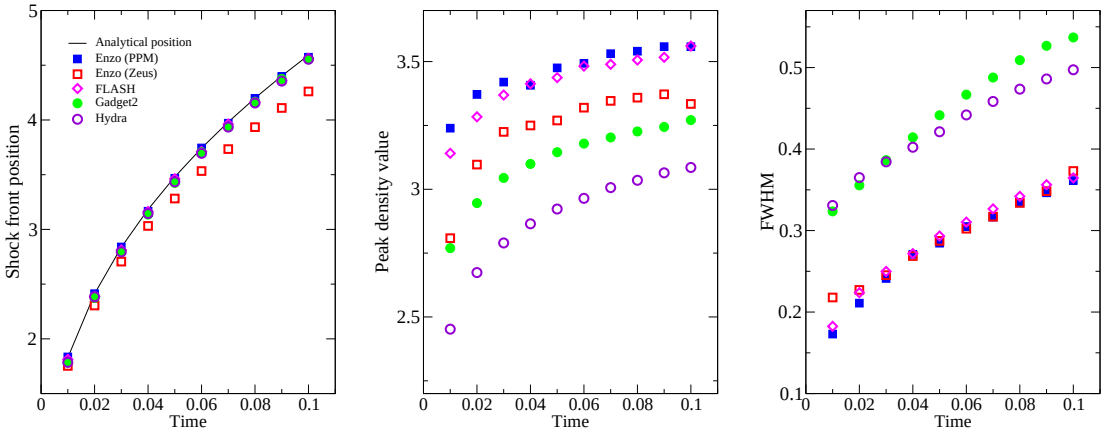

Figure 4.8: For the Sedov blast test, from left to right; position of shock front over time, the maximum density value and the width of the shock front at half maximum.

peak density values are remarkably small.

It is difficult to undertake a "number of particles per grid cell" comparison similar to the one above for this test for two reasons: firstly, the initial conditions rapidly become impossible to set up in a spherically symmetric way for the grid codes if they are not allowed to refine heavily. Our initial condition set-up inserts the energy into a region of radius 3.5 grid cells (or 180 cells). If fewer than this number are used the cells do not represent a spherical energy input well. Without refinement a region encompassing this many cells rapidly becomes large, whereupon the initial conditions no longer represent a Sedov explosion. In addition, small numbers of large cells simply cannot be expected to follow a spherical blast and little useful is learnt. Secondly, the SPH codes are helped at late times because the Sedov blast "sweeps up" the particles it encounters (as it is supposed to) which effectively increases the number of SPH resolution elements in the shock front, as is to be expected given the density enhancement. The AMR codes naturally account for this by adding extra layers of refinement. However, if an unrefined comparison is attempted this high level of refinement has to be present everywhere and at all times, so an incredible number of cells are needed. For this particular test we find that we need static grids of order 250^3 to reproduce a blast with similar resolution to an SPH model with only 100^3 particles. The quantity that should be directly compared is the number of cells per particle in the region of interest. As above we find that roughly one SPH particle per AMR cell is required to obtain similar effective resolutions.

4.4.2.3 Code-Specific Issues for the Sedov Blast

Figure 4.9 shows the density projection of the Sedov blast test performed by *Enzo (Zeus)* at early times. The problem becomes immediately apparent; instead of being spherical, the shock-front is an asymmetrical diamond shape. Over time the shock-front becomes spherical, as can be seen in Figure 4.7, and the dramatic energy loss stops, but by this stage 25% of the shock's energy has been lost. The reason *Enzo (Zeus)* fails to produce a spherical shock-front at early times is not obvious. It appears overly sensitive to the initial set-up not being a perfect Sedov blast start (since a point-like energy injection can only be approximate). Many different solutions were attempted, including injecting the energy as a Gaussian rather than 'Top hat' profile and adding it into a larger radius (which helped, but did not greatly improve the situation). Even to recover the pretty miserable results that *Enzo (Zeus)*achieved above we had to use a somewhat dirty fix of increasing the code's quadratic artificial viscosity term, QV. This value is a broadening

parameter that controls how many cells the shock is spread over. By default $QV = 2$, which is used for the tests in sections 4.4.1 and 4.5. Agertz et al. (2007) found that varying QV made little difference to the evolution of the fluid, but just broadened the shock. In this case, we find that increasing the width of the shock decreases the asymmetry at early times, although Figure 4.9 shows the effect was far from perfect. Using the default value of $QV = 2$, the shock-front lags even further behind the result shown in Figure 4.6, which uses $QV = 10$. The increase in QV comes at a price; the spreading of the shock over more cells weakens it, causing the peak density to be lowered. Raising QV beyond 10 improves the position of the shock further, but the value now becomes unphysically high and we do not recommend using it.

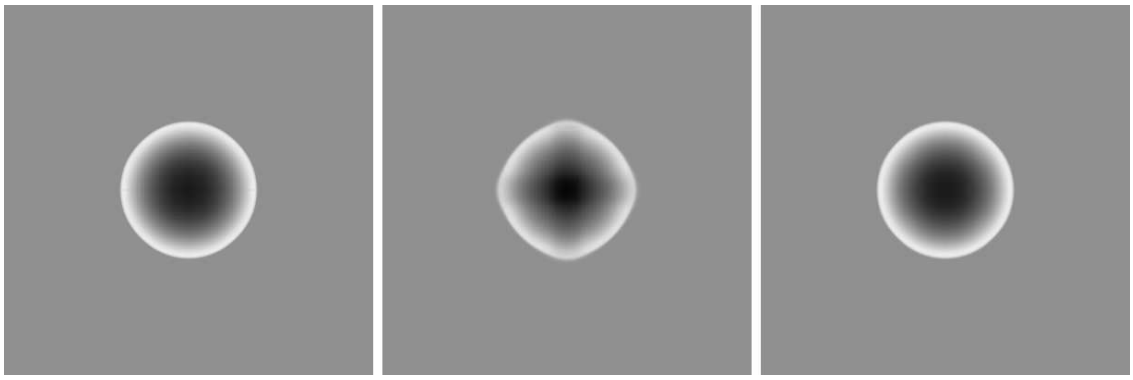

Figure 4.9: Density projection showing the early evolution of the Sedov blast test at t = 0.01 for the AMR codes. Left to right shows *Enzo (PPM)*, *Enzo (Zeus)* and *Flash*. The projected density range is $[10^{0.87}, 10^{1.1}]$. At this time, the asymmetry of the shock-front with *Enzo (Zeus)* is clearly visible. Both SPH codes give spherically symmetric results.

The Sedov blast has already been used by Springel and Hernquist (2002) to test *Gadget2*, and indeed for the parameters Springel and Hernquist (2002) used *Gadget2* works beautifully and does not exhibit any spurious entropy driven bubbles. In order to investigate their origin we tried injecting the energy into a single point particle or smoothing the injection profile using a Gaussian function (with a cut-off at a few percent of its central value) rather than a top hat, and also changing the number of SPH neighbours in order to help numerical convergence. None of these changes makes any practical difference to our test and the entropy driven bubbles still occurred. In all the Gadget2 runs performed for this test, the artificial viscosity parameter ArtBulkVisc has been set to the frequently

used value of 1; there are hints that a choice for a higher value might lead to a sensible reduction of the entropy driven bubbles (see Pakmor et al., in preparation). It appears that in extreme shocks the viscosity implementation employed by *Gadget2* does not sufficiently prevent particle inter-penetration and we warn users of *Gadget2* to be wary of the generation of spurious entropy scatter in the vicinity of extreme shocks.

4.5 Gravitational Tests

In this section, we move away from the formation and resolution of shocks to look at a new aspect of the codes; how they deal with gravity. While treatment of fluids is important, few astrophysical simulations can be performed without a self-gravitating gas. However, adding self-gravity, where every fluid element is affected by every other, dramatically complicates the situation and it is not possible to design a test with an exact analytical solution anymore. Since it is still essential for the purpose of this comparison that our problems remain well-posed, we select situations in which the correct behaviour of the system is known, even if it cannot be mathematically expressed. To do this, we perform two tests; the first of these concerns a static gas profile in equilibrium. Gravity acts to try and collapse the gas, while pressure pushes it outwards. While these forces remain perfectly balanced, the gas remains at rest. This situation is analogous to a relaxed galaxy cluster and requires the code to resolve the gas density over many orders of magnitude. The second test involves the same cluster translating through the box. By using periodic boundary conditions, the cluster's velocity is chosen such that it should return to its original position after 1 Gyr. With no external forces, the cluster should remain in hydrostatic equilibrium and retain its profile during the translation.

4.5.1 Initial Conditions for the Cluster

The model used for the galaxy cluster is the King model (King, 1966; Padmanabhan, 2002), which was chosen because it possesses a finite radial cut-off, and is therefore a well defined problem for a code comparison. Its form is based on the distribution function:

$$f(\epsilon) = \begin{cases} \rho_c (2\pi\sigma^2)^{-3/2} \left(e^{\epsilon/\sigma^2} - 1 \right) & \epsilon \geq 0, \\ 0 & \epsilon < 0 \end{cases} \tag{4.3}$$

where $\epsilon = \Psi - \frac{1}{2}v^2$, is the coordinate change for the shifted energy, ρ_c is the central density and σ is related to (but not equal to) the velocity dispersion. The resulting

density distribution of this cluster vanishes at the *tidal radius*, r_t. Integrating over all velocities yields a density distribution:

$$\rho(\Psi) = \rho_c \left[e^{\frac{\Psi}{\sigma^2}} \mathrm{erf}\left(\sqrt{\frac{\Psi}{\sigma^2}}\right) - \sqrt{\frac{4\Psi}{\pi\sigma^2}}\left(1 + \frac{2\Psi}{3\sigma^2}\right)\right]. \tag{4.4}$$

Putting this into the Poisson equation results in a second order ODE which can be solved numerically. This model has three independent parameters, the mass of the cluster, the tidal radius and the concentration $c = \log_{10}(r_t/r_0)$, where r_0 is the central or *King radius*. For this problem, we selected a concentration of 3, $r_t = 1\,\mathrm{Mpc}$ and a cluster mass of $10^{14}\,\mathrm{M}_\odot$. This results in a King radius of 1 kpc. Therefore, to successfully maintain hydrostatic equilibrium, the codes must be able to model the cluster out to 1 Mpc while resolving the 1 kpc core. This makes it a particularly challenging test.

4.5.2 The Static Cluster

4.5.2.1 Resolution of the Cluster

The two key requirements for success in this test are to be able to resolve the core and to have an accurate gravitational solver. Figure 4.10 shows this case in point. It plots the density profile of the cluster at the start of the simulation and after 1 Gyr for steadily increasing levels of resolution. Although the King model does not have an analytical solution, a one-dimensional numerical solution for the cluster's profile can be achieved from a simple numerical integration. This is shown in each plot as the solid line. In the top left image, the cluster is modelled on a static grid of size 100^3. The initial match to the profile is good up to densities of 10^{-4}, but the inner kiloparsec that contains the core is totally unresolved. The effect is for the pressure to dominate over the gravity and the cluster expands, dropping the density in the core still further. After 1 Gyr, the cluster's profile is barely recognisable, despite the lack of external forces. Moving one plot to the right in Figure 4.10, we see the results of adding a sub-grid into the area that contains the highest density, i.e. the centre of the cluster. The density in the central region is now followed up to $0.15\,\mathrm{M}_\odot\mathrm{pc}^{-3}$. The cluster is still not balanced and deviates away from the profile, but the shift is markedly less. As we continue to add in levels of refinement, we see the central density rise to meet the numerical expectation as the core becomes more resolved. The profile of the cluster changes progressively less until we reach five levels of refinement, when the change becomes almost undetectable except at the very centre.

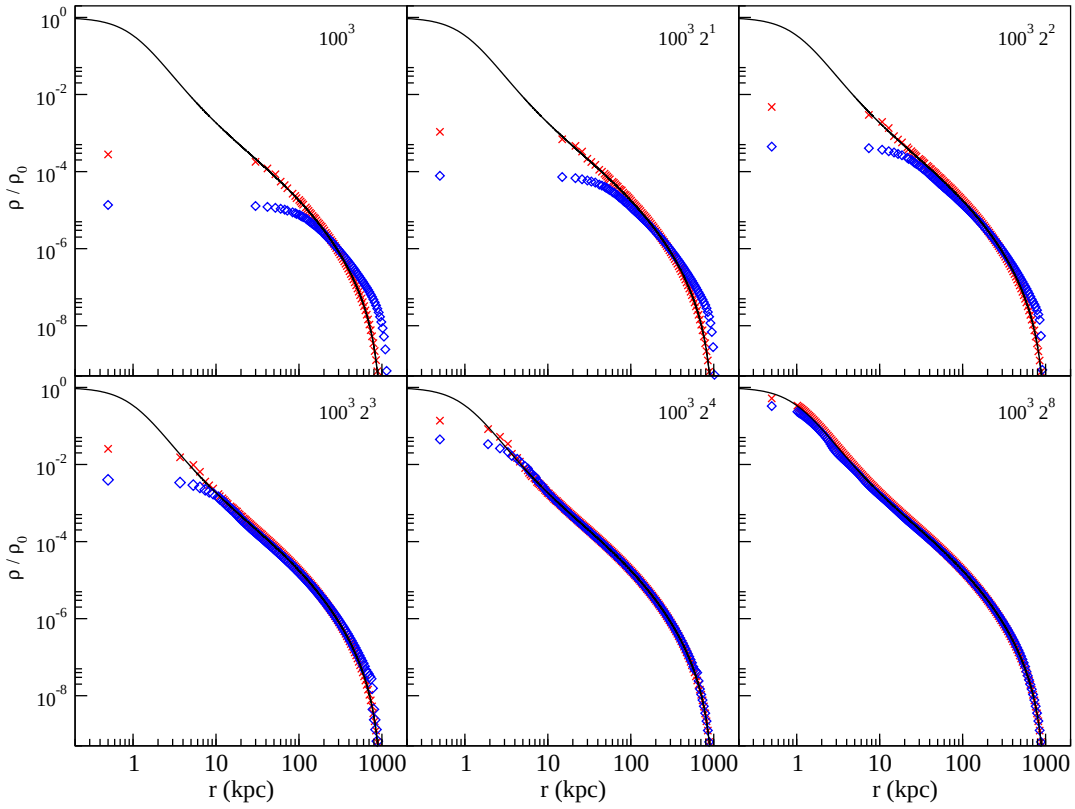

Figure 4.10: Density profiles of the static King-model cluster run with *Enzo (Zeus)* with increasing resolution. The solid line shows the 1D calculated numerical solution for the profile, with the red crosses and open blue boxes indicating the simulated profile at 0 Gyr and 1 Gyr, respectively. The number to the top right of each panel indicates the size of the root mesh and the number of refinements (i.e. the bottom right panel has eight levels of refinement each increasing in resolution by a factor of 2).

4.5.2.2 General Results

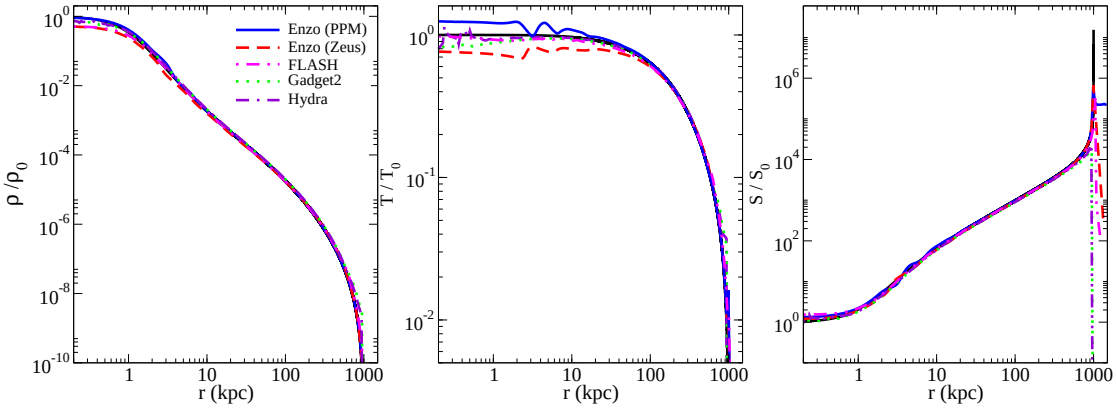

Figure 4.11: Profiles from the static King-model cluster after 1 Gyr. Left to right shows density, temperature and entropy. The black line shows the expected profile. In this test, *Enzo* used an inital grid of 100^3 and 8 levels of refinement (minimum cell size, $\Delta x_{\rm min}$, of 0.12 kpc), *Flash* used an initial grid of 128^3 with 7 levels of refinement ($\Delta x_{\rm min} = 0.18$ kpc) and *Gadget2* and *Hydra* used 100,000 particles.

Figure 4.11 shows the density, temperature and entropy profiles for the cluster after 1 Gyr. For this test, *Enzo* was run with an initial grid of 100^3 and eight additional levels of refinement, each reducing the cell size by a factor of two. These sub-grids were placed anywhere where the cell mass was above a critical value. The cluster was set up in a box of size 3 Mpc with isolated gravitational boundary conditions. This gave a minimum cell size in the core of 0.12 kpc. *Flash* ran with the same boxsize and with periodic boundary conditions. It used a slightly larger initial grid of 128^3 (since the gravity solver requires factors of two) and included seven additional levels of refinement, each of which reduced the cell size by a factor of two. This gave a minimum cell size in the core of 0.18 kpc. *Hydra* and *Gadget2* radially perturbed a glass of 100,000 particles to the desired density profile within a periodic 3 Mpc box.

As can be seen in the density profile in Figure 4.11, with sufficient central resolution all the codes are successful at keeping the cluster in equilibrium and resolve the core well. All the codes show negligible deviations from the static King profile after 1 Gyr, matching the analytical profile over seven orders of magnitude, down to densities of 10^{-10}. At the low density edge of the cluster, small deviations ($\sim 15\%$) from the profile are seen as the cluster edge diffuses into the background. *Enzo (PPM)*, *Enzo (Zeus)* and *Flash* produce

the closest fit to the numerical solution at these densities, owing to the larger choice of initial grid which fixes the minimum resolution. *Hydra* and *Gadget2* show small deviations close to the cluster's tidal radius as they start to run out of particles.

Unlike the shock tests in section 4.4, all our codes in this test and in section 4.5.3 refine based on increasing mass. Although grid based codes can make resolution choices from other parameters, mass in a grid cell (i.e. density) is the most used for simulations on scales above a few parsecs. However, in contrast to the particle based codes, a grid code's minimum resolution is always known, since it is fixed by the specified minimum grid size. Depending on the simulation type, a fixed minimum resolution is either a huge advantage or a drain on computational time. In simulations where the densest structures are the primary focus, SPH's ability to sweep all the particles into these areas means that high resolution can be gained quickly and efficiently. To gain the same efficiency from a grid code, a careful choice of minimum resolution grid and refinement criteria must be made. Both the test in this section and the following one are examples of this problem type and were significantly more time consuming for grid based codes than for particle based ones. This is in contrast to the shock tests which ran faster. However, if the focus of a simulation is the surrounding structure of a dense object, then the ability to specify a minimum resolution is extremely useful and time efficient. In a particle scheme this would normally be obtained by adding enough additional particles such that both the dense and surrounding medium were resolved; a solution which may significantly add to the expense of the simulation.

Figure 4.11 demonstrates the two SPH codes achieve a similar result with 100,000 (46^3) particles to that obtained by the AMR codes with initial meshes of around 1 million cells and 8 or 9 levels of refinement. To achieve this with a static mesh would be impossible, as the mesh would need to be $25,600^3$. Not surprisingly then, given the complication of placing and interpolating between refinements the SPH codes are much faster. We note that in the SPH runs the distance from the centre to the 32nd particle is 0.4 kpc, which is very close to twice the minimum cell size of the AMR runs in the core. Again, we see that in order to obtain convergence between the AMR and SPH codes we require roughly one particle per cell in the region of interest. Unfortunately for the AMR codes, for this particular problem which includes a wide density range the ability of SPH codes to naturally increase resolution with mass is very advantageous.

Finally for this section, Figure 4.11 also clearly shows that both of the *Enzo* implementations include noticeable ringing in the temperature profile which is reflected in the

entropy and less obviously density profiles. This causes a deviation from the analytical solution of around ±20% in the inner 2 kpc (10 cells). This ringing is not present in the analytic solution and doesn't appear in either *Flash* or the SPH codes which closely resemble both each other and the calculated solution. The exceptional agreement between *FLASH* and the other codes clearly indicates how well the new hybrid FFT-multigrid solver works and reassuringly there is no observable distortion to the profile from the large scale deviations seen in chapter 3.

4.5.3 The Translating Cluster

Using the stable clusters developed above we can test the Galilean invariance of the codes by giving them a velocity relative to the static simulation volume. This is a commonly encountered situation for cosmological simulations where large objects often move at many hundreds of kilometres per second relative to the background. At sufficiently high velocities it is well known and straightforward to demonstrate that mesh based codes are not Galilean invariant, whereas particle based methods are (e.g. Wadsley et al., 2008). What we wish to investigate is the size of these effects for typical astrophysical objects moving at typical astrophysical velocities.

For this test the cluster was given a bulk velocity such that, in 1 Gyr, it moved around the simulation box once, returning to its original starting position. This is shown visually in Figure 4.12 which also demonstrates the box's boundary conditions which are periodic. Since there are no external forces acting on the cluster, the end profile should be identical to the initial one.

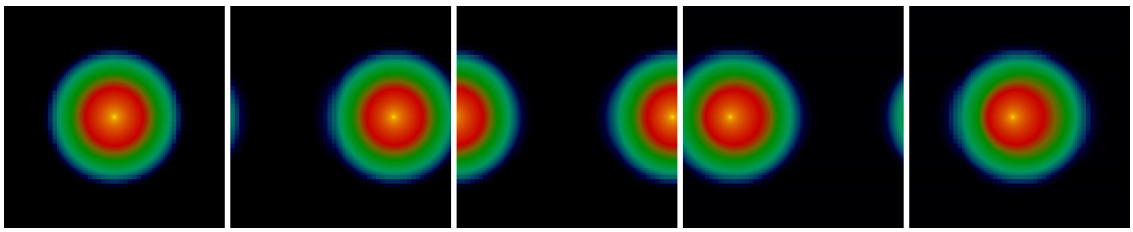

Figure 4.12: Density projections of the cluster over the course of 1 Gyr in which is moves once around the simulation box. Images taken at 0, 250, 500, 750, 1000 Myrs with projected density range $[10^{8.4}, 10^{16.5}]\,\mathrm{M}_\odot\,\mathrm{Mpc}^{-2}$. Yellow and red shows higher density regions than green while black is very low density. (Images produced with *Enzo (Zeus)*).

The time consuming nature of this test for grid codes also became an issue. *Enzo (Zeus)*

and *Enzo (PPM)* lowered their initial grid to 50^3, but added an additional level of refinement maintaining a minimum cell size of 0.12 kpc. *Flash* managed, after a long run, to maintain the same resolution as for the static test, with a minimum cell size of 0.18 kpc. The two SPH codes also had the same set-up as for the static case.

Figure 4.13 shows the residue remaining if the initial configuration is subtracted from the final one. Another fundamental difference between grid based and particle based techniques is visible here in the background medium. Unlike a particle code, grid codes cannot have zero density and energy cells. Therefore, a cool, low density gas surrounds the cluster in this test when run by both *Enzo* and *Flash*. In the case of Figure 4.12, which only contains results from *Enzo (Zeus)*, the minimum value shown is the background density, which is therefore displayed in black. From looking at the cluster core in Figure 4.13, clearly neither of the *Enzo* simulations returns to the correct position, with a particularly significant lag in the case of *Enzo (PPM)*. In both these cases, the cluster as a whole correctly returns to the centre of the simulation box, but the central core does not, leaving it off-set from the centre of the cluster. This can be seen clearly in the last image of Figure 4.12. After 1 Gyr, *Enzo (PPM)* has a core 176 kpc from the original start position while *Enzo (Zeus)* finishes with its core 93 kpc short of the initial position. The outer regions of the cluster also show distortion due to the lower resolution of the initial grid. These problems don't appear to occur for either the SPH codes or *Flash*. *Flash* shows a small amount of distortion in the outer region of the cluster, although the central region stays largely intact with the core only 1.7 kpc from the centre of the box. Both SPH codes show very similar results with *Hydra* getting its core closest to the original start position with a 0.4 kpc off-set and Gadget2 with a 25 kpc off-set. In lower density regions, we see some distortion to the cluster's structure which becomes more marked at higher radii. This is due to the number of particles significantly decreasing as we move away from the core.

Figure 4.14 is equivalent to Figure 4.11 except that now the cluster has moved once around the periodic box. All codes maintain the density profile of the cluster extremely well, with the only deviations appearing in the core. Here, *Gadget2* performs best, resolving the core in good agreement with the analytical prediction. All other codes struggle to resolve the core after 1 Gyr, with *Flash* struggling the most and underestimating the core density by a factor of 10. Lower resolution runs (minimum cell size of 0.23 kpc) using *Enzo* also showed a similar drop in core density in this test, whereas a much smaller difference was noticed for the static cluster at the same resolution. This change in core density

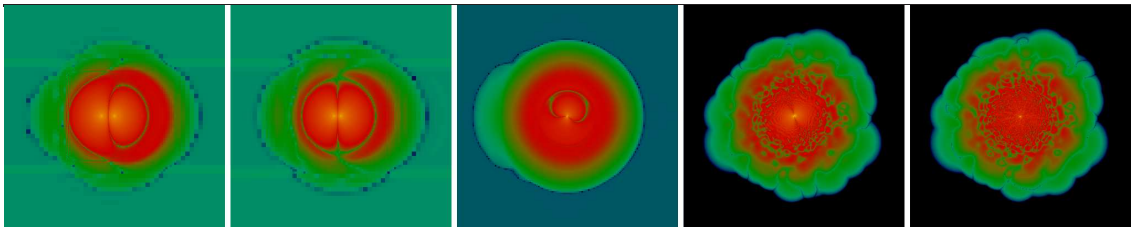

Figure 4.13: Image subtractions of the density projections at the start and end of the translating cluster test. From left to right shows *Enzo (PPM)*, *Enzo (Zeus)*, *Flash*, *Gadget2* and *Hydra*. The projected density range is $[10^4, 10^{18.6}]\,\mathrm{M}_\odot\ \mathrm{Mpc}^{-2}$.

was seen, but not understood, in the Santa Barbara Comparison Project (Frenk et al., 1999) which observed the lower cluster densities in AMR codes. Without an analytical prediction for what the correct density should be, it was impossible to determine which of the recorded densities was more accurate. This test highlights two important effects controlling these core properties; resolution and advection. In the first instance, it is important to resolve the cluster core otherwise its size will be overestimated, leading to a lower central density. This is demonstrated clearest in Figure 4.10 where in the case with four levels of refinement, the cluster remained intact but the central density still dropped. In the second case, the explicit movement of the fluid through the grid (i.e. advection) in AMR can (but not necessarily does) cause a drop in the central density. This is an explanation for why the static and translating king models differ in the AMR case, even at the same resolution. We see from Figure 4.14 that *Flash* suffers most from advection issues whereas *Enzo (PPM)* shows minimal difference from the static model, following the analytic profiles at least as well as the SPH codes. In both these areas, SPH's ability to follow the mass allows it to excel quickly and efficiently compared with AMR codes who have to be more careful when selecting their refinement criteria.

While this test provides clues towards the deviation in cluster properties in the Santa Barbara Comparison Project, it should be noted that other effects can also be at work. In cosmological models, for instance, turbulence plays an important role as investigated by Wadsley et al. (2008) and which I myself highlight in chapter 6. The difference between the entropy cores in the AMR code *Enzo* and the SPH code *Hydra* in figure 4.14 for instance are almost identical and clearly do not indicate the large entropy difference seen by Frenk et al. (1999). I therefore extend this work in chapter 6 to investigate the effects of Galilean non-invariance for a multitude of advection geometries and find that although non-zero, these contributions, along with the effects of limited gravitational force resolution, are

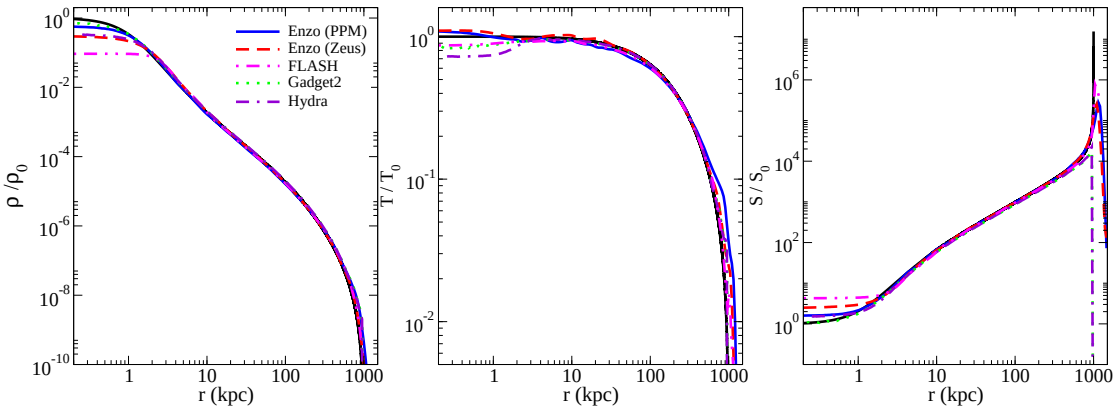

Figure 4.14: Profiles from the translating King-model cluster after 1 Gyr. Left to right shows density, temperature and entropy. The black line shows the expected profile. In this test, *Enzo* used an initial grid of 50^3 and 9 levels of refinement (minimum cell size, $\Delta x_{\rm min}$, of 0.12 kpc), *Flash* used an initial grid of 128^3 with 7 levels of refinement ($\Delta x_{\rm min} = 0.18$ kpc) and *Gadget2* and *Hydra* used 100,000 particles.

not the root cause of the entropy cores seen in the Santa Barbara Comparison Project.

The temperature plot in the second panel of Figure 4.11 shows that *Flash*, *Gadget2* and *Hydra* all underestimate the core temperature whereas *Enzo (PPM)* and *Enzo (Zeus)* overestimate it slightly. Additionally, all the AMR code overestimate the cluster temperature at its edge with *Enzo (PPM)* having the poorest fit. The core's entropy is also overestimated in the case of the *Flash* and *Enzo (Zeus)*; unsurprising since this depends on the inverse of the density.

4.5.3.1 Gravitational Tests: Code-Specific Issues

For both the static and translating cluster tests presented above, *Enzo* was run on a single processor. When using the parallelised version of *Enzo*, the gravity solver suffers from excess noise in the velocity motions which mix the gas, reducing the entropy in the cluster centre. The result is a density drop in the cluster core and an expansion of its outer regions, unrelated to the issues discussed in section 4.5.3. The resulting density profiles using the parallelised version of *Enzo* are shown in Figure 4.15. In other situations (e.g. Tasker and Bryan, 2008) the gravity solver has been tested successfully to give identical results in serial and parallel, but an improved, less noisy, gravity solver is currently being developed to solve this problem where is occurs.

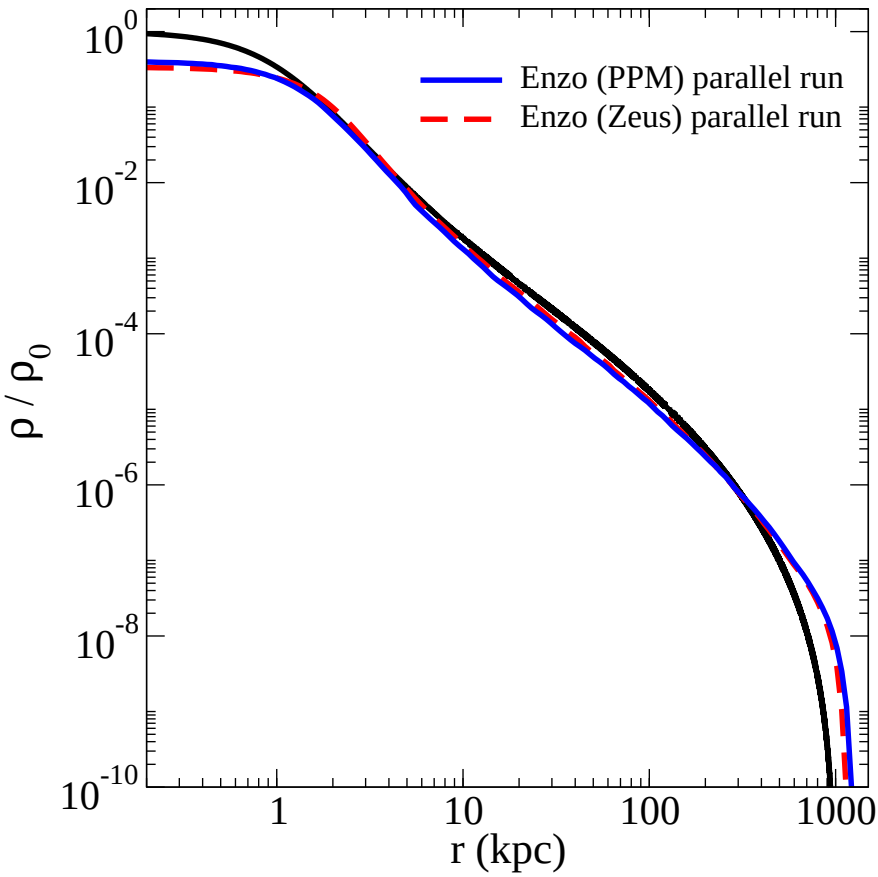

Figure 4.15: Density profile from the translating king model after 1 Gyr run with *Enzo* in parallel. Run on multiple processors, the spurious velocity motions cause the gas to become over-mixed and the cluster density to drop in the core and expand at the cluster edge. The black line shows the expected profile.

Enzo (PPM) also experienced problems dealing with the background medium. The sharp drop in conditions between the cluster edge and low density background gas can cause *Enzo (PPM)* to artificially produce negative densities and energies. This was corrected for by setting a minimum density of $0.1\,\mathrm{M}_{\odot}\mathrm{Mpc}^{-3}$, which negative densities were automatically set to. Having to approximate densities multiple times during the run is a likely reason for the momentum loss that is clearly observed in Figure 4.14.

The default multigrid algorithm used in the publicly available version of *Flash 2.5*, proved too slow to be used practically. In order to overcome this we implemented a hybrid FFTW based multigrid gravity solver, resulting in a vast reduction in the runtime of the simulations. The latest version of *Flash 3.0* includes a more efficient multigrid algorithm and a nested FFTW gravity solver is currently under development for release in the near future (similar to the one used here).

For *Gadget2* we encountered a problem in that although the radial profiles were very stable the cluster as a whole tended to drift around over time. This is due to the difficulty of very accurately determining the lowest order term in the gravitational force expansion for a tree code. Each individual term includes a small error, with these errors largely but not exactly uncorrelated. Under these conditions the total momentum of the system is not guaranteed to be conserved exactly and a "random walk" occurs. As the configuration is designed to be entirely static the direction of the residual force is highly correlated from one step to the next and despite recovering the correct value to one part in 10^8 this still leads to a net drift. It is possible to circumvent this, as shown above, by dramatically reducing the opening angle for the tree but this rapidly negates the advantage of using a tree in the first place. For more normal simulations this tiny zeroth order force error is of course negligible as the random motion of the particles disorders the direction of the drift error as time progresses. P^3M based gravity solvers like that employed by *Hydra* do not suffer from this problem as the zeroth order term in the Fourier transform is automatically zero.

4.6 Discussion and Conclusion

We have presented four well posed tests with known solutions which we have used to compare four well used astrophysical codes; *Enzo, Flash, Gadget2* and *Hydra*. We have examined each code's ability to resolve strong shocks and incorporate an accurate gravity solver capable of resolving forces over many orders of magnitude in density.

The tests presented here were specifically designed to be difficult and push the codes to their limit; appropriately so since many situations in astrophysics demand extreme physical conditions. Despite these requirements, all the codes did pass our tests satisfactorily and what we highlight here is the strengths and weaknesses of the different techniques.

Significant issues were *Enzo (Zeus)*'s failure to produce a spherical shock, a problem that leads to a lack of energy conservation and to poor recovery of the position of a Sedov blast. For extreme shocks the artificial viscosity implementation of *Gadget2* leads to spurious entropy driven bubbles and for very static configurations the centre-of-mass can slowly drift due to the difficulty of recovering the zeroth order term in the tree force. For a translating cluster, both implementations of *Enzo* appear to lose momentum over time.

All our tests lead to the conclusion that in order to achieve roughly the same resolution in grid and particle codes a good rule of thumb is to require one particle per cell in the region of interest. This is difficult to achieve for the SPH codes over shock fronts or voids and likewise harder for AMR codes to match in the centre of collapsed objects. This is reflected strongly in our results where the AMR codes largely excel at the shock tests in section 4.4 and the SPH codes at the gravitationally bound cluster tests in section 4.5. For a cosmological simulation these criteria would equate to having the minimum mesh cell size about the same size as the gravitational softening or half the SPH search length.

As implied by the relation above, for strong shocks SPH codes require significantly more effort than AMR codes. This is because AMR codes have the ability to add extra resolution in regions of rapidly changing density, whereas SPH codes can effectively only refine with the density itself, so they end up undertaking lots of unnecessary work far from the shock where the forces are small in regions of high but uniform density. However, for models such as the King sphere, where AMR codes traditionally refine using a density criterion anyway, SPH codes can achieving high resolution far more efficiently in these dense structures because they do this very naturally by following individual particles. For AMR codes to achieve similar results, high resolution and a significant increase in computational time is required. In these cases AMR codes would still have an advantage if it was necessary to resolve the low density region, where the SPH codes simply run out of particles and have to smooth over very large volumes.

Additional care must be taken for the parallel implementations of *Hydra* and *Enzo*. *Hydra*'s parallel version was not used in this chapter and further testing would be needed to ensure that it performs correctly. The gravity solver in *Enzo* was demonstrated to

produce different results in serial and parallel in the King model tests (although not in other situations) and this should be explicitly tested when running a simulation with this code. An updated gravity solver for *Enzo* which rectifies this problem is currently being developed. While serial runs allow much parameter space exploration, rapid code development and inter-comparison, true state-of-the-art large simulations are confined to codes that can utilise high performance computing facilities and successfully run on a large number of processors.

To a large extent the choice of simulation code largely comes down to the problem being attacked. For problems with large dynamic range or when the object is rapidly translating across the volume particle based methods are much less computationally expensive in order to achieve the same result. Conversely, for problems where the interesting physics is in regions of rapidly changing density rather than in high density regions themselves, AMR codes excel thanks to their more adaptive refinement criteria.

The computational time required for each code to perform these tests varied greatly. Ideally, elapsed time is largely irrelevant since the required resolution and problem size should be set by the appropriate physics being tackled rather than available computer resources. However, we include Table 4.2 for completeness and to give the reader an indication of times involved. In Table 4.2, the computational systems referred to are; 'UF HPC' is the University of Florida's high performance computing center which uses (typically) InfiniBand-connected 2.2 GHz AMD Opteron duel cores with 6 Gb RAM. 'UF Astro' is the University of Florida Astronomy's minicluster with 2.66 GHz Intel quad-cores and 4 Gb RAM. 'Columbia' is the Columbia University Astronomy's cluster with gigabit-connected 2.2 GHz Opteron cores and 3 Gb RAM. 'Miranda' is at the University of Durham and consists of Myrinet-connected 2.6 GHz Opteron cores with 4 Gb RAM. 'Nottingham' is the Nottingham University HPC with gigabit-connected 2.2 GHz Opteron cores with 2 Gb of RAM. Readers should note that comparing CPU times between simulations performed on disparate computers is fraught with danger. For true runtime performance, the codes would need to be bench marked on identical systems which was beyond the scope of this chapter. We therefore include this table purely for guidance and warn against over interpretation. † The times quoted for the static and translating cluster runs in *Flash* are estimated total runtimes based on the speed up attained with the latest version of the FFTW gravity solver. The original runs were performed using an early version of the hybrid FFTW gravity solver which was still under development and as such was not optimised. The original runs took place on 24 processors and runtimes

for the pre-development level FFTW solver were 217 hrs 42 mins for the static cluster and 564 hrs 5mins for the translating cluster. This drastic improvements arises from a decrease in the memory requirements and improved communication patterns. A reduction in computational time in the Sedov blast test was additionally achieved by using a simplified Riemann solver that was not required to follow multiple fluids with different adiabatic indices (a feature the default configuration of *Flash* has implemented to model thermonuclear flashes). This reduced the memory overheads.

While this chapter attempts to cover the major features of astrophysical simulations, it does comprise only four tests. Further examples would extend this chapter beyond its scope (and readability), but the differences between codes cannot be fully catalogued without further testing. This chapter then, is designed as a starting point for a suite of tests to be developed from which codes can be quantitatively compared for the jobs they are intended for. To assist groups wishing to run these tests on their own code and to compare new updates, we are making these results and initial conditions available on the web at `http://www.astro.ufl.edu/codecomparison`.

Table 4.2: Computational time to perform the presented tests, including the number of processors used and the type of system.

Riemann Shock Tube: Planar	hh:mm	no. processors	system
Enzo (PPM)	00:36	16	UF HPC (Opteron)
Enzo (Zeus)	04:00	1	UF Astro (Intel Xeon)
Flash	00:24	32	Miranda (Opteron)
Hydra (10^6 particles)	00:36	1	Nottingham (Opteron)
Gadget2 (10^6 particles)	01:31	2	Nottingham (Opteron)
Riemann Shock Tube: Oblique			
Enzo (PPM)	01:22	16	UF HPC (Opteron)
Enzo (Zeus)	05:06	8	UF HPC (Opteron)
Flash	00:26	45	Miranda (Opteron)
Hydra (10^6 particles)	00:36	1	Nottingham (Opteron)
Gadget2 (10^6 particles)	01:30	2	Nottingham (Opteron)
Sedov Blast Test			
Enzo (PPM)	00:24	16	UF HPC (Opteron)
Enzo (Zeus)	04:41	6	UF HPC (Operton)
Flash	15:33	16	Miranda (Opteron)
Hydra	06:49	1	Nottingham (Opteron)
Gadget2	19:41	4	Nottingham (Opteron)
Static Cluster Test			
Enzo (PPM)	61:50	1	UF HPC (Opteron)
Enzo (Zeus)	23:49	1	UF HPC (Opteron)
Flash	111:01 †	12	Miranda (Opteron)
Hydra	37:51	1	Nottingham (Opteron)
Gadget2	09:48	4	Nottingham (Opteron)
Translating Cluster Test			
Enzo (PPM)	78:50	1	Columbia (Opteron)
Enzo (Zeus)	70:45	1	UF Astro (Intel Xeon)
Flash	374:12 †	12	Miranda (Opteron)
Hydra	39:23	1	Nottingham (Opteron)
Gadget2	09:18	4	Nottingham (Opteron)

Chapter 5

Ram Pressure Stripping the Hot Gaseous Halos of Galaxies in Groups and Clusters

5.1 Foreword

Material in this chapter was published as part of the McCarthy et al. (2008b) paper. The *GADGET-2* simulations along with the derivation of the analytic ram pressure stripping formulae were developed by my co-authors whilst all material pertaining to the *FLASH* code was undertaken exclusively by myself.

5.2 Introduction

Despite successes of semi-analytic models, there still exist many discrepancies between observation and theory still to be explained. One such discrepancy arises when looking at the colour of galaxies within clusters, many of which in semi-analytic models are predicted to be far redder than observations (see, e.g., Weinmann et al., 2006; Baldry et al., 2006). It is believed that this may arise through the over simplified instantaneous ram pressure stripping prescription that is presently adopted. This assumes that when galaxies fall in and cross the virial radius of the galaxy group or cluster, their hot gaseous halo is immediately stripped off through ram pressure. This removal of gas for star formation can lead to the stellar population becoming dominated by old red stars. Many studies have looked at the stripping of cold gas from disk galaxies when they fall into cluster environments (see, e.g., Quilis et al., 2000; Okamoto and Nagashima, 2003; Roediger and Brüggen, 2006, 2007, 2008) as a means to account for the observed low star formation

rates and dominant red stellar populations. However little work has been done relating to the hot gaseous halo that are found to exist around most cluster galaxies (Sun et al., 2007). As the hot gaseous halo presents a large reservoir of gas which through radiative cooling can be used to form luminous young blue stars, it presents a means to prolong the star formation in galaxies once they have fallen into the cluster.

In this chapter I begin by outlining the initial conditions for the model galaxies which we subject to ram pressure forces in both uniform density runs and by sending them on orbits through a live galaxy group or cluster halo (§ 5.3). I also describe the general technique used in the SPH simulations to determine the mass loss rate due to ram pressure stripping (§ 5.3.1).

In § 5.4 I derive our simple analytic formula to describe the ram pressure stripping of a spherical gas cloud as it is subjected to a wind. Our model makes a series of assumptions which allow it to neglect contributions to the stripping of material due to hydrodynamic instabilities. In order to confirm that these assumptions are upheld, we restrict the galaxy to group mass ratio such that the Kelvin-Helmholtz timescale is much larger than the duration of our simulations (§ 5.5.1). As the SPH simulations used are known to spuriously suppress the formation of turbulent instabilities, I perform identical ram pressure stripping simulations in *FLASH*. By contrasting the results with those in *GADGET-2* I am able to confirm whether hydrodynamic instabilities have time to develop and offset our results or whether both codes give good agreement within our carefully selected regime (§ 5.5.2).

By performing a series of simple three dimensional simulations in which a galaxy travels through a uniform density medium, it is possible to fix the parameters in our analytic model. A suite of SPH simulations are then used to test the robustness of the analytic model for a wide range of orbits, velocities and mass ratios between the galaxy and host cluster, which I summarise in § 5.6. I conclude our findings in § 5.7.

5.3 Initial Conditions for the Simulations

The galaxies in this study are represented by spherically-symmetric systems composed of a realistic mixture of dark matter and diffuse baryons. The dark matter is assumed to follow the NFW distribution (Navarro et al., 1996, 1997):

$$\rho(r) = \frac{\rho_s}{(r/r_s)(1 + r/r_s)^2} \tag{5.1}$$

where $\rho_s = M_s/(4\pi r_s^3)$ and

$$M_s = \frac{M_{200}}{\ln(1 + r_{200}/r_s) - (r_{200}/r_s)/(1 + r_{200}/r_s)} \quad . \tag{5.2}$$

Here, r_{200} is the radius within which the mean density is 200 times the critical density, $\rho_{\rm crit}$, and $M_{200} \equiv M(r_{200}) = (4/3)\pi r_{200}^3 \times 200\rho_{\rm crit}$.

The only 'free' parameter of the NFW profile is the scale radius, r_s which can be expressed in terms of a concentration parameter, $c_{200} \equiv r_{200}/r_s$. By default, we adopt the mean mass-concentration ($M_{200} - c_{200}$) relation derived from the *Millennium Simulation* (Springel et al., 2005b) by Neto et al. (2007). This is similar to the earlier relationship derived by Eke et al. (2001).

For simplicity we initially assign the baryons in the halo the same density profile as the dark matter and fix the ratio of the gas to total mass to the universal ratio $f_b = \Omega_b/\Omega_m = 0.022h^{-2}/0.3 = 0.141$. Here, h is the Hubble constant in units of 100 km s^{-1} Mpc^{-1}. The other properties of the diffuse gas (i.e., temperature and pressure profiles) are fixed by ensuring the gas is initially gravitationally bound and in hydrostatic equilibrium,

$$\frac{dP(r)}{dr} = -\frac{GM(r)}{r^2}\rho_{\rm gas}(r) \quad . \tag{5.3}$$

Although the assumption that the baryons trace the dark matter is reasonable for most of the baryons in the cluster, in practice due to physical effects not modelled here such as star formation and feedback, it will definitely deviate within the galactic core. Despite not modelling the physics within the core, the aim is more to produce a generic model for ram pressure stripping that can be applied to a wide range of scenarios with the free parameters being fixed through simulation.

We implement two suites of simulations, the first involves the passing of a galaxy through a uniform medium. As these runs are simplified, we also run a second suite of "2-system" runs. These 2-system runs involve the passing of the galaxy through a live galaxy group or cluster environment using a series of velocities, orbital parameters and system masses. The more massive system is set to have a total mass of $M_{200} = 10^{14}{\rm M}_\odot$, while the less massive systems have masses in the range $2 \times 10^{12}{\rm M}_\odot \leq M_{200} \leq 10^{13}{\rm M}_\odot$ (i.e., mass ratios from 50:1 to 10:1). These are equivalent in mass to a normal elliptical galaxy falling into a moderate group or low mass cluster.

The default gas particle mass in the SPH runs, $m_{\rm gas}$, is set to $2 \times 10^8 f_b$ M$_\odot$, while the default dark matter particle mass, $m_{\rm dm}$, is set to $2 \times 10^8(1 - f_b)$ M$_\odot$. These masses are

fixed for the 2-system runs for both the galaxy and the group into which they fall giving half a million particles within r_{200} each. A gravitational softening of 5 kpc is adopted for all of our runs and note that our results are robust to varying of the softening length.

Standard SPH parameters (see, e.g., Springel, 2005) are adopted with 32 neighbours within the particle smoothing length. Although an artificial viscosity of 0.8 is adopted, the results are shown to be consistent for a variety of reasonable viscosities between 0 and 0.8 (see McCarthy et al., 2008b, for more details).

The simulations are run for a maximum duration of 10 Gyr in the case of the 2-system runs or until a convergent result is achieved in the case of the uniform medium runs.

5.3.1 Measuring the Ram Pressure Stripping

The standard approach for measuring the ram pressure stripping in the SPH runs involves computing the mass that remains gravitationally bound to the galaxy as a function of time. We determine the number of particles (both gas and dark matter) that are bound according to the iterative technique outlined in Tormen et al. (1998) and Hayashi et al. (2003).

We start with an initial galaxy in which all particles are initially bound. As the simulation progresses we recalculate the potential, kinetic and for the gas, the internal energies of each of the particles. As the total kinetic and internal energy exceeds the potential energy, we discard it. Once a particle has been stripped by this means we do not allow it to rejoin the galaxy at later times and thus the bound mass of particles must monotonically reduce with time. This ensures that our bound gas mass at the end is a conservative lower limit.

5.4 An Analytic Model for Ram Pressure Stripping

The prescription we use is similar to the seminal work of Gunn and Gott (1972) which described the loss of gas when a disk galaxy travels face on through the ICM. The ram pressure of the wind is defined as $P_{ram} = \rho_{ICM} v^2_{\rm orb}$ where ρ_{ICM} is the density of the ICM and v^2_{orb} is the velocity of the galaxy with respect to the ICM as it passes along its orbit. They derive the restoring gravitational force per unit area on the disk to be $2\pi G \Sigma_* \Sigma_{\rm gas}$ where Σ_* and $\Sigma_{\rm gas}$ are respectively the stellar and gaseous surface densities of the disk. When the ram pressure exceeds the restoring gravitational force in the disk, material will be stripped off of the galaxy. As the outer regions are less tightly bound than those on

the inside, it therefore makes sense that ram pressure stripping affects the outskirts of the disk most.

We prolong this model to that of a spherically-symmetric gaseous halo by considering gas in a projected annulus between radii R and $R + dR$ (see figure 5.1). The projected area of this annulus, dA, is $2\pi RdR$ giving a ram pressure force on the annulus of $F_{\rm ram} = P_{\rm ram}dA$.

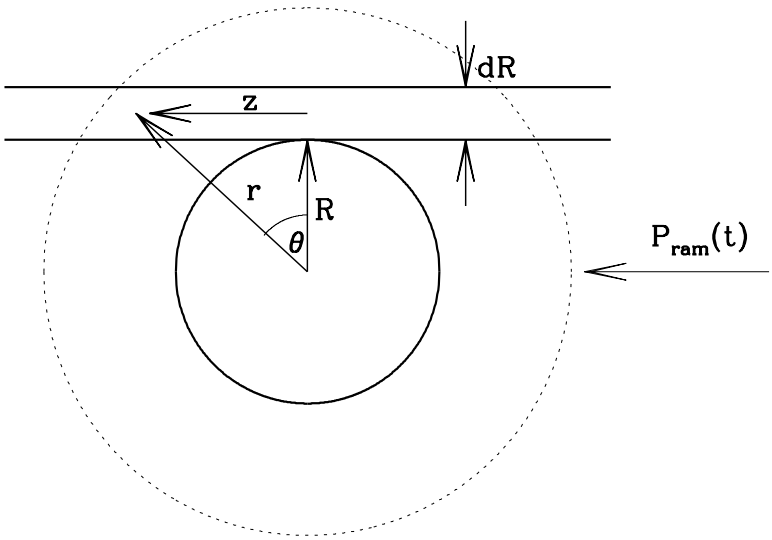

Figure 5.1: A schematic diagram of the ram pressure stripping of a spherically symmetric gas distribution. Here, the ram pressure force is directed from left to right and we consider the ratio of the ram pressure force to the gravitational restoring force per unit area for a projected annulus of width dR at the outer edge (radius R) of the gaseous halo of the galaxy (reproduced from McCarthy et al., 2008b).

The applied ram pressure force acts in the "z-direction," opposite to the motion of the galaxy through the ICM; $v_{\rm orb}$. The maximum gravitational restoring force, $F_{\rm grav}$, can be written approximately as $g_{\rm max}(R)\Sigma_{\rm gas}(R)dA$, where $g_{\rm max}(R)$ is the maximum restoring acceleration in the z direction and $\Sigma_{\rm gas}(R)$ is the projected surface density of the gas in

the annulus. If the force due to ram pressure exceeds the maximum gravitational restoring force then the annuli will be displaced in the z-direction. We can therefore express the ram pressure stripping condition as:

$$\rho_{\rm ICM} v_{\rm orb}^2 > g_{\rm max}(R)\Sigma_{\rm gas}(R) \quad . \tag{5.4}$$

If the gas density and total mass profiles of the galaxy can be represented by simple power-laws (as they are in the simulations we present) it is straightforward to evaluate the right-hand side of equation (5.4). In the case of a singular isothermal sphere, for example, where $\rho_{\rm gas}(r) \propto r^{-2}$ and $M_{\rm gal}(r) \propto r$ (where $M_{\rm gal}(r)$ is the *total* mass within radius r), we find $g_{\rm max}(R) = GM_{\rm gal}(R)/(2R^2)$ and $\Sigma_{\rm gas}(R) = \pi\rho_{\rm gas}(R)R$. This leads to the following stripping condition

$$P_{\rm ram}(t) > \frac{\pi}{2}\frac{GM_{\rm gal}(R)\rho_{\rm gas}(R)}{R} \quad . \tag{5.5}$$

For more general gas density and total mass profiles, the condition for ram pressure stripping may be expressed as

$$P_{\rm ram}(t) > \alpha\frac{GM_{\rm gal}(R)\rho_{\rm gas}(R)}{R} \quad , \tag{5.6}$$

where α is a geometric constant of order unity dependent upon the precise shape of the gas density and total mass profiles of the galaxy. We note that equation (5.6) is similar to the analytic stripping conditions derived previously by Gisler (1976) and Sarazin (1988) for elliptical galaxies. According to equation (5.6) all of the gas beyond the 3D "stripping radius," $R_{\rm strip}$, where the ram pressure exceeds the gravitational restoring force per unit area, will be stripped. We assume that the properties of the dark matter and gas interior to the stripping radius are unmodified by the stripping. The left-hand side of equation (5.6) indicates that in general, ram pressure is a function of time (i.e., for non-circular orbits).

This model does imply a series of simplifying assumptions. Most notably the lack of any shock heating of the halo which could quickly heat the halo gas until it becomes unbound, increasing the rate at which material is lost. However cosmological simulations have shown, small dense low-mass virialised systems (as in our setup) can penetrate all the way down to the cluster core without being significantly shocked (see, e.g., Motl et al., 2004; Murray and Lin, 2004; Poole et al., 2006; McCarthy et al., 2007; Dekel and Birnboim, 2008). In fact McCarthy et al. (2007) finds that most of the thermalisation of energy occurs in the ICM of the group or cluster whilst the colder galaxy sinks down into the core. McCarthy et al. (2007) does however observe a linear increase in the fraction

of energy that is thermalised in the gas of the less massive system (our galaxy in this case) with the ratio between the mass of the less massive system and the total mass of both systems. Thus shock heating is expected to become important for systems when the mass of the galaxy becomes comparable to the mass of the group. To overcome this we constrain our 2-system runs to use galaxies with masses lower than 10% of the group mass.

Another effect our model neglects is Tidal stripping which could alter the properties interior to the stripping radius. Again however this only proves to be significant when the galaxy mass is more than $\sim$ 10% of the mass of the group or cluster into which it falls (McCarthy et al., 2008b). As we restrict our 2-system simulations to mass ratios $\geq$ 10:1, then ram pressure stripping ends up being considerably more efficient than tidal stripping.

Finally, our simulations lack a prescription for the formation of Kelvin-Helmholtz and Rayleigh-Taylor instabilities. Once established, these hydrodynamic instabilities can lead to the complete disruption of the halo. As these are the focus of my involvement in the project I will discuss in more detail efforts made to ensure that they do not influence the results of our work and the checks we make to ensure that this is true in § 5.5.

5.5 Formation of Kelvin-Helmholtz Instabilities

The formation of Kelvin-Helmholtz instabilities at the interface between the hot halo gas of the galaxy and the ICM, present a means by which the entire halo can be completely disrupted. Their onset would therefore void the assumption that the halo properties interior to the stripping radius are unaffected. The production suite of simulations presented in McCarthy et al. (2008b) uses the SPH code *GADGET-2* due to its speed and low memory consumption. Unfortunately Agertz et al. (2007) has demonstrated that SPH codes spuriously suppress the formation of Kelvin-Helmholtz and Rayleigh-Taylor instabilities. They find this is due to erroneous pressure forces that arise when the density is miscalculated at fluid interfaces.

To ensure that our model remains valid and that we do not overestimate the survivability of the hot galaxy haloes, we adopt two approaches. First we restrict our SPH investigation to galaxies where their mass and thus gravitational potential is deep enough to strongly bind the halo gas and suppress the formation of instabilities (see § 5.5.1). Secondly I perform identical simulations in *FLASH* to contrast with the SPH ram pressure

stripping observed in *GADGET-2*. I present the results of these tests in § 5.5.2. As the density in an AMR code is calculated in a completely different fashion, such errors do not arise and they are able to resolve such turbulent instabilities. Figure 5.2 clearly shows that for a particular "blob" of gas subjected to a wind, whilst it remains intact within an SPH code, instabilities in the AMR code completely disrupt it over the course of a few Kelvin-Helmholtz times, $\tau_{\rm KH}$. Direct comparison to an AMR code will therefore highlight if instabilities should be present but are suppressed in our SPH suite.

5.5.1 Minimising the Effect of Hydrodynamic Instabilities

As Agertz et al. (2007) find that SPH and AMR based codes agree very well with each other on timescales less than the Kelvin-Helmholtz timescale, it would make most sense to restrict the simulations we perform to galaxy types where the Kelvin-Helmholtz timescale is large relative to the simulation time span. Following the approach of Mori and Burkert (2000) (see also Nulsen, 1982; Murray et al., 1993; Mayer et al., 2006), the Kelvin-Helmholtz timescale (including the stabilising effects of gravity) can be estimated as:

$$\begin{aligned} \tau_{\rm KH} &= \frac{F M_0}{\dot{M}_{\rm KH}} \\ &= 2.19 \times 10^9 \left(\frac{F}{0.1}\right)\left(\frac{M_0}{10^9\ M_\odot}\right)^{1/7} \\ &\quad \times \left(\frac{n_{\rm ICM}}{10^{-4}\ {\rm cm}^{-3}}\right)^{-1}\left(\frac{v_{\rm gal}}{10^3\ {\rm km\ s}^{-1}}\right)^{-1} {\rm yr}\ , \end{aligned} \tag{5.7}$$

where F is the baryon fraction of the galaxy, M_0 is the total mass of the galaxy within the radius down to which the galaxy has been stripped by ram pressure, $n_{\rm ICM}$ is the number density of hydrogen atoms in the ICM, and $v_{\rm orb}$ is the velocity of the galaxy with respect to the ICM.

For the default 2-system run we estimate from equation 5.7 that the Kelvin-Helmholtz timescale at pericentre is approximately 4.5 Gyr (i.e., comparable to the duration of our 2-system simulations). As most of the simulation will be spent at greater orbital radii, far from pericentre, the actual value of $\tau_{\rm KH}$ will be much larger and thus Kelvin-Helmholtz instabilities will have insufficient time to develop before the simulation concludes.

5.5.2 Comparison of Ram Pressure Stripping in *GADGET-2* and FLASH

By comparing the results from the Lagrangian SPH *GADGET-2* code to those obtained using the Eulerian *FLASH* AMR code I can confirm whether the formation of Kelvin-

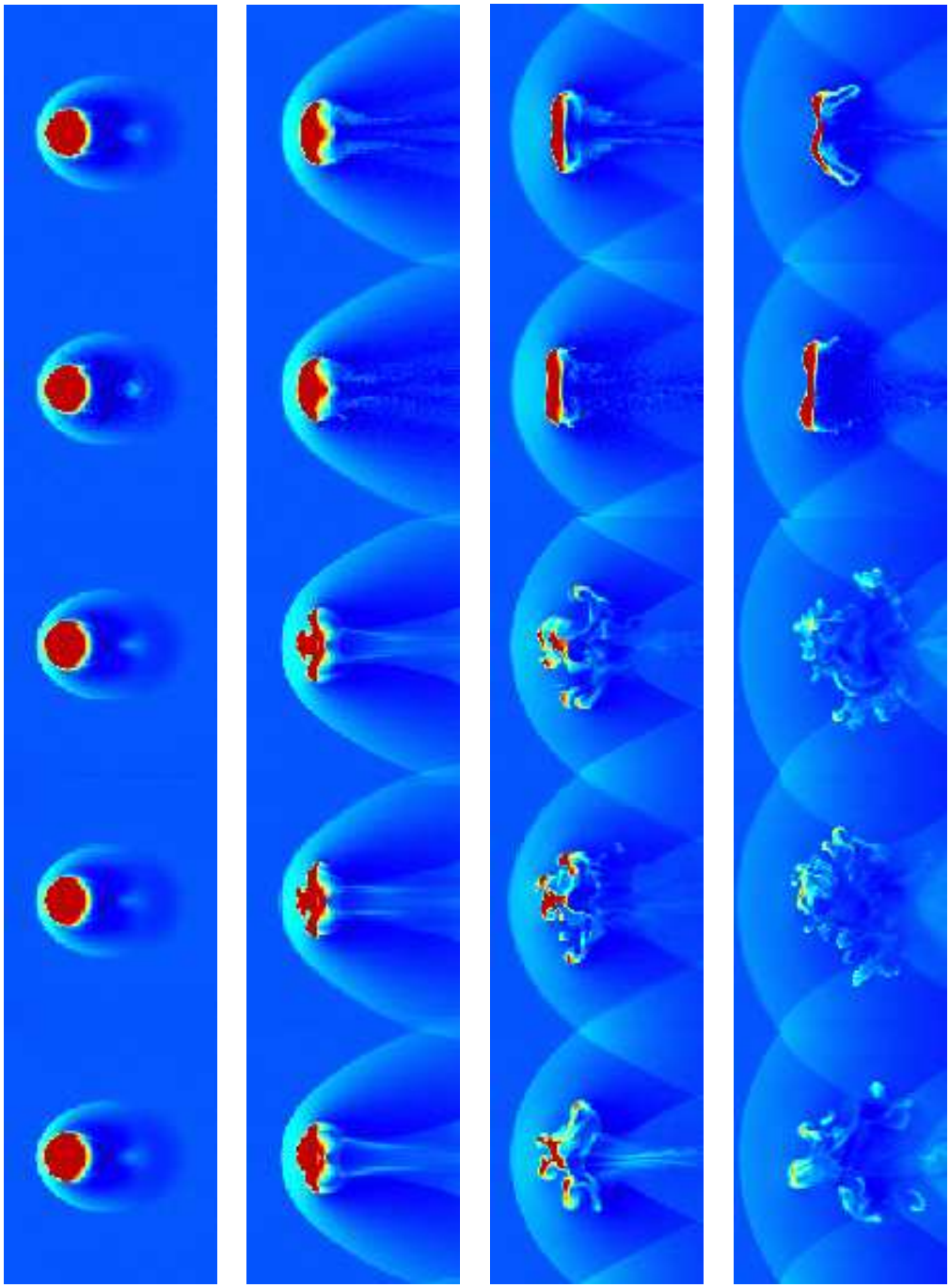

Figure 5.2: Gas density slices through the centre of the cloud at $t = 0.25, 1.0, 1.75$ and $2.5\,\tau_{\rm KH}$. From top to bottom we show Gasoline (SPH), GADGET-2 (SPH), Enzo (AMR), FLASH (AMR) and ART-Hydro (AMR). The grid simulations clearly show dynamical instabilities and complete fragmentation after $2.5\,\tau_{\rm KH}$, unlike the SPH simulations in which most of the gas remains in a single cold dense blob. Reproduced from figure 4, Agertz et al. (2007).

Helmholtz and Rayleigh-Taylor instabilities have been suppressed within the main suite of runs.

The characteristics of the *GADGET-2* simulation are given in § 5.3. In *FLASH*, I have tried three different versions of the same uniform medium run. In the first version, the galaxy is moved across a periodic box filled with a static background medium (as in the *GADGET-2* simulation) and the computational volume is resolved with a fixed 256^3 base grid. This yields a spatial resolution comparable to that of the *GADGET-2* run. In the second version, I take advantage of the AMR capability of *FLASH*, using a base grid of 64^3 cells and allowing up to two levels of refinement. This significantly speeds up the calculation. Finally, the third version is the same as the first version except that the galaxy is placed in the centre of the box and is assigned zero bulk velocity while the uniform background medium is assigned a velocity of -1000 km s^{-1}. Encouragingly, I find that all three versions of the FLASH run yield virtually identical results. For clarity I show only the results from the first version along with those from *GADGET-2* in figure 5.3.

For all of the *GADGET-2* SPH simulations presented in McCarthy et al. (2008b), the bound mass of gas is determined by calculating the centre of mass of the galaxy, computing energies in this frame, throwing out unbound particles, recomputing the centre of mass, and so on until no further particles are identified as being unbound. Under this iterative scenario, once a particle is stripped it can never be re-accreted. The $M(t)$ curves are necessarily monotonically decreasing in this case. Unfortunately, it is not trivial to implement this type of algorithm for the *FLASH* simulation since it is not a Lagrangian code. Instead, I assume that all of the dark matter remains gravitationally bound which according to the SPH simulations in McCarthy et al. (2008b) is proven to be generally correct. I then use the dark matter halo to calculate the centre of mass of the galaxy and determine which of the gas cells in the box are gravitationally bound to this dark matter halo. Under this scenario, gas that was once stripped can potentially be re-accreted. Therefore, a direct comparison between the default *GADGET-2* result and *FLASH* result should be treated with caution. Fortunately, however, it is straightforward to apply the same simplified bound mass algorithm used for the *FLASH* run to the *GADGET-2* run and we have done this.

In figure 5.3, we compare the bound mass of gas as a function of time for the *GADGET-2* and *FLASH* runs. The plot demonstrates that when both runs are treated on an equal footing, using the same algorithm for computing the bound mass of gas, the agreement between them is superb. This offers clear evidence that both Kelvin-Helmholtz and

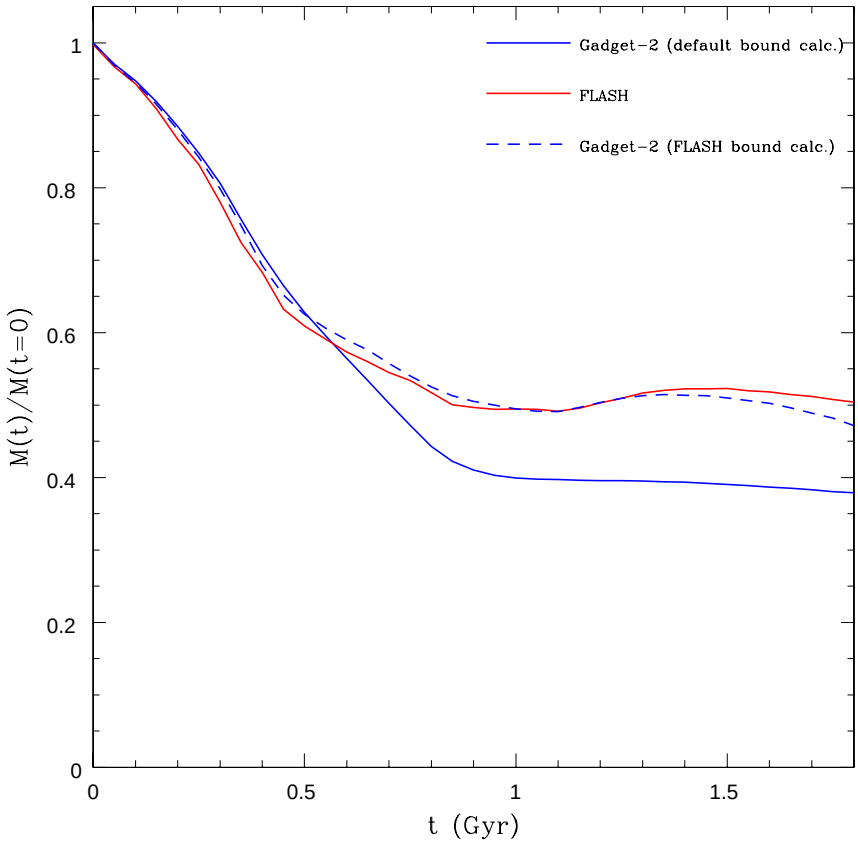

Figure 5.3: A comparison of the *GADGET-2* and *FLASH* results for the bound mass of gas for the uniform medium run. The solid blue curve corresponds to applying the default iterative bound mass algorithm described in § 5.3.1 to the *GADGET-2* run. The solid red curve are results of the *FLASH* code. The dashed blue curve corresponds to the case when we apply the same bound mass algorithm used for the *FLASH* run (see text) to the *GADGET-2* run. This demonstrates that when the *GADGET-2* and *FLASH* runs are treated on an equal footing the agreement between the two is excellent. (figure reproduced from figure 12, McCarthy et al., 2008b)

Rayleigh-Taylor instabilities which the SPH code would suppress, have not developed and thus the assumptions in our analytic model are found to hold. Furthermore such excellent agreement between the SPH and AMR simulations shows firm evidence that our suite of SPH simulations are reliable.

5.6 Comparison of the Analytic Model to Simulation

In McCarthy et al. (2008b), once the input parameters for the analytic model are fixed using one of the uniform density runs, our analytical model only requires information relating to the density profile (gas and dark matter) of the galaxy, the density profile of the ICM and the cluster orbit in order to produce exceptional agreement with results from the 2-system runs. With the exception of runs where the mass of the galaxy is more than $\sim 10\%$ of the mass of the group and its orbit is highly non-radial our analytic model agrees with the observed mass loss to within $\approx 10\%$. Such a breakdown for more massive galaxies is expected as the assumptions imposed in the creation of our analytic model break down. For higher mass galaxies it will most likely be necessary to incorporate the effects of tidal stripping and shock heating. But for galaxies within the limits of our models specified operating range, we find the results hold for a variety of orbital parameters, mass concentrations and galaxy to group mass ratios.

5.7 Conclusions

I present the derivation of a simple analytical model which for galaxies with masses less than $\sim 10\%$ of the mass of the group or cluster into which they fall gives an accurate description of the mass loss due to ram pressure stripping as a function of time. We find this to be true for a wide variety of orbits, mass concentrations and galaxy to group mass ratios. The model only requires information on the density profile (gas and dark matter) of the galaxy, the density profile of the ICM and the cluster orbit, making it ideal for implementation into semi-analytic models. Its adoption in place of the present instantaneous stripping approximation which starves galaxies of gas as soon as they cross the cluster's virial radius, has resulted in vastly improved galaxy colours, star formation rates and morphologies (see Font et al., 2008).

I confirm that the use of SPH simulations to determine the correct level of ram pressure stripping and key assumptions made in the formulation of the analytic model are

correct. I find that when identical analysis algorithms which allow for the re-accretion of previously stripped material are used in both *FLASH* and *GADGET-2*, the results show exceptional agreement. This clearly confirms the validity of both the analytical model and the robustness of the results obtained from the SPH simulations.

Chapter 6

On the Origin of Cores in Simulated Galaxy Clusters

6.1 Foreword

This chapter was published as a paper in Mitchell et al. (2009). The *FLASH* simulations presented in this chapter were exclusively performed by myself whilst the *GADGET-2* simulations present work undertaken by my colleague Dr. Ian McCarthy. The analysis presents a joint work with both myself and Dr. Ian McCarthy contributing in roughly equal measure. The introduction was predominantly written by my colleague along with an outline of the initial conditions whilst the description of the codes and the majority of material presented in sections 6.4 to 6.5.4 represents material I have written but was reviewed by my colleague before submission. The summary presents joint work written by both authors. I have added additional material to the afterword in section 6.7 both for completeness and in response to more recent work published by Springel (2010). For brevity this material was previously left out of the original publication.

6.2 Introduction

There has been a great deal of attention devoted in recent years to the properties of the hot X-ray-emitting plasma (the intracluster medium, hereafter ICM) in the central regions of massive galaxy groups and clusters. To a large extent, the focus has been on the competition between radiative cooling losses and various mechanisms that could be heating the gas and therefore (at least partially) offsetting the effects of cooling. Prior to the launch of the *Chandra* and *XMM-Newton* X-ray observatories, it was generally accepted that large quantities of the ICM should be cooling down to very low energies,

where it would cease to emit X-rays and eventually condense out into molecular clouds and form stars (Fabian, 1994). However, the amount of cold gas and star formation actually observed in most systems is well below what was expected based on analyses of *ROSAT*, *ASCA*, and *Einstein* X-ray data (e.g. Voit and Donahue, 1995; Edge, 2001). New high spatial and spectral resolution data from *Chandra* and *XMM-Newton* has shown that, while cooling is clearly important in many groups and clusters (so-called 'cool core' systems), relatively little gas is actually cooling out of the X-ray band to temperatures below $\sim 10^7$ K (Peterson et al., 2003).

It is now widely believed that some energetic form of non-gravitational heating is compensating for the losses due to cooling. Indeed, it seems likely that such heating goes beyond a simple compensation for the radiated energy. As well as having a profound affect on the properties of the ICM, it seems that the heat input also has important consequences for the formation and evolution of the central brightest cluster galaxy (BCG) and therefore for the bright end of the galaxy luminosity function (e.g. Benson et al., 2003; Bower et al., 2006, 2008). The thermal state of the ICM therefore provides an important probe of these processes and a concerted theoretical effort has been undertaken to explore the effects of various heating sources (e.g., supermassive black holes, supernovae, thermal conduction, dynamical friction heating of orbiting satellites) using analytic and semi-analytic models, in addition to idealised and full cosmological hydrodynamic simulations (e.g. Binney and Tabor, 1995; Narayan and Medvedev, 2001; Ruszkowski et al., 2004; McCarthy et al., 2004; Kim et al., 2005; Voit and Donahue, 2005; Sijacki et al., 2008). In most of these approaches, it is implicitly assumed that *gravitationally-induced* heating (e.g., via hydrodynamic shocks, turbulent mixing, energy-exchange between the gas and dark matter) that occurs during mergers/accretion is understood and treated with a sufficient level of accuracy. For example, most current analytic and semi-analytic models of groups and clusters attempt to take the effects of gravitational heating into account by assuming distributions for the gas that are taken from (or inspired by) non-radiative cosmological hydrodynamic simulations — the assumption being that these simulations accurately and self-consistently track gravitational processes.

However, it has been known for some time that, even in the case of identical initial setups, the results of non-radiative cosmological simulations depend on the numerical scheme adopted for tracking the gas hydrodynamics. In particular, mesh-based Eulerian (such as adaptive mesh refinement, hereafter AMR) codes appear to systematically produce higher entropy (lower density) gas cores within groups and clusters than do particle-

based Lagrangian (such as smoothed particle hydrodynamics, hereafter SPH) codes (see, e.g. Frenk et al., 1999; Ascasibar et al., 2003; Voit et al., 2005; Dolag et al., 2005; O'Shea et al., 2005). This may be regarded as somewhat surprising, given the ability of these codes to accurately reproduce a variety of different analytically solvable test problems (e.g., Sod shocks, Sedov blasts, the gravitational collapse of a uniform sphere; see, e.g., chapter 4), although clearly hierarchical structure formation is a more complex and challenging test of the codes. At present, the origin of the cores and the discrepancy in their amplitudes between Eulerian and Lagrangian codes in cosmologically-simulated groups and clusters is unclear. There have been suggestions that it could be the result of insufficient resolution in the mesh simulations, artificial entropy generation in the SPH simulations, Galilean non-invariance of the mesh simulations, and differences in the amount of mixing in the SPH and mesh simulations (e.g. Dolag et al., 2005; Wadsley et al., 2008). We explore all of these potential causes in § 6.5.

Clearly, though, this matter is worth further investigation, as it potentially has important implications for the competition between heating and cooling in groups and clusters (the cooling time of the ICM has a steep dependence on its core entropy) and the bright end of the galaxy luminosity function. And it is important to consider that the total heating is not merely the sum of the gravitational and non-gravitational heating terms. The Rankine-Hugoniot jump conditions tell us that the efficiency of shock heating (either gravitational or non-gravitational in origin) *depends on the density of the gas at the time of heating* (see, e.g., the discussion in McCarthy et al. (2008a)). This implies that if gas has been heated before being shocked, the entropy generated in the shock can actually be amplified (Voit et al., 2003; Voit and Ponman, 2003; Borgani et al., 2005; Younger and Bryan, 2007). The point is that gravitational and non-gravitational heating will couple together in complex ways, so it is important that we are confident that gravitational heating is being handled with sufficient accuracy in the simulations.

A major difficulty in studying the origin of the cores in cosmological simulations is that the group and cluster environment can be extremely complex, with many hundreds of substructures (depending on the numerical resolution) orbiting about at any given time. Furthermore, such simulations can be quite computationally-expensive if one wishes to resolve in detail the innermost regions of groups and clusters (note that, typically, the simulated cores have sizes $\lesssim 0.1 r_{200}$). An alternative approach, which we adopt in the present study, is to use idealised simulations of binary mergers to study the relevant gravitational heating processes. The advantages of such an approach are obviously that

the environment is much cleaner, therefore offering a better chance of isolating the key processes at play, and that the systems are fully resolved from the onset of the simulation. The relatively modest computational expense of such idealised simulations also puts us in a position to be able to vary the relevant physical and numerical parameters in a systematic way and to study their affects.

Idealised merger simulations have been used extensively to study a variety of phenomena, such as the disruption of cooling flows (Gómez et al., 2002; Ritchie and Thomas, 2002; Poole et al., 2008), the intrinsic scatter in cluster X-ray and Sunyaev-Zel'dovich effect scaling relations (Ricker and Sarazin, 2001; Poole et al., 2007), the generation of cold fronts and related phenomena (e.g. Ascasibar and Markevitch, 2006), and the ram pressure stripping of orbiting galaxies (e.g. Mori and Burkert, 2000; McCarthy et al., 2008b). However, to our knowledge, idealised merger simulations have not been used to elucidate the important issue raised above, nor have they even been used to demonstrate whether or not this issue even exists in non-cosmological simulations.

In the present study, we perform a detailed comparison of idealised binary mergers run with the widely-used public simulations codes *FLASH* (an AMR code) and *GADGET-2* (a SPH code). The chapter is organised as follows. In § 6.3 I give a brief description of the simulation codes and the relevant adopted numerical parameters. In addition, we describe the initial conditions (e.g., structure of the merging systems, mass ratio, orbit) of our idealised simulations. In § 6.4, we present a detailed comparison of results from the *FLASH* and *GADGET-2* runs and confirm that there is a significant difference in the amount of central entropy generated with the two codes. In § 6.5 we explore several possible causes for the differences we see. Finally, in § 6.6, we summarise and discuss our findings.

6.3 Simulations

6.3.1 The Codes

Below, I provide brief descriptions of the *GADGET-2* and *FLASH* hydrodynamic codes used in this study and the parameters we have adopted. The interested reader is referred to Springel et al. (2001) and Springel (2005) for in-depth descriptions of *GADGET-2* and to Fryxell et al. (2000) for the *FLASH* code. I also describe the codes and their main features in chapter 2. Both codes are representative examples of their respective AMR and SPH hydrodynamic formulations, as has been shown in the recent code comparison

of Tasker et al. (2008, see chapter 4).

6.3.1.1 *FLASH*

In this study I use *FLASH* version 2.5 and refer the reader back to chapter 2 for a more detailed overview of the *FLASH* architecture.

As shown in chapter 3 we have modified the gravity solver in *FLASH* to use an FFT on top of the multigrid solver (written by T. Theuns). This results in a vast reduction in the time spent calculating the self-gravity of the simulation relative to the publicly available version. I have rigorously tested the new algorithm against the default multigrid solver, more tests are presented in chapter 4.

The maximum number of Newton-Raphson iterations permitted within the Riemann solver was increased in order to allow it to deal with particularly sharp shocks and discontinuities whilst the default tolerance was maintained. The hydrodynamic algorithm also adopted the default settings with periodic boundary conditions being applied to the gas as well as to the gravity solver.

To identify regions of rapid flow change I incorporate *FLASH*'s adapted Löhner (1987) error estimator refinement criteria (see § 2.6.1). With this applied to the density as is common place, I imposed the additional user-defined criteria whereby the density has to exceed a threshold of $200\rho_c$, below which the refinement is set to the minimum 64^3 mesh. This restricts the refinement to the interior of the clusters and, as I demonstrate below, was found to yield nearly identical results to uniform grid runs with resolution equal to the maximum resolution in the equivalent AMR run.

Each block contains the same number of cells, $N_x = 16$ cells in each dimension in our runs. The maximum allowed level of refinement, l, is one of the parameters of the run. At refinement level l, a fully refined AMR grid will have $N_x\, 2^{l-1}$ cells on a side.

All the *FLASH* merger runs are simulated in 20 Mpc on a side periodic boxes in a non-expanding (Newtonian) space and are run for a duration of $\simeq 10$ Gyr. By default, our *FLASH* simulations are run with a maximum of $l = 6$ levels of refinement (512^3 cells when fully refined), corresponding to a minimum cell size of ≈ 39 kpc, which is small in comparison to the entropy cores produced in non-radiative cosmological simulations of clusters (but note we explicitly test the effects of resolution in § 6.4). The simulations include non-radiative hydrodynamics and gravity.

6.3.1.2 GADGET-2

GADGET-2 is a publicly available TreeSPH code designed for the simulation of cosmological structure formation. By default, the code implements the entropy conserving SPH formalism proposed by Springel and Hernquist (2002). The code is massively parallel, has been highly optimised and is very memory efficient. This has led to it being used for some of the largest cosmological simulations to date, including the first N-body simulation with more than 10^{10} dark matter particles (the *Millennium Simulation*; Springel et al., 2005b).

The SPH formalism is inherently Lagrangian in nature and fully adaptive, with 'refinement' based on the density. The particles represent discrete mass elements with the fluid variables being obtained through a kernel interpolation technique (Lucy, 1977; Gingold and Monaghan, 1977; Monaghan, 1992). The entropy injected through shocks is captured through the use of an artificial viscosity term (see, e.g., Monaghan, 1997). We will explore the sensitivity of our merger simulation results to the artificial viscosity in § 6.5.3.

By default, gravity is solved through the use of a combined tree particle-mesh (TreePM) approach. The TreePM method allows for substantial speed ups over the traditional tree algorithm by calculating long range forces with the particle-mesh approach using Fast Fourier techniques. Higher gravitational spatial resolution is then achieved by applying the tree over small scales only, maintaining the dynamic range of the tree technique. This allows *GADGET-2* to vastly exceed the gravitational resolving power of mesh codes which rely on the particle-mesh technique alone and are thus limited to the minimum cell spacing.

We adopt the following numerical parameters by default for our *GADGET-2* runs (but note that most of these are systematically varied in § 6.4). The artificial bulk viscosity, $\alpha_{\rm visc}$, is set to 0.8. The number of SPH smoothing neighbours, $N_{\rm sph}$, is set to 32. Each of our $10^{15} M_{\odot}$ model clusters (see § 6.3.2) are comprised of 5×10^5 gas and dark matter particles within r_{200}, and the gas to total mass ratio is 0.141. Thus, the particle masses are $m_{\rm gas} = 2.83 \times 10^8 M_{\odot}$ and $m_{\rm dm} = 1.72 \times 10^9 M_{\odot}$. The gravitational softening length is set to 10 kpc, which corresponds to $\approx 5 \times 10^{-3} r_{200}$ initially. All the *GADGET-2* merger runs are simulated in 20 Mpc on a side periodic boxes in a non-expanding (Newtonian) space and are run for a duration of $\simeq$ 10 Gyr. The simulations include basic hydrodynamics only (i.e., are non-radiative).

6.3.2 Initial Conditions

In our simulations, the galaxy clusters are initially represented by spherically-symmetric systems composed of a realistic mixture of dark matter and gas.

The dark matter is assumed to follow a NFW distribution (Navarro et al., 1996, 1997):

$$\rho(r) = \frac{\rho_s}{(r/r_s)(1+r/r_s)^2} \tag{6.1}$$

where $\rho_s = M_s/(4\pi r_s^3)$ and

$$M_s = \frac{M_{200}}{\ln(1+r_{200}/r_s) - (r_{200}/r_s)/(1+r_{200}/r_s)} \quad . \tag{6.2}$$

Here, r_{200} is the radius within which the mean density is 200 times the critical density, $\rho_{\rm crit}$, and $M_{200} \equiv M(r_{200}) = (4/3)\pi r_{200}^3 \times 200\rho_{\rm crit}$.

The dark matter distribution is fully specified once an appropriate scale radius (r_s) is selected. The scale radius can be expressed in terms of the halo concentration $c_{200} = r_{200}/r_s$. We adopt a concentration of $c_{200} = 4$ for all our systems. This value is typical of massive clusters formed in ΛCDM cosmological simulations (e.g., Neto et al., 2007).

In order to maintain the desired NFW configuration, appropriate velocities must be assigned to each dark matter particle. For this, we follow the method outlined in McCarthy et al. (2007). Briefly, the three velocity components are selected randomly from a Gaussian distribution whose width is given by the local velocity dispersion [i.e, $\sigma(r)$]. The velocity dispersion profile itself is determined by solving the Jeans equation for the mass density distribution given in eqn. (6.1). As in McCarthy et al. (2007), the dark matter haloes are run separately in isolation in *GADGET-2* for many dynamical times to ensure that they have fully relaxed.

For the gaseous component, we assume a power-law configuration for the entropy[1], $K \equiv P\rho_{\rm gas}^{-5/3}$, by default. In particular,

$$\frac{K(r)}{K_{200}} = 1.47\left(\frac{r}{r_{200}}\right)^{1.22}, \tag{6.3}$$

where the 'virial entropy', K_{200}, is given by

$$K_{200} \equiv \frac{GM_{200}}{2\ r_{200}} \frac{1}{(200\ \rho_{\rm crit})^{2/3}}\,. \tag{6.4}$$

[1]Note, the quantity K is the not the actual thermodynamic specific entropy (s) of the gas, but is related to it via the simple relation $s \propto \ln K^{3/2}$. However, for historical reasons we will refer to K as the entropy.

This distribution matches the entropy profiles of groups and clusters formed in the non-radiative cosmological simulations of Voit et al. (2005) (VKB05) for $r \gtrsim 0.1 r_{200}$. It is noteworthy that VKB05 find that this distribution approximately matches the entropy profiles of both SPH (the *GADGET-2* code) and AMR (the *ENZO* code) simulations. Within $0.1 r_{200}$, however, the AMR and SPH simulations show evidence for entropy cores, but of systematically different amplitudes. We initialise our systems without an entropy core [i.e., eqn. (6.3) is assumed to hold over all radii initially] to see, first, if such cores are established during the merging process and, if so, whether the amplitudes differ between the SPH and AMR runs, as they do in cosmological simulations. We leave it for future work to explore the differences that result (if any) between SPH and AMR codes when large cores are already present in the initial systems (e.g., the merger of two 'non-cool core' systems).

With the mass distribution of dark matter established (i.e., after having run the dark matter haloes in isolation) and an entropy distribution for the gas given by eqn. (6.3), we numerically compute the radial gas pressure profile (and therefore also the gas density and temperature profiles), taking into account the self-gravity of the gas, by simultaneously solving the equations of hydrostatic equilibrium and mass continuity:

$$\frac{\mathrm{dlog}P}{\mathrm{dlog}M_{\rm gas}} = -\frac{GM_{\rm gas}M_{\rm tot}}{4\pi r^4 P} \tag{6.5}$$

$$\frac{\mathrm{dlog}r}{\mathrm{dlog}M_{\rm gas}} = \frac{M_{\rm gas}}{4\pi r^3}\left(\frac{K}{P}\right)^{3/5} \tag{6.6}$$

Two boundary conditions are required to solve these equations. The first condition is that $r(M_{\rm gas} = 0) = 0$. The second condition is that the total mass of hot gas within r_{200} yields a realistic baryon fraction of $M_{\rm gas}/M_{tot} = 0.141$. In order to meet the second condition, we choose a value for $P(M_{\rm gas} = 0)$ and propagate the solution outwards to r_{200}. We then iteratively vary the inner pressure until the desired baryon fraction is achieved.

For the *GADGET-2* simulations, the gas particle positions are assigned by radially morphing a glass distribution until the desired gas mass profile is obtained (see McCarthy et al., 2007). The entropy (or equivalently internal energy per unit mass) of each particle is specified by interpolating with eqn. (6.3). For the *FLASH* simulations, the gas density and entropy of each grid cell is computed by using eqn. (6.3) and interpolating within the radial gas density profile resulting from the solution of eqns. (6.5) and (6.6).

Thus, for both the *GADGET-2* and *FLASH* simulations we start with identical dark matter haloes (using particle positions and velocities from the *GADGET-2* isolated runs

with a gravitational softening length of 10kpc) and gas haloes, which have been established by interpolating within radial profiles that have been computed numerically under the assumption that the gas is initially in hydrostatic equilibrium within the dark matter halo. Note that when varying the resolution of the *FLASH* simulations we simply change the maximum number of refinements l – we do not vary the number of dark matter particles – in this way, the initial dark matter distribution is always the same as in the low resolution *GADGET-2* run in all the simulations we have run.

In both the *GADGET-2* and *FLASH* simulations, the gaseous haloes are surrounded by a low density pressure-confining gaseous medium that prevents the systems from expanding prior to the collision (i.e., so that in the case of an isolated halo the object would be static) but otherwise it has a negligible dynamical effect on the system.

Isolated gas+DM haloes were run in both *GADGET-2* and *FLASH* for 10 Gyr in order to test the stability of the initial gas and dark matter haloes. Although deviations in the central entropy develop over the course of the isolated simulations, indicating the systems are not in perfect equilibrium initially, they are small in amplitude (the central entropy increases by $< 10\%$ over 10 Gyr), especially in comparison to the factor of $\sim 2-3$ jump in the central entropy that occurs as a result of shock heating during the merger. Furthermore, we note that the amplitude of the deviations in the isolated runs are significantly decreased as the resolution of these runs is increased. Our merger simulations, however, are numerically converged (see § 6.4), indicating that the deviations have a negligible effect on merger simulation results and the conclusions we have drawn from them.

6.4 Idealised Cluster Mergers

The existence of a discrepancy between the inner properties of the gas in groups and clusters formed in AMR and SPH cosmological simulations was first noticed in the Santa Barbara code comparison of Frenk et al. (1999). It was subsequently verified in several works, including Dolag et al. (2005); O'Shea et al. (2005); Kravtsov et al. (2005), and VKB05. The latter study in particular clearly demonstrated, using a relatively large sample of ~ 60 simulated groups and clusters, that those systems formed in the AMR simulations had systematically larger entropy cores than their SPH counterparts. Since this effect was observed in cosmological simulations, it was generally thought that the discrepancy was due to insufficient resolution in the mesh codes at high redshift (we note,

however, that VKB05 argued against resolution being the cause). This would result in under-resolved small scale structure formation in the early universe. This explanation is consistent with the fact that in the Santa Barbara comparison the entropy core amplitude tended to be larger for the lower resolution mesh code runs. Our first aim is therefore to determine whether the effect is indeed due to resolution limitations, or if it is due to a more fundamental difference between the two types of code. We test this using identical idealised binary mergers of spherically-symmetric clusters in *GADGET-2* and *FLASH* where it is possible to explore the effects of finite resolution with relatively modest computational expense (compared to full cosmological simulations).

6.4.1 A Significant Discrepancy

As a starting point, we investigate the generation of entropy cores in a head on merger between two identical $10^{15} M_{\odot}$ clusters, each colliding with an initial speed of $0.5 V_{\rm circ}(r_{200}) \simeq$ 722 km/s [i.e., the initial *relative* velocity is $V_{\rm circ}(r_{200})$, which is typical of merging systems in cosmological simulations; see, e.g., Benson (2005)]. The system is initialised such that the two clusters are just barely touching (i.e., their centres are separated by $2r_{200}$). The simulations are run for a duration of 10 Gyr, by the end of which the merged system has relaxed and there is very little entropy generation ongoing.

Our idealised test gives very similar results to non-radiative cosmological simulations – there is a distinct difference in the amplitude of the entropy cores in the AMR and SPH simulations, with the entropy in the mesh code a factor ~ 2 higher than the SPH code. It is evident that the difference between the codes is captured in a single merger event. An immediate question is whether this is the result of the different effective resolutions of the codes. Resolution tests can be seen in the left hand panel of Figure 6.1, where we plot the resulting radial entropy distributions.

For *GADGET-2*, we compare runs with 10^5, 10^6 (the default), and 10^7 particles. The minimum SPH smoothing lengths of these simulations throughout the runs are approximately 25, 11, and 5 kpc, respectively. For *FLASH* I compare AMR runs with minimum cell sizes of $\approx$ 78, 39 (the default), 19.5, and 9.8 kpc and a uniform grid run with the default 39 kpc cell size. The simulation characteristics for these head on mergers are presented in Table 6.1. To make a direct comparison with the cosmological results of VKB05 (see their Fig. 5), we normalise the entropy by the initial 'virial' entropy (K_{200}; see eqn. 6.4) and the radius by the initial virial radius, r_{200} (for reference, $r_{200,i} \simeq 2062$ kpc).

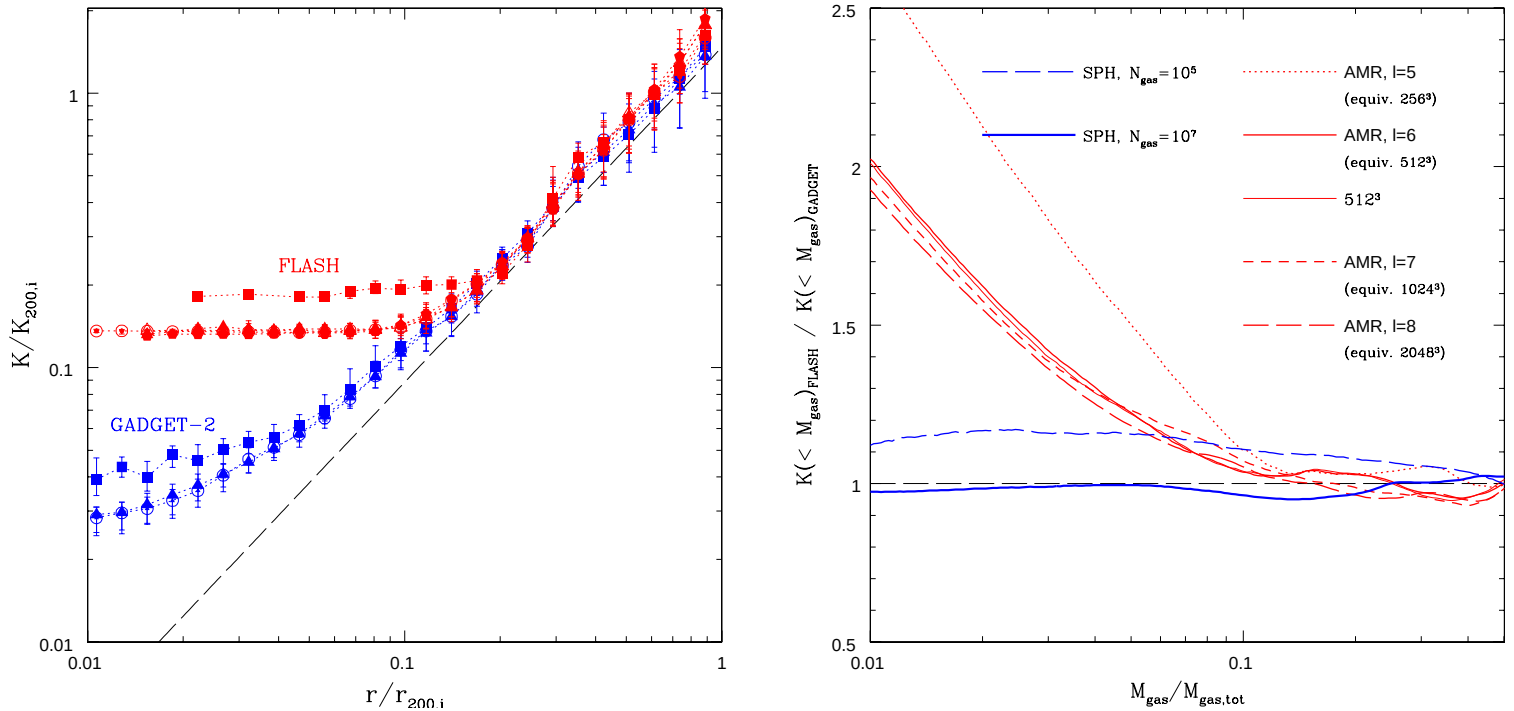

Figure 6.1: Plots showing the entropy cores formed in idealised head on mergers of equal mass ($10^{15} M_{\odot}$) clusters in *FLASH* and *GADGET-2* simulations. The left hand panel shows the final radial entropy distribution, where data points are the median entropy value in radial bins and error bars correspond to the 25^{th} and 75^{th} percentiles. The dashed black line represents the initial power-law configuration. Blue solid squares, triangles, and open circles represent the low (10^5 gas particles), default (10^6), and high resolution (10^7) *GADGET-2* runs. Red solid squares, triangles, pentagons, and open circles represent *FLASH* AMR runs with $l = 5$, 6 (default), 7, and 8, respectively. Open red triangles represent a uniform 512^3 *FLASH* run. The right hand panel presents the results in a slightly different way: it shows the entropy as a function of enclosed gas mass $K(< M_{\rm gas})$. The results (see legend) have been normalised to the final distribution of the default *GADGET-2* run (dashed black line). Clearly the results for both converge but on very different solutions.

Table 6.1: Characteristics of the head on simulations presented in § 6.4.1

FLASH sim.	No. cells	Max. spatial res. (kpc)
$l = 5$	equiv. 256^3	78
$l = 6$ (default)	equiv. 512^3	39
$l = 7$	equiv. 1024^3	19.5
$l = 8$	equiv. 2048^3	9.8
512^3	512^3	39

GADGET-2 sim.	No. gas particles	Max. spatial res. (kpc)
low res.	10^5	≈ 25
default	10^6	≈ 11
hi res.	10^7	≈ 5

The plot clearly shows that the simulations converge on two distinctly different solutions within the inner ten percent of r_{200}, whereas the entropy at large radii shows relatively good agreement between the two codes. The simulations performed for the resolution test span a factor of 8 in spatial resolution in *FLASH* and approximately a factor of 5 in *GADGET-2*. The *FLASH* AMR runs effectively converge after reaching a peak resolution equivalent to a 512^3 run (i.e., a peak spatial resolution of ≈ 39 kpc or $\approx 0.019 r_{200}$). I have also tried a *FLASH* run with a uniform (as opposed to adaptive) 512^3 grid and the results essentially trace the AMR run with an equivalent peak resolution. This reassures us that my AMR refinement criteria is correctly capturing all regions of significance. The lowest resolution SPH run, which only has 5×10^4 gas particles within r_{200} initially, has a slightly higher final central entropy than the default and high resolution SPH runs. This may not be surprising given the tests and modelling presented in Steinmetz and White (1997). These authors demonstrated that with such small particle numbers, two-body heating will be important if the mass of a dark matter particle is significantly above the mass of a gas particle. The *GADGET-2* runs converge, however, when the number of gas and dark matter particles are increased by an order magnitude (i.e., as in our default run), yielding a maximum spatial resolution of ≈ 11 kpc (here we use the minimum SPH smoothing length as a measure of the maximum spatial resolution).

A comparison of the left hand panel of Fig. 6.1 to Fig. 5 of VKB05 reveals a remarkable

correspondence between the results of our idealised merger simulations and those of their cosmological simulations (which spanned system masses of $\sim 10^{13-15} M_{\odot}$). They find that the ratio of the AMR and SPH core amplitudes is ~ 2 in both the idealised and cosmological simulations. This difference is also seen in the Santa Barbara comparison of Frenk et al. (1999) when comparisons are made between the SPH simulations and the highest resolution AMR simulations carried out in that study (ie., the 'Bryan' AMR results)[2]. This consistency presumably indicates that whatever mechanism is responsible for the differing core amplitudes in the cosmological simulations is also responsible for the differing core amplitudes in our idealised simulations. This is encouraging, as it implies the generation of the entropy cores can be studied with idealised simulations. As outlined in § 6.2, the advantage of idealised simulations over cosmological simulations is their relative simplicity. This gives us hope that we can use idealised simulations to track down the underlying cause of the discrepancy between particle-based and mesh-based hydrodynamic codes.

The right hand panel of Figure 6.1 shows the resulting entropy distributions plotted in a slightly different fashion. Here we plot the entropy as a function of 'enclosed' gas mass $K(< M_{\rm gas})$. This is constructed by simply ranking the particles/cells by entropy in ascending order and then summing the masses of the particles/cells [the inverse, $M_{\rm gas}(K)$, would therefore be the total mass of gas with entropy lower than K]. Convective stability ensures that, eventually when the system is fully relaxed, the lowest-entropy gas will be located at the very centre of the potential well, while the highest entropy gas will be located at the system periphery. $K(< M_{\rm gas})$ is therefore arguably a more fundamental quantity than $K(r)$ and we adopt this test throughout the rest of the chapter. It is also noteworthy that in order to compute $K(< M_{\rm gas})$ one does not first need to select a reference point (e.g., the centre of mass or the position of the particle with the lowest potential energy) or to bin the particles/cells in any way, both of which could introduce ambiguities in the comparison between the SPH and AMR simulations (albeit likely minor ones).

In the right hand panel of Figure 6.1, we plot the resulting $K(< M_{\rm gas})$ distributions normalised to the final entropy distribution of the default *GADGET-2* run. Here we see that the lowest-entropy gas in the *FLASH* runs have a higher entropy, by a factor of

[2]We note, however, that the lower resolution AMR simulations in that study produced larger entropy cores, which suggests that they may not have been numerically converged (as in the case of $l = 5$ AMR run in Fig. 6.1).

$\approx 1.9 - 2.0$, than the lowest-entropy gas in the default *GADGET-2* run. Naively, looking at the right hand panel of Figure 6.1 one might conclude that the discrepancy is fairly minor, given that $\sim 95\%$ of the gas has been heated to a similar degree in the SPH and AMR simulations. But it is important to keep in mind that it is the properties of the lowest-entropy gas in particular that are most relevant to the issue of heating vs. cooling in groups and clusters (and indeed in haloes of all masses), since this is the gas that has the shortest cooling time.

The agreement between our results and those from cosmological simulations (e.g., Frenk et al., 1999, VKB05) is striking. The convergence of the entropy distributions in our idealised simulations negates the explanation that inadequate resolution of the high redshift universe in cosmological AMR simulations is the root cause of the discrepancy between the entropy cores in SPH and AMR simulations (although we note that some of the lower resolution AMR simulations in the study of Frenk et al. may not have been fully converged and therefore the discrepancy may have been somewhat exaggerated in that study for those simulations). We therefore conclude that the higher entropy generation in AMR codes relative to SPH codes within the cores of groups and clusters arises out of a more fundamental difference in the adopted algorithms. Below we examine in more detail how the entropy is generated during the merging process in the simulations and we then systematically explore several possible causes for the differences in the simulations.

6.4.2 An Overview of Heating in the Simulations

We have demonstrated that the entropy generation that takes place in our idealised mergers is robust to our choice of resolution, yet a difference persists in the amount of central entropy that is generated in the SPH and mesh simulations. We now examine the entropy generation as a function of time in the simulations, which may provide clues to the origin of the difference between the codes.

Figure 6.2 shows $\log(K)$ in a slice through the centre of the default *FLASH* simulation at times 0, 1, 2, 3, 4, 5, 7 and 10 Gyr. This may be compared to Figure 6.3, which shows the entropy distribution of the simulations as a function of time (this figure is described in detail below). Briefly, as the cores approach each other, a relatively gentle shock front forms between the touching edges of the clusters, with gas being forced out perpendicular to the collision axis. Strong heating does not actually occur until approximately the time when the cores collide, roughly 1.8 Gyrs into the run. The shock generated through the core collision propagates outwards, heating material in the outer regions of the system.

This heating causes the gas to expand and actually overshoot hydrostatic equilibrium. Eventually, the gas, which remains gravitationally bound to the system, begins to fall back onto the system, producing a series of weaker secondary shocks. Gas at the outskirts of the system, which is the least bound, takes the longest to re-accrete. This dependence of the time for gas to be re-accreted upon the distance from the centre results in a more gradual increase in entropy than seen in the initial core collision. In a *qualitative* sense, the heating process that takes place in the *FLASH* simulations is therefore very similar to that seen in the *GADGET-2* simulations (see §3 of McCarthy et al. 2007 for an overview of the entropy evolution in idealised *GADGET-2* mergers).

The top left panel in Figure 6.3 shows the ratio of $K(< M_{\rm gas})$ in the default *FLASH* run relative to $K(< M_{\rm gas})$ in the default *GADGET-2* run. The various curves represent the ratio at different times during the simulations (see figure key — note that these correspond to the same outputs displayed in Figure 6.2). It can clearly be seen that the bulk of the difference in the *final* entropy distributions of the simulations is established around the time of core collision. The ratio of the central entropy in the *FLASH* simulation to the central entropy in the *GADGET-2* simulation converges after $\approx$ 4 Gyr. The top right panel shows the time evolution of the lowest-entropy gas only in both the *GADGET-2* and *FLASH* runs. Here we see there are similar trends with time, in the sense that there are two main entropy generation episodes (core collision and re-accretion), but that the entropy generated in the first event is much larger in the *FLASH* run than in the *GADGET-2* run. Far outside the core, however, the results are very similar. For completeness, the bottom two panels show $K(< M_{\rm gas})$ at different times for the *GADGET-2* and *FLASH* runs separately.

The small initial drop in the central entropy at 1 Gyr in the *FLASH* run (see bottom left panel) is most likely due to interpolation errors at low resolution. This drop in entropy should not physically occur without cooling processes (which are not included in our simulations), but there is nothing to prevent a dip from occurring in the simulations due to numerical inaccuracies (the second law of thermodynamics is not explicitly hardwired into the mesh code). At low resolutions, small violations in entropy conservation can occur due to inaccurate interpolations made by the code. We have verified that the small drop in entropy does not occur in the higher resolution *FLASH* runs. We note that while this effect is present in default *FLASH* run, it is small and as demonstrated in Fig. 1 the default run is numerically converged.

It is interesting that the *FLASH* to *GADGET-2* central entropy ratio converges rela-

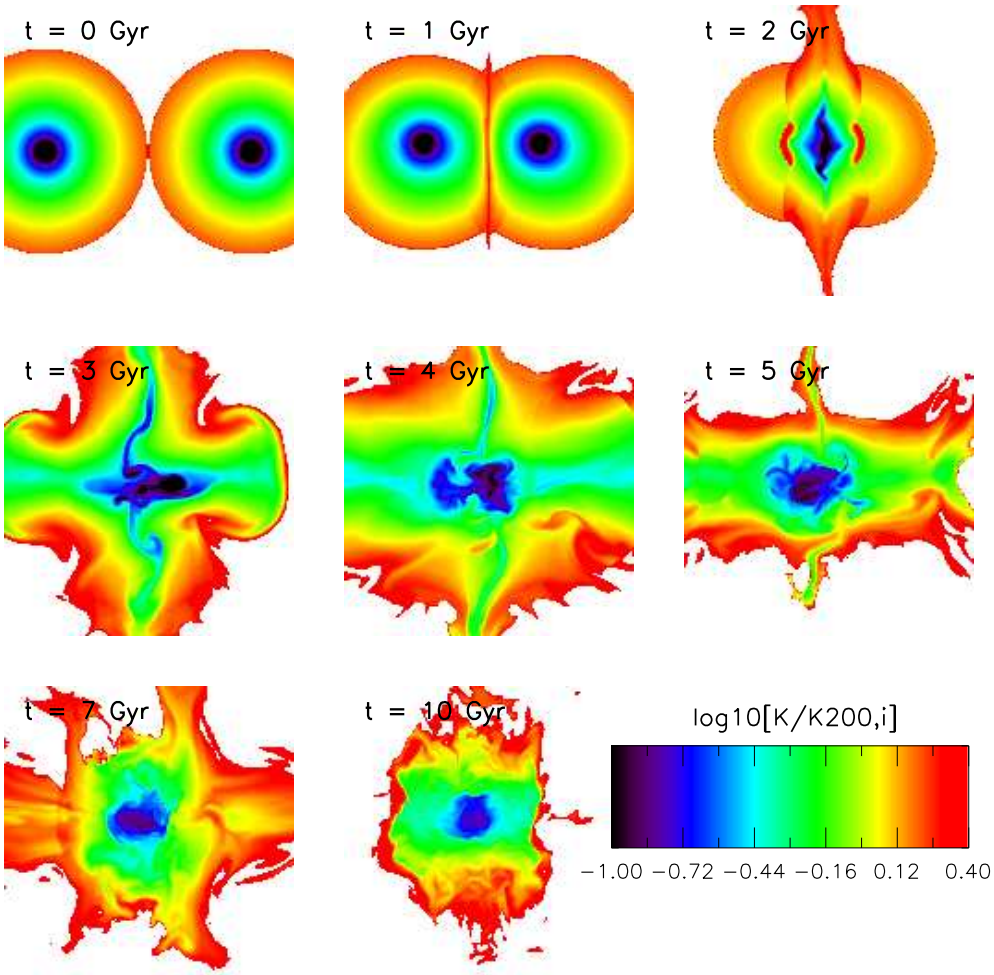

Figure 6.2: Logarithmic entropy slices (i.e., thickness of zero) through the centre of the default *FLASH* merger simulation (with $l = 6$) at times 0, 1, 2, 3, 4, 5, 7 and 10 Gyr. The lowest entropy material is shown in blue, increasing in entropy through green, yellow to red. Each panel is 6 Mpc on a side. Significant entropy is generated at $t \approx 2$ Gyr when the cores collide and gas is squirted out, and again later on when this gas re-accretes.

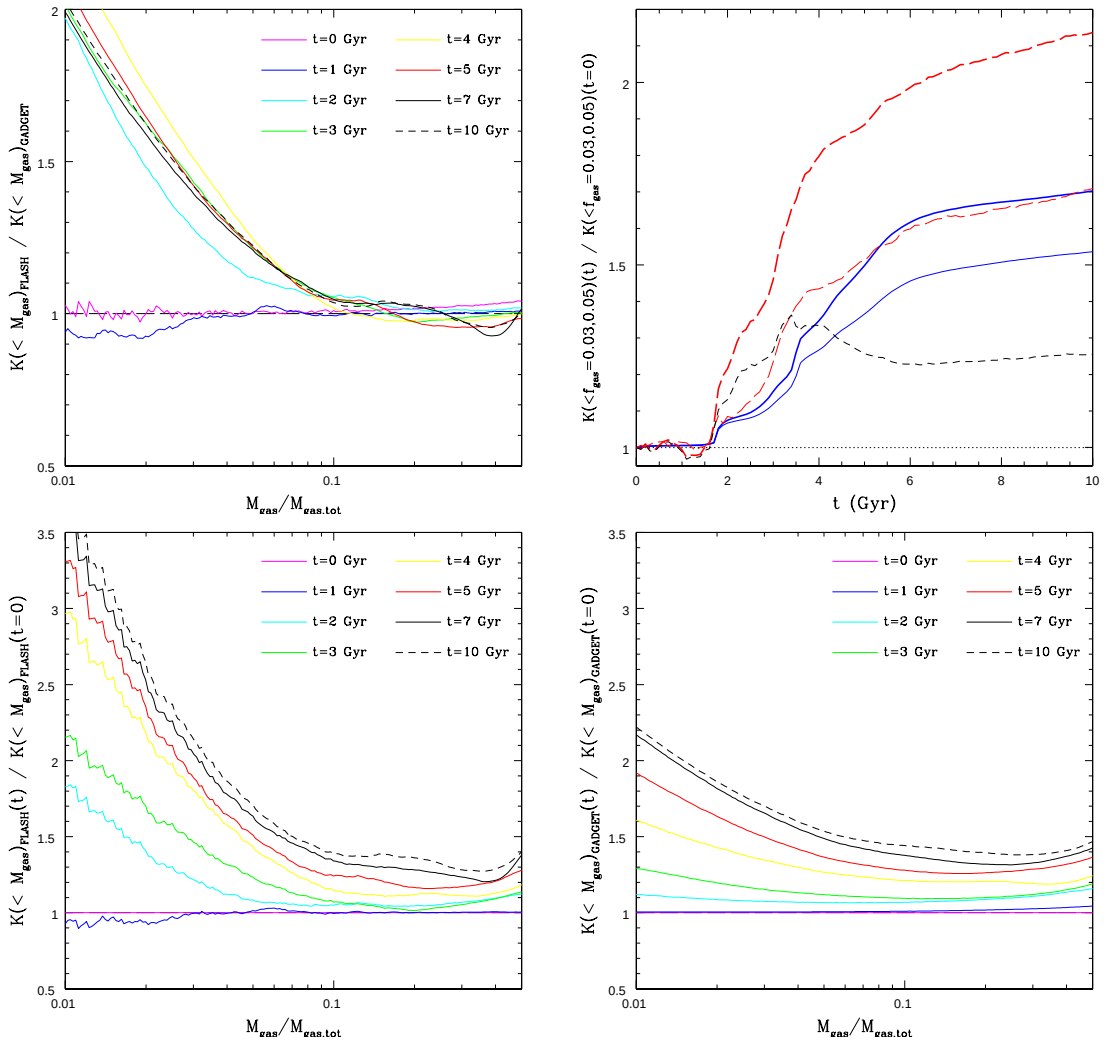

Figure 6.3: Time-dependence of entropy generation in the default *GADGET-2* and *FLASH* runs. The top left panel shows the ratio of $K(< M_{\rm gas})$ in the *FLASH* run to that in the *GADGET-2* run. The various curves represent the ratio at different times during the simulations (see legend). Top right panel shows the time evolution of the lowest-entropy gas only. Shown are $K(< M_{\rm gas}/M_{\rm gas,tot} = 0.03)$ (thick curves) and $K(< M_{\rm gas}/M_{\rm gas,tot} = 0.05)$ (thin curves) for the *FLASH* (long-dashed red curves) and *GADGET-2* (solid blue curves) runs (i.e., having sorted the gas particles/cells by entropy, we show the evolution of the entropy that encloses 3% and 5% of the total gas mass). The curves have been normalised to their initial values at the start of the simulations. The short-dashed black curve represents the ratio of *FLASH* to *GADGET-2* entropies enclosing 3% of the total gas mass. The bottom two panels show the $K(< M_{\rm gas})$ distributions for the default *FLASH* and *GADGET-2* runs separately, at different times during the simulation. Together, these plots illustrate that the difference in the final entropy distributions of the *FLASH* and *GADGET-2* runs is primarily established around the time of core collision ($\sim 2-3$ Gyr). It is worth noting, however, that significant entropy generation continues after this time, but it occurs in nearly the same fashion in the AMR and SPH runs.

tively early on in the simulations. This is in spite of the fact that a significant fraction of the entropy that is generated in both simulations is actually generated at later times, during the re-accretion phase. Evidently, this phase occurs in a very similar fashion in both simulations. In § 6.5, we will return to the point that the difference between the results of the AMR and SPH simulations arises around the time of core collision.

6.4.3 Alternative Setups

It is important to verify that the conclusions we have drawn from our default setup are not unique to that specific initial configuration. Using a suite of merger simulations of varying mass ratio and orbital parameters, McCarthy et al. (2007) demonstrated that the entropy generation that takes place does so in a qualitatively similar manner to that described above in all their simulations. However, these authors examined only SPH simulations. We have therefore run several additional merger simulations in both *GADGET-2* and *FLASH* to check the robustness of our conclusions. All of these mergers are carried out using the same resolution as adopted for the default *GADGET-2* and *FLASH* runs.

In Figure 6.4, we plot the final *FLASH* to *GADGET-2* $K(< M_{\rm gas})$ ratio for equal mass mergers with varying orbital parameters (see figure caption). In all cases, *FLASH* systematically produces larger entropy cores than *GADGET-2* and by a similar factor to that seen in the default merger setup. Interestingly, the off-axis case results in a somewhat larger central entropy discrepancy between *GADGET-2* and *FLASH* even though the bulk energetics of this merger are the same as for the default case. A fundamental difference between the off-axis case and the default run is that the former takes a longer time for the cores to collide and subsequently relax (but note by the end of the off-axis simulation there is very little ongoing entropy generation, as in the default case). This may suggest that the timescale over which entropy is generated plays some role in setting the magnitude of the discrepancy between the AMR and SPH simulations. For example, one possibility is that 'pre-shocking' due to the artificial viscosity i.e., entropy generation during the early phases of the collision when the interaction is subsonic or mildly transonic) in the SPH simulations becomes more relevant over longer timescales. Another possibility is that mixing, which is expected to be more prevalent in Eulerian mesh simulations than in SPH simulations, plays a larger role if the two clusters spend more time in orbit about each other before relaxing into a single merged system (of course, one also expects enhanced mixing in the off-axis case simply because of the geometry). We explore these and other possible causes of the difference in § 6.5.

In addition to varying the orbital parameters, we have also experimented with colliding a cluster composed of dark matter only with another cluster composed of a realistic mixture of gas and dark matter (in this case, we simulated the head on merger of two equal mass $10^{15} M_{\odot}$ clusters with an initial relative velocity of $\simeq 1444$ km/s). Obviously, this is not an astrophysically reasonable setup. However, a number of studies have suggested that there is a link between the entropy core in clusters formed in non-radiative cosmological simulations and the amount of energy exchanged between the gas and the dark matter in these systems (e.g., Lin et al., 2006; McCarthy et al., 2007). It is therefore interesting to see whether this experiment exposes any significant differences with respect to the results of our default merger simulation.

The dotted magenta curve in Figure 6.4 represents the final *FLASH* to *GADGET-2* $K(< M_{\rm gas})$ ratio for the case where a dark matter only cluster merges with another cluster composed of both gas and dark matter. The results of this test are remarkably similar to that of our default merger case. This indicates that the mechanism responsible for the difference in heating in the mesh and SPH simulations in the default merger simulation is also operating in this setup. Although this does not pin down the difference between the mesh and SPH simulations, it does suggest that the difference has little to do with differences in the properties of the large hydrodynamic shock that occurs at core collision, as there is no corresponding large hydrodynamic shock in the case where one cluster is composed entirely of dark matter. However, it is clear from Figure 6.3 that the difference between the default mesh and SPH simulations is established around the time of core collision, implying that some source of heating other than the large hydrodynamic shock is operating at this time (at least in the *FLASH* simulation). We return to this point in § 6.5.

6.5 What Causes the Difference?

There are fundamental differences between Eulerian mesh-based and Lagrangian particle-based codes in terms of how they compute the hydrodynamic and gravitational forces. Ideally, in the limit of sufficiently high resolution, the two techniques would yield identical results for a given initial setup. Indeed, both techniques have been shown to match with high accuracy a variety of test problems with known analytic solutions. However, as has been demonstrated above (and in other recent studies; e.g., Agertz et al., 2007; Trac et al., 2007; Wadsley et al., 2008) differences that do not appear to depend on resolution present

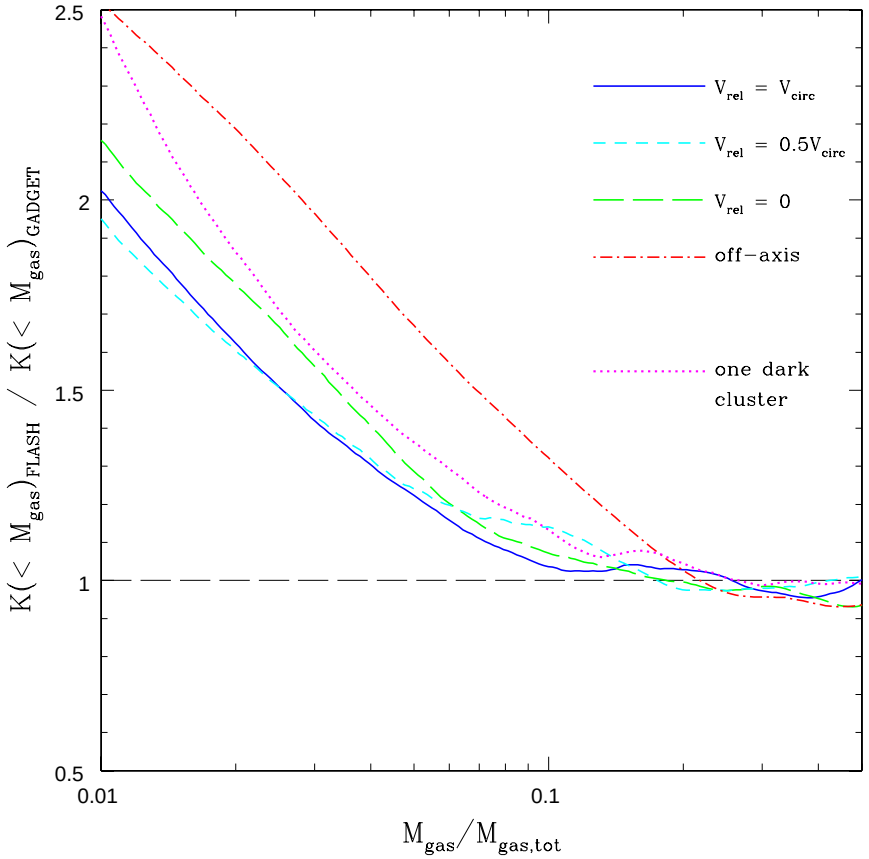

Figure 6.4: The ratio of *FLASH* to *GADGET-2* final entropy distributions for equal mass mergers of varying initial orbital parameters. The solid blue curve represents the default setup (head on collision with an initial relative velocity of $V_{\rm circ}(r_{200})$). The long-dashed green and short-dashed cyan curves represent head on collisions with initial relative velocities of $0.5V_{\rm circ}(r_{200})$ and 0 (i.e., at rest initially). The dot dashed red curve represents an off-axis collision with an initial relative radial velocity of $\simeq 0.95V_{\rm circ}(r_{200})$ and an initial relative tangential velocity of $\simeq 0.312V_{\rm circ}(r_{200})$ (i.e., the total energy is equivalent to that of the default setup). Also shown (dotted magenta curve), is the entropy ratio of a run where one of the clusters is composed of dark matter only and the other of a realistic mixture of gas and dark matter (see text). All these simulations result in a comparable difference in entropy profile between the mesh code and the SPH code.

themselves in certain complex, but astrophysically-relevant, circumstances.

In what follows, we explore several different possible causes for why the central heating that takes place in mesh simulations exceeds that in the SPH simulations. The possible causes we explore include:

- § 6.5.1 A difference in gravity solvers - Most currently popular mesh codes (including *FLASH* and *ENZO*) use a particle-mesh (PM) approach to calculate the gravitational force. To accurately capture short range forces it is therefore necessary to have a finely-sampled mesh. By contrast, particle-based codes (such as *GADGET-2* and *GASOLINE*) often make use of tree algorithms or combined tree particle-mesh (TreePM) algorithms, where the tree is used to compute the short range forces and a mesh is used to compute long range forces. Since the gravitational potential can vary rapidly during major mergers and large quantities of mass can temporarily be compressed into small volumes, it is conceivable differences in the gravity solvers and/or the adopted gravitational force resolution could give rise to different amounts of entropy generation in the simulations.

- § 6.5.2 Galilean non-invariance of mesh codes - Given explicit dependencies in the Riemann solver's input states, all Eulerian mesh codes are inherently not Galilean invariant to some degree. This can lead to spurious entropy generation in the cores of systems as they merely translate across the simulation volume (e.g., chapter 4).

- § 6.5.3 'Pre-shocking' in the SPH runs - Artificial viscosity is required in SPH codes to capture the effects of shock heating. However, the artificial viscosity can in principle lead to entropy production in regions where no shocks should be present (e.g., Dolag et al., 2005). If such pre-shocking is significant prior to core collision in our SPH simulations, it could result in a reduced efficiency of the primary shock.

- § 6.5.4 A difference in the amount of mixing in SPH and mesh codes - Mixing will be suppressed in standard SPH implementations where steep density gradients are present, since Rayleigh-Taylor and Kelvin-Helmholtz instabilities are artificially damped in such circumstances (e.g., Agertz et al., 2007). In addition, the standard implementation of artificial viscosity will damp out even *subsonic* motions in SPH simulations, thereby inhibiting mixing (Dolag et al., 2005). On the other hand, one expects there to be some degree of over-mixing in mesh codes, since fluids are implicitly assumed to be fully mixed on scales smaller than the minimum cell size.

We now investigate each of these possible causes in turn. We do not claim that these are the only possible causes for the differences we see in the simulations. They do, however, represent the most commonly invoked possible solutions (along with hydrodynamic resolution, which we explored in § 6.4) to the entropy core discrepancy between SPH and mesh codes.

6.5.1 Is it due to a Difference in the Gravity Solvers?

In the *FLASH* simulations, gravity is computed using a standard particle-mesh approach. With this approach, the gravitational force will be computed accurately only on scales larger than the finest cell size. By contrast, the *GADGET-2* simulations make use of a combined TreePM approach, where the tree algorithm computes the short range gravitational forces and the particle-mesh algorithm is used only to compute long range forces. To test whether or not differences in the gravity solvers (and/or gravitational force resolution) are important, we compare the final mass distributions of the dark matter in our simulations. The distribution of the dark matter should be insensitive to the properties of the diffuse baryonic component, since its contribution to the overall mass budget is small by comparison to the dark matter.[3] Thus, the final distribution of the dark matter tells us primarily about the gravitational interaction alone between the two clusters.

Figure 6.5 shows the ratio of the final *FLASH* dark matter mass profiles to the final *GADGET-2* dark matter mass profile. Recall that in all runs the number of dark matter particles is the same. The differences that are seen in this figure result from solving for the gravitational potential on a finer mesh. For the lowest resolution *FLASH* run, we see that the final dark matter mass profile deviates significantly from that of the default *GADGET-2* run for $r \lesssim 0.04 r_{200,i}$. However, this should not be surprising, as the minimum cell size in the default *FLASH* run is $\sim 0.02 r_{200,i}$. By increasing the maximum refinement level, l, we see that the discrepancy between the final *FLASH* and *GADGET-2* dark matter mass profiles is limited to smaller and smaller radii. With $l = 8$, the minimum cell size is equivalent to the gravitational softening length adopted in the default *GADGET-2* run. In this case, the final dark matter mass distribution agrees with that of the default *GADGET-2* run to within a few percent at all radii beyond a few softening lengths (or a few cell sizes), which is all that should be reasonably expected. A comparison of the various *FLASH* runs with one another (compare, e.g., the default *FLASH* run with the $l = 8$ run,

[3] We have explicitly verified this by running a merger between clusters with gas mass fractions that are a factor of 10 lower than assumed in our default run.

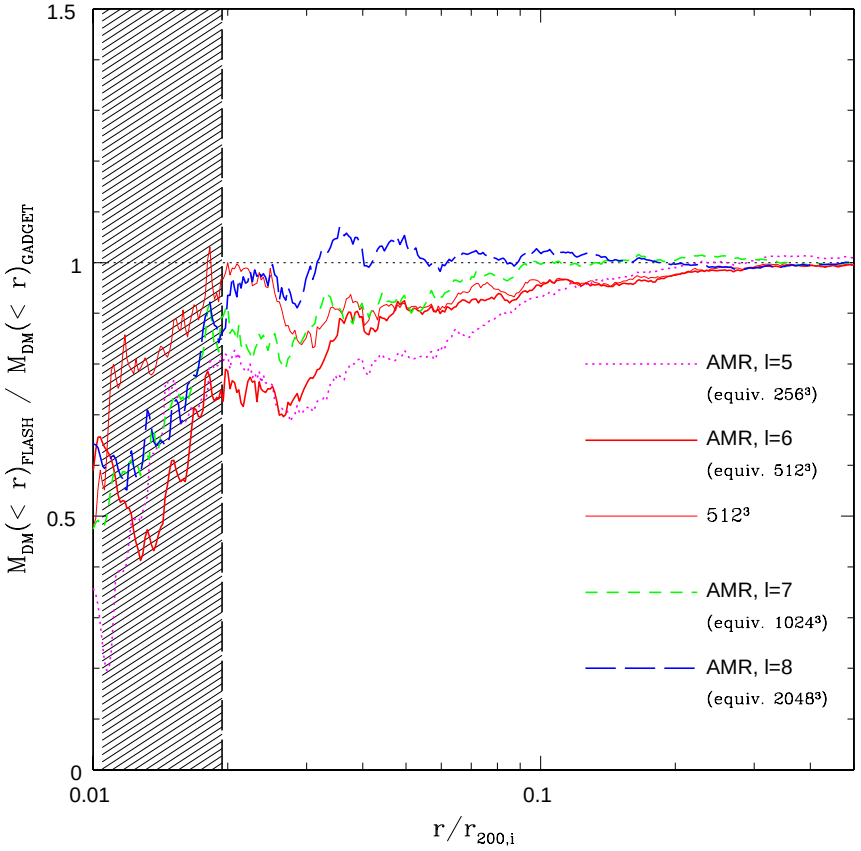

Figure 6.5: A plot comparing the resulting dark matter mass distributions for the default merger setup at 10 Gyr. The dark matter mass profiles have been normalised to the final dark matter mass profile of the default resolution *GADGET-2* run. The dotted magenta, solid red, short-dashed green, and long-dashed blue curves represent the *FLASH* AMR runs with $l = 5$, 6, 7, and 8, respectively, which correspond to peak grid cell sizes of $\approx$ 78, 39, 19.5, and 9.8 kpc (respectively). The thin solid red curve represents the uniform 512^3 *FLASH* run. The gravitational softening length adopted for the *GADGET-2* run is 10 kpc. For reference, $r_{200,i} \simeq 2062$ kpc. The vertical dashed line indicates four softening lengths. The *FLASH* dark matter distribution converges to the *GADGET-2* result when the numerical resolutions become similar: the observed differences in gas entropy are not due to differences in the underlying dark matter dynamics.

for which there is a $\sim 6\%$ discrepancy out as far as $0.1r_{200}$) may suggest a somewhat slower rate of convergence to the default *GADGET-2* result than one might naively have expected. Given that I have tested the new FFT gravity solver against both the default multigrid solver and a range of simple analytic problems and confirmed its accuracy to a much higher level than this, we speculate that the slow rate of convergence is due to the relatively small number of dark matter particles used in the mesh simulations. In the future, it would be useful to vary the number of dark matter particles in the mesh simulations to verify this hypothesis.

In summary, we find that the resulting dark matter distributions agree very well in the *GADGET-2* and *FLASH* simulations when the effective resolutions are comparable. The intrinsic differences between the solvers therefore appear to be minor. More importantly for our purposes, even though the gravitational force resolution for the default *FLASH* run is not as high as for the default *GADGET-2* run, this has no important consequences for the comparison of the final entropy distributions of the gas. It is important to note that, even though the final mass distribution in the *FLASH* simulations shows small differences between $l = 6$ and 8, Figure 6.1 shows that the entropy distribution is converged for $l \geq 6$ and is not at all affected by the improvement in the gravitational potential.

6.5.2 Is it due to Galilean Non-Invariance of Grid Hydrodynamics?

Due to the nature of Riemann solvers (which are a fundamental feature of AMR codes), it is possible for the evolution of a system to be Galilean non-invariant. This arises from the fixed position of the grid relative to the fluid. The Riemann shock tube initial conditions are constructed by determining the amount of material that can influence the cell boundary from either side, within a given time step based on the sound speed. The Riemann problem is then solved at the boundary based on the fluid properties either side of the boundary. By applying a bulk velocity to the medium in a given direction, the nature of the solution changes. Although an ideal solver would be able to decouple the bulk velocity from the velocity discontinuity at the shock, the discrete nature of the problem means that the code may not be Galilean invariant. Since one expects large bulk motions to be relevant for cosmological structure formation, and clearly is quite relevant for our merger simulations, it is important to quantify what affects (if any) Galilean non-invariance has on our AMR simulations.

I have tested the Galilean non-invariance of our *FLASH* simulations in two ways. In the first test, I simulate an isolated cluster moving across the mesh with an (initial) bulk

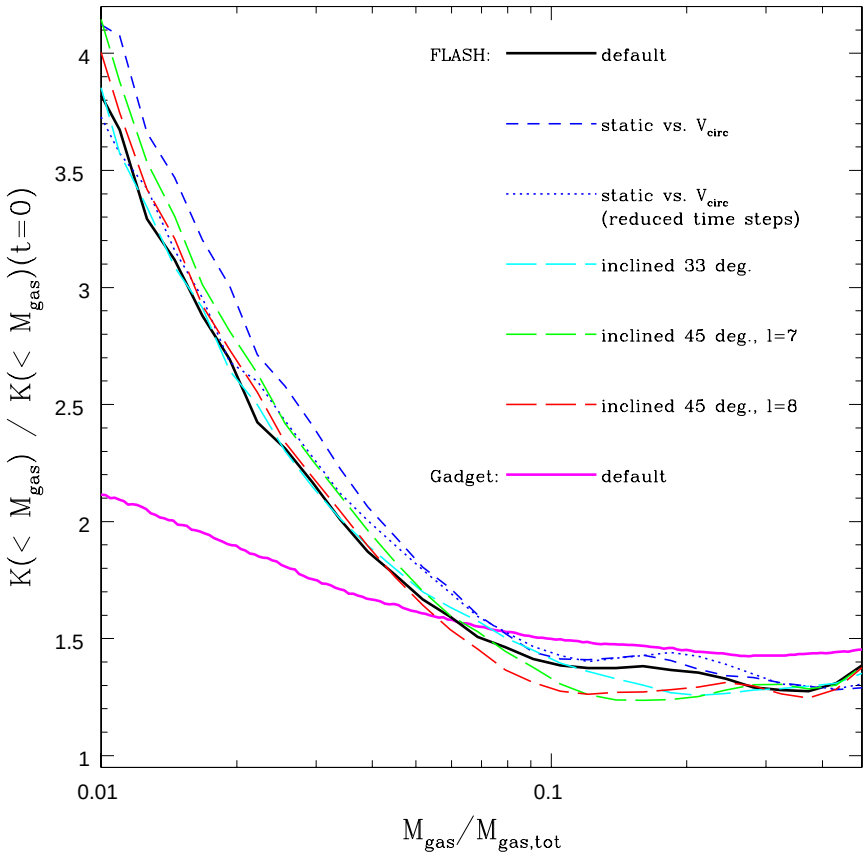

Figure 6.6: Testing the effects of Galilean non-invariance on the *FLASH* merger simulations. Plotted is the final entropy distribution, normalised to the initial one, for the default *FLASH* merger simulation and various different 'takes' on the default run. The solid black curve represents the default run, the short-dashed blue curves represents a merger where one cluster is held static and the other given a bulk velocity twice that in the default run (i.e., the relative velocity is unchanged from the default run), and the dotted blue curve represents this same merger but with the size of the time steps reduced by an order of magnitude. The dashed cyan, green, and red lines represent mergers that take place on an oblique angle to mesh at 33 degrees, 45 degrees with $l = 7$ and 45 degrees with $l = 8$, respectively. The default *GADGET-2* simulation is shown for comparison (solid magenta line). This comparison illustrates that the effects of Galilean non-invariance on the resulting entropy distribution are minor and do not account for the difference in the entropy core amplitudes of the mesh and SPH simulations.

velocity of $V_{\rm circ}(r_{200})$ ($\simeq 1444$ km/s) and compare it to an isolated cluster with zero bulk velocity. This is similar to the king profile tests carried out in chapter 4. In agreement with results in chapter 4 I find that there is some spurious generation of entropy in the very central regions ($M_{\rm gas}/M_{\rm gas,tot} \lesssim 0.03$) of the isolated cluster that was given an initial bulk motion. However, after ≈ 2 Gyr of evolution (i.e., the time when the clusters collide in our default merger simulation), the increase in the central entropy is only $\sim 10\%$. This is small in comparison to the $\sim 300\%$ jump that takes place at core collision in our merger simulations. This suggests that spurious entropy generation prior to the merger is minimal and does not account for the difference we see between the SPH and AMR simulations.

In the second test, I consider different implementations of the default merger simulation. In one case, instead of giving both systems equal but opposite bulk velocities (each with magnitude $0.5V_{\rm circ}$), I fix one and give the other an initial velocity that is twice the default value, so that the relative velocity is unchanged. (I also tried reducing the size of the time steps for this simulation by an order magnitude.) In addition, I have tried mergers that take place at oblique angles relative to the grid. If the merger is well-resolved and the dynamics are Galilean invariant, all these simulations should yield the same result.

Figure 6.6 shows the resulting entropy distributions for these different runs. The results of this test confirm what was found above; i.e., that there is some dependence on the reference frame adopted, but that this effect is minor in general (the central entropy is modified by $\lesssim 10\%$) and does not account for the discrepancy we see between entropy core amplitudes in the default *GADGET-2* and *FLASH* simulations.

6.5.3 Is it due to 'Pre-Shocking' in SPH?

Artificial viscosity is required in SPH codes in order to handle hydrodynamic shocks. The artificial viscosity acts as an excess pressure in the equation of motion, converting gas kinetic energy into internal energy, and therefore raising the entropy of the gas. In standard SPH implementations, the magnitude of the artificial viscosity is fixed in both space and time for particles that are approaching one another (it being set to zero otherwise). This implies that even in cases where the Mach number is less than unity, i.e., where formally a shock should not exist, (spurious) entropy generation can occur. This raises the possibility that significant 'pre-shocking' could occur in our SPH merger simulations. This may have the effect of reducing the efficiency of the large shock that occurs at core collision and could therefore potentially explain the discrepancy between

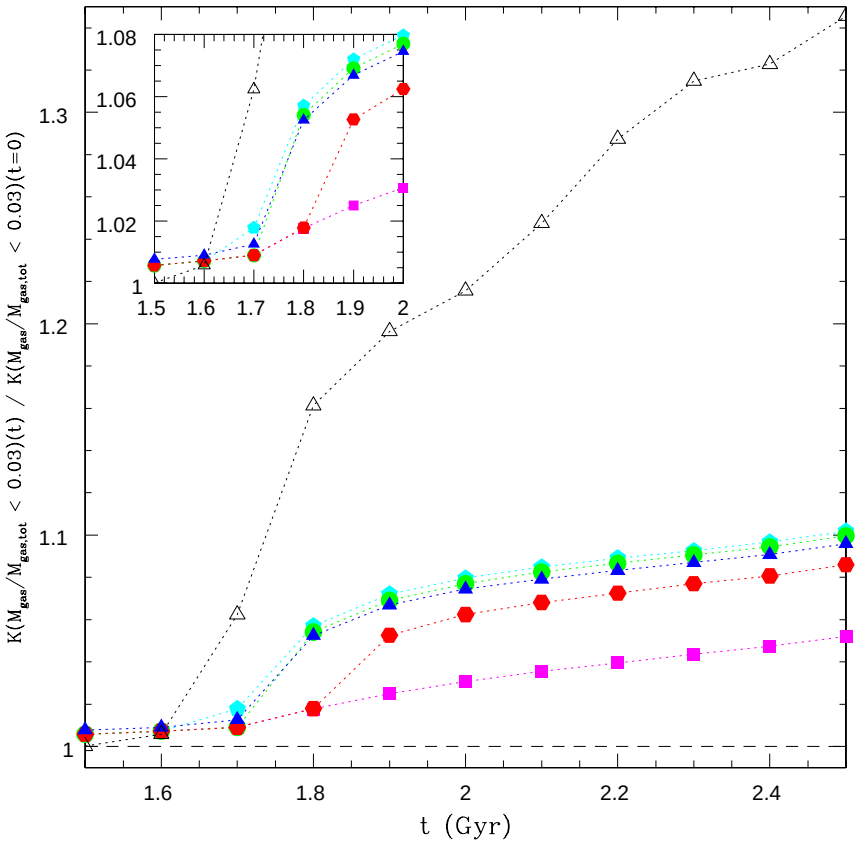

Figure 6.7: Testing the effects of pre-shocking due to artificial viscosity in the default *GADGET-2* merger simulation. This plot shows the evolution of the central entropy (enclosing 3% of the gas mass) around the time of first core collision. The solid blue triangles represent the default *GADGET-2* simulation. The solid cyan points, solid green points, and solid red points represent runs where the artificial viscosity is kept at a very low level ($\alpha_{\rm visc} = 0.05$) until $t \approx 1.6$, 1.7, 1.8 Gyr, respectively, at which point the artificial viscosity is set back to its default value. The solid magenta squares represent a run with low artificial viscosity throughout, and the open triangles represent the default *FLASH* simulation. Reducing the value of the artificial viscosity parameter before the cores collide delays the increase in entropy (cyan, green and solid red), however as soon as the original value is restored, the entropy K increases to a level nearly independent of when α was restored. Therefore pre-shocking has little affect on the post-shock value of K.

the mesh and SPH simulations.

Dolag et al. (2005) raised this possibility and tested it in SPH cosmological simulations of massive galaxy clusters. They implemented a new variable artificial viscosity scheme by embedding an on-the-fly shock detection algorithm in *GADGET-2* that indicates if particles are in a supersonic flow or not. If so, the artificial viscosity is set to a typical value, if not the artificial viscosity is greatly reduced. This new implementation should significantly reduce the amount of pre-shocking that takes place during formation of the clusters. The resulting clusters indeed had somewhat higher central entropies relative to clusters simulated with the standard artificial viscosity implementation (although the new scheme does not appear to fully alleviate the discrepancy between mesh and SPH codes). However, whether the central entropy was raised because of the reduction in pre-shocking or if it was due to an increase in the amount of mixing is unclear.

Our idealised mergers offer an interesting opportunity to re-examine this test. In particular, because of the symmetrical geometry of the merger, little or no mixing is expected until the cores collide, as prior to this time there is no interpenetration of the gas particles belonging to the two clusters (we have verified this). This means that we are in a position to isolate the effects of pre-shocking from mixing early on in the simulations. To do so, we have devised a crude method meant to mimic the variable artificial viscosity scheme of Dolag et al. (2005). In particular, we run the default merger with a low artificial viscosity (with $\alpha_{\rm visc} = 0.05$, i.e., approximately the minimum value adopted by Dolag et al.) until the cores collide, at which point we switch the viscosity back to its default value. We then examine the amount of entropy generated in the large shock.

Figure 6.7 shows the evolution of the central entropy around the time of core collision. Shown are a few different runs where we switch the artificial viscosity back to its default value at different times (since the exact time of 'core collision' is somewhat ill-defined). Here we see that prior to the large shock very little entropy has been generated, which is expected given the low artificial viscosity adopted up to this point. A comparison of these runs to the default *GADGET-2* simulation (see inset in Figure 6.7) shows that there is evidence for a small amount of pre-shocking in the default run. However, we find that for the cases where the artificial viscosity is set to a low value, the resulting entropy jump (after the viscosity is switched back to the default value) is nearly the same as in the default merger simulation. In other words, pre-shocking appears to have had a minimal effect on the strength of the heating that occurs at core collision in the default SPH simulation. This argues against pre-shocking as the cause of the difference we see

between the mesh and SPH codes.

Lastly, we have also tried varying $\alpha_{\rm visc}$ over the range 0.5 and 1.0 (i.e., values typically adopted in SPH studies; Springel, 2005) for the default *GADGET-2* run. We find that the SPH results are robust to variations in $\alpha_{\rm visc}$ and cannot reconcile the differences between SPH and AMR results.

6.5.4 Is it due to a Difference in the Amount of Mixing in SPH and Mesh Codes?

Our experiments with off-axis collisions and collisions with a cluster containing only dark matter suggest that mixing plays an important role in generating the differences between the codes. Several recent studies (e.g., Dolag et al., 2005; Wadsley et al., 2008) have argued that mixing is handled poorly in standard implementations of SPH, both because (standard) artificial viscosity acts to damp turbulent motions and because the growth of KH and RT instabilities is inhibited in regions where steep density gradients are present (Agertz et al., 2007). Using cosmological SPH simulations that have been modified in order to enhance mixing[4], Dolag et al. (2005) and Wadsley et al. (2008) have shown that it is possible to generate higher central entropies in their galaxy clusters (relative to clusters simulated using standard implementations of SPH), yielding closer agreement with the results of cosmological mesh simulations. This is certainly suggestive that mixing may be the primary cause of the discrepancy between mesh and SPH codes. However, these authors did not run mesh simulations of galaxy clusters and therefore did not perform a direct comparison of the amount of mixing in SPH vs. mesh simulations of clusters. Even if one were to directly compare cosmological SPH and mesh cluster simulations, the complexity of the cosmological environment and the hierarchical growth of clusters would make it difficult to clearly demonstrate that mixing is indeed the difference.

Our idealised mergers offer a potentially much cleaner way to test the mixing hypothesis. To do so, I re-run the default *FLASH* merger simulation but this time I include a large number of 'tracer particles', which are massless and follow the hydrodynamic flow of the gas during the simulation. The tracer particles are advanced using a second order accurate predictor-corrector time advancement scheme with the particle velocities being

[4]Note the that modifications implemented by Dolag et al. (2005) and Wadsley et al. (2008) differ. As described in §4.3, Dolag et al. (2005) implemented a variable artificial viscosity, whereas Wadsley et al. (2008) introduced a turbulent heat flux term to the Lagrangian energy equation in an attempt to explicitly model turbulent dissipation.

interpolated from the grid (further details are given in the *FLASH* manual (version 2.5) at: http://flash.uchicago.edu/). Each tracer particle has a unique ID that is preserved throughout the simulation, allowing us to track the gas in a Lagrangian fashion, precisely as is done in Lagrangian SPH simulations. To simplify the comparison further, I initially distribute the tracer particles within the two clusters in our *FLASH* simulation in exactly the same way as the particles in our initial *GADGET-2* setup.

In the left hand panel of Figure 6.8, we plot the final vs. the initial entropy of particles in the default *GADGET-2* and *FLASH* merger simulations. This plot clearly demonstrates that the lowest-entropy gas is preferentially heated in both simulations, however the degree of heating of that gas in the mesh simulation is much higher than in the SPH simulation. Consistent with our analysis in §3, we find that the bulk of this difference is established around the time of core collision. It is also interesting that the scatter in the final entropy (for a given initial entropy) is much larger in the mesh simulation. The larger scatter implies that convective mixing is more prevalent in the mesh simulation. At or immediately following core collision ($t \approx 2 - 3$ Gyr), there is an indication that, typically, gas initially at the very centre of the two clusters (which initially had the lowest entropy) has been heated more strongly than gas further out [compare, e.g., the median $K(t = 5\mathrm{Gyr})$ at $K(t = 0)/K_{200,i} \approx 0.02$ to the median at $K(t = 0)/K_{200,i} \approx 0.08$]. Such an entropy inversion does not occur in the SPH simulations and likely signals that the extra mixing in the mesh simulation has boosted entropy production.

In the right hand panel of Figure 6.8 we plot the final vs. the initial enclosed gas mass of particles in the default *GADGET-2* and *FLASH* merger simulations. The enclosed gas mass of each particle (or tracer particle) is calculated by summing the masses of all other particles (or cells) with entropies lower than the particle under consideration[5]. This plot confirms our mixing expectations based on the entropy plot in the left hand panel. In particular, only a small amount of mass mixing is seen in the SPH simulation, whereas in the mesh simulation the central $\sim 5\%$ of the gas mass has been fully mixed.

The higher degree of mixing in the *FLASH* simulation is shown pictorially in Figure 6.9. The left panel shows the final spatial distribution of the initially lowest-entropy particles

[5]In convective equilibrium, the enclosed gas mass calculated in this way also corresponds to the total mass of gas of all other particles (or cells) within the cluster-centric radius (or at lower, more negative gravitational potential energies) of the particle under consideration. We have verified this for the final output when the merged system has relaxed by, instead of summing the masses of all particles with entropy lower than K_i, by summing the masses of all particles with potentials lower than Φ_i for the ith particle.

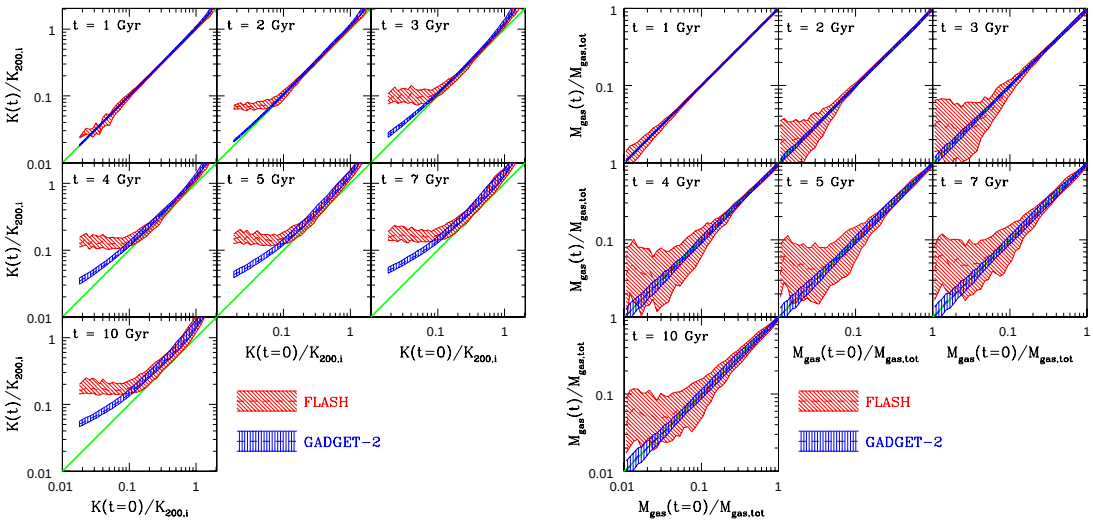

Figure 6.8: Quantifying the amount of heating and mixing in the default *GADGET-2* and *FLASH* merger simulations. *Left:* The entropy of particles (tracer particles in the case of *FLASH*) at time t vs. the initial entropy of those particles. The solid green line is the line of equality [$K(t) = K(t = 0)$; i.e., no heating]. The shaded blue and red regions represent the distributions from the *GADGET-2* and *FLASH* simulations, respectively. They enclose 50% of the particles; i.e., the lower/upper bounds represent the 25th/75th percentiles for $K(t)$ at fixed $K(t = 0)$. The dashed blue and red lines represent the median $K(t)$ at fixed $K(t = 0)$. The central entropy in the *FLASH* runs increases significantly more than in the *GADGET-2* run when the cores collide, at $t \sim 2$ Gyr, the increase in entropy later is similar between the two codes. The scatter in *FLASH* entropy is also much larger than in *GADGET-2*. *Right:* The enclosed gas mass of particles (tracers particles in the case of *FLASH*) at time t vs. the initial enclosed mass of those particles. The enclosed gas mass of each particle is calculated by summing the masses of all other particles (or cells) with entropies lower than the particle under consideration. The solid green line is the line of equality [$M_{\rm gas}(t) = M_{\rm gas}(t = 0)$; i.e., no mass mixing]. The shaded blue and red regions represent the distributions from the *GADGET-2* and *FLASH* simulations, respectively. They enclose 50% of the particles; i.e., the lower/upper bounds represent the 25th/75th percentiles for $M_{\rm gas}(t)$ at fixed $M_{\rm gas}(t = 0)$. The dashed blue and red lines represent the median $M_{\rm gas}(t)$ at fixed $M_{\rm gas}(t = 0)$. Particles in *FLASH* mix much more than in *GADGET-2*.

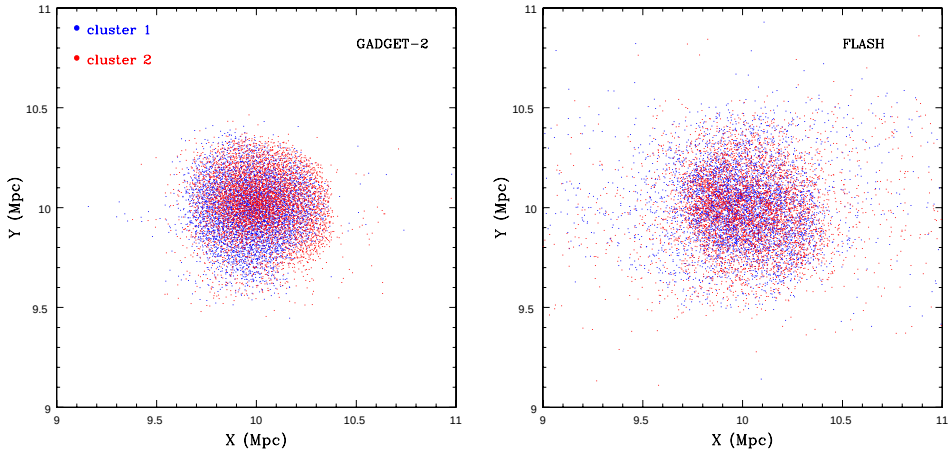

Figure 6.9: The final spatial distribution of particles (tracer particles in the case of *FLASH*) with the lowest initial entropies (we select the central 5% of particles/tracer particles in both clusters). The blue points represent particles belonging to one of the clusters and the red points represent particles belonging to the other. *Left:* The low resolution *GADGET-2* simulation. *Right:* The default *FLASH* simulation. There is a high degree of mixing in the mesh simulation, whereas there remain two distinct blobs corresponding to the infallen clusters in the SPH simulation. The difference arises immediately following core collision through the turbulent mixing that it drives.

in *GADGET-2* simulation, while the right panel is the analogous plot for tracer particles in the *FLASH* simulation (see figure caption). The larger degree of mixing in the mesh simulation relative to the SPH simulation is clearly evident. In the *FLASH* simulation, particles from the two clusters are intermingled in the final state, while distinct red and blue regions are readily apparent in the SPH calculation, a difference which arises immediately following core collision.

The increased mixing boosts entropy production in the *FLASH* simulations, but what is the origin of the increased mixing? We now return to the point raised in § 6.4, that the bulk of the difference between mesh and SPH simulation is established around the time of core collision. This is in spite of the fact that significant entropy generation proceeds in both simulations until $t \sim 6$ Gyr. Evidently, both codes treat the entropy generation in the re-accretion phase in a very similar manner. What is different about the initial core collision phase? As pointed out recently by Agertz et al. (2007), SPH suppresses

the growth of instabilities in regions where steep density gradients are present due to spurious pressure forces acting on the particles. Could this effect be responsible for the difference we see? To test this idea, we generate 2D projected entropy maps of the SPH and mesh simulations to search for signs of clear instability development. In the case of the SPH simulation, we first smooth the particle entropies (and densities) onto a 3D grid using the SPH smoothing kernel and the smoothing lengths of each particle computed by *GADGET-2*. We then compute a gas mass-weighted projected entropy by projecting along the z-axis. In the case of the *FLASH* simulation, the cell entropies and densities are interpolated onto a 3D grid and projected in the same manner as for the *GADGET-2* simulation.

Figure 6.10 shows a snap shot of the two simulations at $t = 2.3$ Gyr, just after core collision. Large vortices and eddies are easily visible in the projected entropy map of the *FLASH* simulation but none are evident in the *GADGET-2* simulation. In order to study the duration of these eddies, we have generated 100 such snap shots for each simulation, separated by fixed 0.1 Gyr intervals. Analysing the projected entropy maps as a movie[6], we find that these large vortices and eddies persist in the *FLASH* simulation from $t \approx 1.8 - 3.2$ Gyr. This corresponds very well with the timescale over which the difference between the SPH and mesh codes is established (see, e.g., the dashed black curve in the top right panel of Figure 6.3).

We therefore conclude that extra mixing in the mesh simulations, brought on by the growth of instabilities around the time of core collision, is largely responsible for the difference in the final entropy core amplitudes between the mesh and SPH simulations. Physically, one expects the development of such instabilities, since the KH timescale, $\tau_{\rm KH}$, is relatively short at around the time of core collision. We therefore conclude that there is a degree of under-mixing in the SPH simulations[7] Whether or not the *FLASH* simulations yield the correct result, however, is harder to ascertain. As fluids are *forced* to numerically mix on scales smaller than the minimum cell size, it is possible that there is a non-negligible degree of over-mixing in the mesh simulations. Our resolution tests (see § 6.4) show evidence for the default mesh simulation being converged, but it may be

[6]For movies see *"Research: Cores in Simulated Clusters"* at http://www.icc.dur.ac.uk/

[7]But we note that very high resolution 2D 'blob' simulations carried out by Springel (2005) do clearly show evidence for vortices. It is presently unclear if these are a consequence of the very different physical setup explored in that study (note that the gas density gradients are much smaller than in the present study) or the extremely high resolution used in their 2D simulations, and whether or not these vortices lead to enhanced mixing and entropy production.

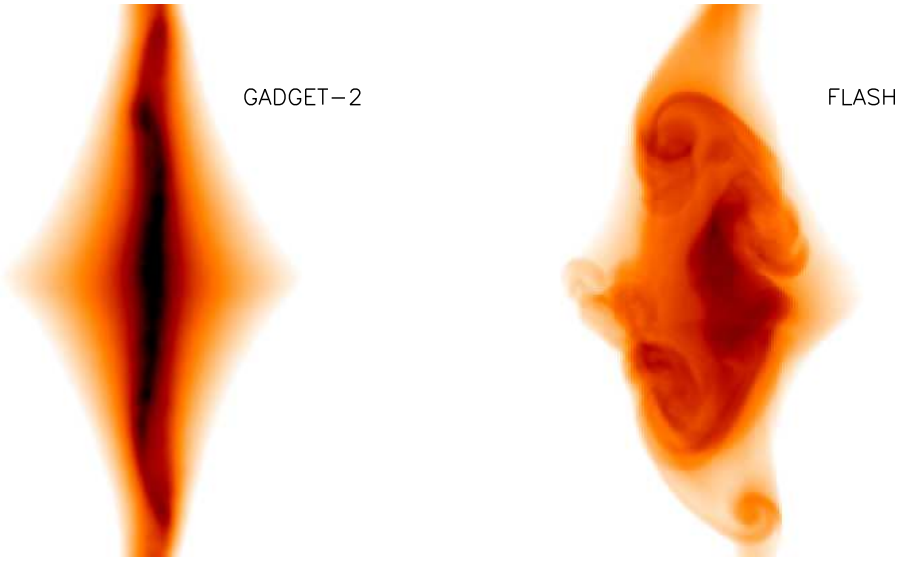

Figure 6.10: Logarithmic projected entropy maps of the default *GADGET-2* simulation and *FLASH* simulation with $l = 8$ at $t = 2.3$ Gyr, just after the collision of the cores. Note that the peak spatial resolutions of the two simulations are similar (approx. 10 kpc; we also note that the *median* SPH smoothing length for the default *GADGET-2* run is $\approx$ 20 kpc within the mixing region, $r \lesssim 200$ kpc.) To highlight the central regions, we have reset the value of any pixel with projected entropy greater than $0.5K_{200,i}$ to $0.5K_{200,i}$. In these maps, the minimum entropy (black) is $\approx 0.07K_{200,i}$ and the maximum entropy (white) is $0.5K_{200,i}$. The maps are 2 Mpc on a side, and project over a depth of 2 Mpc. Note that the images are not directly comparable with figure 6.2, where a slice is shown. *Left:* the *GADGET-2* simulation. *Right:* the *FLASH* simulation. The *FLASH* entropy distribution is characterised by vortices on a range of scales, which mix gas with different entropies. These vortices are mostly absent in the *GADGET-2* simulation.

that the resolution needs to be increased by much larger factors than we have tried (or are presently accessible with current hardware and software) in order to see a difference.

6.6 Summary and Discussion

In this chapter, we set out to investigate the origin of the discrepancy in the entropy structure of clusters formed in Eulerian mesh-based simulations compared to those formed in Lagrangian SPH simulations. While SPH simulations form clusters with almost power-law entropy distributions down to small radii, Eulerian simulations form much larger cores with the entropy distribution being truncated at significantly higher values. Previously it has been suspected that this discrepancy arose from the limited resolution of the mesh based methods, making it impossible for such codes to accurately trace the formation of dense gas structures at high redshift.

By running simulations of the merging of idealised clusters, we have shown that this is not the origin of the discrepancy. We used the *GADGET-2* code (Springel, 2005) to compute the SPH solution and the *FLASH* code (Fryxell et al., 2000) to compute the Eulerian mesh solution. In these idealised simulations, the initial gas density structure is resolved from the onset of the simulations, yet the final entropy distributions are significantly different. The magnitude of the difference generated in idealised mergers is comparable to that seen in the final clusters formed in full cosmological simulations. A resolution study shows that the discrepancy in the idealised simulations cannot be attributed to a difference in the effective resolutions of the simulations. Thus, the origin of the discrepancy must lie in the code's different treatments of gravity and/or hydrodynamics.

We considered various causes in some detail. We found that the difference was *not* due to:

- The use of different gravity solvers. The two codes differ in that *GADGET-2* uses a TreePM method to determine forces, while *FLASH* uses the PM method alone. The different force resolutions of the codes could plausibly lead to differences in the energy transfer between gas and dark matter. Yet we find that the dark matter distributions produced by the two codes are almost identical when the mesh code is run at comparable resolution to the SPH code.

- Galilean non-invariance of mesh codes. I investigate whether the results are changed if we change the rest-frame defined by the hydrodynamic mesh. Although I find that

an artificial core can be generated in this way in the mesh code, its size is much smaller than the core formed once the clusters collide, and is not enough to explain the difference between *FLASH* and *GADGET-2*. We show that most of this entropy difference is generated in the space of ~ 1 Gyr when the cluster cores first collide.

- Pre-shocking in SPH. We consider the possibility that the artificial viscosity of the SPH method might generate entropy in the flow prior to the core collision, thus reducing the efficiency with which entropy is generated later. By greatly reducing the artificial viscosity ahead of the core collision, we show that this effect is negligible.

Having shown that none of these numerical issues can explain the difference of the final entropy distributions, we investigated the role of fluid mixing in the two codes. Several recent studies (e.g., Dolag et al., 2005; Wadsley et al., 2008) have argued that if one increases the amount of mixing in SPH simulations the result is larger cluster entropy cores that resemble the AMR results. While this is certainly suggestive, it does not clearly demonstrate that it is the enhanced mixing in mesh simulations that is indeed the main driver of the difference (a larger entropy core in the mesh simulations need not necessarily have been established by mixing). By injecting tracer particles into our *FLASH* simulations, we have been able to make an explicit comparison of the amount of mixing in the SPH and mesh simulations of clusters. We find very substantial differences. In the SPH computation, there is a very close relation between the initial entropy of a particle and its final entropy. In contrast, tracer particles in the *FLASH* simulation only show a close connection for high initial entropies. The lowest $\sim 5\%$ of gas (by initial entropy) is completely mixed in the *FLASH* simulation. We conclude that mixing and fluid instabilities are the cause of the discrepancy between the simulation methods.

The origin of this mixing is closely connected to the suppression of turbulence in SPH codes compared to the Eulerian methods. This can easily be seen by comparing the flow structure when the clusters collide: while the *FLASH* image is dominated by large scale eddies, these are absent from the SPH realisation (see Figure 6.10). It is now established that SPH codes tend to suppress the growth of Kelvin-Helmholtz instabilities in shear flows, and this seems to be the origin of the differences in our simulation results (e.g., Agertz et al., 2007). These structures result in entropy generation through mixing, an irreversible process whose role is underestimated by the SPH method. Of course, it is not clear that the turbulent structures are correctly captured in the mesh simulations (Iapichino and Niemeyer, 2008; Wadsley et al., 2008). The mesh forces fluids to be mixed

on the scale of individual cells. In nature, this is achieved through turbulent cascades that mix material on progressively smaller and smaller scales: the mesh code may well overestimate the speed and effectiveness of this process. Ultimately, deep X-ray observations may be able to tell us whether the mixing that occurs in the mesh simulations is too efficient. An attempt at studying large-scale turbulence in clusters was made recently by Schuecker et al. (2004). Their analysis of *XMM-Newton* observations of the Coma cluster indicated the presence of a scale-invariant pressure fluctuation spectrum on scales of 40-90 kpc and found that it could be well described by a projected Kolmogorov/Oboukhov-type turbulence spectrum. If the observed pressure fluctuations are indeed driven by scale-invariant turbulence, this would suggest that current mesh simulations have the resolution required to accurately treat the turbulent mixing process. Alternatively, several authors have suggested that ICM may be highly viscous (eg., Fabian et al., 2003a) with the result that fluid instabilities will be strongly suppressed by physical processes. This might favour the use of SPH methods which include a physical viscosity (Sijacki and Springel, 2006).

It is a significant advance that we now understand the origin of this long standing discrepancy. Our work also has several important implications. Firstly, as outlined in § 6.2, there has been much discussion in the recent literature on the competition between heating and cooling in galaxy groups and clusters. The current consensus is that heating from AGN is approximately sufficient to offset cooling losses in observed cool core clusters (e.g., McNamara and Nulsen, 2007). However, observed present-day AGN power output seems energetically incapable of explaining the large number of systems that do not possess cool cores[8] (McCarthy et al., 2008b). Recent high resolution X-ray observations demonstrate that these systems have higher central entropies than typical cool core clusters (e.g., Dunn and Fabian, 2008). One way of getting around the energetics issue is to invoke an early episode of preheating (e.g., Kaiser, 1991; Evrard and Henry, 1991). Energetically, it is more efficient to raise the entropy of the (proto-)ICM prior to it having fallen into the cluster potential well, as its density would have been much lower than it is today (McCarthy et al., 2008b). Preheating remains an attractive explanation for these systems.

However, as we have seen from our idealised merger simulations, the amount of central

[8]Recent estimates suggest that $\sim 50\%$ of all massive X-ray clusters in flux-limited samples do not have cool cores (e.g., Chen et al., 2007). Since at fixed mass cool core clusters tend to be more luminous than non-cool core clusters, the fraction of non-cool core cluster in flux-limited samples may actually be an underestimate of the true fraction.

entropy generated in our mesh simulations is significant and is even comparable to the levels observed in the central regions of non-cool core clusters. It is therefore tempting to invoke mergers and the mixing they induce as an explanation for these systems. However, before a definitive statement to this effect can be made, much larger regions of parameter space should be explored. In particular, a much larger range of impact parameters and mass ratios is required, in addition to switching on the effects of radiative cooling (which we have neglected in the present study). This would be the mesh code analog of the SPH study carried out by Poole et al. (2006) (see also Poole et al., 2008). We leave this for future work. Alternatively, large cosmological mesh simulations, which self-consistently track the hierarchical growth of clusters, would be useful for testing the merger hypothesis. Indeed, Burns et al. (2008) have recently carried out a large mesh cosmological simulation (with the *ENZO* code) and argue that mergers at high redshift play an important role in the establishment of present-day entropy cores. However, these results appear to be at odds with the cosmological mesh simulations (run with the *ART* code) of Nagai et al. (2007) (see also Kravtsov et al., 2005). These authors find that most of their clusters have large cooling flows at the present-day, similar to what is seen in some SPH cosmological simulations (e.g., Kay et al., 2004; Borgani et al., 2006). On the other hand, the SPH simulations of Kereš et al. (2009) appear to yield clusters with large entropy cores. This may be ascribed to the lack of effective feedback in their simulations, as radiative cooling selectively removes the lowest entropy gas (see, e.g., Bryan, 2000; Voit et al., 2002), leaving only high entropy (long cooling time) gas remaining in the simulated clusters. However, all the simulations just mentioned suffer from the overcooling problem (Balogh et al., 2001), so it is not clear to what extent the large entropy cores in clusters *in either mesh or SPH* simulations are produced by shock heating, overcooling, or both. All of these simulations implement different prescriptions for radiative cooling (e.g., metal-dependent or not), star formation, and feedback, and this may lie at the heart of the different findings. A new generation of cosmological code comparisons will be essential in sorting out these apparently discrepant findings. The focus should not only be on understanding the differences in the hot gas properties, but also on the distribution and amount of stellar matter, as the evolution of the cold and hot baryons are obviously intimately linked. Reasonably tight limits on the amount of baryonic mass in the form of stars now exists (see, e.g., Balogh et al., 2008) and provides a useful target for the next generation of simulations. At present, merger-induced mixing as an explanation for intrinsic scatter in the hot gas properties of groups and clusters remains an open question.

Secondly, we have learnt a great deal about the nature of gas accretion and the development of hot gas haloes from SPH simulations of the universe. Since we now see that these simulations may underestimate the degree of mixing that occurs, which of these results are robust, which need revision? For example, Kereš et al. (2005) (among others) have argued that cold accretion by galaxies plays a dominant role in fuelling the star formation in galaxies. Is it plausible that turbulent eddies could disrupt and mix such cold streams as they try to penetrate through the hot halo? We can estimate the significance of the effect by comparing the Kelvin-Helmholtz timescale, $\tau_{\rm KH}$, with the free-fall time, $\tau_{\rm FF}$. The KH timescale is given by (see, e.g., Nulsen, 1982; Price, 2008)

$$\tau_{\rm KH} \equiv \frac{2\pi}{\omega} \tag{6.7}$$

where

$$\omega = k\frac{(\rho\rho')^{1/2}v_{\rm rel}}{(\rho+\rho')} \tag{6.8}$$

and ρ is the density of the hot halo, ρ' is the density of the cold stream, k is the wave number of the instability, and $v_{\rm rel}$ the velocity of the stream relative to the hot halo. If the stream and hot halo are in approximate pressure equilibrium, this implies a large density contrast (e.g., a 10^4 K stream falling into a 10^6 K hot halo of a Milky Way-type system would imply a density contrast of 100). In the limit of $\rho' \gg \rho$ and recognising that the mode responsible for the destruction of the stream is comparable to the size of the stream (i.e., $k \sim 2\pi/r'$), eqns. (6.7) and (6.8) reduce to:

$$\tau_{\rm KH} \approx \frac{r'}{v_{\rm rel}}\left(\frac{\rho'}{\rho}\right)^{1/2} \tag{6.9}$$

Adopting $\rho'/\rho = 100$, $r' = 100$ kpc, and $v_{\rm rel} = 200$ km/s (perhaps typical numbers for a cold stream falling into a Milky Way-type system), we find $\tau_{\rm KH} \sim 5$ Gyr. The free-fall time, $\tau_{\rm FF} = R_{\rm vir}/V_{\rm circ}(R_{\rm vir})$ [where $R_{\rm vir}$ is the virial radius of main system and $V_{\rm circ}(R_{\rm vir})$ is the circular velocity of the main system at its virial radius], is ~ 1 Gyr for a Milky Way-type system with mass $M_{\rm vir} \sim 10^{12}M_\odot$. On this basis, it seems that the stream would be stable because of the large density contrast in the flows. It is clear, however, that the universality of these effects need to be treated with caution, as the free-fall and KH timescales are not vastly discrepant. High resolution mesh simulations (cosmological or idealised) of Milky Way-like systems would provide a valuable check of the SPH results.

Finally, the SPH method has great advantages in terms of computational speed, effective resolution and Galilean invariance. Is it therefore possible to keep these advantages and add additional small scale transport processes to the code in order to offset the suppression of mixing? Wadsley et al. (2008) and Price (2008) have presented possible approaches based on including a thermal diffusion term in the SPH equations. Although the approaches differ in their mathematical details, the overall effect is the same. However, it is not yet clear how well this approach will work in cosmological simulations that include cooling (and feedback), since the thermal diffusion must be carefully controlled to avoid unphysical suppression of cooling in hydrostatic regions (e.g., Dolag et al., 2005). One possibility might be to incorporate such terms as a negative surface tension in regions of large entropy contrast (Hu and Adams, 2006). An alternative approach is to combine the best features of the SPH method, such as the way that it continuously adapts to the local gas density and its flow, with the advantage of a Riemann based method of solving the fluid dynamic equations (e.g., Inutsuka, 2002).

Clearly, there is a great need to find simple problems in which to test these codes: simple shock tube experiments are not sufficient because they do not include the disordered fluid motions that are responsible for generating the entropy core. Idealised mergers represent a step forward, but the problem is still not sufficiently simple that it is possible to use self-similar scaling techniques (e.g., Bertschinger, 1985, 1989) to establish the correct solution . One possibility is to consider the generation of turbulent eddies in a fluctuating gravitational potential. We have begun such experiments, but (although the fluid flow patterns are clearly different) simply passing a gravitational potential through a uniform plasma at constant velocity does not expose the differences between SPH and Eulerian mesh based methods that we see in the idealised merger case. We will tackle the minimum complexity that is needed to generate these differences in a future paper.

6.7 Afterword

6.7.1 Does Numerical Noise in the Particle-Mesh Approach Induce Excess Mixing in AMR Codes?

It is a well known fact that flows where the kinetic energy greatly dominates the internal energy, i.e. those that are very cold and move supersonically with respect to the mesh, often exhibit spurious heating in adiabatic regions of the flow (see, e.g. Ryu et al., 1993; Bryan et al., 1995). This arises from finite discretisation errors that lead to small amounts

of dissipation occurring within the cold gas. As any small advection errors within a numerical scheme can result in kinetic energy instantly being transferred to the internal energy, it leads to inaccurate temperatures and internal energies. As gravitational collapse in early structure formation involves large amounts of cold gas moving at high velocities, this problem can become quite significant. To overcome this, *FLASH* separately evolves the internal energy whenever it falls below a given fraction of the kinetic energy (10^{-4} by default) instead of obtaining it from the difference between the total and kinetic energy. This ensures that rounding errors are minimised, allowing for correct energy evolution.

Since the publication of the other material in this chapter (i.e. that which was published in Mitchell et al., 2009), a new Galilean invariant moving mesh code, *AREPO*, has been released. In Springel (2010), the author claims that the code is able to reduce spurious entropy generation due to weak shocks and that it is these weak shocks that represent the excess entropy generation in AMR codes. If correct, this result could help us to ascertain the correct amount of entropy generation that should occur within cosmological simulations.

The new hybrid code offers the benefit of a fully Galilean invariant hydrodynamics prescription, similar to an SPH code, along with the ability to model the formation of fluid instabilities as in a conventional AMR code. I refer interested readers to Springel (2010) for a detailed description of the *AREPO* code but note that it makes use of an unstructured mesh generated using Voronoi tessellation and each cell has roughly equal mass. The code allows for advection of mass, momentum and energy using the Euler equations for compressible fluid flow like an AMR code and incorporates the use of a Riemann solver. However, similar to an SPH code the cells are able to advect with the bulk fluid flow and it refines dynamically in regions of increasing density. *AREPO* therefore seems ideally suited for cosmological applications where bulk velocities can become highly supersonic and refinement is often best concentrated in the densest regions.

Springel (2010) claims that spurious generation of entropy occurs within AMR codes when collisionless systems such as stars and dark matter which dominate the gravitational potential, are represented by a discrete mass distribution. This produces a relatively noisy gravitational potential with bumpy regions resulting from the discretised mapping of particle properties to the mesh. When coupled to the hydrodynamics, this drives stochastic mixing of the gas and spurious entropy generation as the hydrodynamics dissipate the small scale velocity fluctuations. This is most likely suppressed in an SPH code due to the smoothing effects of the Kernel.

Springel (2010) enables the *AREPO* code to remove spurious entropy generation in these weak shocks by advecting an additional variable; an entropic function $S = P/\rho^{\gamma}$. The entropic function is directly related to the actual thermodynamic entropy, S_{therm} by

$$\frac{S_{therm}}{N} = \frac{3}{2}\left[\ln\frac{S}{N} + \ln\left(\frac{2\pi m^{5/3}}{h^2}\right) + \frac{5}{3}\right] \tag{6.10}$$

where N is the number of atoms, m their mass and h is Planck's constant. Normally the advected entropy is discarded and recalculated based on the Riemann solver output densities and pressures. This is to be expected given that the Euler hydrodynamic equations automatically deal with the generation of entropy within shocks. However, in the absence of shocks the entropy remains constant. By applying a suitable criterion, it is possible to discard the updated energy from the Riemann solver and use the advected entropy in its place to calculate the pressure and energy in regions of cold flows. Springel (2010) implements a simple Mach number criterion, comparing the greatest Mach number across the cell's faces to a threshold Mach number. In this "entropy-energy" formalism, if the cell's greatest Mach number is less than the threshold, then the entropy is used to update the cell's thermodynamic quantities. In all other regions the Riemann solver solution is maintained.

To demonstrate the effect that this has upon entropy generation in cosmological simulations, Springel (2010) runs the Santa Barbara cluster in the *AREPO* code. He applies both the "total-energy" approach where the internal energy is based purely on the difference between the total and kinetic energy, and the above "entropy-energy" scheme using the entropy when the local Mach number falls below the threshold; $M_{thresh} = 1.1$. The resulting entropy as a function of radius is shown in the right hand panel of figure 6.11 (plot obtained from figure 45, Springel, 2010) along with the results from the original Santa Barbara code comparison displayed in the left hand panel (plot obtained from figure 18, Frenk et al., 1999). The green, red and dark blue lines in the right hand panel (RHP) of figure 6.11 show the "entropy-energy" implementation in *AREPO* at different levels of resolution; 32^3, 64^3 and 128^3 cells respectively and clearly show good convergence. The open blue circles represent the high resolution 128^3 run using the "total-energy" formalism whilst the light blue diamonds represent the results obtained from the *GADGET-2* SPH code using 128^3 particles.

The different symbols in the left hand panel (LHP) of figure 6.11 correspond to the entropy values obtained by different authors using different SPH and AMR codes (see legend).

The simulations of Couchman (*Hydra*), Jenkins & Pearce (parallel *Hydra*), Evrard (SPH P^3M), Navarro (*Grape* + SPH), Steinmetz (*Grape* + SPH), Owen (Adaptive SPH), Wadsley (P^3M SPH) and Gnedin (Smooth Lagrangian Hydrodynamics) represent different SPH implementations. The simulations of Cen (*TVD*), Pen (Moving Mesh Hydrodynamics), Yepes (*PM-FCT*) and Bryan (*SAMR* - predecessor to the *ENZO* code presented in Chapter 4) represent different AMR codes. I refer the reader to Frenk et al. (1999) for a detailed description of each code. It is worth noting that the simulations of Jenkins & Pearce are performed in the parallel Hydra code which was found in chapter 4 to produce excellent agreement with the *GADGET-2* code. This is evident from the strong correlation between the 128^3 *GADGET-2* simulation in the RHP of figure 6.11 and the *Hydra* results in the LHP (open stars).

Comparison of the different energy schemes in the moving mesh code reveals that whilst both produce higher entropy cores than in SPH, the "energy-entropy" method is able to cut the entropy generation relative to the "total-energy" implementation by over $\sim 40\%$. This brings the "entropy-energy" results much closer into line with SPH. Springel (2010) claims that this reduction in the entropy generated is conclusive proof that weak shocks due to noisy gravitational potentials lead to spurious generation of entropy within AMR codes. However I find that simulations run with a smoothed dark matter distribution using Paul Ricker's "`SmoothMapParticlesToMesh`" routine (see chapter section 2.4) give identical entropy profiles. As this removes small scale anisotropies brought on through the mapping of discrete particle properties to the mesh, it would suggest that such local variations in the gravitational potential are not responsible for driving the increased mixing seen in the mesh codes. It also worth noting that the entropy core in the *AREPO* "total-energy" implementation is $\sim 20\%$ higher than in the *SAMR* code. Clearly this requires more careful investigation.

To conclusively determine whether the collisionless component generates an excess of weak shocks, we perform mergers between clusters composed entirely of gas. These simulations use the same default cluster setup but adopt clusters composed purely of gas instead of a realistic mixture both of gas and dark matter. The gas maintains a power-law profile as before but now constitutes the total mass of the cluster (i.e. the Baryon fraction is increased to one). This allows for the removal of the discrete collisionless component and ensures that the gravitational potential is modelled on the same grid as the hydrodynamic flow. Clearly if the mapping of particle properties to and from the mesh induces numerical noise, leading to spurious mixing and entropy generation, then gas-only simulations should

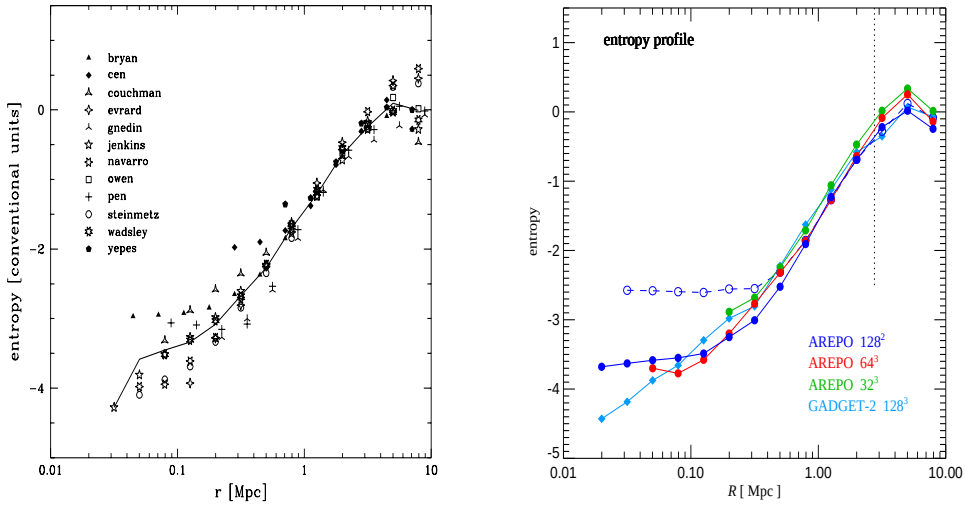

Figure 6.11: Plots of entropy as a function of radius for the Santa Barbara code comparison using different SPH and AMR codes (left hand panel, figure 18 from Frenk et al., 1999) and for the cluster run with the *AREPO* moving mesh code (right hand panel, figure 45 from Springel, 2010). Different symbols in the left hand panel (LHP) of correspond to profiles obtained by different authors using different SPH and AMR codes (see legend). The *SAMR* code of Bryan is predecessor to the *ENZO* AMR code. The simulations of Jenkins & Pearce (open stars) use the parallel SPH Hydra code which according to chapter 4 agrees well with *GADGET-2*. This is evident from the excellent agreement between the 128^3 particle *GADGET-2* simulation in the RHP (light blue diamonds) and *Hydra* in the LHP (open stars). The green, red and dark blue lines in the right hand panel (RHP) show the "entropy-energy" implementation in *AREPO* at different levels of resolution; 32^3, 64^3 and 128^3 cells respectively. Open blue circles represent the high resolution 128^3 run using the "total-energy" formalism. Both static and moving mesh codes generate entropy cores with the "total-energy" *AREPO* implementation agreeing to within $\sim \pm 20\%$ with the *SAMR* results. The *AREPO* "entropy-energy" implementation has a much lower entropy core closer inline with the SPH codes which lack a core.

possess much smaller cores. The resulting entropy at time $t = 10\ Gyr$, normalised to the initial profile at time $t = 0\ Gyr$ is plotted in figure 6.12 as a function of the fraction of the enclosed gas mass. The gas-only runs for *FLASH* (blue) and *GADGET-2* (green) are plotted along side the default *FLASH* simulation (red) which includes a realistic mixture of gas and dark matter. Clearly, not only does the *FLASH* gas-only run generate a large entropy core within the central ten percent of the total cluster gas mass, its amplitude is much greater than for the default simulation. The *GADGET-2* simulation on the other hand converges upon a similar solution to the default configuration with a collisionless component. Thus the amplitude of the entropy difference between SPH and AMR cores is more than doubled when using a pure hydrodynamic implementation.

Based upon these results it is clear that the collisionless component plays no significant role in the generation of these entropy cores. This would seem to be supported by the comparatively slow convergence of the dark matter (§ 6.5.1) relative to the entropy (§ 6.4.1). Interestingly it is noted by Springel (2010) that during purely hydrodynamic simulations of the formation of a Kelvin-Helmholtz instability, the new *AREPO* code demonstrates less mixing than that of a static AMR code. As this test lacks any gravity to generate weak shocks, it would appear that the difference in the levels of mixing and thus entropy generation, is a fundamental property of the hydrodynamic implementation. It is currently unknown which of the two mesh types provides a more accurate result; is the structured AMR mesh too inflexible to correctly follow the fluid flow or do the unstructured mesh faces bias the flow direction and suppress the formation of vortices? Clearly further detailed study into these effects is required before we can truly ascertain the correct solution. However I can confirm that the increased generation of entropy in the formation of our simulated clusters does not occur due to the presence of a noisy gravitational potential from mapping of particle properties to the mesh.

6.7.2 Conclusion

Determining the correct physical level of mixing may prove to be non-trivial. Springel (2010) has claimed that the level of mixing in AMR codes is too high based upon evidence from his new Galilean invariant mesh code, *AREPO*. When he runs the Santa Barbara cluster in *AREPO* he finds that the entropy generation is less than in conventional fixed mesh codes, though still greater than in the *GADGET-2* SPH code. He claims the greater entropy generation in AMR codes is caused by spurious noise in the gravitational potential as particle properties are mapped to the mesh. This drives stochastic mixing and weak

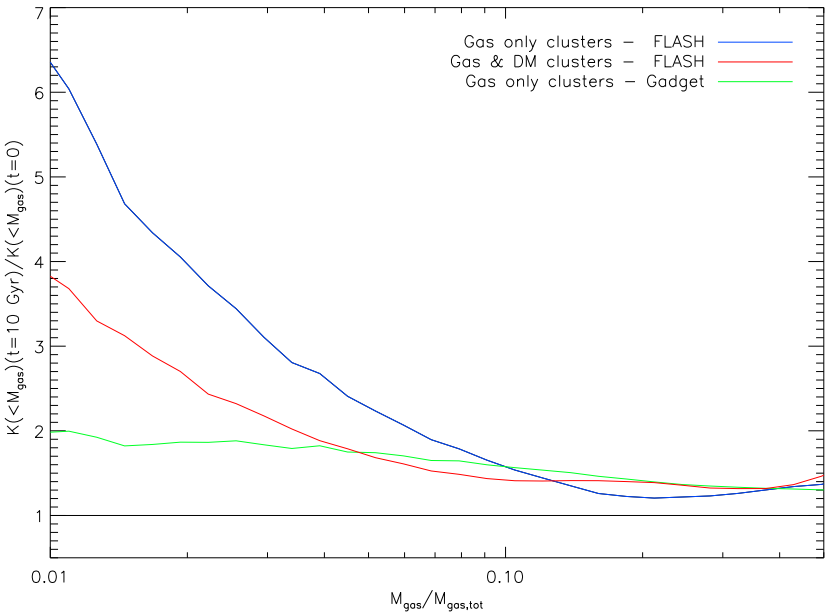

Figure 6.12: A plot demonstrating the effect of running with and without a particle based dark matter component in *FLASH* and *GADGET-2*. The entropy of the enclosed gas mass at time $t = 10$ Gyr normalised to the initial entropy at time $t = 0$ Gyr is plotted as a function of enclosed gas mass. The default *FLASH* simulation (blue line) contains a realistic mix of both gas and dark matter whilst the red and green lines correspond to *FLASH* and *GADGET-2* runs using clusters composed entirely of gas. These have the same total mass, initial power-law entropy profile and collision setup but lack the addition of any particle components. The gas only run in *GADGET-2* reproduces results for runs with both gas and dark matter indicating no change in the entropy generated whilst the difference between *FLASH* and *GADGET-2* more than doubles when the gas fraction is increased to one. Clearly a noisy gravitational potential due to the mapping of particle properties to the mesh is not responsible for the cores in AMR simulations.

shocks. In SPH codes these weak shocks would be suppressed by the smoothing Kernel. By implementing an "energy-entropy" scheme which is able to update the hydrodynamic variables in regions of weak shocks using the conserved entropy in place of the Riemann solution, he is able to largely suppress most of the entropy generation. Currently the switch between using the entropy and the Riemann solution to update the hydrodynamic variables is based on an arbitrary Mach number threshold.

We test this hypothesis by running gas only cluster mergers with no particle based dark matter component. If the noise in the gravitational potential is responsible for the excess entropy in the AMR codes then this should result in a large decrease in the core amplitude. However we find that the entropy difference between the two codes is amplified, offering firm proof that the generation of entropy is not due to noise in the gravitational potential but more an intrinsic property of the hydrodynamic implementation. Indeed it is interesting to note that for simple Kelvin-Helmholtz tests in *AREPO*, Springel (2010) finds that the moving mesh code generates less mixing than a conventional AMR code, an effect which is currently not understood.

Chapter 7

Sub-Grid Models for Star Formation in FLASH

7.1 Introduction

Star formation is perhaps the greatest observable property of any galaxy and most modern surveys have thrown up a wealth of information on their luminosities and spectral types. Recent works have shown a deep interest in the ability of supernovae to help regulate cooling and star formation in galaxies in an effort to improve properties of model galaxies (see, e.g., Booth et al., 2007; Dalla Vecchia and Schaye, 2008; Dubois and Teyssier, 2008; Schaye et al., 2010).

Supernova driven winds may reduce the production of new stars and help regulate star formation over the lifetime of the galaxy. However the resolution necessary to self-consistently model them is beyond current capabilities. Galaxies are typically tens of thousands of kilo-parsecs across whilst the coalescence of multiple supernova driven bubbles barely spans ~ 200 pc. An isolated galaxy simulation will therefore need to resolve more than two orders of magnitude to correctly resolve individual superbubbles. To scale such simulations up to cosmological scales spanning many giga-parsecs would require hardware beyond what is currently available. To overcome such limitations it is necessary to adopt sub-grid prescriptions. These allow us to overcome resolution limitations but require us to iterate simulations until they reproduce observations.

An increasing number of investigations are beginning to show major differences between the relative performance of different hydrodynamic implementations. Agertz et al. (2007) for instance has highlighted the spurious suppression of instabilities in SPH codes which AMR codes are able to resolve and the work I present in chapter 6 has shown how significant the different levels of mixing in the two codes can be when performing

mergers. As models are often validated or refuted through comparison to work by the wider scientific community, this mandates the comparison of different sub-grid models embedded in different hydrodynamic implementations. Given we now know that various hydrodynamic codes perform differently at a fundamental level, it raises the question as to whether identical sub-grid models will yield the same results when run in different codes? If there are indeed differences implicit to the numerical schemes then it presents considerable problems when trying to compare them. As the current public release version of *FLASH* lacks any modules for star formation and cooling, I will attempt to tackle these two issues simultaneously. I will implement sub-grid models for cooling, star formation and supernova feedback into the *FLASH* AMR code in as near identical fashion to those in the *GADGET-2* SPH code and contrast the results. This will allow for a better understanding of the different hydrodynamics algorithms and any potential limitations that they may face.

I will structure this chapter as follows. In § 7.2 I describe the star formation recipe of Schaye and Dalla Vecchia (2008) along with my motivation for adopting it. I will outline the polytropic equation of state that Schaye and Dalla Vecchia (2008) use to model the unresolved multi-phase ISM in § 7.2.1 and explain their choice of parameters. I will also briefly outline the cooling routines that they use in § 7.2.2.

In § 7.3 I describe the difficulties faced when implementing their SPH star formation model into the *FLASH* AMR code and the solutions I adopt along with preliminary tests to evaluate its performance (§ 7.4).

In § 7.5 I summarise the initial conditions used in the model disk galaxy simulations which Schaye and Dalla Vecchia (2008) uses to test their star formation prescription. Small modifications are required to enable *FLASH* to run the model disk galaxy which I detail in § 7.6 and I contrast the results from *FLASH* and *GADGET-2* in § 7.7. I conclude my findings in § 7.8.

7.2 Star Formation of Schaye and Dalla Vecchia (2008)

One of the most sophisticated star formation models currently available is that of Schaye and Dalla Vecchia (2008). Implemented into state-of-the-art cosmological simulations run by the Virgo Consortium; the *Galaxies-Intergalactic Medium Interaction Calculation* (*GIMIC*, see chapter 1.7.5 Crain et al., 2009) and the *Over-Whelmingly Large Simulations* (*OWLS*, see Schaye et al., 2010; Navarro et al., 2009), it allows any arbitrary star formation

law written in terms of surface densities to be accurately reproduced within a three dimensional simulation. Part of my motivation for using the same sub-grid physics in *FLASH* as in the *OWLS* and *GIMIC* simulations, is that it would permit *FLASH* to self-consistently follow on from their high-redshift outputs, allowing smaller and more detailed simulations to be built upon their cosmological framework.

Observationally, Kennicutt and Schmidt showed that the projected surface star formation density for galaxies per unit time, $\dot{\Sigma}_*$, when averaged over large scales follows the empirical relation in equation (7.1) where $\Sigma_{g,c}$ is the threshold surface density below which star formation ceases (see, e.g., Kennicutt, 1998a,b; Kennicutt et al., 2007, 2009).

$$\dot{\Sigma}_* = \left\{ \begin{array}{ll} 0 & if\ \Sigma_g < \Sigma_{g,c} \\ A(\Sigma_g/1\mathrm{M}_\odot\,\mathrm{pc}^{-2})^n & if\ \Sigma_g \geq \Sigma_{g,c} \end{array} \right\} \tag{7.1}$$

The results from Kennicutt (1998b) are shown in figure 7.1 for a sample of 97 spiral galaxies spanning five orders of magnitude. Kennicutt (1998b) clearly showed that there was a strong linear relation in logarithmic space between the star formation rate and surface gas density of galaxies. He found the data was best fit by the relation

$$\dot{\Sigma}_* = (2.5 \pm 0.7) \times 10^{-4}\mathrm{M}_\odot\,\mathrm{yr}^{-1}\,\mathrm{kpc}^{-2} \left(\frac{\Sigma_g}{1\mathrm{M}_\odot\,\mathrm{pc}^{-2}} \right)^{(1.4\pm0.15)} \tag{7.2}$$

This relation was found to hold for a range of galaxies, from strong star bursts (black filled squares in figure 7.1) to relatively quiet quiescently forming disk galaxies (dark filled circles in figure 7.1). The exponent, n, and the normalisation, A, are the two variable parameters that we adopt in our sub-grid star formation recipe and can be changed to fit any observed Kennicutt-Schmidt style star formation law. This relation can also be expressed in terms of a gas consumption timescale

$$\begin{array}{rl} t_g \equiv \frac{\Sigma_g}{\dot{\Sigma}_*} & = A^{-1} \left(1\mathrm{M}_\odot\,\mathrm{pc}^{-2}\right)^n \Sigma_g^{(1-n)} \\ & = (4 \pm 1) \times 10^9 \mathrm{yr} \left(\frac{\Sigma_g}{1\mathrm{M}_\odot\,\mathrm{pc}^{-2}} \right)^{-(0.4\pm0.15)} \end{array} \tag{7.3}$$

Early theoretical work by Toomre (1964) described the onset of gravitational instability in an arbitrary, highly flattened stellar system that is differentially rotating. He derived the two following stability criteria for a gas disk (7.4) and for a stellar disk (7.5):

$$Q \equiv \frac{c_s k}{\pi G \Sigma_g} > 1 \tag{7.4}$$

$$Q \equiv \frac{\sigma_R k}{3.36 G \Sigma_*} > 1 \tag{7.5}$$

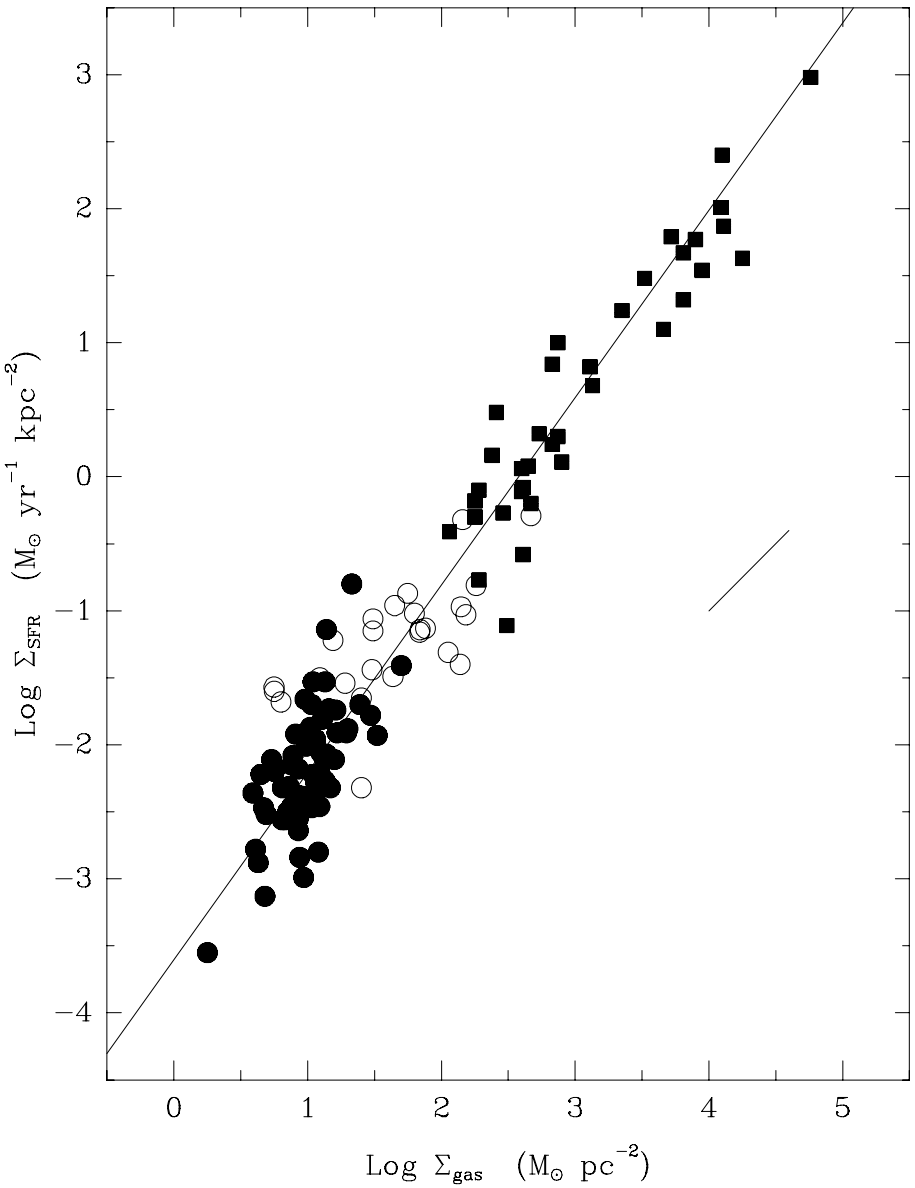

Figure 7.1: Surface star formation rate plotted against gas surface density from Kennicutt's study of 97 spiral galaxies. The study shows the composite star formation law for normal disk (solid circles) and starburst (squares) galaxy samples. Open circles show the star formation rates and gas densities for the centres of the normal disk galaxies. Plot reproduced from figure 6 in Kennicutt (1998b).

where c_s the sound speed, k is the epicyclic-frequency of the perturbation, σ_R is the radial velocity dispersion, G the gravitational constant, Σ_g the surface gas density of the disk and Σ_* the stellar surface density. Both formulae show similar behaviour between a gas and collisionless stellar component. If the Toomre parameter Q is less than one then both pressure and rotational support are insufficient to prevent the growth of instabilities (see Toomre, 1964; Binney and Tremaine, 1987). Toomre's criterion was substantiated with observational evidence when Kennicutt (1989) compared a sample of 15 disk galaxies and found $Q \approx 2$, and later by Martin and Kennicutt (2001) who obtained $Q \approx 1.89$ for a sample of 32 galaxies.

Schaye (2004) explains the sharp cut-off in the star formation rate as a sudden phase transition between a single phase warm ISM to a multi-phase ISM. As material cools below 10^4 K to form molecular clouds, it reduces the thermal velocity dispersion and thus the effective Toomre Q value. Thus "hot" disks can be summarised as having large velocity dispersions and high Q, whilst "cool" disks have low dispersion and low Q. The existence of thermal-gravitational instabilities requires that most star formation recipes include both a density and temperature (or pressure) threshold.

As molecular clouds are very compact relative to the volumes they occupy, their collisional cross section is extremely small and thus to a large extent they can be treated as collisionless. To this end the star forming phase of the ISM is effectively self-gravitating. As observations show that the gas fraction in most galaxies is observed to be very low ($f_g \leqslant 0.3$, see McGaugh and de Blok, 1997) a proportion of which in the form of molecular clouds, it becomes clear that most of the mass in a galactic disk is in either a collisionless stellar or molecular cloud form. Based upon this Schaye and Dalla Vecchia (2008) relate the Kennicutt-Schmidt surface density law to a volume density law by assuming that if self-gravity is important then the density will typically fluctuate over the local Jeans length scale. For self-gravitating disks, this implies that the scale height of the disk will be of order the local Jeans scale and thus the gas column density is of order the "Jeans column density."

$$\Sigma_g \sim \Sigma_{g,J} \equiv \rho_g \lambda_J \tag{7.6}$$

$$= \left(\frac{\gamma k_B}{\mu G X}\right)^{1/2} (f n_H T)^{1/2} \tag{7.7}$$

$$= \left(\frac{\gamma}{G}\right)^{1/2} (f_g P_{tot})^{1/2} \tag{7.8}$$

where γ is the ratio of specific heats, X the hydrogen mass fraction, $f \equiv f_g/f_{th}$, with f_g the

mass fraction in gas (within a scale height of the gas) and f_{th} the fraction of the mid-plane pressure that is thermal. P_{tot} is the total mid-plane pressure including both thermal and non-thermal pressure components. Using equation (7.7), assuming a gas fraction $f_g = 0.3$ and a mean molecular weight of $\mu = 0.59$ (most of the gas is assumed ionised around the star forming threshold), it is possible to convert the observed density cut-off of $\Sigma_{g,c} = 7.3 M_\odot\ pc^{-2}$ into a volume number density of $\rho_{g,c} = 1.672 \times 10^{-25}$ g cm^{-3} (equivalent to $n_H = 0.1$ cm^{-3}) along with a pressure threshold of $P_{tot,c}/k_B = 2.3 \times 10^3$ cm^{-3} K.

Using equation (7.8), the gas consumption timescale (7.3) can be written as

$$t_g = A^{-1} \left(1 \mathrm{M}_\odot\, \mathrm{pc}^{-2}\right)^n \left(\frac{\gamma}{G} f_g P_{tot}\right)^{(1-n)/2} \tag{7.9}$$

thus the Schmidt law can be written as

$$\dot{\rho_*} \equiv \frac{\rho_g}{t_g} = A \left(1 \mathrm{M}_\odot\, \mathrm{pc}^{-2}\right)^{-n} \left(\frac{\gamma}{G} f_g P_{tot}\right)^{(n-1)/2} \rho_g \tag{7.10}$$

where t_g is the gas consumption timescale. The mass of stars which are formed in a particular cell over a given timestep is

$$\dot{m}_* \equiv \frac{m_g}{t_g} = m_g \frac{\dot{\rho_*}}{\rho_g} \tag{7.11}$$

$$= A (1 \mathrm{M}_\odot\, \mathrm{pc}^{-2})^{-n} m_g \left(\frac{\gamma}{G} f_g P_{tot}\right)^{(n-1)/2} \tag{7.12}$$

where m_g represents the cell or particle gas mass for which we are computing $\dot{m}_*$. The number of stars formed is then determined through a stochastic approach. The SPH prescription determines the probability of converting an entire gas particle of mass m_g into a star particle in a timestep of size Δt to be

$$Prob = min\left(\frac{\Delta t}{t_g}, 1\right) = min\left(\frac{\dot{m}_* \Delta t}{m_g}, 1\right) \tag{7.13}$$

where the mass of the star and gas particle are identical.

As the Schaye and Dalla Vecchia (2008) star formation recipe outlined above is derived through analytic relations, only two parameters exist, the Kennicutt-Schmidt law exponent, n, and the normalisation, A. As these values are constrained observationally, there exist no "free" parameters which can be varied by the user until the model reproduces some set of observations. Schaye and Dalla Vecchia (2008) demonstrates that this lack of free parameters is a key advantage with their model successfully reproducing the input star formation law irrespective of the chosen equation of state. This will prove to be

extremely advantageous when running very large simulations since it will remove the need to run several different star formation laws in the hope that one will mimic observations. As a result it has been adopted as the star formation recipe of choice for the *GIMIC* and *OWLS* simulations.

7.2.1 Modelling the Multiphase ISM with a Polytropic Equation of State

Hubble space telescope images of the Eagle and Horse head nebula clearly reveal an intricate web of sub-structures with very dense molecular clouds enshrouding the next generation of proto-stars, interspersed with hot stellar winds and supernova driven gas outflows. Simulations which fail to resolve these sub-parsec scale regimes typically represent this multiphase interstellar medium as a single fluid with a mean density, temperature, internal energy and pressure. Supernova driven winds should drive mass out through the under-dense channels between compact molecular clouds, but if the multiphase ISM is represented by a single fluid then the steep density contrasts are lost and ejected material may experience greater resistance as it attempts to leave the nebula. In addition we cannot resolve the peak densities that molecular clouds collapse down to before forming stars. Thus any density thresholds that we impose must correspond to spatial averages in order to make any physical sense.

Resolving the multi-phase ISM in the simple disk galaxy simulations presented in this chapter, would require a minimum of ten levels of refinement to resolve 100 parsec diameter molecular clouds. At this level of resolution at least three cells would be placed across them. Although it is hard to predict the exact pattern of refinement for such a simulation, a uniform mesh would require more than eleven terabytes of memory. Naturally it is hoped that the adaptive nature of the AMR code would be able to reduce the need for a uniform mesh and reduce memory consumption, but nevertheless the calculation with current resources would prove unfeasible. This would be especially true if we wished to extend the simulation up to galaxy group and cluster scales.

To compensate for our current inability to self-consistently model the multiphase medium, polytropic equations of state are often adopted (see, e.g., Yepes et al., 1997; Springel and Hernquist, 2003; Ocvirk et al., 2008; Dubois and Teyssier, 2008). These approximate the pressure generated due to sub-grid processes as a simple function of density with a given polytropic index, γ_{eff}:

$$P_{tot} = P_{tot,c} \left(\frac{\rho_g}{\rho_{g,c}} \right)^{\gamma_{eff}} \tag{7.14}$$

Here the total pressure, P_{tot}, represents both the thermal and non-thermal pressure contributions within the disk. The thermal pressure component due to the gas as used by the hydrodynamic algorithms is given by $P_{th} = f_g P_{tot}$. A comparison can be made to the normal equation of state for an ideal gas

$$P = \rho \varepsilon_{int} (\gamma - 1) \tag{7.15}$$

which scales proportionally to density and has a dependence upon the internal energy, ε_{int}, and thus temperature. It is important to note that the ratio of specific heats, γ and the polytropic index, γ_{eff}, are not necessarily the same. A lack of dependence upon temperature in the polytropic equation of state is preferred as it is not clear what effective temperature a multi-phase medium composed of both hot and cold gas phases should have.

In order for material to pass onto the effective equation of state, its density must exceed the star forming threshold density, $\rho_{g,c}$ and its temperature must have dropped below a threshold temperature, T_c, which Schaye and Dalla Vecchia (2008) set to $10^5 K$. They stress that the results are insensitive to the exact value chosen for the temperature threshold as long as it lies between the point in most simulations where the cooling time from the numerical cooling routines rises sharply ($\sim 10^4$ K) and where it is a minimum (10^5 K). For consistency I maintain the same temperature and density thresholds in *FLASH* as Schaye and Dalla Vecchia (2008) do in *GADGET-2*.

The value of the polytropic index directly controls the amount of pressure support against gravitational collapse and as such its value should be carefully chosen. For a polytropic equation of state as in equation (7.14), substitution into the Jeans equation (1.24) reveals that the Jeans length and mass scale as

$$\begin{aligned} \lambda_J &\propto f_g^{1/2} \rho_g^{(\gamma_{eff}-2)/2} \\ M_J &\propto f_g^{3/2} \rho_g^{(3\gamma_{eff}-4)/2} \end{aligned} \tag{7.16}$$

Thus a value for $\gamma_{eff} = 2$ gives a constant Jeans length and a Jeans mass, $M_J \propto \rho_g$. For $\gamma_{eff} \geq 2$ self gravity alone should be insufficient to promote further collapse. For $4/3 < \gamma_{eff} < 2$ gas clouds should be able to collapse without fragmenting whilst for $\gamma_{eff} < 4/3$ collapse can occur with the possibility of fragmentation.

Schaye and Dalla Vecchia (2008) opt to implement a value of $\gamma_{eff} = 4/3$ in order to give a Jeans mass which is independent of the gas density and a Jeans length that scales

as $\lambda_J \propto \rho_g^{-1/3}$. In SPH this proves to be important as the spatial resolution scales with the SPH smoothing Kernel and it in turn can be fixed so as to ensure that it scales in-line with the Jeans length. As the SPH smoothing Kernel $h \propto (m_g/\rho_g)^{1/3} \propto (m_g/M_J)^{1/3}\lambda_J$ then for a suitable choice of the particle mass, m_g, as the ratio of m_g/M_J is independent of density, then so too is h/λ_J.

As gas collapses and is crushed to greater densities, the cooling rate will increase and thus the pressure will drop rapidly. By making the polytropic equation of state dependent purely on the density, it ensures that increased density leads to an increase in the pressure which resists collapse. This resists an unphysical runaway collapse scenario. The excess pressure relative to a standard ideal equation of state represents the radiative contributions from stars and the energy injection from supernova which support the ISM. High resolution simulations would require radiative transfer in order to fully incorporate these effects.

7.2.2 Metal Dependent Cooling

In order to incorporate the effects of metal dependent cooling, Schaye and Dalla Vecchia (2008) chose to implement the cooling routines of Wiersma et al. (2009). These make use of pre-calculated cooling rates obtained using the CLOUDY code (Ferland et al., 1998), tabulated for a range of different densities, temperatures and metallicities. They include all of the atomic processes outlined in § 1.3.3 along with the effects of a meta-galactic ionising UV/X-ray background which can ionise heavy elements and reduce the effective cooling rates by up to an order of magnitude for temperatures and densities typical of the shock heated ISM. As the cooling rate within a given particle or cell is obtained by simple interpolation from the values within the cooling table, the routines are highly efficient allowing them to be used for very large simulations. In addition, this also makes them very easy to implement into the *FLASH* code for which I confirmed the obtained and expected cooling rates for a series of densities, temperatures and metallicities.

7.3 Implementing Star Formation in FLASH

Although simple to implement in an SPH code, correct implementation of the star formation routines in the *FLASH* code is non-trivial due to fundamental differences in the two hydrodynamic schemes. SPH codes benefit from identical fixed particle masses for the gas and stars whereas the mass of gas in an AMR code's cell varies depending upon density

and resolution. As a consequence it is possible for very small, highly refined cells in dense star forming regions to have masses smaller than those of the star particles. If we formed the star particle then it would result in unphysical negative densities as mass is removed from the cell. This would lead to a loss of convergence in the Riemann solver. If however we choose not to produce stars when this happens, it will lead to an underproduction of stars in the densest and most highly refined regions of the simulation. This manifests itself as a strong mesh dependence as I demonstrate in § 7.3.2.

The creation of star particles presents another problem not present in SPH codes. As gas particles are converted directly into stars in *GADGET-2*, there is no need for material to pass off of the effective equation of state. However in *FLASH*, once mass has been removed from a cell to form stars, the cell will be below the density threshold and should be removed from the polytropic equation of state. As I show in § 7.3.3 due to the multitude of different adiabats that the gas can cool through, this process is poorly defined.

7.3.1 Stochastic Approach to Star Formation

The star formation prescription of Schaye and Dalla Vecchia (2008) works well in SPH as the masses of the star and gas particles are identical. The robust nature of SPH allows for the direct conversion of gas particles into star particles ensuring that we always obtain reasonable formation probabilities. AMR codes on the other hand have variable mass cells. As a given cell may be many times more massive than the star particles it forms, it is necessary for each cell to be able to spawn multiple star particles at each timestep. Schaye and Dalla Vecchia (2008) simply propose scaling the probability according to

$$Prob \rightarrow Prob' = Prob\ m_g/m_*. \tag{7.17}$$

In this case a full integer number of particles can be formed and the non-integer remainder is dealt with stochastically. An additional star particle will be formed if the remainder is greater than a number drawn at random from a uniform distribution of numbers between 0 and 1. This should lead to the correct number of stars being formed providing that the sampling statistics are adequate. In practice I find that it typically leads to RMS deviations between the final total expected and formed stellar masses, of order a few percent. I opt out of using variable mass star particles in order to constrain the number of very low mass particles that we produce, to ensure I use the same star particle masses as the SPH simulations of Schaye and Dalla Vecchia (2008) and to ensure that we do not encounter any two body heating effects brought on by small mass particles orbiting much

higher mass particles (see Steinmetz and White, 1997).

This scheme can be improved upon by instead drawing a random number from a Poisson distribution with mean equal to the probability given by equation (7.17). This integer random number then determines the number of star particles to be made. Averaging over time should therefore give the correct total stellar mass. For my fiducial simulations, this resulted in an RMS deviation between the final total stellar mass actually formed ($\sum m_*$) and that expected over the duration of the simulation $\left(\int^{t_{sim}} \dot{m}_* \, dt\right)$ of less than 0.49%, giving good agreement without the need for variable mass stars. For this reason I choose to adopt this technique for all of the simulations presented in this chapter.

Once formed, stars are allocated random positions within the cell and given the velocity of the gas on the mesh which is mapped to their positions using quadratic interpolants.

7.3.2 Countering Mesh Dependencies with Deficit Tracer Particles

The reduction of gas mass in a cell due to star formation presents the first hurdle that AMR codes must overcome. In SPH codes, the direct conversion of gas particles into star particles ensures that the ratio of the gas to star particle mass is of order unity and thus equation (7.17) will work well for all density regimes and at all levels of resolution. AMR codes on the other hand rely upon being able to spawn star particles of a given mass (in this case they are identical in mass to those in the SPH simulations for consistency) with the mass of the cell from which they were spawned being decreased by an equal amount. The mass of a cell in an AMR code is variable and depends upon both the density and the volume which is in turn dependent upon the refinement level. As the gas becomes denser, the Jeans length decreases forcing the AMR code to refine up. This inevitably leads to complications as cells reach high levels of star formation brought on by high densities but for which their small volumes provide insufficient mass to accommodate their star formation rates. As reducing the cell to a vacuum state or to unphysical negative densities results in the Riemann solver failing to converge, a minimum density floor, ρ_{floor}, must be imposed. I set $\rho_{floor} = 10^{-31}$ g cm^{-3} ($n_H \approx 10^{-8}$ cm^{-3}) which is equal to the lowest physical density in my simulation's gaseous halo (see § 7.6.2), yet still high enough to ensure that we do not incur Riemann solver problems.

Initial approaches which only formed stars in cells with sufficient mass, resulted in undesirable mesh effects coming into play as can be seen in the left hand panel of figure 7.2. Here the increase in density as we tend towards the centre of the disk galaxy (at x/y = 30 kpc) results in an increase in the resolution as we track the reduced Jeans length.

Therefore if we head away from the centre of the simulation, the density drops quickly and thus so does the mass at a given level of refinement. This results in cells becoming unable to form stars even though they are dense enough to do so. As we head further out from the centre of the simulation, the refinement level drops at $x = y = 22.5$ kpc and 37.5 kpc, leading the cell volume to increase by a factor of eight and allowing cells to resume star formation. This obviously results in strong mesh effects being visible and renders the implementation heavily resolution dependent.

In order to overcome this resolution dependence I developed the following approach. I permit all cells that have a density above the star forming density threshold to form stars. Where cells contain insufficient mass to form the star, the density is reduced to ρ_{floor} and the remaining mass required to form the star is tagged as a stellar mass deficit. This is placed on a special "mass deficit tracer" particle which is spawned by the cell at the position of the star. The mass deficit tracer particle is given the same velocity as the gas on the mesh and advects with the gas. By allowing the star to form whilst reducing the density (and thus pressure), material from neighbouring cells should flow into the cell both through gravitational attraction and diffusion. At each subsequent timestep, the density of the cell in which the mass deficit tracer particle resides is checked and any excess mass above ρ_{floor} is removed until the mass deficit is repaid in full. At this point the mass deficit tracer particle is removed.

This approach removes the need to wait for individual cells to become massive enough to form stars which otherwise leads to mesh dependencies. This does temporarily increasing the effective total mass in the simulation since the deficit particles are not included in the gravity calculation. This leaves slightly more gas free to form stars than should technically be available however the effect on the total mass in the simulation is found to be extremely small. Tracer particles are only formed in regions where the gas density has been dropped to ρ_{floor}, preventing further star formation and they each represent only a fraction of m_*. As the probability of forming a star is also linked to the cell mass (see equation 7.12), low mass cells are unlikely to produce large numbers of stars.

An alternative solution would be to remove gas evenly from neighbouring cells but this could lead to mass being removed from non-star forming regions, de-localising the consumption of gas. This scheme also presents numerical limitations. If we consider a region of dense but small cells, all of which are star forming, then the formation of stars in one cell could remove mass from the other neighbouring cells. This would then reduce their densities to below the star formation threshold. As we loop over cells in a block, stars

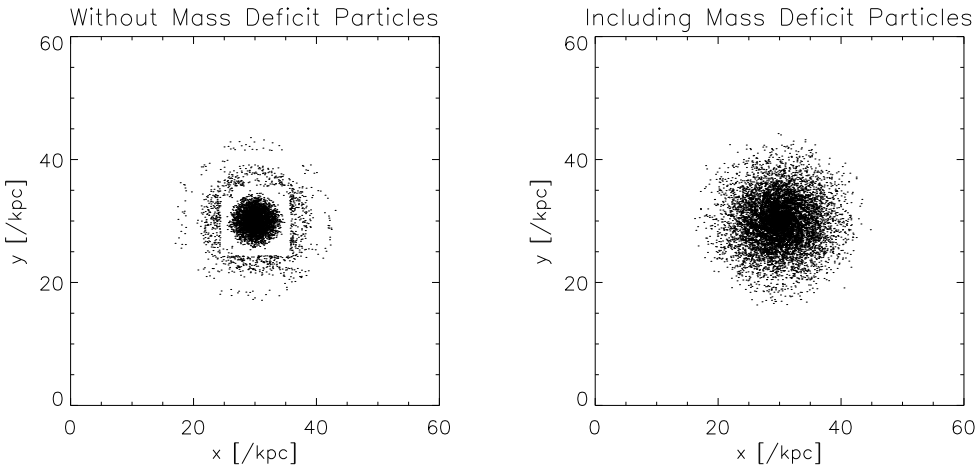

Figure 7.2: Plot of star particle positions of all newly formed stars since the start of the simulation for the face on disk galaxy in *FLASH*. The ability of mass deficit tracer particles to remove any mesh dependencies upon star formation is demonstrated. The simulation in the left hand panel prevents stars from being created in cells that have insufficient mass to form them without reducing the density below ρ_{floor} ($n_H \approx 10^{-8}$ cm^{-3}). The model in the right hand panel permits star formation in these cells and instead places the deficit on special tracer particles, to be recovered later. Clearly the left-hand panel shows strong mesh dependence with the refinement pattern being visible in the distribution of stars. The addition of deficit particles however removes this mesh dependency and a smooth distribution can be seen. (Stars present in the bulge and disk in the initial conditions are not plotted for clarity.)

would be preferentially formed in the first cells leading to further mesh dependencies. This effect could be made worse when considering cells in a dense region split across different blocks on different processors. The simultaneous formation of stars on different blocks and the reduction of mass in neighbouring cells could lead to mass being removed from guardcells. This would imply that the mass in a cell could be reduced due to local star formation and then again by star formation on a neighbouring block. Without a communication intensive algorithm to prevent this, which would again introduce mesh dependencies, it could lead to unphysical negative densities. The use of mass deficit tracer particles allows the removal of mass to arise in the regions where star formation is occurring without biasing neighbouring cell properties. As stars typically form in dense regions at the bottom of gravitational potential wells, mass will continue to flow into the cell to cancel out the deficit.

Analysis of simulations presented in § 7.7 indicated that the mean mass deficit accounted for only 0.007% of the mass of the gas in the disk with a peak deficit of 0.01%. This amounts to a peak mass in deficit tracer particles of 21.6 m_* spread throughout the disk. Mass deficit tracer particles typically only last for ~ 1.2 Myrs, which for the average timestep of just over $100,000$ years is equivalent to around only 12 timesteps. The odd exceptional deficit tracer particle remained for up to 8.4 Myrs (~ 84 timesteps) but was then promptly removed from the system. These time frames are very small compared to the simulated lifetime of the disk (400 Myrs) and that of the lifetime of stars before they eventually go supernova (30 Myrs). As such they are removed before they affect important galactic dynamics. With this in mind, I conclude that the use of mass deficit tracer particles to remove resolution dependence from my numerical scheme is justified and that any adverse effects are negligible.

7.3.3 The Polytropic Equation of State in FLASH

In *FLASH*, material can openly mix between cells allowing material on the effective equation of state to intermix with non-star forming material. This is different to the SPH formalism where star formation is a property bound locally to each particle and thus material is either on the effective equation of state or it is not. To track which material lies on the effective equation of state I introduce a mass scalar which advects with the fluid (see chapter 2.2). The mass scalar is normalised to lie between 0 and 1 and represents the fraction of gas in a cell that lies on the effective equation of state. When this fraction drops below a user defined tracer fluid threshold (0.5 by default) as a result of

advection of material into a less dense neighbouring cell, the material is considered to have passed off the effective equation of state. The only other way to leave the polytropic equation of state is for the density to drop below the star forming threshold, $\rho_{g,c}$, after a star is created. I have modified the *FLASH EOS* modules to use either the polytropic or standard ideal gas equations of state based upon whether the mass scalar is above or below the tracer fluid threshold. Material can pass onto the effective equation of state after cooling reduces the temperature in the *source terms module*. The relative state of material is also checked prior to forming stars in the *star formation module*. If material is not already on the effective equation of state but is both cool and dense enough, it is moved onto it. After stars are made (if any), the density is compared to the threshold density and if below it, the material is assigned to the standard equation of state. I prevent material from passing between the two different equations of state except during these two algorithms. This ensures that the equation of state does not alter during the three successive directional sweeps in the hydrodynamics algorithm. This allows for more consistent evolution of the hydrodynamics solver.

FLASH faces the problem of determining what energy and temperature gas in a cell should have when it leaves the effective equation of state. As can be seen in figure 7.3, material on the standard equation of state for an ideal gas (black points) can cool down onto the effective equation of state along several adiabats, each with different temperature (left panel) and pressure profiles (right panel) for a given density. In the left hand panel, material towards the left, at high temperature and low density represents material in the halo. As material cools and collapses into denser structures within the disk, the temperature range narrows and funnels down onto the effective equation of state which begins at densities greater than $\rho_{g,c} = 1.627 \times 10^{-25}$ g cm^{-3} ($n_H = 0.1$ cm^{-3}) and temperatures below $T_c = 10^4$K. Although the behaviour of the gas once on the effective equation of state is well defined (blue points), moving off the equation of state is less well defined as there is no ideal way to determine which adiabat the gas should follow. I therefore opt to update the pressure once material has left the effective equation of state by continuing to use the polytropic equation of state with the new density. By using equation (7.15) I can then calculate a new internal energy. As can be seen from the left hand panel of figure 7.3, this can lead to material which has just left the polytropic equation of state (green points) being cooler relative to some of the more energetic adiabats. It does however offer a much more robust and well defined solution. Since the cooling routines incorporate the Haardt and Madau (2001) model for the UV background radiation from galaxies and quasars,

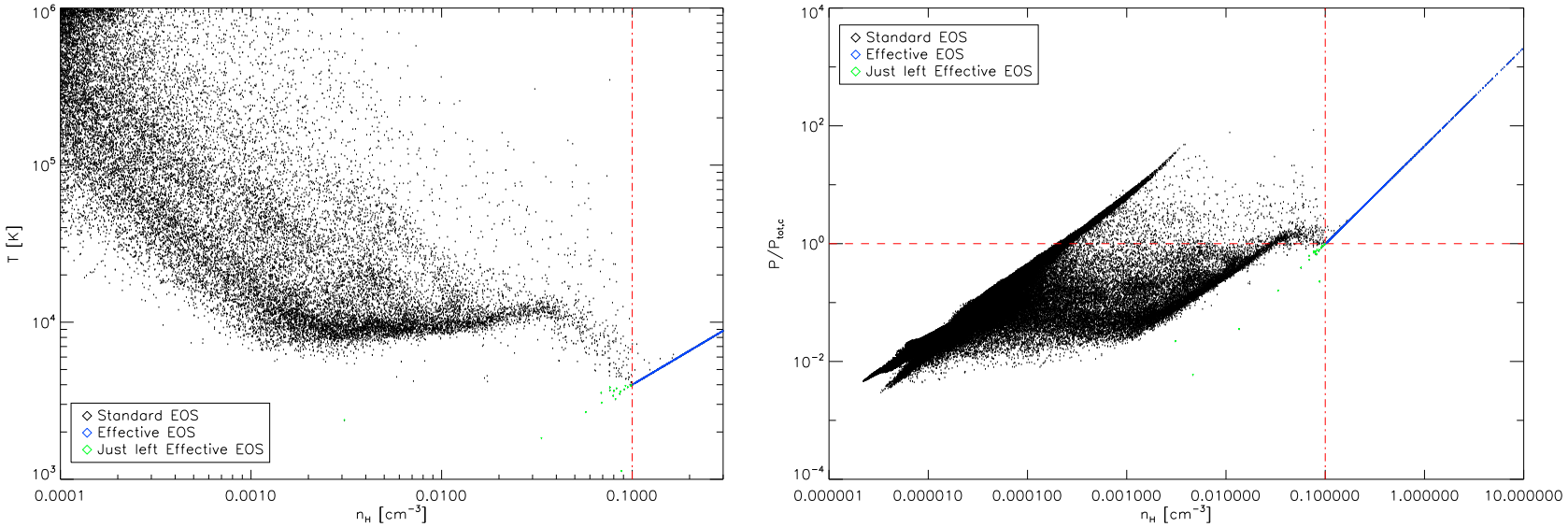

Figure 7.3: Temperature (left panel) and the pressure as a function of the critical pressure (right panel) are plotted against density for the default disk galaxy simulation. Black points represent material governed by the ideal gas law, blue - those on the polytropic equation of state and green points -those that have recently left the effective equation of state. The density threshold for star formation is plotted (red dot-dashed line) along with the pressure threshold (red dashed line). Although behaviour of the gas is well defined on the effective equation of state (marked in blue) at $n_H \geq 0.1$ cm^{-3}, material can cool onto the effective equation of state across a multitude of different adiabats with different temperatures and pressures (see black points). As can be seen from the left hand panel, both the temperature and density thresholds must be obeyed in order to pass onto the effective equation of state.

these cells will be quickly heated back in line with other cells at an equivalent density (shown in black).

7.4 Preliminary Testing of the FLASH Modules

To check the ability of my *FLASH* star formation implementation without the complexities in a real galaxy I initially tested my star formation prescription on a grid of cells each with a different volume density. This allowed the cells to span an equivalent surface density range from the threshold density to 30 $M_\odot\,pc^{-2}$ in a volume of 16 kpc^3 giving cell masses between $14,131$ and $890,000$ $M_\odot$. I evolve the simulation in the absence of any hydrodynamics, preventing gas from mixing, shocking and star particles from passing between cells. A test of the star formation schemes convergence with increasing mass resolution is shown in figure 7.4. The resulting star formation rate as a function of surface density is plotted for a suite of runs using different star particle masses covering six orders of magnitude.

In order to convert the gas volume density into a surface density I first project the gas density along the chosen observation sight line (i.e. along the z-axis). Once the density has been integrated I divide the density through by the Jeans length to get the projected surface density over a single scale length of the disk (see equation 7.6). The simulation is evolved for a single timestep (121,000 years) during which the number of star particles created is used to directly infer the star formation rate at a given surface density. The scheme shows good convergence providing that the star particle mass is kept below around $10^5 M_\odot$ at which point the number of stars created in a single timestep is too low for adequate sampling statistics to be gathered. With more massive particles the quality of the fit rapidly diminishes. Below this threshold, despite a small amount of scatter which reduces in line with increasing resolution, the data for all particle masses follow the input Kennicutt-Schmidt law very well with the fit improving as the particle mass is reduced as would be expected.

The quality of the fit improves dramatically when integrating over longer timescales which lead to better sampling statistics. This is apparent from figure 7.5 for which the simulation was evolved for ten timesteps using a star particle mass of $m_* = 10^5$ $M_\odot$. The range in surface densities was also increased to span $\Sigma_g = 1 - 100$. Although the density within the cells will decrease slightly with time, the short duration of the simulation presented here ensures that it is small. The surface density at which star formation

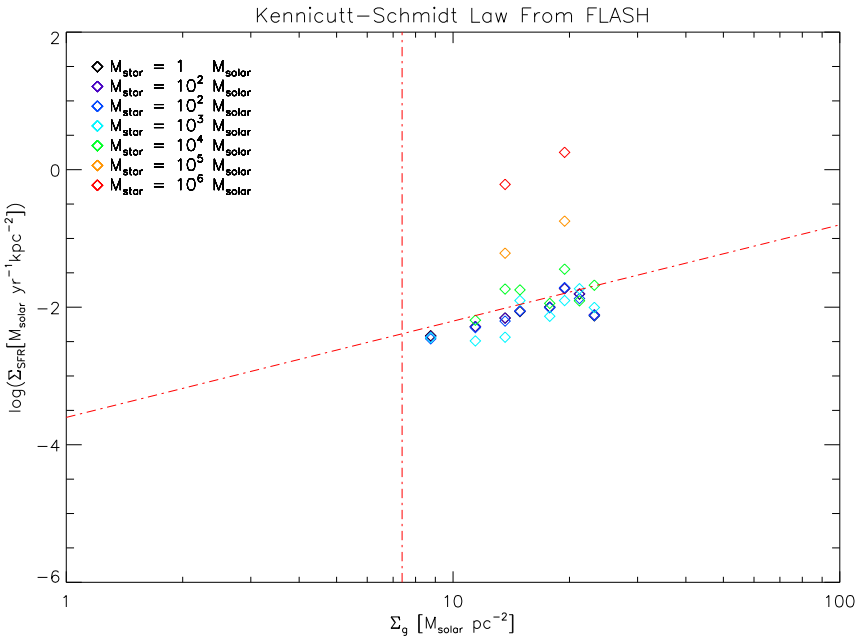

Figure 7.4: Surface star formation rate over a single timestep as a function of surface density using varying star particle masses. The quality with which the input Kennicutt-Schmidt law and surface density threshold (red dot-dashed lines) is reproduced is very good for star particle masses between 1 and $10^4 M_\odot$. Clearly the numerical stability of the scheme is good until a minimum star particle mass threshold is surpassed after which sampling statistics are reduced and the quality of the fit rapidly degrades. These deviations are however restricted to short term localised deviations and when averaging over time as is shown in figure 7.5, they return the correct Kennicutt-Schmidt star formation law.

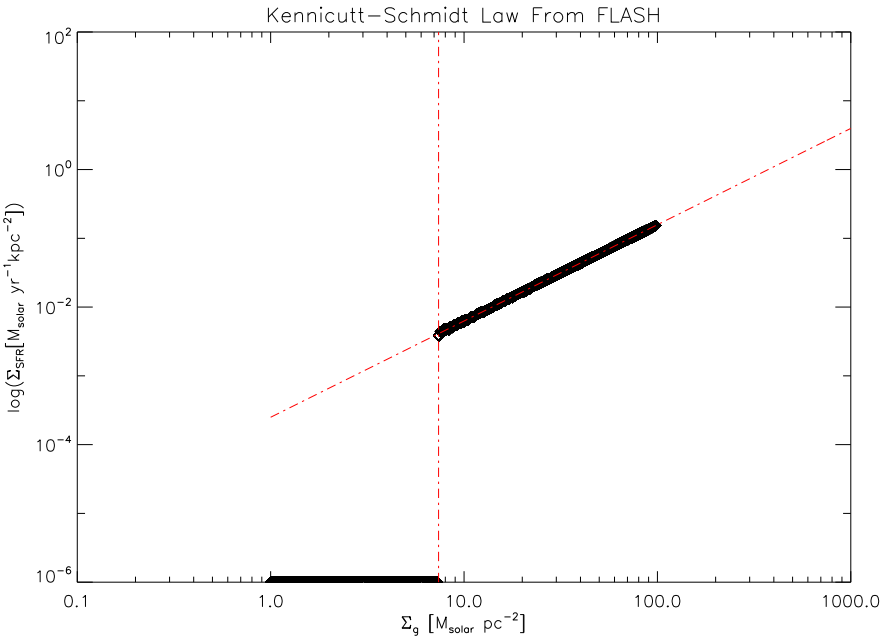

Figure 7.5: Surface star formation rate as a function of surface density. This characteristic Kennicutt-Schmidt plot exactly reproduces the input slope and threshold surface density at which star formation is observed to cut-off (red-dashed lines). The results come from a test grid of cells each with varying volume densities evolved in the absence of hydrodynamics to prevent gas and particle mixing.

truncates and the slope exactly replicate the input Kennicutt-Schmidt law. The lack of binning ensures that the threshold density gives a sharp density cut-off and removes any scatter from the plot. As I show later in § 7.7 the way in which results are binned can introduce some spread into the results and smear out the sharp density cut-off but the Kennicutt-Schmidt law is consistently reproduced very well. This tests confirms the performance of the star formation algorithm in a simple unambiguous way.

7.5 Disk Galaxy Initial Conditions in Gadget

Through a comprehensive suite of disk galaxy simulations Schaye and Dalla Vecchia (2008) are able to test the performance of their star formation algorithms in the SPH code *GADGET-2*. The same disk galaxies are also used later to test the supernova feedback recipes of Dalla Vecchia and Schaye (2008) (§ 8.7). The SPH initial conditions for the tests in both papers were generated using a code provided by Volker Springel with Schaye and Dalla Vecchia (2008) and Dalla Vecchia and Schaye (2008) referring readers to Springel

et al. (2005a) for a detailed description of the algorithm. In summary the galactic disk is composed an exponential gas and stellar disk along with a stellar bulge, all of which are embedded in a live dark matter halo.

The dark matter follows a Hernquist profile (Hernquist, 1990),

$$\rho_{DM}(r) = \frac{M_{DM}}{2\pi} \frac{a}{r\,(r+a)^3} \tag{7.18}$$

with a cumulative mass profile of $M(< r) = M_{DM} r^2/(r+a)^2$. This form of halo profile has the advantages of matching the NFW halo fitting profile (Navarro et al., 1997) in the inner regions ($r \ll r_{200}$) whilst the mass profile actually converges at a finite radius due to the sharper decline in the outer regions. In analogy with the NFW profile, they define a halo concentration $c = r_{200}/r_s$ where r_s is the scale radius of the equivalent NFW halo. The value of the constant a is then given by

$$a = r_s \sqrt{2\left[\ln(1+c) - c/(1+c)\right]} \tag{7.19}$$

In the models presented by Schaye and Dalla Vecchia (2008), they adopt a Hernquist profile normalised to give an NFW profile with concentration $c = 9$ and a total mass within the virial radius equal to the total mass of the Hernquist profile. For the $10^{12}\mathrm{M}_{\odot}$ halo used for the simulations in § 7.7 this gives an NFW virial radius of 163 h^{-1} kpc (using a value of $h = 0.73$). For the simulations presented in chapter 8.7 the same concentration ($c = 9$) is used so the $10^{10}\mathrm{M}_{\odot}$ and $10^{12}\mathrm{M}_{\odot}$ haloes have effective NFW virial radii of 35.1 h^{-1} kpc and 163 h^{-1} kpc respectively. The simulations take place inside a 20 Mpc^3 box.

The stellar bulge like the dark matter follows a Hernquist profile, contains 1.4% of the total mass of the system and has no net rotation. The disk, like the dark matter halo has a net spin with the halo possessing a dimensionless spin parameter $\lambda = 0.033$. Schaye and Dalla Vecchia (2008) adopts a disk with a radial scale length of 2.47 h^{-1} kpc, computed by relating it to the angular momentum following Mo et al. (1998); Springel et al. (2005a) where the circular velocity at the virial radius is 163 kms^{-1}. The stellar bulge has a scale length one tenth of that of the disk. The disk contains both four percent of the total angular momentum and four percent of the total mass of the system. The stellar disk has a vertical profile given by an isothermal sheet with a constant scale height ten percent of that of the radial disk (0.247 h^{-1} kpc). The vertical gas distribution is setup in hydrostatic equilibrium using an iterative technique.

The SPH simulations set the baryonic particle mass for the 10^{12} h^{-1} $M_{\odot}$ galaxy to

$$m_b = \frac{M_J(\rho_{g,c})}{N_{resol} N_{ngb}} \tag{7.20}$$

where $M_J(\rho_{g,c})$ is the Jeans mass at the star formation threshold for their fiducial model, N_{ngb} is the number of particles within the smoothing Kernel and N_{resol} is the factor by which the Jeans length exceeds the Kernel mass. The 10^{10} h^{-1} $M_{\odot}$ galaxy used in § 8.7 contains the same number of particles and are therefore one hundred times lighter. In the SPH simulations they fix the ratio of the Jeans length to the SPH smoothing length to be independent of density such that

$$\frac{h}{\lambda_J} = \left(\frac{1}{8 N_{resol}} \right)^{1/3} \tag{7.21}$$

They choose to adopt $N_{resol} = 6$ and therefore they resolve the Jeans length by at least three smoothing lengths. For $N_{ngb} = 48$ this gives a baryonic particle mass of 5.1×10^4 h^{-1} $M_{\odot}$.

7.6 Disk Galaxy Initial Conditions in FLASH

To present as fair a comparison as possible in *FLASH*, I use the exact same initial conditions as in the *GADGET-2* runs, mapping them to the mesh with software kindly provided by Milan Raicevic (Durham University). The technique involves an initial loop over all gas particles, comparing their smoothing lengths to the size of the cells in the block to which they currently reside. If the smoothing length is smaller than the cell size, then the block is refined up and the new cell size compared. This process is repeated until the cell size is comparable to the local SPH smoothing lengths of the gas particles or until the maximum specified level of refinement is reached. At this point, the particle mass is mapped to the mesh using a Gaussian which closely approximates the SPH smoothing Kernel (Monaghan, 1992). The Gaussian is truncated at 2 σ and normalised so that the enclosed area beneath it equals unity. The cell's density on the mesh is then found by dividing through the mass that has been mapped to it, by its volume. All other properties are then mapped to the cells on the mesh weighting them according to the mean particle density in a given cell. All of my simulations are run from initial conditions mapped to a mesh with a peak refinement level of eight, giving an effective 2048^3 mesh, this is then derefined at start up to the desired maximum refinement level.

As *FLASH* does not actively adapt in the same manor as *GADGET-2* I implement a separate refinement criterion as outlined in chapter 2.6. This refines any region in which

the cell size is larger than a user defined fraction of the local Jeans length. Suites of numerical tests run in chapter 4 have shown that comparable results can be obtained between SPH and AMR codes for a range of simulated phenomena, when the smoothing length is equal in size to the minimum cell size (see chapter 4). With this in mind I adopt the same ratio of 0.257 between the Jeans length and the minimum cell size, as between the Jeans and smoothing lengths in the *GADGET-2* simulations. In regions where derefinement will not drop the resolution below 0.257 Jeans lengths per cell, I allow the blocks to derefine providing that it is not in conflict with any other refinement criteria.

Each block within the *FLASH* code contains 16^3 cells and I enforce a minimum level of refinement of three giving an effective root grid of 64^3 cells. By default the simulations are performed in a $60\,\mathrm{kpc}^3$ box which allows us to fully resolve the disk at comparable spatial scales to those in the SPH simulations whilst requiring modest resources to run. This box size is considerably smaller than the volume used in the SPH simulations since it would require around 16 levels of refinement to resolve the same spatial resolution. Given the limits to the FFTW size and the memory available to perform such tests, this was impractical and would have run extremely slow with current algorithms. The reduced box size instead allows me to use more modest levels of refinement with my simulations ranging between four to eight levels. As I cannot resolve the full live dark matter halo I chose to adopt an analytic halo which I justify in § 7.6.1. As the absence of a gaseous halo would also lead to Riemann solver failures, I have to implement a very tenuous gaseous halo. I chose to implement a halo which is initially in hydrostatic equilibrium with the dark matter halo and for which the total halo gas mass is extremely small, ensuring negligible accretion onto the disk.

As I am using a much smaller box than in the SPH simulations it is possible that boundary effects may become significant. If using periodic boundary conditions, any expansion of the disk could drive pressure waves through the disk and halo gas which upon reaching the boundary would be allowed to drive back onto the opposite side of the disk. To prevent interactions such as this and mimic the isolated conditions of the SPH disk galaxy, I adopt outflow boundary conditions. These permit material to leave the simulation volume and thus relieve these pressure forces. Once material has left the volume, information pertaining to it is lost and it may not re-enter the simulation. To confirm that the boundary conditions and the box size make no difference to the results obtained in § 7.7 I performed identical simulations with the default galaxy using periodic and outflow boundary conditions and found that the star formation rates and morphology

were unaffected. The same is true if the box size is doubled from 60 to 120 kpc^3, ensuring that interactions within the disk are not influenced by the limited box size.

The Fast Fourier Transform gravity solver I use requires periodic boundary conditions for the gravity in order to operate effectively. Although it is possible to perform FFTWs with isolated boundary conditions, this requires substantially more memory and would present further impediments to the maximum box size that I can resolve.

I run all of the simulations in *FLASH*, unless stated otherwise with identical input parameters to their SPH counterparts. These include a threshold star formation surface density of $7.3\,\mathrm{M}_\odot\,\mathrm{pc}^{-2}$ equivalent to a volume density threshold of $n_H = 0.1$ cm^{-3}, the observed Kennicutt-Schmidt law (equation 7.2), a gas fraction of $f_g = 0.3$ (equal to the initial gas fraction of the disk) and a polytropic equation of state normalised to $T/\mu = 10^4 K/0.59$ at the threshold.

7.6.1 Analytic Dark Matter Potential

The SPH simulations benefit greatly from the ability to run gas and dark matter in total isolation inside of an arbitrarily large box. The run time for the gravitational tree algorithm is independent of the particle separation and is instead only dependent upon the total number of particles. This allows them to be spread out over a very large spatial range; from the compact disk to the sparse edges of the live dark matter halo. This ability to model particle dynamics over such a wide range of scales and maintain performance that scales well with the number of particles has allowed SPH codes to perform the largest cosmological simulations to date (e.g. Springel et al., 2008; Kuhlen et al., 2008). This does however present challenges when trying to mimic these runs in an AMR code. In *FLASH* we must enclose the entire particle setup inside of the simulated volume and then refine up to the desired level at the centre. As this requires starting from a limited root grid, we would need to span distances ranging from the 20 Mpc box down to mere tens of parsecs to resolve the Jeans length within the disk. The fifteen levels of refinement needed to jump six orders of magnitude and accurately model the dark matter halo and disk in a self-consistent fashion would clearly be unfeasible. This is especially true if we wish to perform a suite of runs. There is also the question of how well would we resolve gravitational forces acting on dark matter particles in areas of low resolution if we restrict refinement to around the galactic disk? If we do not restrict refinement to the disk, then the resources required would be impractical. Hence in order to overcome these problems I opt for the use of an analytic dark matter potential, allowing me to drastically reduce

down to kiloparsec scales the size of the volume which I simulate.

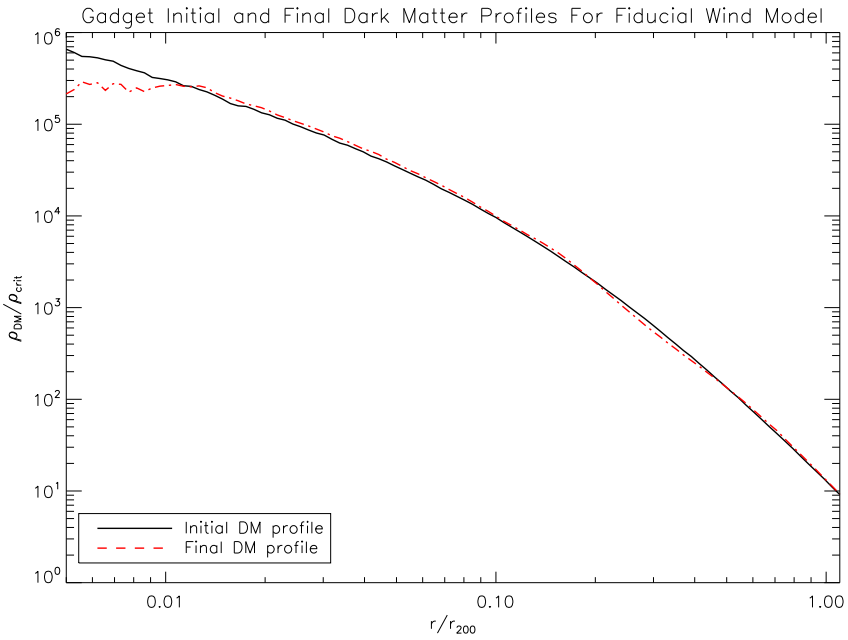

Figure 7.6: Comparison of the initial and final dark matter profiles in the *GADGET-2* fiducial wind simulation. The final configuration clearly shows excellent agreement with the initial profile with deviations only occurring below $\sim 1\%$ of the virial radius. This clearly demonstrates that the dark matter profile can be well approximated by an analytic dark matter halo, removing the need for an extremely large simulation box to enclose the dark matter and thus allowing for a much more focused higher resolution simulation of the disk itself.

Figure 7.6 clearly shows that in the fiducial *GADGET-2* simulation including both winds and star formation, the final dark matter profile is only perturbed from the initial configuration within the central one percent of the virial radius and shows excellent agreement at all greater radii. The displacement of dark matter within the central parsec of the galactic halo is an effect recently observed by Governato et al. (2010) in which supernova feedback is able to selectively remove gas with the lowest angular momentum. Observations of the velocity curves of dwarf galaxies have shown that they feature shallower dark matter density profiles than those obtained in dark matter only simulations (de Blok et al., 2008; Simon et al., 2005; Swaters et al., 2003). Governato et al. (2010) shows that bulk outflow of material through supernova winds and the orbital energy loss of gas clouds through dynamical friction allows for the transfer of energy to the dark matter component, a process which can cause the central dark matter profile to expand.

The dark matter profile will thus be perturbed less in the absence of feedback. This result is found to hold for all of the *GADGET-2* simulations in Schaye and Dalla Vecchia (2008) and Dalla Vecchia and Schaye (2008). I therefore conclude that adopting an analytic Hernquist dark matter profile is an ideal alternative to a more computationally demanding live halo. This is especially true as it would not be guaranteed to be any more accurate relative to the high gravitational force resolution present in *GADGET-2*. This is further reflected in the similarities between the *GADGET-2* and *FLASH* simulations shown later in § 7.7. I therefore adopt the same halo parameters as the SPH simulations but choose to maintain a constant analytic dark matter potential. The gas and stellar components however are self-gravitating and continue to contribute to the gravitational dynamics within the simulated volume.

7.6.2 Gaseous Halo in FLASH

Although the original SPH simulations were performed in isolation with no gaseous halo it is necessary for me to implement a gaseous halo around the galactic disk in the AMR simulations. As previously mentioned, AMR codes are unable to model vacuum conditions as inappropriate interpolations in the PPM scheme cause the Riemann solver to fail to converge (see § 2.2). As the purpose of these simulations is to reproduce those performed in *GADGET-2* as closely as possible, it is necessary to keep the mass of the halo to a bare minimum in order to ensure that the accretion of halo gas onto the disk is minimal. The drop in density and pressure between the boundary of the disk and start of the halo need to be as large as possible without causing convergence problems for the Riemann solver.

Given that the Hernquist profile matches the NFW fitting formula at radii less than r_{200}, I adopt the complimentary isothermal gas profile as used by Wu et al. (2000).

$$\frac{\rho_{gas}}{\rho_0} = \left(1 + \frac{r}{r_s}\right)^{\eta/(r/r_s)} \tag{7.22}$$

where $\eta = \frac{\mu m_H \alpha}{k_B T}$ is a dimensionless parameter which defines the slope of the density profile and the parameter ρ_0 was chosen as in Wu et al. (2000) to satisfy the relation

$$\frac{M_{gas}^{200}}{M_{DM}^{200}} = \frac{\Omega_b}{\Omega - \Omega_b} \tag{7.23}$$

M_{DM}^{200} denotes the mass of a sphere for which the mean enclosed dark matter mass density is 200 times the critical density. At early times this will prevent accretion of halo gas onto the disk though cooling will undoubtedly tend to result in low levels of accretion at late times.

For a box size of 60 kpc^3 this results in the total mass of the gaseous halo being equal to only 0.0003% of the mass of the gas in the disk thereby ensuring that any accretion onto the disk is negligible. Consequently the effective star formation rates in the AMR and SPH simulations should be unaffected by accretion of halo gas onto the disk. Although this means that the disk galaxy is not very realistic, my emphasis lies upon comparison of the differences in the hydrodynamics between the two numerical schemes.

To ensure that material within the disk is not pressure confined and that any material ejected from the disk due to supernova winds is not subject to excess ram pressure from the halo (relevant for chapter 8), I configure it such that the pressure drops by at least four orders of magnitude and the density by six orders of magnitude at the boundary between the halo and disk gas. Thus any material forced out of the disk should feel no resistance and be free to leave the galaxy as in the SPH simulations. I have confirmed this by increasing the halo density by two orders of magnitude and no discernible differences where observed in the global star formation rates or the galaxy structure for both the simulations in § 7.7.

As the net cooling rate is very low at low densities and temperatures (see chapter 1.3.3) I prevent rapid cooling of the halo by choosing a temperature of 64eV ($\approx 7.4 \times 10^5$ K) which yields a value for $\eta = 17.58$. I fix $\rho_0 = 3 \times 10^{-38}$ g cm^{-3} ($n_H = 2 \times 10^{-14}$ cm^{-3}).

7.7 Comparison of SPH and AMR Star Formation

I initially run with an identical setup to the SPH simulations in order to ascertain whether there is any underlying differences between the two hydrodynamic implementations. The fiducial simulations of Schaye and Dalla Vecchia (2008) adopt a polytropic index of 4/3 which I therefore use in my first suite of runs. The gas fraction is assumed to be one third and the polytropic equation of state is normalised to $T/\mu = 10^4 K/0.59$ at the threshold. Both suites of simulations use a Kennicutt-Schmidt normalisation of $A = 2.5 \times 10^{-4}$ M$_\odot$ yr^{-1} kpc^{-2}, a power-law exponent, $n = 1.4$ and a surface density threshold of 7.3 M$_\odot$ pc^{-2} equivalent to a volume density threshold of $n_H = 0.1$ cm^{-3}. I run the *FLASH* simulations for 400 Myrs.

Initially at low levels of resolution (lref 5 and 6) the *FLASH* runs appear to give reasonable star formation rates - comparable to the fiducial *GADGET-2* run. However when performed at increasing levels of refinement, the *FLASH* simulations display a complete lack of convergence. This is clearly visible in figure 7.7 where the global star formation

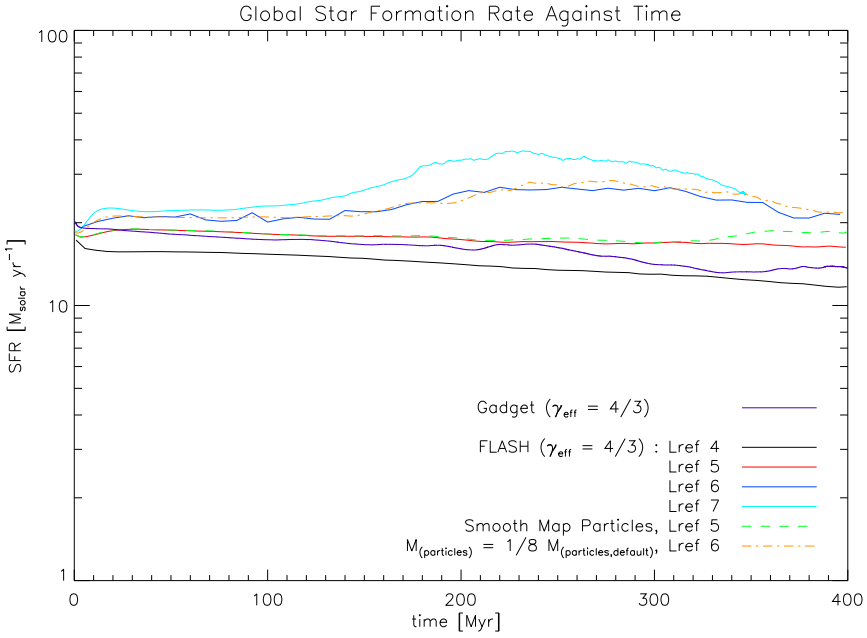

Figure 7.7: Global star formation as a function of time for the 10^{12} h^{-1} $M_{\odot}$ galaxy with $\gamma_{eff} = 4/3$. The different lines correspond to different levels of peak resolution with levels 'lref 4' (black), 'lref 5' (red), 'lref 6' (blue), 'lref 7' (green) corresponding to effective mesh sizes of 128^3, 256^3, 512^3 and 1024^3 cells respectively giving peak spatial resolutions of 469, 234, 117 and 59 parsecs. The purple line corresponds to the equivalent SPH run. The *FLASH* runs clearly fail to converge upon a final solution, with increasing refinement driving the peak star formation up - particularly at late times between 200 and 300 Myrs. The dot-dashed orange line represents the same peak resolution as the Lref 6 run in blue but uses star particles with 1/8 the default particle mass. The orange and blue lines are in good agreement showing that lack of convergence is not related to particle mass. The dashed green line is equivalent to the lref 5 run (red) but uses the `SmoothMapParticles` routine to smooth the particle distribution when mapping it to the grid. Both lines show strong agreement indicating the mapping of discrete particle properties to the mesh is not responsible for the over-dense knots shown in figure 7.8.

rate is plotted as a function of time. The black, red, blue and green lines correspond to increasing levels of refinement with effective peak grid resolutions of 256^3 (lref 4), 512^3 (lref 5), 1024^3 (lref 6) and 2048^3 (lref 7) respectively. These give spatial scales of 468 pc, 234 pc, 117 pc and 58 pc with the smallest smoothing length in the fiducial *GADGET-2* simulation being 44 pc. As we increase the resolution, the star formation rate after around 150 Myrs begins to increase, continually diverging from the lower resolution runs. I terminated the $lref = 7$ simulation after 340 Myrs once it was clear that the simulation continued to show no sign of convergence. At this point it had taken 120.3 hours across 32 processors, whilst at the refinement level below it ran to completion in only 31.6 hours on 24 processors. Running simulations at and above this level of refinement has proven to be prohibitively expensive. Fast gravitational calculation requires that the FFT is at, or as close to the peak level of refinement as possible in order to minimise iterations on the mesh. However the size of the FFT is limited by the memory available and a compromise has to be made between resources used and runtime taken. The FFT performed across a uniform mesh at the refinement level of eight requires 192 Gigabytes of memory alone without taking into account the rest of the *FLASH* architecture. Consequently simulations run at and above this refinement level must prolong and iterate the obtained potential which is computationally expensive (see chapter 3).

The cause of this lack of convergence can be best seen in the comparative images of the face on projected gas density shown in figure 7.8. The *GADGET-2* simulations (top) clearly show the development of smooth spiral arms that gradually build up as the simulation progresses whereas in *FLASH* the spiral structure is disturbed by the formation of extremely dense knots which disrupt the spiral arms. These knots are found to form when the precessing spiral arms come into contact with each other, the material sticking and collapsing into a dense knot. As the simulation evolves, these dense knots collide, merge and tear the fragile spiral arms apart.

One possible cause for these dense knots could arise in the mapping of discrete particle masses to the AMR mesh. If the particle resolution is too low relative to the grid resolution, then the mapping of particles to a single cell could result in a particle mass distribution that appears to be lumpy and overly concentrated in a few cells as opposed to the smooth stellar mass distribution which it should represent. This could result in overly dense gas deposits concentrating around strong star forming regions.

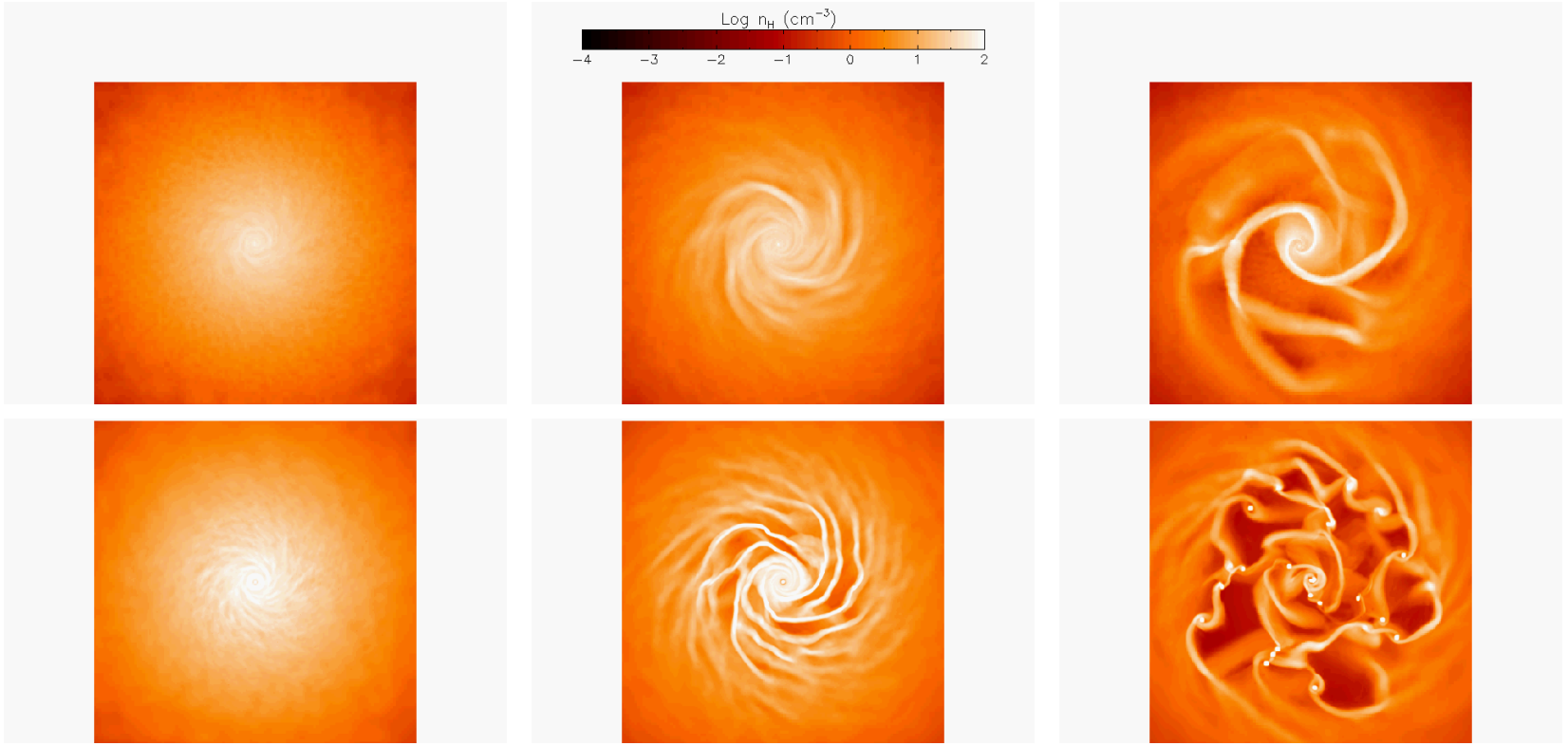

Figure 7.8: Logarithmic face on projected gas density for the fiducial model disk galaxy in *GADGET-2* (top) and *FLASH* (bottom). The images show the galaxy at times 25, 100 and 250 Myrs heading from left to right. Both galaxies were run using a polytropic index of $\gamma_{eff} = 4/3$. Although the galaxies run in the two codes show some initial similarities up to 100 Myrs with similar spiral structure, the *FLASH* galaxy quickly develops very dense knots at later times. These form when the precessing spiral arms make contact with each other, the material sticking to form dense knots which disturb the spiral arms. These extremely dense knots raise the global star formation significantly and become denser as resolution is increased, preventing convergence. *GADGET-2* does not form dense knots when spiral arms come into contact.

In order to check this I first run the same simulations with the `SmoothMapParticlesToMesh` routine written by Paul Ricker (see chapter 2.6). By restricting the stellar density once mapped to the mesh, down the refinement tree until no cell in the block lacks a stellar contribution, the star particle mass is smoothed out across coarser blocks. The results are then prolonged back up the refinement tree to produce a smooth stellar density distribution. The mapped density is representative of the discrete particle distribution but lacks any empty cells or spuriously sharp contrasts between neighbouring cells. This was tested at a series of levels of refinement and the result for lref = 5 is shown plotted (dashed green line) in figure 7.7 against the same simulation run without it (red). Clearly there is very little deviation, a result which holds at the other levels of refinement (not plotted for clarity). An additional check of the particle resolution was performed by reducing the star particle mass by a factor of eight. The result is plotted as the dot-dashed orange line in figure 7.7 against the equivalent run in blue which uses the default particle mass and is run at the same level of resolution. Clearly both profiles trace each other extremely well, a conclusion which holds for the other levels of resolution, indicating that discrete particle resolution effects are not to blame.

In order to ensure that the star formation model is correctly implemented and reproducing the observed input Kennicutt-Schmidt law, the surface star formation rate as a function of surface density is plotted in figure 7.9. The plot is generated in the same manor as those presented in Schaye and Dalla Vecchia (2008) by projecting the gas density and star formation rate along the minor axis and taking a series of radial annuli containing a fixed gas mass equal to $2000m_*$. The integrated surface densities are given per unit Jeans length assuming that the disk scale height fluctuates on order of the Jeans scale (see equation 7.6). The different points correspond to different resolution runs; lref 4 (black), lref 5 (red), lref 6 (green) and lref 7 (blue). The runs are very well converged and show good agreement with the input Kennicutt-Schmidt law and density threshold at which star formation truncates at all resolutions (red dashed lines).

The minor divergence from the ideal slope at very high densities and the smearing out of the otherwise sharp star formation surface density cut-off, results from binning effects and improves as the size of the annuli are decreased. As the $\gamma_{eff} = 4/3$ *FLASH* simulations clearly show very dense knots in the centre of large swept up voids, binning these regions smears out the star formation rate and density. Evidently the star formation recipe works well at all levels of resolution and shows very little change in the quality of the fit as resolution is decreased.

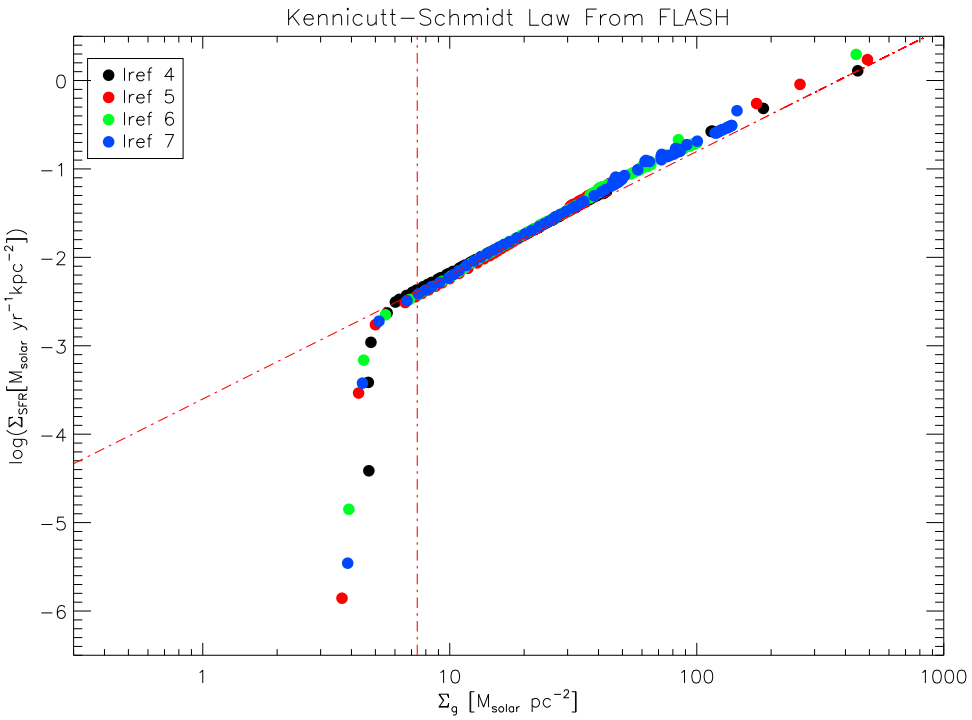

Figure 7.9: Kennicutt-Schmidt plot of the surface star formation rate as a function of surface gas density for the 10^{12} h^{-1} $M_{\odot}$ galaxy with a polytropic index of 4/3. The different points correspond to different resolution runs; lref 4 (black), lref 5 (red), lref 6 (green) and lref 7(blue). The runs are converged and show good agreement with the input Kennicutt-Schmidt law and surface density threshold at which star formation cuts off. Clearly the star formation recipe works well at all levels of resolution.

A comparison in figure 7.10 of the Kennicutt-Schmidt plots obtained from *FLASH* (left hand panel) with those from the default *GADGET-2* run (right hand panel) shows strong agreement between the two codes. Both show a slight smearing of the density threshold and a minor elevation of the star formation rate at high density due to the effects of binning but the fundamental properties of the input star formation law are obeyed. Although this plot fails to highlight the origin of the difference between the two codes, it does indicate that the star formation recipes are identical and that the star formation rates are correct for the gas densities present in the simulations. Clearly from this the difference between the two codes must arise due to the polytropic equation of state which I will now investigate.

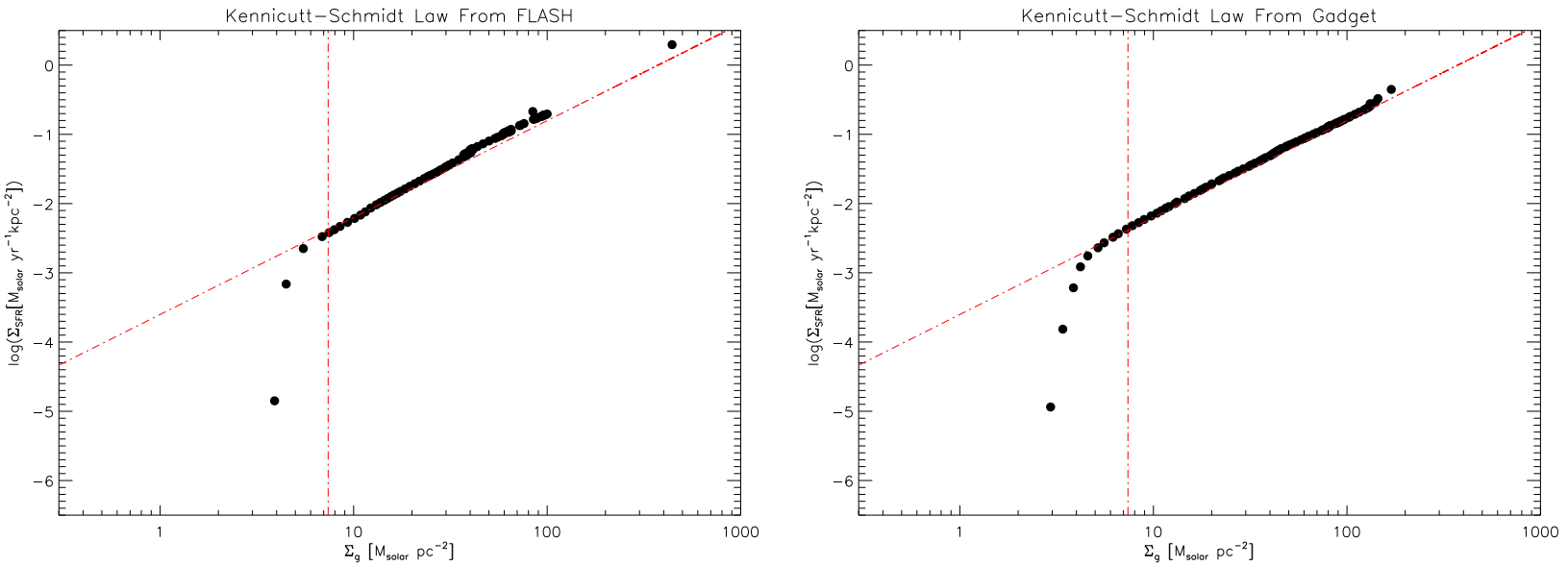

Figure 7.10: Kennicutt-Schmidt plots of the surface star formation rate per unit area against gas surface density at $t = 100\,Myr$. Both *FLASH* (right hand panel) and *GADGET-2* (left hand panel) produce equally excellent fits to the input Kennicutt-Schmidt laws with only minor smearing of the threshold density at which star formation ceases due to binning effects.

7.7.1 Effect of Increasing the Polytropic Index in FLASH

In order to confirm whether the polytropic equation of state is the origin of the differences between the SPH and AMR codes I implement a higher polytropic index in *FLASH*. The higher polytropic index of 5/3 results in greater pressure at a given density when on the effective equation of state. This gives more pressure support against the collapse of gas, raising the Jeans length and mass which scale with the polytropic index (see equations 7.16). Whilst a polytropic index of 4/3 leads to a Jeans length that scales as $\lambda_J \propto \rho^{-1/3}$ and a Jeans mass that is independent of density (see equations 7.16), increasing its value to 5/3 leads to a Jeans length that decreases slower with increasing density ($\lambda_J \propto f_g \rho^{-1/6}$) and a Jeans mass that now increases with increasing density ($M_J \propto f_g \rho^{1/2}$). This helps stabilise the accretion of mass onto dense regions which could otherwise runaway and form the extremely dense observed knots. Now as dense regions form, the Jeans length increases to encompass more of the over-density, allowing density fluctuations to be countered by pressure waves that travel across the region. This smoothes out the growth of perturbations before they can collapse to form extremely dense knots.

The global star formation rate as a function of time in *FLASH* using the higher polytropic index of 5/3 instead of 4/3 is shown in figure 7.11. The different lines correspond to different peak levels of resolution with *FLASH* runs at refinement levels of four (black), five (red), six (blue), seven (green) and eight (cyan) being plotted against the default *GADGET-2* simulation (purple dot-dashed line). The highest resolution *FLASH* run has a peak spatial resolution of 29 parsecs, in slight excess of the ~ 31 pc peak smoothing length in the *GADGET-2* simulation. Unlike for the lower polytropic index, the *FLASH* results now quickly converge at very modest resolution (lref5 and above). The star formation rate is approximately 15% higher in *FLASH* but it evolves identically to that in *GADGET-2*, decreasing smoothly with time as gas is slowly consumed. Despite different pressure laws being used, the results show surprising agreement. Indeed it could be argued that we would expect *FLASH* to give a slightly higher star formation rate as the density contrasts obtained within an AMR code will be higher than in an SPH code where the density field is smoothed.

The agreement is evident in the face-on projected gas density shown in figure 7.12. The results for *GADGET-2* (top) and *FLASH* (bottom) are plotted at times 25, 100 and 250 Myrs from left to right. Remarkable agreement is visible between both *FLASH* and

GADGET-2 with almost identical spiral structure being present in both codes at all times. The *FLASH* spiral arms are slightly denser than their SPH equivalents and additional spiral structure is resolved towards the outer edges of the disk whereas it is smoothed over in the SPH simulations - this may be the source of the increased star formation rate in *FLASH*. Clearly this indicates exceptional agreement between the two codes with well converged star formation rates and almost identical structure but for two distinctly different polytropic indices.

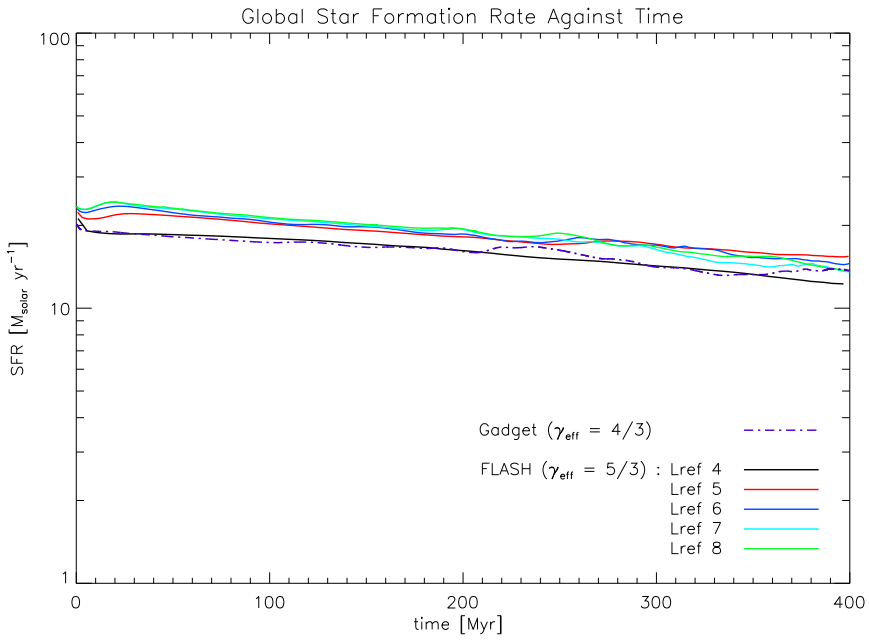

Figure 7.11: Global star formation as a function of time for the 10^{12} h^{-1} $M_{\odot}$ galaxy with $\gamma_{eff} = 5/3$. The different lines correspond to different levels of peak resolution with levels 'lref 4' (black), 'lref 5' (red), 'lref 6' (blue), 'lref 7' (green) and 'Lref 8' (cyan) corresponding to effective mesh sizes of 128^3, 256^3, 512^3, 1024^3 and 2048^3 cells respectively giving effective peak spatial resolutions of 469, 234, 117, 59 and 29 parsecs. This plot demonstrates that the lack of convergence seen in the $\gamma_{eff} = 4/3$ case (figure 7.7) is unique to that polytropic index with the $\gamma_{eff} = 5/3$ case quickly converging upon a final solution by the relatively modest 'lref 5.' The *GADGET-2* $gamma_{eff} = 4/3$ case is shown for comparison (purple dot-dashed line) and although the two runs use different polytropic indices, they yield very similar star formation rates with *FLASH* only showing a slightly higher level of star formation.

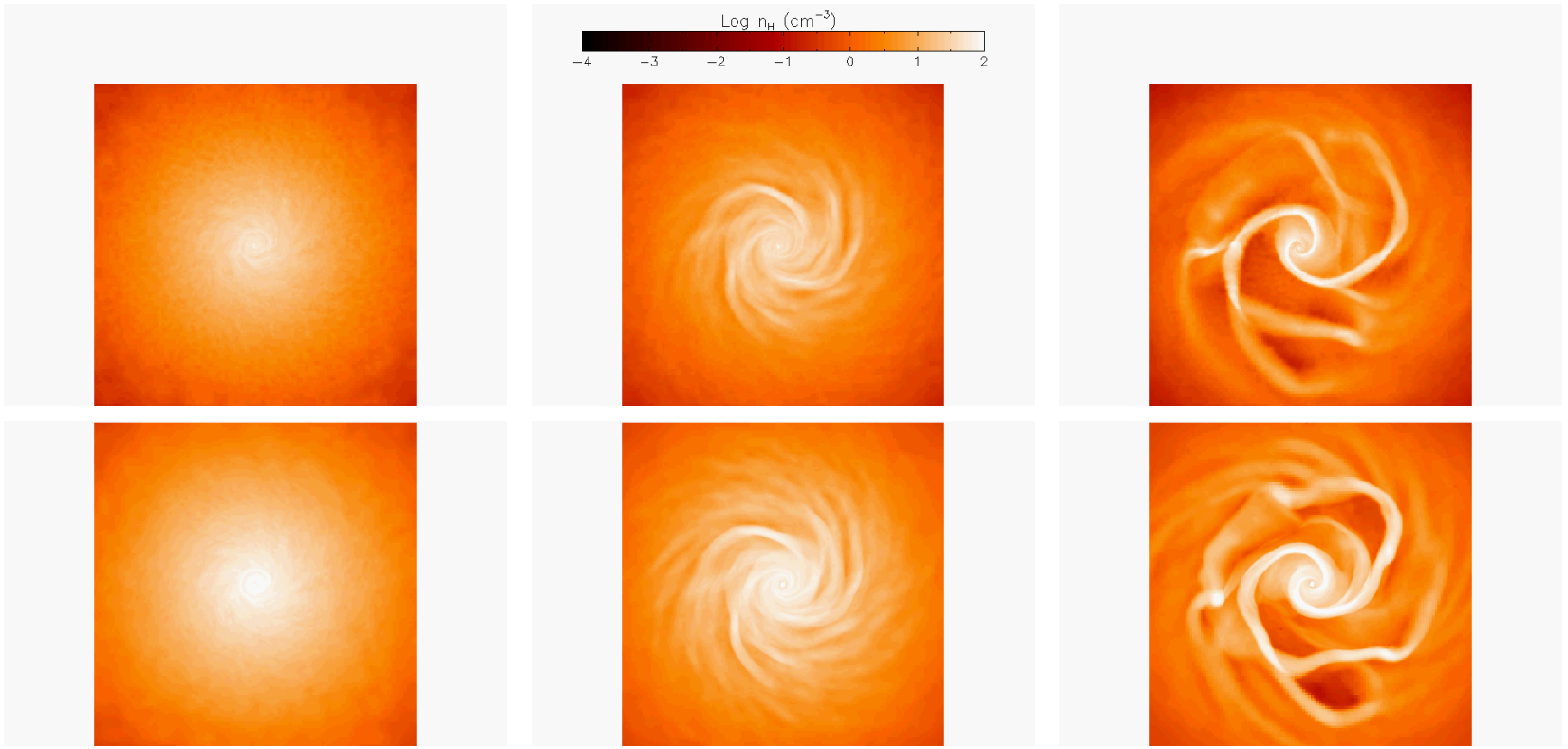

Figure 7.12: Logarithmic face on projected gas density images for the fiducial model disk galaxy in *GADGET-2* (top) and *FLASH* (bottom). The images show the galaxy at times 25, 100 and 250 Myrs heading from left to right. The galaxies run in the two codes show almost identical spiral structure with the *FLASH* simulations resolving slightly more structure in the outer regions of the disk which is smoothed over and suppressed in *GADGET-2*. Despite identical disks being formed, the polytropic indices are different with *GADGET-2* using $\gamma_{eff} = 4/3$ whilst *FLASH* adopts the higher value of $\gamma_{eff} = 5/3$ to provide greater pressure support against gravitational collapse, preventing the formation of extremely dense knots seen in figure 7.8.

Ideally increasing the resolution should allow us to converge upon a final solution, beyond which any further increases in refinement should improve the factor by which we resolve the Jeans length. Figure 7.13 highlights this by showing the ratio of the minimum Jeans length in the densest cells to their cell size, plotted against gas density. The coloured points represent the outcome of the same simulation run at different levels of refinement with the following peak spatial resolutions; black - 468 pc (Lref 4), red - 234 pc (lref 5), blue - 117 pc (lref 6), green 58 pc (lref 7) and cyan 29 pc (lref 8). The dashed line connects points from the simulations run with the lower polytropic index, $\gamma_{eff} = 4/3$, whilst the solid line connects simulations with the higher polytropic index, $\gamma_{eff} = 5/3$.

For simulations using the higher polytropic index, as resolution increases the maximum density quickly tails off and seems to converge upon a maximum density with the increasing Jeans mass helping the system to stabilise. Increasing the resolution therefore gives an increase in the ratio by which the Jeans length is resolved as opposed to further collapse.

Simulations using the lower polytropic index display little sign of convergence with increases in resolution merely extending the peak density that cells reach. As the star formation rate $\dot{m}_* \propto \rho^{1.4}$, an increase in density leads to a more rapid increase in the star formation rate. This produces the lack of convergence in the global star formation rate seen in figure 7.7. Successively increasing the resolution by a factor of two increases the peak density on average by around an order of magnitude, whilst the Jeans resolution only improves by $\sim 50\%$. As the resolution increases, so too does the density, reducing the Jeans length and accelerating further collapse. This runaway collapse model shows poor convergence with greater refinement not guaranteeing better Jeans length resolution or a converged star formation rate.

I note that for both equations of state the Jeans length in the highest resolution simulations is resolved by at least the minimum three cells that is imposed in the SPH simulations of Schaye and Dalla Vecchia (2008). This however does not guarantee that an increase in resolution will not lead to new structure forming as is seen in the case of the lower polytropic index.

As the higher polytropic index runs converge well before reaching a comparable spatial resolution to that in *GADGET-2*, it can be concluded that future simulations need not necessarily use comparable spatial resolutions to obtain the same results. Ultimately, in *FLASH* the Jeans mass is stable for the higher polytropic index of 5/3 whereas it decreases with increased resolution when using the lower polytropic index. This allows material to

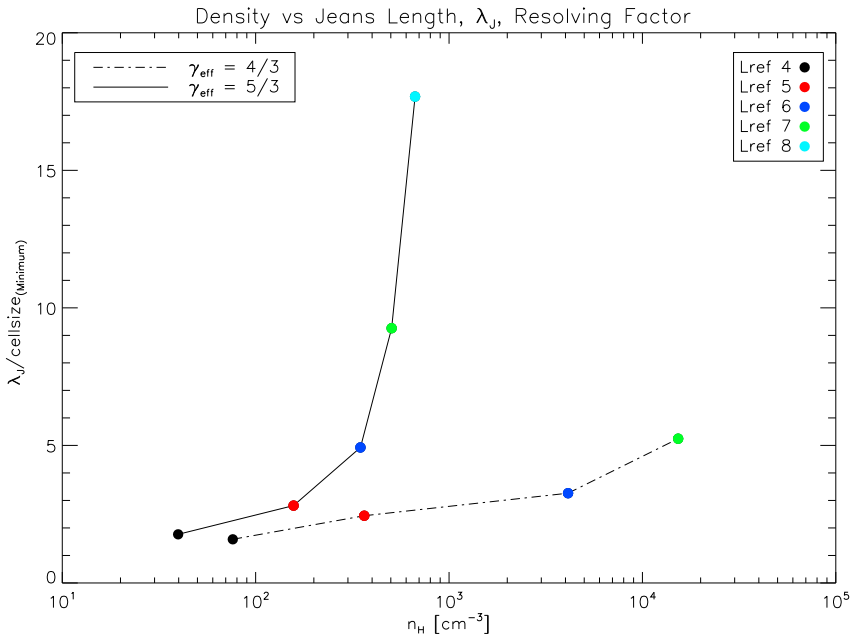

Figure 7.13: Ratio of the minimum Jeans length in the densest cells to their cell size, plotted against gas density. The coloured points represent the outcome of the same simulation run at different levels of refinement with the following peak spatial resolutions; black - 468pc (Lref 4), red - 234pc (lref 5), blue - 117pc (lref 6), green 58pc (lref 7) and cyan 29pc (lref 8). The dashed line connects points from the simulations run with $\gamma_{eff} = 4/3$ whilst the solid line connects simulations with $\gamma_{eff} = 5/3$. As resolution increases, simulations with $\gamma_{eff} = 5/3$ show convergence upon a maximum cell density and thus the Jeans length is resolved with an increasing number of cells. However runs with $\gamma_{eff} = 4/3$ lack convergence and as resolution increases, material is channelled into denser cells, so the Jeans length resolution does not show a marked increase between runs. Typically doubling the resolution increases the peak density by an order of magnitude whilst the Jeans length resolution only improves by $\sim 50\%$. We however note that the Jeans length in the highest resolution simulations for both runs, is resolved by at least the minimum three cells that is imposed in the SPH simulations of Schaye and Dalla Vecchia (2008).

collapse into denser structures within one or two cells.

Figure 7.14 further supports this by showing the mass fraction of the total gas mass at a given density for the $10^{12}\,M_{\odot}$ galaxy using a polytropic index of 4/3 (top) and 5/3 (bottom). The different lines represent increasing levels of refinement; lref 4 (black), lref 5 (red), lref 6 (blue) and lref 7 (green). In the case of a polytropic index of 4/3, as we tend to higher resolutions, material in the disk is swept into much smaller and denser structures. This leads to further a decrease in the Jeans length and creates a runaway collapse. This leads to a lack of convergence in the density profile at densities above the equation of state threshold, $\rho_{g,c} \approx 2 \times 10^{-25}\mathrm{g/cm}^3$ ($n_H = 0.1$ cm^{-3}). Thus it is unlikely that even with extremely high resolution we can converge upon a given solution. The purpose of the polytropic equation of state is to remove the need to use extremely high levels of resolution and as such it should not be necessary to resolve the true multi-phase nature of the ISM. However the scheme must be numerically stable as is the case for the higher polytropic index.

The mass fraction as a function of density for the higher polytropic index can be seen to converge upon a final density distribution. With each successive increase in resolution the mass distribution takes on a smoother more normal distribution.

It is possible that the difference in the way the two codes react when given the same polytropic index arises from the way that they model the Jeans length. Theoretical derivations usually assume a one dimensional perturbation or a symmetrical spherical mass distribution but with an isolated disk, few particles lie above or below the disk. As the Kernel operates over a fixed number of nearest neighbours, the spatial distribution of particles will affect the maximum length of the smoothing length. If a compact spherical distribution is flattened out into a disk, it may increase the smoothing length. Thus it is possible that the *GADGET-2* refinement scheme adopted here which relates the Jeans length to the smoothing length, fails to fully resolve the Jeans scale at all points. This could lead to smoothing out of the otherwise over-dense knots which are observable in *FLASH* since the refinement criteria is independent of the way in which mass is distributed.

It would appear that excellent agreement can be achieved between the two codes using polytropic indices of 4/3 and 5/3 in *FLASH* and *GADGET-2* respectively. In figure 7.15, to determine whether *GADGET-2* is capable of forming such dense knots and see whether a general relation applies, I show projected density images for *GADGET-2* and *FLASH* simulations using polytropic indices of one and 4/3 respectively. The effective equation of state in *GADGET-2* should now lead to a Jeans length and mass that decrease with

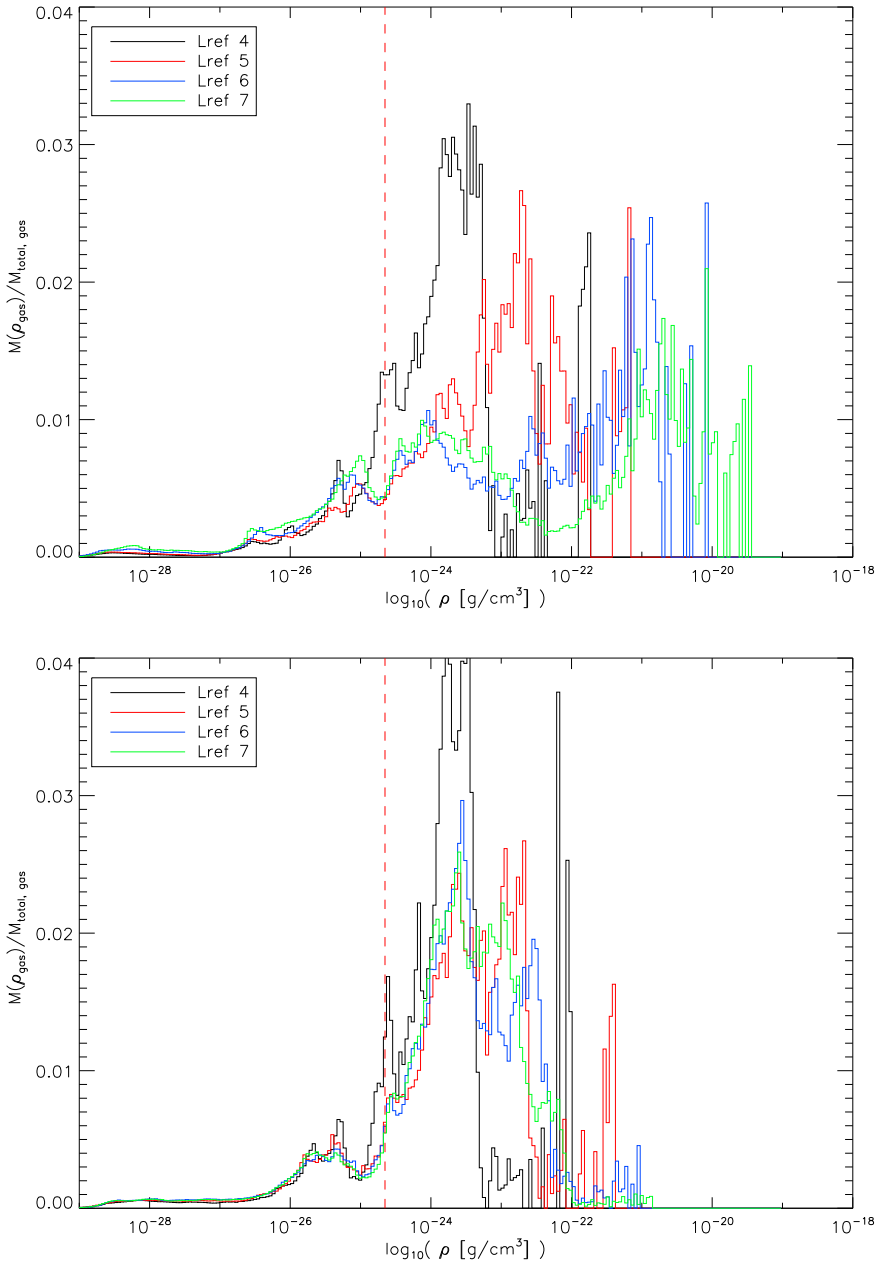

Figure 7.14: Plots of the fraction of the total mass gas at a given density as a function of density for the 10^{12} $h{-}1$ $M_\odot$ galaxy with $\gamma_{eff} = 4/3$ (top) and $\gamma_{eff} = 5/3$ (bottom). The different lines represent increasing levels of refinement; 'lref 4' (black), 'lref 5' (red), 'lref 6' (blue) and 'lref 7' (green). As we tend to higher resolutions for the lower polytropic index, material in the disk is forced into much smaller denser structures, with the general density profile showing no convergence to any given solution beyond densities comparable to the star forming threshold $\rho_{g,c} \approx 2 \times 10^{-25}$g/cm^3 ($n_H = 0.1$ cm^{-3}) (red dashed line). The mass distribution as a function of density can be seen to converge when a higher polytropic index is used. Although an increase in resolution leads to a slight increase in the densest cells, the bulk of the mass remains in a set distribution.

increasing density, leading to the observed fragmentation of the spiral arms and the formation of dense knots. Now both codes produce dense knots as the spiral arms come into contact. Clearly the SPH code does not entirely suppress their formation like it can with hydrodynamic instabilities. However, unlike figure 7.12 the results between the two codes no longer show as much similarity. The formation of the dense knots is found to occur much earlier in *GADGET-2* with many being in place by 100 Myrs. These then go onto sweep up more of the mass in the simulation resulting in the final image at 250 Myrs appearing less well resolved. This happens as even though all images are reproduced with the same resolution, most of the particles that are used to define features of the disk have been swept up into a few very dense regions. The AMR code however can maintain resolution in these less dense regions as well as in the very dense knots.

Similarity between the two codes when using these different pressure laws is not to be expected although it does prevent me from determining a universal relation between the pressure laws in the two hydrodynamic implementations. It would appear that the difference in the two codes is more subtle and would hint at it being a dynamic effect like the smoothing length and variable resolution in the SPH codes. However the results here alone are insufficient to conclusively ascertain the origin.

Such a difference between the way in which pressure forces are dealt with in SPH and AMR codes is not entirely new. Agertz et al. (2007) highlighted the generation of spurious pressure forces in SPH codes around shearing surfaces which suppress the formation of Kelvin-Helmholtz instabilities. The formation of hydrodynamic instabilities can drive turbulent mixing which I have shown in chapter 6 can be a key mechanism for generating entropy cores in galaxy cluster mergers within AMR codes. As these instabilities are suppressed in SPH codes they tend to produce much cooler denser cores. Apparently subtle differences between these different hydrodynamic implementations can therefore have very profound implications upon simulation results. Clearly the nature of these different implementations requires very careful further investigation in order to improve the way in which we develop future models. With the vast resources that are being invested into the latest state-of-the-art simulations, it would prove wise to evaluate differences between the hydrodynamic implementations so that numerical artefacts are not mistaken for genuine physical effects.

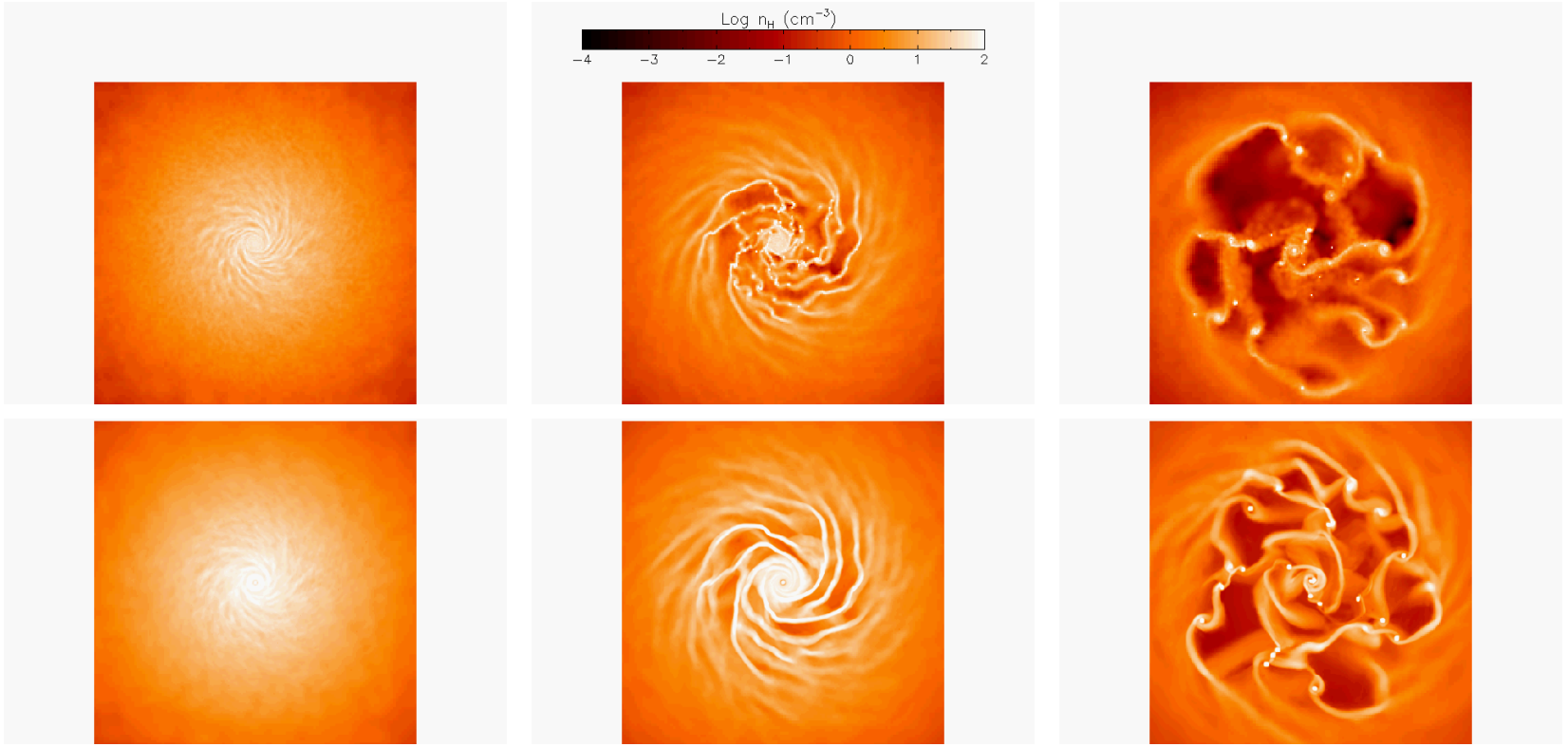

Figure 7.15: Logarithmic face on projected gas density for the model disk galaxy in *GADGET-2* (top) and *FLASH* (bottom). The images show the galaxy at times 25, 100 and 250 Myrs heading from left to right. The two codes use different polytropic indices with *FLASH* using $\gamma_{eff} = 4/3$ and *GADGET-2* using a lower index of $\gamma_{eff} = 1$. All images are shown with the spatial same resolution, however as most of the mass in the SPH simulation is swept up into the dense knots by 250 Myrs, the less dense regions appear less resolved than in the *FLASH* code. Both codes show fragmentation and form dense knots but they evolve earlier on in the SPH code.

7.8 Conclusions

I have successfully implemented the star formation recipe of Schaye and Dalla Vecchia (2008) into the *FLASH* code along with a polytropic equation of state and the cooling routines of Wiersma et al. (2009). I test the star formation recipe and the polytropic equation of state using simplified box models in the absence of hydrodynamics to prevent mixing and particle migration between cells. These have shown that the model is able to reproduce the input Kennicutt-Schmidt law at all levels of resolution and that particle masses below 10^6 $M_\odot$ allow for accurate reproduction of the Kennicutt-Schmidt relation even over a single timestep.

Comparison to the results obtained in Schaye and Dalla Vecchia (2008) was made through the use of a 10^{12} h^{-1} $\mathrm{M}_\odot$ disk galaxy. Initial simulations using the fiducial polytropic index of 4/3 indicated that whilst the *GADGET-2* simulations produce smooth spiral arms and a well converged global star formation rate, those in *FLASH* do not. The *FLASH* AMR implementation instead generates very high density knots when spiral arms come into contact with each other, the material sticking and condensing down into bound structures which then proceed to merge and disturb the remaining spiral arms. I hypothesise that this may be due to the way in which the smoothing length and Jeans length are inter-related, with the ability to resolve the Jeans length dependent upon the spatial distribution of particles.

I confirm the observed discrepancies are not due to differences in the star formation recipe by contrasting the Kennicutt-Schmidt plots, firstly for *FLASH* over a range of resolutions and then between *FLASH* and the default *GADGET-2* simulation. The results are independent of resolution and show almost identical behaviour between *FLASH* and *GADGET-2*. This leads to the realisation that the difference must lie in the polytropic equation of state which is used to model the density pressure relation in star forming regions where the ISM should be multi-phase.

In a thin disk, the smoothing length which should be a given fraction of the Jeans length, could be drawn out relative to a more spatially compact spherical particle distribution. Such an effect could cause the Jeans length to be under-resolved leading to a suppression of the formation of very dense knots. This would not be present in an AMR code as the density and pressure are properties local to the cell so both they and the refinement criteria are not dependent upon the spatial distribution of neighbours. Although this theory seems feasible, I cannot currently present substantial evidence to conclusively

prove it and thus future investigation should be undertaken to try and isolate the cause.

Unlike hydrodynamic instabilities which are completely suppressed in SPH codes, the formation of these very dense knots is possible in *GADGET-2*, providing we reduce the polytropic index. By contrasting polytropic indices of one and 4/3 in *GADGET-2* and *FLASH* respectively, I note similar formation of dense knots. However they are seen to develop earlier in the SPH simulation and sweep up a much greater fraction of the mass than in the AMR code. As the two pressure laws are quite different at this point, with the *GADGET-2* index of one leading to a Jeans length and mass that reduce as the density increases, this is not unexpected.

Exceptional agreement with the fiducial *GADGET-2* simulations can however be achieved by increasing the pressure support against gravitational collapse in *FLASH* by using a greater polytropic index of 5/3. This corresponds to a Jeans length that decreases less rapidly with increasing density and a Jeans mass that increases with density, stabilising the collapse of material into dense knots against a runaway effect. This results in excellent convergence of the star formation properties of the galaxy with increasing resolution. Spiral structure for the higher polytropic index of 5/3 is almost identical to that observed in the *GADGET-2* simulations which use the lower default polytropic index of 4/3.

This shows that the star formation recipe of Schaye and Dalla Vecchia (2008) has been implemented in a consistent fashion into the *FLASH* AMR framework with near identical results being obtained. The need to vary the polytropic pressure relation however clearly indicates yet another way in which a subtle difference between the hydrodynamic implementations can incur dramatic differences in simulation results even when using identical sub-grid prescriptions.

In order for AMR codes to robustly model the multi-phase ISM with a polytropic equation of state they should adopt values for the polytropic index of 5/3 or greater in order to attain convergence in their simulations.

Chapter 8

Sub-Grid Models for Supernova Feedback in FLASH

8.1 Introduction

As stars age and eventually die in cataclysmic supernova, they inject enormous amounts of energy and metals into the local ISM which influence the future star forming properties of the galaxy (see, e.g. Aguirre et al., 2001). Supernova driven winds and the induced shock heating of the ISM present a means with which to regulate the formation of stars, requiring the hot gas to cool before star formation can resume. Without supernova feedback, cosmological simulations develop strong cooling flows and higher star formation efficiencies than are observed (see, e.g., White and Frenk, 1991). In the case of material which is blown out of the galaxy, that which remains bound turns around and rains back down creating a "galactic fountain." This provides a regulated fuel supply for future star formation. In less tightly bound structures such as dwarf galaxies, supernova driven winds may have sufficient kinetic energy to escape from the galactic potential, causing a more abrupt end to star formation and allowing the spectrum to become dominated by the older redder stellar populations.

In practice it has proven difficult to implement effective feedback mechanisms into simulations. Current cosmological runs still fail to counter cooling flows which has led to overly massive galaxies being formed (see, e.g. Tonnesen et al., 2007). The problem ultimately arises due to the current lack of resolution in cosmological simulations with which to resolve individual supernova driven bubbles and the multi-phase ISM. Even superbubbles formed when many individual supernova coalesce are beyond the resolution of most cosmological simulations. If codes inject the energy from single or even multiple supernova into a given cell, the energy will be distributed over a much larger volume

of gas than it would in reality. This can lead the energy to be radiated away in a few timesteps before it has time to undergo the adiabatic expansion phase and convert the thermal energy into kinetic form (see, e.g., Katz et al., 1996; Balogh et al., 2001). Mechanisms proposed to overcome this include temporarily turning off cooling (see, e.g., Mori et al., 1997; Thacker and Couchman, 2000; Brook et al., 2004; Stinson et al., 2006) and decoupling the supernova wind from the hydrodynamics (see, e.g., Springel and Hernquist, 2003). As normally the expanding supernova shell sweeps up material as it travels away from the dying star, it's effective "mass loading" - the amount of material which is swept up in the wind, increases. Thus simulations which decouple the wind from the hydrodynamics may give the wrong effective mass loading. Such measures will prevent supernova feedback from influencing galaxy morphology in any way other than removing gas for further star formation. They are unable to blow supernova bubbles, leave cavities in the ISM and drive compression waves which may be an important component in the formation of spiral arms (see, e.g. chapter 1, Binney and Tremaine, 1987). The more recent model of Dalla Vecchia and Schaye (2008) which has been used in the *OWLS* and *GIMIC* simulations presents an alternative approach whereby the supernova driven wind remains coupled to the ISM and instead of directly injecting thermal energy, they assume that a fraction is converted into kinetic energy whilst the rest is lost through radiative processes. This model offers many advantages over other prescriptions with Dalla Vecchia and Schaye (2008) showing that the resulting mass loading (the fraction of mass swept up in the wind) is greater than the input mass loading. This implies that the supernova driven material sweeps up additional matter as it attempts to force its way out of the galaxy. They test their feedback prescription on both a 10^{10} $M_{\odot}$ dwarf galaxy and a larger 10^{12} $M_{\odot}$ Milky Way like galaxy. By keeping the feedback coupled to the hydrodynamics, the winds are able to puff up the disk and dramatically change the galaxy morphology. The dwarf galaxy undergoes a morphological change towards becoming an irregular as the supernova bubbles blow large cavities in the disk, whilst in the case of the larger disk galaxy a significant fraction of the mass outflow remains bound due to the greater gravitational potential and turns around to fall back in on the galaxy, generating a galactic fountain. They find that the infall is highly clumpy due to clouds that have formed through thermal-gravitational instability in the wind. This causes the outflow to decrease with time since mass expelled at late times must push through those layers which where cast off earlier and are now falling back in. The star formation efficiency is greatly reduced in both with a much sharper initial decline in the case of the dwarf galaxy.

In the previous chapter 7 I revealed how distinct differences in the hydrodynamic implementation are able to influence the performance of sub-grid models. I showed that with identical sub-grid recipes for star formation and a polytropic equation of state, the different hydrodynamic algorithms can lead to dramatically different galaxy morphologies. In this chapter I continue the comparison between the *GADGET-2* SPH and *FLASH* AMR codes once the kinetic supernova feedback prescription of Dalla Vecchia and Schaye (2008) is included. I continue to adopt the star formation and polytropic equation of state of Schaye and Dalla Vecchia (2008) as detailed in the previous chapter, using a polytropic index of 4/3 for *GADGET-2* and a value of 5/3 for *FLASH* since these were found to result in very similar galaxy properties.

I begin by describing the kinetic supernova feedback recipe of Dalla Vecchia and Schaye (2008) in § 8.2 along with an outline of the initial conditions of their simulations in § 8.3. I then describe the *FLASH* initial conditions and any modifications necessary to run the code in § 8.4. As the two hydrodynamic schemes are so fundamentally different I develop a series of feedback models in *FLASH*(§ 8.5). These include a pure kinetic feedback mechanism (§ 8.5.1) and a feedback model that includes a thermal component to help counter dissipation of thermal energy by the polytropic equation of state (§ 8.6.3).

I highlight in § 8.5.4 the problems with variable mass cells in an AMR code and how they affect the SPH feedback mechanism of Dalla Vecchia and Schaye (2008). I also describe the difficulties of tracking the wind in § 8.5.5.

I initially test the performance of my different feedback models using the same $10^{12}\ h^{-1}\ \mathrm{M}_{\odot}$ galaxy as in chapter 7 to ascertain the robustness of my feedback model (§ 8.6). By contrasting the results from different schemes in the *FLASH* code with the observed feedback efficiency in *GADGET-2*, I am able to highlight differences in the two codes. I then compare the effects of feedback in the two hydrodynamics codes upon the morphologies, net mass outflow rates and outflow wind speeds in both a Milky Way like $10^{12}\ h^{-1}\ \mathrm{M}_{\odot}$ galaxy and a smaller $10^{10}\ h^{-1}\ \mathrm{M}_{\odot}$ dwarf galaxy. I contrast the results in § 8.7. I conclude my findings in § 8.9.

8.2 Kinetic Supernova Feedback Recipe of Dalla Vecchia and Schaye (2008)

Dalla Vecchia and Schaye (2008) begin by assuming that a given fraction of the energy, f_w, is converted to kinetic energy and carried away in the supernova driven wind. The

remainder is implicitly assumed to be lost through radiative processes. It is these radiative losses which represent part of the energy needed to provide the pressure support that the polytropic equation of state mimics.

As in Aguirre et al. (2001), the kinetic feedback is specified through two variable parameters, the initial mass loading of the wind, and the wind velocity, v_w. Dalla Vecchia and Schaye (2008) choose to express the initial mass loading of the wind, $\dot{M}_w$, as a fraction of the star formation rate, $\dot{M}_*$, using a dimensionless mass loading parameter, η:

$$\dot{M}_w \equiv \eta \dot{M}_* \tag{8.1}$$

Assuming that core collapse supernova inject a kinetic energy of ϵ_{SN} per solar mass of stars formed, then the fraction of energy in the wind is

$$f_w = \frac{\eta v_w^2}{2\epsilon_{SN}} \approx 0.4 \left(\frac{\eta}{2}\right) \left(\frac{v_w}{600 \text{ kms}^{-1}}\right)^2 \left(\frac{\epsilon_{SN}}{1.8 \times 10^{49} \text{erg M}_\odot^{-1}}\right) \tag{8.2}$$

The value of $\epsilon_{SN} \approx 1.8 \times 10^{49}$ erg $\text{M}_\odot^{-1}$ is appropriate for a Chabrier (2003) initial mass function (IMF) and a stellar mass range of $0.1 - 100\text{M}_\odot$, if all stars in the mass range $6 - 100\text{M}_\odot$ end their lives as core collapse supernova imparting the canonical supernova energy of 10^{51} ergs each (Springel and Hernquist, 2003). Alternatively a Salpeter (1955) IMF would give $\epsilon_{SN} \approx 1.1 \times 10^{49}$ ergs $\text{M}_\odot^{-1}$. Dalla Vecchia and Schaye (2008) choose values of $v_w = 600\text{kms}^{-1}$ and $\eta = 2$ as their fiducial values, implying that the wind carries around 40% of the energy with the rest being lost radiatively.

They confirm that these values are consistent with observations of local starburst galaxies which have wind speeds that increase with galaxy mass (or star formation rates) from tens to thousands of kilometres per second, whilst the mass in cold outflowing gas ranges from $\eta \sim 0.01$ to 10 (see Veilleux et al., 2005; Martin, 2005; Rupke et al., 2002; Heckman et al., 2000). These values are found to hold at high redshift, with starburst galaxies at redshift $z \sim 3$ typically having wind velocities of hundreds to thousands of kilometres per second (Shapley et al., 2003). In particular, Schaye et al. (2010) shows that the use of the values of $\eta = 2$ and $v_w = 600$ kms^{-1} are able to reproduce the observed peak in the cosmic star formation rate at redshift two.

When stars reach an age $t_{SN} = 3 \times 10^7 yr$, corresponding to the maximum lifetime of a star that ends its life as a core collapse supernova, it is allowed to inject kinetic energy into the surrounding cells. They adopt a stochastic approach in which they loop over each particle within a given number of nearest neighbours with a probability of kicking each particle of

$$Prob = \frac{\eta \; m_*}{\sum_{i}^{N_{gb}} m_{g,i}} \tag{8.3}$$

where m_* is the mass of the star particle going supernova, $m_{g,i}$ is the mass of the i^{th} gas particle (or cell in *FLASH*) with them looping over N_{gb} neighbouring particles. In the SPH code $N_{gb} = 48$, corresponding to a fixed number of particles within the SPH smoothing Kernel. When particles are kicked, they are given a randomly orientated velocity with magnitude equal to the chosen wind speed. As star particles have the same mass as the gas particles, this ensures that on average a number of particles equal to the chosen mass loading will be kicked with the velocity of the wind every time a star goes supernova. As the size of a supernova blast wave is still many times smaller than the resolution reached in current simulations, the model does not attempt to evenly kick particles out in a spherically expanding shell but instead aims to kick sufficient particles with adequate momentum to escape the galactic disk and to drag the ISM up with them. Dalla Vecchia and Schaye (2008) try kicking particles orientated both away from the originating supernova as well as randomly and indicate that it makes no difference to the results. Looping over particles within the smoothing length provides sufficient mass to ensure enough particles will be kicked, prevents the kicking of gas from being de-localised from the supernova and removes the need for more communication intensive algorithms that would be required if you wanted to kick more distant neighbours.

In order to avoid the unphysical formation of stars in gas that has been kicked at high velocity the SPH simulations explicitly tag recently kicked particles as "wind" particles. These remain in the wind for a period of time $t_w = 1.5 \times 10^7 yr$ during which they continue to contribute to the hydrodynamic forces but cannot form stars or be kicked again until they leave the wind. They may leave the wind early if the density of the region they enter is less than ten percent of the threshold density for star formation.

Dalla Vecchia and Schaye (2008) tests the performance of the model on a $10^{10}\ h^{-1}\ \mathrm{M}_\odot$ dwarf galaxy and a $10^{12}\ h^{-1}\ \mathrm{M}_\odot$ Milky Way like galaxy, both of which use the same initial disk setup as in the star formation tests presented in Schaye and Dalla Vecchia (2008) which I briefly outline in § 8.3. For simplicity metal enrichment and the return of mass to the ISM is neglected when stars go supernova.

8.3 Initial Conditions in GADGET

Dalla Vecchia and Schaye (2008) adopt the same model disk galaxies as used in Schaye and Dalla Vecchia (2008). They consist of an exponential gas and stellar disk along with a spherical stellar bulge and a dark matter halo which follow Hernquist (1990) profiles. The disk gas fraction is initially set to one third. The dark matter profile is scaled to match an NFW profile within the virial radius with concentration, $c = 9$. The 10^{10} h^{-1} $M_{\odot}$ dwarf galaxy (we use $h = 0.73$) has an effective NFW virial radius of 35.1 h^{-1} kpc whilst the larger 10^{12} h^{-1} $M_{\odot}$ Milky Way like galaxy has a virial radius of 163 h^{-1} kpc.

They adopt a density threshold for star formation of $n_H = 0.1$ cm^{-3} and a temperature threshold of $T/\mu = 10^4 K/0.59$. The simulations feature radiative cooling and heating in the presence of the redshift zero UV background from galaxies and quasars. The simulations are setup with primordial abundances of hydrogen and helium and no metals are present or injected through stellar processes for simplicity.

Both the *GADGET-2* and *FLASH* simulations presented in § 8.6 adopt the same Kennicutt-Schmidt normalisation of $2.5 \times 10^{-4} M_{\odot}$ yr^{-1} kpc^{-2} and exponent, $n = 1.4$ as was used by Schaye and Dalla Vecchia (2008). This ensures that the results from my feedback models can be contrasted directly with my findings in chapter 7 without feedback.

The simulations presented in Dalla Vecchia and Schaye (2008) use a Chabrier IMF which extends the Salpeter (1955) IMF down to lower mass stars and uses improved observations. As the Kennicutt law in equation (7.2) assumes a Salpeter IMF, the new Kennicutt-Schmidt law normalisation is reduced to give a value of $1.5 \times 10^4 M_{\odot}$ yr^{-1} kpc^{-2}. Both the *FLASH* and *GADGET-2* simulations presented in § 8.7 adopt this revised value for the Kennicutt-Schmidt law normalisation to remove the need to re-run the *GADGET-2* feedback simulations with the original normalisation. As Schaye and Dalla Vecchia (2008) show that their star formation model can robustly reproduce any input star formation law, this should affect both codes identically. This reduced normalisation will result in a lower star formation and thus supernova rate for a given surface density. The same exponent, $n = 1.4$ is maintained.

The *GADGET-2* simulations are run in an isolated 20 Mpc^3 box and use a particle mass of 5.1×10^4 h^{-1} $M_{\odot}$ for baryonic material (stars and gas). The dark matter particles are 4.6 more massive than the gas particles. The total number of particles in each simulations is 4,800,736, of which 225,437 are gas particles in the disk. Again, like the simulations

presented in chapter 7, Dalla Vecchia and Schaye (2008) fix the ratio of the smoothing length to the Jeans length, $\lambda_J/h = 0.275$. For the 10^{12} h^{-1} $M_\odot$ galaxy this gives a peak spatial resolution of 44 pc and for the 10^{10} h^{-1} $M_\odot$ dwarf a peak resolution of 15pc.

8.4 Initial Conditions in FLASH

8.4.1 Gaseous Halo, Box Size and Boundary Conditions

As outlined in chapter 7, the *FLASH* simulations feature an additional tenuous gaseous halo since the Riemann solvers in AMR codes experience convergence problems when dealing with vacuum conditions. The pressure and density initially drop by four and six orders of magnitude respectively at the boundary between the disk and the halo. This ensures that the ram pressure of material leaving the disk will be much greater than any resistive pressure from the halo. The dark matter is modelled with an analytic Hernquist (1990) profile instead of a live halo as in the *GADGET-2* simulations.

As hardware constraints prevent the *FLASH* simulations from being run in a 20 Mpc^3 box, I instead adopt a 120 kpc^3 box for the 10^{12} h^{-1} $M_\odot$ galaxy at a refinement level of seven ($\equiv 1024^3$ cells) giving a peak spatial resolution of 58pc. This is comparable to the converged simulations in chapter 7 and the peak spatial resolution in the SPH simulations. This box size encloses 27% of the galaxy's virial radius. I choose to simulate the dwarf galaxy in a 60 kpc^3 box at a refinement level of six giving an effective peak spatial resolution of 58pc, identical to the higher mass galaxy. This allows me to determine the effect of feedback in a dwarf galaxy with a comparable level of resolution to the higher mass galaxy as would be the case in a true cosmological simulation. Although my *FLASH* dwarf galaxy does not have comparable spatial resolution to the SPH equivalent, I find that feedback in the dwarf galaxy causes it to become highly unstable as soon as feedback begins. This makes it impractical to run more computationally demanding simulations. The choice of box size for the dwarf galaxy allows the box to enclose 63% of the galaxy's virial radius. I vary the input mass loading and wind velocity in both disk galaxy simulations which are run for 400 Myrs.

Given that my *FLASH* simulations take place inside a 120 kpc^3 box, it is possible that boundary effects could influence my results. My first concern is whether interactions at the boundary could artificially reflect material back down onto the disk. This would increase the star formation rate and inhibit outflowing winds. The vast 20 Mpc^3 box in *GADGET-2* ensures that ejected material cannot contact the box edges and so only

material which actually remains gravitationally bound can fall back onto the disk at late times.

To overcome this I impose outflow boundary conditions which allow material to leave through the simulation box boundaries. Boundary effects are unlikely to influence behaviour in the disk anyway as material ejected from the galaxy with a wind velocity of $v_w = 600$ kms^{-1} at time $t = 0$, would only have travelled 248 kpc by the end of these simulations. It will therefore only have just bounced back to the centre of the 120 kpc^{-3} box by the end of the simulation. Furthermore the sound speed with which pressure waves can traverse the halo gas typically peaks at around 131 kms^{-1}. Such a relatively low velocity ensures that interactions at the boundary will be unable to influence the behaviour of material in and around the star forming regions of the disk. I confirm this by further halving the *FLASH* box size to 60 kpc^3. This should lead to more interaction of outflowing gas with the boundary, however as expected I observe no change in the global star formation rate. I can therefore conclude that the boundary conditions will not artificially suppress the efficiency of the supernova feedback in *FLASH*.

The reduced box size however does have one main drawback. In the simulations of Dalla Vecchia and Schaye (2008), they observe natural turnaround of some of the outflowing material at late times. This then falls back in on the galaxy creating a galactic fountain. The turnaround of material generally occurs at larger radii than those that are enclosed within my *FLASH* simulations. Although I use outflow boundary conditions to prevent material artificially being reflected back from the boundary, it does prevent material that may naturally turnaround outside of the box from falling back onto the galaxy. As any information pertaining to material that leaves the box is lost, this means that my simulations are not expected to be able to form the galactic fountain that Dalla Vecchia and Schaye (2008) observe. Fortunately Dalla Vecchia and Schaye (2008) only observes inflow of material after 330 Myrs in the larger galaxy and after 400 Myrs in the dwarf galaxy. As such this limitation is unlikely to present a significant bias upon my *FLASH* simulations, especially at early times.

8.4.2 Choice of Polytropic Index

In chapter 7 I showed that substantial differences arose when using the same polytropic index in both the *FLASH* and *GADGET-2* codes. When using a polytropic index of 4/3 (as adopted in Schaye and Dalla Vecchia, 2008; Dalla Vecchia and Schaye, 2008) the AMR code produces very compact dense knots of gas. These compact knots are found to

become denser as the resolution is increased. Since the star formation rate scales with the density ($\dot{m}_* \propto \rho^{1.4}$), this prevents the global star formation rate from converging as higher densities lead to higher star formation rates. The SPH code however does not form these dense knots of gas and instead produces smoother spiral structures. As I do not wish to run the *GADGET-2* simulations again using a higher polytropic index and as I have already shown in chapter 7 that using identical polytropic indices will give substantially different results, I opt to use values of 4/3 in *GADGET-2* and the higher value of 5/3 in *FLASH*. These values were found to yield very good agreement both in terms of the global star formation histories and the general morphologies of the galaxies formed. I refer the reader back to chapter 7 for a more detailed overview of the initial conditions and the comparative results. Table 8.1 lists the simulations that I run in § 8.7 along with the different input wind speeds and mass loading parameters.

Table 8.1: Parameters: code used for simulation (*FLASH* or *GADGET-2*), total mass, M_{halo}; input mass loading, η; input wind velocity, v_w; whether the wind model was used (Wind)

Simulation	Code	M_{halo} (h^{-1} $M_\odot$)	η	v_w (kms^{-1})	Wind
Fm12nowind	*FLASH*	10^{12}	-	-	N
Fm12ml1	*FLASH*	10^{12}	1	848	Y
Fm12ml2	*FLASH*	10^{12}	2	600	Y
Fm12ml4	*FLASH*	10^{12}	4	424	Y
Fm10nowind	*FLASH*	10^{10}	-	-	N
Fm10ml1	*FLASH*	10^{10}	1	848	Y
Fm10ml2	*FLASH*	10^{10}	2	600	Y
Gm12nowind	*GADGET-2*	10^{12}	-	-	N
Gm12ml1	*GADGET-2*	10^{12}	1	848	Y
Gm12ml2	*GADGET-2*	10^{12}	2	600	Y
Gm12ml4	*GADGET-2*	10^{12}	4	424	Y
Gm10nowind	*GADGET-2*	10^{10}	-	-	N
Gm10ml1	*GADGET-2*	10^{10}	1	848	Y
Gm10ml2	*GADGET-2*	10^{10}	2	600	Y
Gm10ml4	*GADGET-2*	10^{10}	4	424	Y

8.5 Implementing Kinetic Supernova Feedback into FLASH

By replacing references to the gas particle masses in equation (8.3) with the AMR cell masses the same stochastic approach to kicking cells can be adopted. Although initially the supernova feedback prescription of Dalla Vecchia and Schaye (2008) appears to be relatively easy to implement into the *FLASH* AMR structure, problems arise due to the fundamentally different way in which the two codes represent the gas.

8.5.1 Stochastic Kicking of Cells

As I shall describe in § 8.5.4, the fact that cell masses can vary in an AMR code presents problems when implementing a feedback recipe that was predominantly written for an SPH code. The SPH code uses fixed mass gas particles with comparable masses to the star particles. AMR codes however have variable mass gas cells and so the total gas mass of the enclosed cells can vary greatly. For instance I find that the mass of cells within my disk galaxy simulations span more than three orders of magnitude. To overcome this and ensure that the feedback scheme works well in all density regimes it is necessary to allow the enclosed number of cells to be varied.

The effect of the star particle resolution can be partly removed by first summing the total number of star particles that will go supernova in a given cell over the timestep, n_{SN} and then calculating the cumulative probability

$$Prob = \frac{\eta \; m_* n_{SN}}{\sum_{i}^{N_{gb}} m_{g,i}} \tag{8.4}$$

instead of calculating the probability independently for each supernova. Here $m_{g,i}$ is the mass of gas in the i^{th} cell and N_{gb} is the number of neighbours we loop over. This ensures that both small and high particle masses give the same kick probability.

When looping over a fixed number of neighbouring cells, it is possible to obtain a probability for kicking cells greater than one. This can occur in very low density regions where the mass of gas may be many times lower than the mass of a star. This would imply the need to kicks cells many times and presents two significant drawbacks. Firstly multiple kicks to the same cell can yield extremely high unphysical wind velocities. These would cause the Riemann solver to fail to converge since large velocity differences at the cell boundary will yield poor interpolation results. Even if the Riemann solver was able to work, such wind speeds would still lead to very small timesteps that would make the

simulation too slow to perform. Secondly, if the kick direction is orientated randomly, it is possible for the second kick to oppose the first, yielding a much smaller or zero kick speed. Thus the energy imparted to the ISM would be lower than expected.

Instead I opt to vary the enclosed number of neighbouring cells in the probability calculation in equation (8.4). I first determine whether the probability of kicking the parent cell in which the star particle resides is less than or equal to one,

$$Prob = \frac{\eta \; m_* n_{SN}}{m_{g,parent\ cell}} \tag{8.5}$$

If it is then I kick the parent cell alone with that probability. In very massive cells this allows for the cumulative supernova probability to reach higher values and increases the frequency with which we kick high mass cells. If the probability of kicking the parent cell is greater than one, I guarantee to kick the parent cell. This prevents feedback from becoming de-localised with surrounding cells being kicked in place of the parent cell where feedback should actually initiate. I then increase the number of neighbouring cells I loop over (N_{gb}) until the following probability is less than or equal to one,

$$Prob = \frac{\eta \; m_* n_{SN} - m_{g,parent\ cell}}{\sum_{i}^{Ngb} m_{g,i}} \tag{8.6}$$

This scheme ensures that the probability of kicking cells is as large as possible whilst adapting to enclose sufficient mass to prevent loss of mass loading in less dense regions of the disk. I present a resolution test of the expected and actual mass loading for the simulations in § 8.6.1.

8.5.2 Tracking the Wind Across the Mesh

To prevent cells that have recently been kicked by supernova from receiving additional kicks that could lead to unphysical wind speeds, I tag the wind using a mass tracer, w, which advects with the fluid. Although as I will explain in § 8.5.5 there are difficulties in tracking the wind in an AMR code, I chose to implement the following method.

When the fluid in a cell is kicked, it is placed on the wind by setting the tracer to 1. After each timestep I then age the wind according to

$$w(t) = w(t=0)e^{-t/t_w} \tag{8.7}$$

thus the wind at time $t + \Delta t$ is given by

$$w(t + \Delta t) = \left(1 - \frac{\Delta t}{t_w}\right) w(t) \tag{8.8}$$

where $t_w = 1.5 \times 10^7$ yrs is the same as in the SPH setup. Thus the wind tracer fluid is reduced each step using equation (8.8) until it drops below a user defined threshold at which point the material within the cell is no longer classed as part of the wind.

The wind tracer fluid to specifies the fraction of material which is part of the wind, with equation (8.8) returning a small fraction of the gas to a star forming state each step. When wind material advects into cells with a density less than $0.1\rho_{g,c}$ I remove it from the wind. Supernova are unable to kick cells with more than a user defined fraction of wind material. By default I set this limit to 0.5 but find that varying the limit makes little difference to the resulting global star formation rate. Although not exactly identical to the SPH setup it provides the closest approximation possible.

8.5.3 A Need for Thermal Feedback

As cells can be kicked randomly with a random orientation, most cells are likely to be kicked into the plane of the disk. When this happens, dense cells collide with equally dense cells and the energy imparted through the kick can be lost. Normally this would be thermalised in a shock, however the polytropic equation of state does not explicitly track a temperature and therefore internal energy which is independent of the density. When the gas shocks, the increase in the internal energy is lost when recalculating the effective pressure and corresponding internal energy using equation (7.14). Although *GADGET-2* does not allow for material on the effective equation of state to shock, I allow for the addition of a "thermal" component to the feedback.

I first determine the temperature of the gas according to the internal energy resulting from the shock (as calculated by the Riemann solver) and the effective temperature on the polytropic equation of state

$$T_{eff} = T_c \left(\frac{\rho_g}{\rho_{g,c}} \right)^{(\gamma_{eff}-1)} \tag{8.9}$$

where T_c is the critical temperature at the star formation threshold. If the temperature from the Riemann solver is greater than a user defined multiple (say 1.05) of the effective temperature on the polytropic equation of state, then I remove it from the effective equation of state. The internal energy from the Riemann solver at this point is conserved and the pressure updated according to

$$P = \rho \varepsilon_{int} (\gamma - 1) \tag{8.10}$$

This allows the collision of equation of state gas to effectively shock and leave the polytropic equation of state, terminating star formation in regimes where it should not be

present. This helps prevent the otherwise spurious suppression of feedback by the numerical implementation.

8.5.4 Problems with Variable Mass Cells

In SPH codes the gas and star particles have identical masses and thus the probability obtained from equation (8.3) is of order η/N_{gb}. For default mass loading of two this gives a probability of kicking a gas particle of 1/24. On average this leads to the SPH code kicking a number of particles equal to the mass loading every time a supernova goes off. The only regime in which this system can fail is in strong star forming regions where many gas particles may have already been kicked or converted into stars leaving no gas particles present that are not already part of the wind. Dalla Vecchia and Schaye (2008) record the global expected and imparted mass loading of the wind and confirm that for their simulations this presents a negligible effect.

AMR codes however have variable mass cells which can lead to a series of different regimes for which different ideal star particle masses exist. At the edges of a galactic disk the gas density can be hundreds to thousands of times less dense than at the galactic core. At the centre of the galactic disk this can result in the enclosed gas mass of neighbouring cells being far greater than the mass of the star particle. This leads to extremely small kick probabilities and often star particles will go supernova without kicking any mass. In theory only a few successful kicks should be required since the cells contain more mass and therefore averaging over a large number of potential kicks will yield a mean mass loading in line with expectations.

Initial investigations which applied the feedback approach in equation (8.3) with a fixed number of neighbouring cells found that in practice the probability of kicking the parent cell is so low that within the space of a typical simulation, insufficient attempts to kick it occur. This arises since applying a uniform probability to each of the enclosed cells can allow the few successful kicks to actually move material outside of the disk. Since the disk is only a few cells thick, the chances of kicking a cell outside of the disk are much higher than kicking those within the disk. This de-localisation of the feedback can cause problems with an under-loading of the wind. As this then allows more mass to accrete onto these dense cells, it exacerbates the effect. Thus in very dense regions were the probability of kicking a cell is extremely low, ideally only the parent cell should be included in the calculation.

On the other hand, in the outer regions of the disk, the same star particle may

have many times the mass of the enclosed cell gas masses. This would yield very high probabilities for kicking the cells, possibly leading to all of the cells being kicked. It may even be necessary in the outer regions of the disk to increase the number of cells we loop over to return the correct global mass loading.

Although increasing the star particle mass increases the probability with which supernova can kick cells in the galactic core, it will lead to supernova on the edges of the disk being unable to kick sufficient mass. I also showed in chapter 7 that increasing the star particle mass beyond a threshold value (i.e. that required to kick central cells with sufficient frequency) leads to poor reproduction of the Kennicutt-Schmidt law as the probability of forming individual stars in low mass cells drops and greater sampling statistics are required. This increases the need to use mass deficit tracer particles and ideally should be avoided.

Decreasing the star particle mass so that we are able to kick enough mass at the edges of the disk, results in insufficient mass loading at the centre of the galaxy. Clearly a small star particle mass is favoured for the star formation routines whilst a much larger mass is required for the supernova feedback routines. Changing the star particle mass will therefore not resolve these limitations.

Clearly it is necessary to diverge from the initial approach used by Dalla Vecchia and Schaye (2008) and allow the number of neighbouring cells which we enclose to vary in line with the enclosed mass and probability of kicking a cell. In very dense regions where the probability is low, less cells will be included whilst in low density regions where the probability of kicking a cell is higher, we can include more cells and mass. Although I truncate the loop over neighbouring cells when it exceeds the number of available guardcells, the disk galaxy simulations presented later in this chapter indicated that this has a negligible effect upon the mass loading and that there were always sufficient enclosed cells. I also prevent cells that have already been kicked and are classed as on the wind from featuring in the enclosed mass calculation in equation (8.6) preventing excessive wind velocities. This also proves to have no adverse effect upon the resulting mass loading relative to what was expected.

I do however note that it is possible for some cells to receive two kicks - both due to supernova local to their block and due to other supernova on neighbouring blocks. This happens when kicks are mapped across from neighbouring blocks using the guardcells. Although an explicit routine could have been written to prevent this it would present other problems such as corners of the blocks becoming visible when supernova at the

edges of the block are only able to kick internal instead of a sphere spread across different neighbouring blocks. The routine would also present a significant communication penalty and thus was not implemented

8.5.5 Difficulties Tracking the Time Evolution of the Wind

In the SPH simulations of Dalla Vecchia and Schaye (2008) they prevent repeatedly kicking particles by tagging them as "wind" particles and attaching a unique timer to each wind particle which records how long they have been part of the wind. For a time, $t_w = 1.5 \times 10^7$ yrs they continue to contribute to the hydrodynamic forces but cannot be kicked until they leave the wind or the density drops below ten percent of the star formation threshold.

Copying this explicitly is impossible in any grid code since the fluid mass in *FLASH* is not discrete like in an SPH code. In order to follow the wind I must use an additional mass tracer, w, which advects with the flow to mimic the property of the timer.

My initial scheme prevented the entire contents of the cell from forming stars or being kicked when the wind tracer fluid was above a given threshold (one e-folding time). Although this worked well if the fluid was static and did not mix with other material, in practice the fluid motion mixes both non-wind and wind material, reducing the wind fraction faster than expected. In addition if a fraction of the material from a kicked cell advects into a neighbouring cell, raising the wind fraction above the cut-off threshold for star formation, it will completely terminate star formation within both cells. This can lead to a spurious increase in the total mass of material in the wind. This leads to a strongly cyclic mode of star formation with star formation giving rise to supernova and cutting off further star formation for a time t_w. As the wind ages and gas leaves the wind, star formation resumes at the same level as in the absence of feedback. Such a model does not mimic the properties of the SPH code and therefore I opt to implement the model outlined in § 8.5.2. As I will show in § 8.7, this model reproduces the smooth decrease in the global star formation rate over time in the SPH simulations.

8.6 Investigating Different Approaches to Feedback in FLASH

I now investigate the effect of the different feedback prescriptions in *FLASH* using the same model disk galaxy and star formation parameters as used for the star formation tests in chapter 7.

8.6.1 Resolution Checks: Is the Mass Loading Correct?

Like the simulations presented in chapter 7 which lack feedback, those which include feedback all converge at relatively modest levels of refinement (lref 6 and above).

To confirm that the *FLASH* recipe is kicking sufficient mass and that the feedback routine is converged I plot the ratio between the actual and expected mass loading in figure 8.1. Stochastic variations can initially lead to $\sim 50\%$ more mass being kicked than is expected. As the sampling statistics improve, these variations rapidly drop off with some supernova kicking less mass than on average they should. This allows the ratio between the actual and expected gas mass kicked by supernova to tend back towards unity. The plotted lines correspond to increasing refinement levels; black (lref 4), red (lref 5), blue (lref 6) and green (lref 7). The results are found to converge almost exactly by lref 6 with an RMS deviation of less than $\sim 1\%$, agreeing well with the convergence of the star formation rate previously shown in § 7.7. Higher levels of resolution then lead to more modest reduction in the RMS deviation with the star formation rate also converging by a refinement level of six.

The relatively high deviations in the lowest resolution run (lref 4, black line) arise due to the lower number of cells. As there are less cells to kick and I sum the total number of stars going supernova in each before calculating the kick probability (see § 8.5.1), the sampling statistics are lower. This leads to much larger stochastic variations, especially as each cell is coarser and therefore more massive than in the higher resolution simulations. More timesteps are required to provide the sampling statistics necessary to return the cumulative ratio back to unity as can be seen happening towards the end of the simulation, at which point the deviation has dropped to $\sim -2.5\%$ from the earlier 15% deviations before 100 Myr.

Figure 8.1 clearly shows that the supernova feedback prescription has converged with excellent agreement between the expected and imparted mass loading.

8.6.2 Efficiency of Kinetic Feedback

The global star formation rate as a function of time without feedback as was obtained in chapter 7 (black) and with the kinetic implementation of the feedback prescription (which does not include the thermal component) is plotted in figure 8.2. The equivalent *GADGET-2* simulations with (dashed purple line) and without winds (dot-dashed purple line) are plotted for comparison.

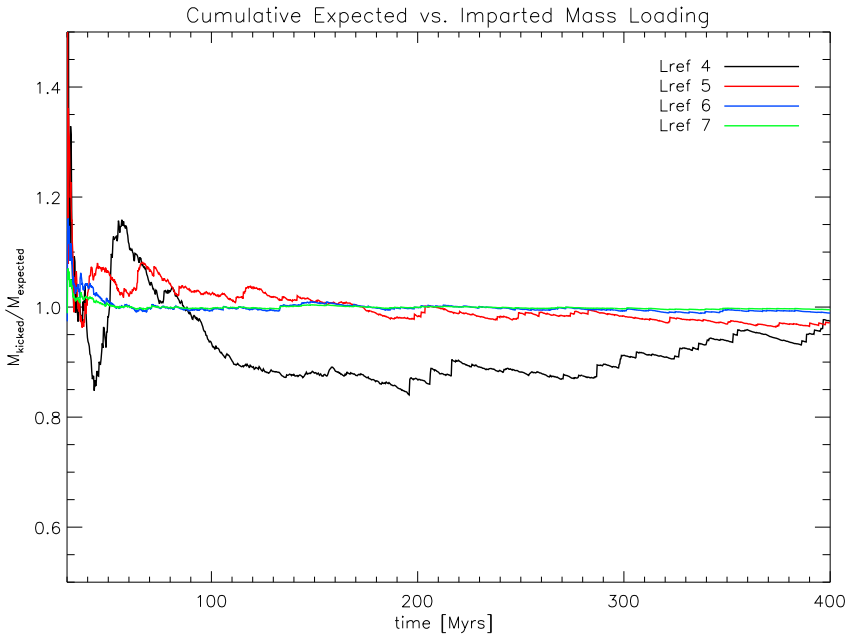

Figure 8.1: Ratio of the mass kicked to that expected against time for different levels of refinement in the *FLASH* code. The lines correspond to refinement levels/peak spatial resolutions; $lref4 \equiv$ 468pc (black), $lref5 \equiv$ 234pc (red), $lref6 \equiv$ 117pc (blue) and $lref7 \equiv$ 58pc (green). Initial deviations from the expected kicked mass are due to the stochastic nature with which particles are kicked. As sampling statistics increase the imparted mass loading quickly converges on the expected mass loading to within less than $\sim 1\%$ for refinement levels $lref6$ and above.

The star formation rate in the absence of feedback is slightly higher in *FLASH* compared to in the *GADGET-2* simulation as was previously observed in chapter 7. This could be partly due to the sharper density contrasts in an AMR code which allow the spiral arms to be thinner and slightly denser. As the star formation rate scales as $\dot{m}_* \propto \rho^{1.4}$ then this can boost the star formation rate. Both codes however use different polytropic indices and although the results are very similar, some small differences are not unexpected.

When feedback is included, the prescription in *GADGET-2* is able to reduce the final global star formation rate by around 50% whilst the same effective prescription in *FLASH* fails to make more than a 5% decrease. Despite the star formation rate and galaxy morphology in the absence of feedback being almost identical (see chapter 7), the efficiency of feedback is dramatically different. The kinetic feedback model in *FLASH* is evidently not as efficient at removing star forming gas from the disk as in the SPH simulation.

As I have already confirmed that the simulations achieve the correct mass loading and

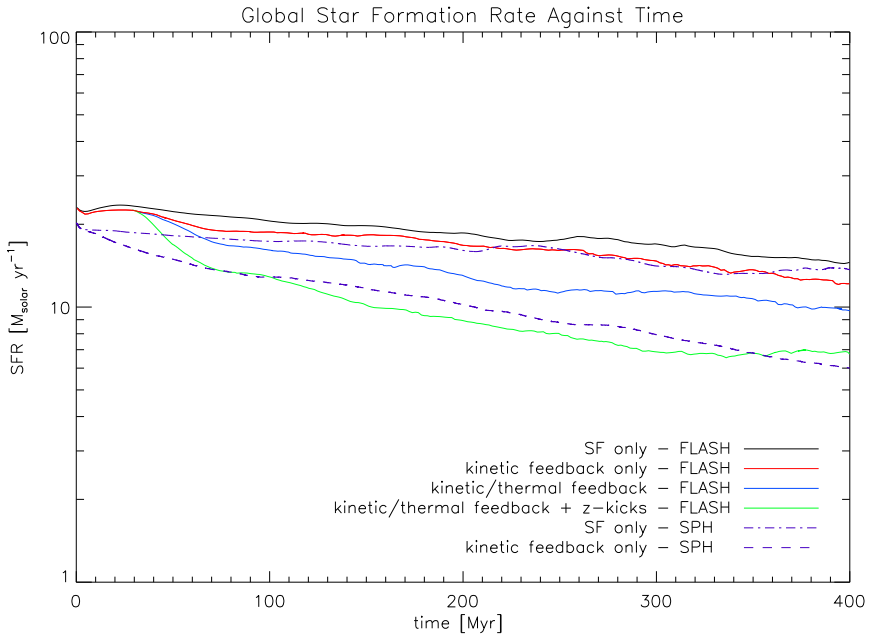

Figure 8.2: Global star formation rate as a function of time for different feedback models. The purple dot-dashed line corresponds to the *GADGET-2* simulation without feedback whilst the dashed purple line corresponds to the *GADGET-2* simulation with feedback. The red line corresponds to a near identical feedback prescription to that in *GADGET-2* implemented into *FLASH* and fails to show more than a 5% decrease in the star formation rate without feedback (black). The implementation of a thermal component to the model which allows the kinetic energy in the wind to thermalise is presented in blue. Although this improves the agreement with the SPH model, only when the cells are kicked orthogonal to the mid-plane do we fully recover the level of feedback efficiency seen in the SPH simulations.

that limitations to the box size do not adversely affect my results there must be another cause for the discrepancy.

8.6.3 Addition of Thermal Feedback

The blue line in figure 8.2 shows the resulting star formation as a function of time once the thermal component to feedback is included. The addition of the thermal component does allows supernova to blow hot under-dense bubbles in the outer regions of the disk, expelling gas for star formation. The central regions however are relatively unaffected due to their high densities which give them very short cooling time scales. After they have

been kicked off the effective equation of state, they are able to cool back onto it within a few million years. Thus despite the greater decline in the star formation rate between 30 and 70 Myrs relative to the SPH purely kinetic feedback configuration, the new "thermal feedback" still fails to recover the *GADGET-2* results. Overall *FLASH* now achieves a reduction in the global star formation rate of almost half that achieved in *GADGET-2*.

This is easiest explained by calculating the cooling time. I begin by considering one of the very dense cells in the centre of the disk with a density of order $n_H = 20$ cm^{-3}. If this cell is kicked with a velocity of $v_w = 600\,\mathrm{kms}^{-1}$ and subsequently collides with an equally dense cell, converting all of the energy into thermal energy, this will result in a temperature of

$$T = \frac{2}{3}\frac{\mu m_p v_w^2}{k_B} = 8.6 \times 10^6 \text{ K} \tag{8.11}$$

Using the look-up tables of Wiersma et al. (2009), a gas composed purely of hydrogen and helium at such a temperature is found to have a cooling function of the order $\Lambda \approx 10^{-23}$ erg s^{-1} cm^3. Such a primordial mixture of hydrogen and helium gives an electron density of

$n_e = 24.96\,\mathrm{cm}^{-3}$ and thus a cooling rate

$$\left|\dot{E}_{cool}\right| = \Lambda(T) n_e \approx 2.5 \times 10^{-22} \text{ erg s}^{-1} \tag{8.12}$$

This yields a cooling time (see Binney and Tremaine, 1987),

$$t_{cool} = \frac{E}{\left|\dot{E}_{cool}\right|} \tag{8.13}$$

where $E = \frac{3}{2}\frac{k_B T}{\mu}$ is the thermal energy of the electrons which we wish to radiate. This gives a cooling time $t_{cool} \approx 3.8 \times 10^5$ yrs. As the smallest simulation timestep is about 10^4 years, this gives a cooling time equivalent to only 24 timesteps. Clearly thermal feedback in such dense cells is inefficient with the thermal energy being radiated away over extremely short timescales before it has time to blow cavities and escape channels in the disk. This prevents gas from escaping from the disk and leads to material returning to the effective equation of state. As star formation resumes, these regions accrete more mass and become denser, making the situation progressively worse.

As energy within the dense central regions of the disk is far too easily dissipated, I try to maximise the efficiency of the feedback by giving all cells that are kicked, velocities away from the mid-plane of the disk (z-kicks). This allows all material to leave in the direction of least resistance, with successive kicks in the disk forming escape channels and

lifting layers of gas off. As the green line in figure 8.2 shows, the resulting star formation rate is comparable to that in *GADGET-2* (purple dashed line) to within around ±16% at all times.

The SPH code is able to produce a similar level of feedback without need for an additional thermal feedback component or escape trajectories that are maximised for efficiency. As these are designed to help prevent dissipation of energy within the dense regions of the disk, it would appear that somehow SPH particles experience dramatically less hydrodynamic resistance when passing through the gas compared to in an AMR code. Interestingly this hints at yet another fundamental difference in the way in which the hydrodynamic implementations differ and holds strong significance for simulations which incorporate sub-grid feedback mechanisms.

8.6.4 Feedback Sensitivity to Mass Loading & Wind Speed

I now test the sensitivity of my feedback model to the amount of energy injected. To do this I increase the energy injected through feedback by an unrealistic factor of four. This can be done by either increasing the mass loading to $\eta = 8$ or by doubling the wind velocity; $v_w = 1200\,\mathrm{kms}^{-1}$. The increased wind speed will have the effect of being able to heat the ISM to higher temperatures and is therefore more likely to drive bubbles in the ISM before it cools. The increased mass loading will result in a greater number of cells being kicked and should result in more material being ejected from the disk, especially around the outer regions of the disk where gas is less confined by dense neighbouring gas.

As can be seen in figure 8.3, the feedback model is exceptionally robust to the varying of the wind speed and mass loading parameters as well as the total energy input to the simulation as neither of the two models make a substantial difference to the global star formation rate. Although they both produce in a sharper drop between 30 and 50 Myrs into the simulation, they ultimately converge within ~ 100 Myrs to the same global star formation rate as for the fiducial wind speed and mass loading values.

Clearly energy is dissipated within the dense disk gas too easily for an increase in either the wind speed or mass loading to make a difference. I now investigate the effect of varying the galaxy mass on the efficiency of the feedback along with its effects upon the galaxy morphology in § 8.7.

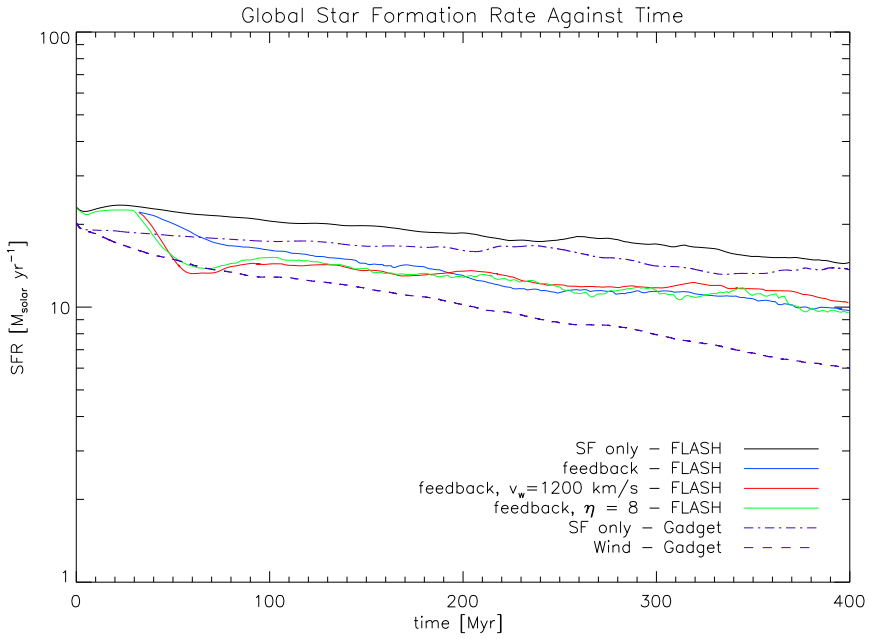

Figure 8.3: Global star formation rate as a function of time. The purple dot-dashed line corresponds to the *GADGET-2* simulation without feedback whilst the dashed purple line corresponds to the *GADGET-2* simulation with kinetic feedback. The blue line corresponds to the feedback model in *FLASH* that allows the winds to thermalise their kinetic energy. The other two lines correspond to runs with four times the injected supernova energy; $\eta = 2$, $v_w = 1200\,\mathrm{kms}^{-1}$ (red) and $\eta = 8$, $v_w = 600\,\mathrm{kms}^{-1}$ (green). The resulting star formation rate in *FLASH* is surprisingly robust to an increase in feedback energy and neither of the two runs are able to replicate the level of efficiency seen in the *GADGET-2* runs.

8.7 Comparison of Feedback in the 10^{10} and 10^{12} $\mathrm{M}_{\odot}$ Galaxies

In the previous section I showed that the kinetic feedback model including a thermal component and kicks orientated tangentially away from the mid-plane of the disk was able to reproduce the feedback efficiency of the SPH simulations in Dalla Vecchia and Schaye (2008). I now use this feedback model and extend the work to examine the effect of supernova feedback in different sized galaxies; a $10^{10}\ h^{-1}\ \mathrm{M}_{\odot}$ dwarf galaxy and a larger $10^{12}\ h^{-1}\ \mathrm{M}_{\odot}$ Milky Way like galaxy.

8.7.1 Morphological Effects of Feedback on the 10^{12} $M_\odot$ Galaxy

Figure 8.4 displays the edge-on (top) and face-on (bottom) logarithmic projected gas density for simulations (see table 8.1) Fm12ml2 (1^{st} column), Fm12nowind (2^{nd} second column), Gm12ml2 (3^{rd} column) and Gm12nowind (4^{th} column) at time t=250 Myr. All images are 45 h^{-1} kpc on a side. The SPH simulations where mapped to the mesh using the same software as was used to map the *GADGET-2* initial conditions to the AMR mesh. For both codes the disk galaxies with only cooling and star formation (Fm12nowind & Gm12nowind) show exceptional agreement both in terms of spiral structure and the peak densities reached. Some additional spiral structure is visible in *FLASH* (Fm12nowind) which tends to be smoothed out in *GADGET-2* (Gm12nowind), to be expected given the greater ability of AMR codes to resolve sharp density contrasts. The *FLASH* disk also appears to have become marginally thinner than that of the *GADGET-2* disk which can also be attributed to this characteristic property of the codes, especially since the disk is only a few cells/smoothing lengths thick.

In the presence of feedback (Fm12ml2 & Gm12ml2), both of the codes show an increased puffing up of the disk and a reduction in the densest regions. Whereas the individual spiral arms in *GADGET-2* are relatively stable in the presence of feedback, a more dramatic change in morphology is brought on in the *FLASH* code. As a result of feedback, smaller spiral arms are destroyed and a single set of dense spiral arms result. Such a dramatic morphological change brought on simply by changing the hydrodynamic implementation, has obvious implications for cosmological simulations.

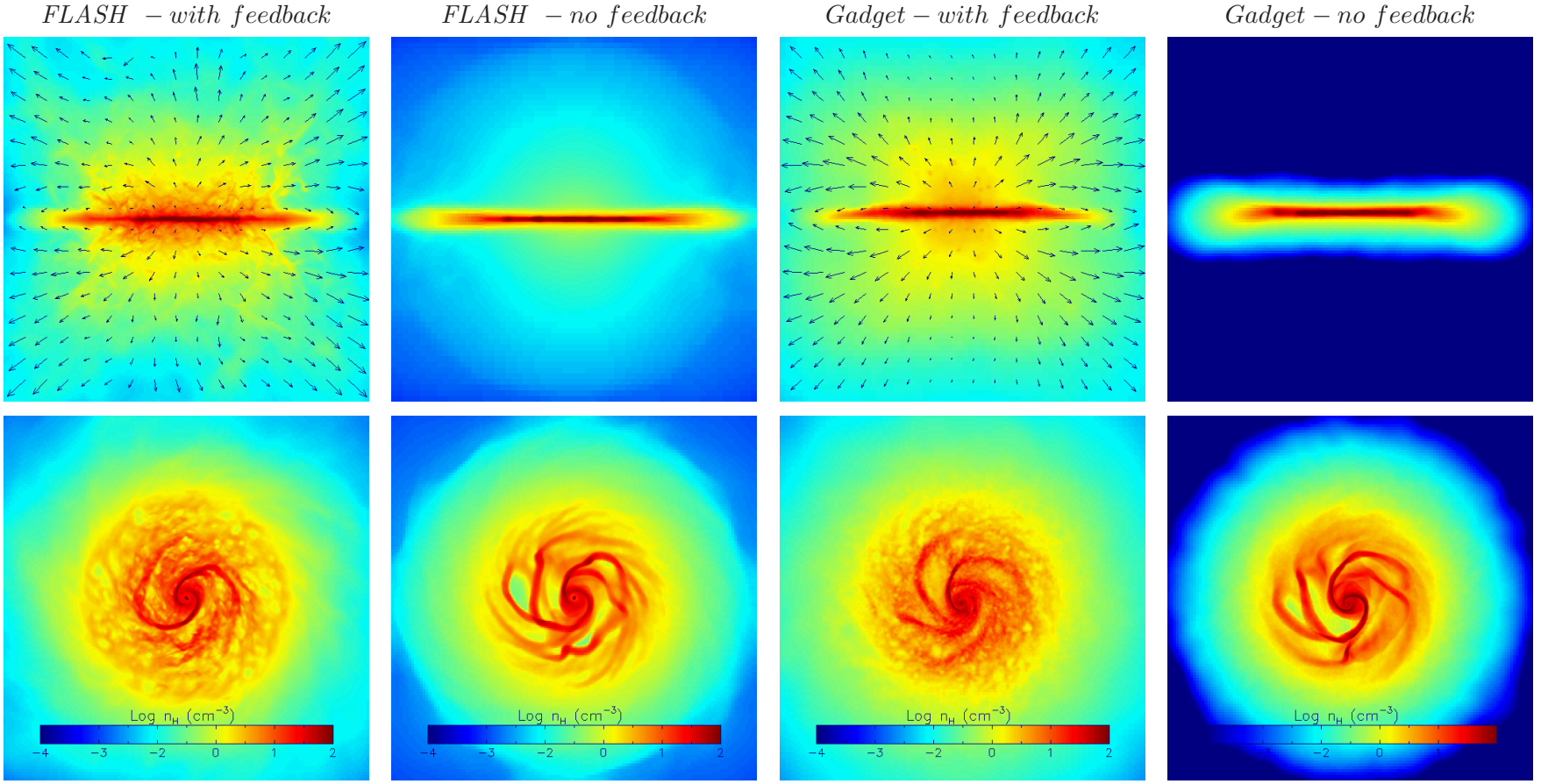

Figure 8.4: Edge-on (top) and face-on (bottom) logarithmic projected gas density for simulations Fm12ml2 (1^{st} column), Fm12nowind (2^{nd} column), Gm12ml2 (3^{rd} column) and Gm12nowind (4^{th} column) at time $t = 250$ Myr. All images use a common colour scheme for the density shown in the colour bars and are 45 h^{-1} kpc on a side. The arrows correspond to the mean projected gas velocity and have peak lengths of 108 kms^{-1} in Fm12ml2 and 292 kms^{-1} in Gm12ml2, with the *GADGET-2* simulation showing a much higher outflow velocity than in the AMR code for the same initial kick speed and mass loading.

8.7.1.1 Ram Pressure in the Gaseous Halo

The vertical distribution of gas above and below the disk is much lumpier in *FLASH* than the smooth halo visible in *GADGET-2*.

FLASH clearly displays a lot more structure with many arcs visible due to the combination of the z-axis kicks and the initial angular momentum of the gas in the disk. This creates a much denser gaseous halo immediately above the disk. As the build up of a dense halo could present resistance to outflowing material, constraining it to the disk and reducing the feedback efficiency, I compare the thermal pressure of the fluid, with the ram pressure of outflowing material

$$P_{ram} = \rho v_w^2 \quad . \tag{8.14}$$

Here ρ is the density of the medium and v_w is the velocity that the gas would be kicked at if we were to place it in the wind. In figure 8.5, the resulting thermal pressure (black) and ram pressure (blue) are plotted for a slice through the galactic core, tangential to the galactic plane at time $t = 250$ Myrs. There is at least an order of magnitude difference between the ram and thermal pressure of the ISM at all radii. As the density of the gas builds towards the galactic disk, its potential ram pressure should it be kicked, builds until the densest cells in the galactic core can present nine orders of magnitude more ram pressure than the ISM alone can resist through thermal pressure. I can therefore conclude that the accumulation of gas in the halo cannot prevent material from leaving the disk.

8.7.1.2 Outflow Geometry

The arrows in figure 8.4 correspond to the mean projected gas velocity and have peak lengths of 108 kms^{-1} in Fm12ml2 and 292 kms^{-1} in Gm12ml2. The *GADGET-2* simulations therefore show a much higher outflow velocity than in the AMR code, for the same initial kick speed and mass loading. The outflow pattern is fairly similar in both codes and apart from the reduced outflow in the plane of the disk, is otherwise isotropic. As the wind in *FLASH* is directed tangentially away from the mid-plane of the disk, the generation of an isotropic flow is somewhat unexpected. It develops as early ejecta slows down and disperses, presenting a denser halo around the centre of the disk, offering greater resistance to subsequent material that is blown out. As material continues to be ejected, it preferentially flows out along directions of least resistance. This increases the outflow angle, creating a more isotropic flow even though I impose tangential kicks away from the mid-plane of the disk and do not randomly orientate the supernova kicks.

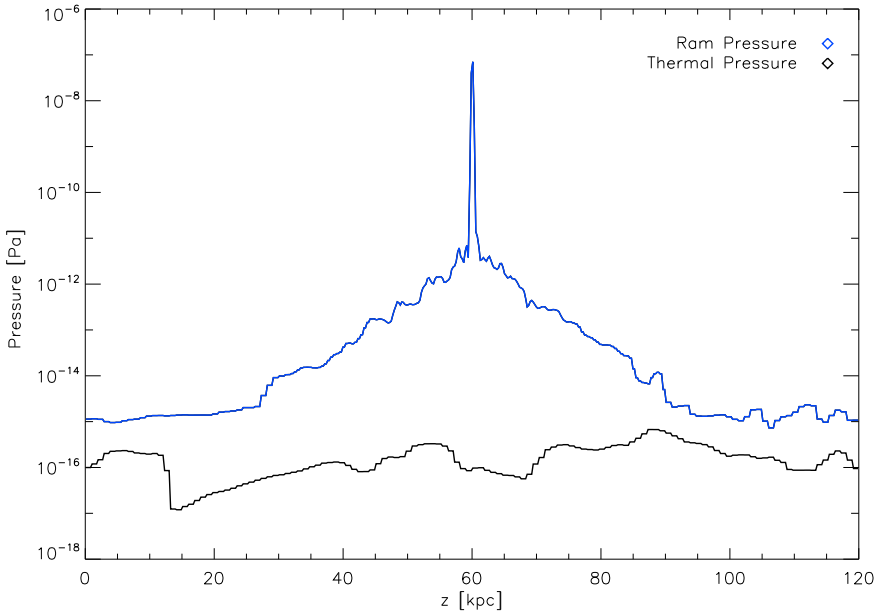

Figure 8.5: Pressure plot for a slice taken through the galactic centre, tangential to the mid-plane of the disk at $t = 250$ Myr. The thermal pressure (black) and ram pressure (blue) that gas would have if it was kicked through supernova feedback are shown. The ram pressure exceeds thermal pressure resistance by at least an order of magnitude at all distances from the galactic plane and by up to nine orders of magnitude in the galactic core. The wind cannot therefore be confined to the disk by the building up of the gas halo.

8.7.1.3 Global Star Formation History

Figure 8.6 shows the global star formation rate as a function of time in *GADGET-2* (left panel) and *FLASH* (right panel). The star formation rate in Fm12nowind initially increases to a peak value of $\sim 14\,\mathrm{M}_\odot\,\mathrm{yr}^{-1}$, slightly higher than the Gm12nowind *GADGET-2* simulation which reaches $\sim 12\,\mathrm{M}_\odot\,\mathrm{yr}^{-1}$. Like the simulations presented in chapter 7 and earlier in § 8.6.2, the ability of the AMR code to resolve sharper density contrasts leads to a slightly thinner, denser disk as is observed in figure 8.4. This results in a higher star formation rate than in *GADGET-2*.

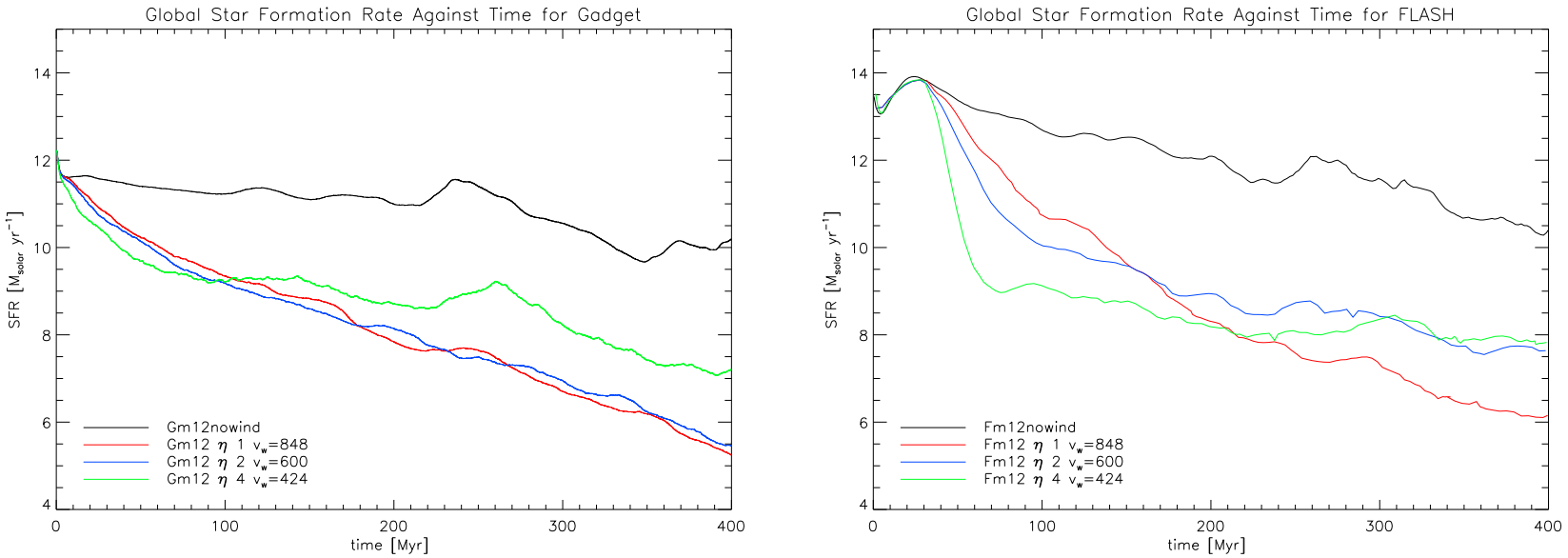

Figure 8.6: Global star formation rate as a function of time for the 10^{12} h^{-1} $M_{\odot}$ galaxy in *GADGET-2* (left panel) and *FLASH* (right panel). In the left panel, the simulations correspond to Gm12nowind (black), Gm12ml1 (red), Gm12ml2 (blue) and Gm12ml4 (green). In the right panel the simulations correspond to Fm12nowind (black), Fm12ml1 (red), Fm12ml2 (blue) and Fm12ml4 (green). Whilst both simulations Gm12ml1 and Gm12ml2 both converge upon a greatly reduced star formation rate, only the *FLASH* run with the greatest initial wind speed is able to match it (Fm12ml1), with the others agreeing well with the lowest wind speed *GADGET-2* run (Gm12ml4).

This increase in the star formation is then followed by a more rapid decline than in the SPH simulations. The sharper decline comes from the destruction of the spiral arms observed in § 8.7.1. Ultimately both codes give comparable global star formation rates by the end of the simulations.

In *GADGET-2* both Gm12ml1 and Gm12ml2 are able to heavily suppress the star formation and converge upon a final rate of around $5.5\,\mathrm{M}_{\odot}\,\mathrm{yr}^{-1}$. Only when using the highest wind velocity (Fm12ml1) is *FLASH* able to reach an equivalent feedback efficiency. The other two *FLASH* runs instead converge on the less efficient Gm12ml4 solution. This suggests that SPH particles are more freely able to exit the disk, their success directly linked to the initial wind speed.

The initial decline of the star formation in the *FLASH* models; Fm12ml2 and Fm12ml4 is much faster than in any of the *GADGET-2* simulations. This would suggest that initially the feedback operates more efficiently in *FLASH* but is then stalled at later times.

8.7.1.4 Disk Density Cross Section

Figure 8.7 displays the mean density along a line running orthogonally through the mid-plane of the disk for the default feedback simulations; Fm12ml2 in *FLASH* (black) and Gm12ml2 in *GADGET-2* (blue) at time $t = 125$ Myr. Both *FLASH* and *GADGET-2* attain the same peak density in the centre of the disk.

In the simulations which lack feedback, *FLASH* produces a slightly thinner disk than *GADGET-2*However in the presence of feedback the puffing up of the disk is more pronounced in *FLASH*, leading to a more gradual change between the disk and the halo gas. *GADGET-2* instead maintains a very thin disk with a well defined edge, implying that the feedback in *FLASH* seems to drag up more material out of the disk. As some of this material is still at densities high enough to form stars, it may be responsible for the observed excess star formation in *FLASH*.

A few kiloparsecs out from the disk, the density of the gas suddenly increases as we enter the halo formed by the outflow. The *GADGET-2* halo density now tails off slower over the next ~ 20 kpc than in the *FLASH* simulation until they both fall away to densities below $n_H = 0.01\ \mathrm{cm}^{-3}$.

As I have already highlighted how the thermal pressure of the halo gas is insufficient to counter the ram pressure of the wind (see figure 8.5), the only other means of stalling the outflow is using the much denser gas within the disk. This suggest that whilst material from the core of the *FLASH* disk attempts to pass out of the galaxy, it collides with other

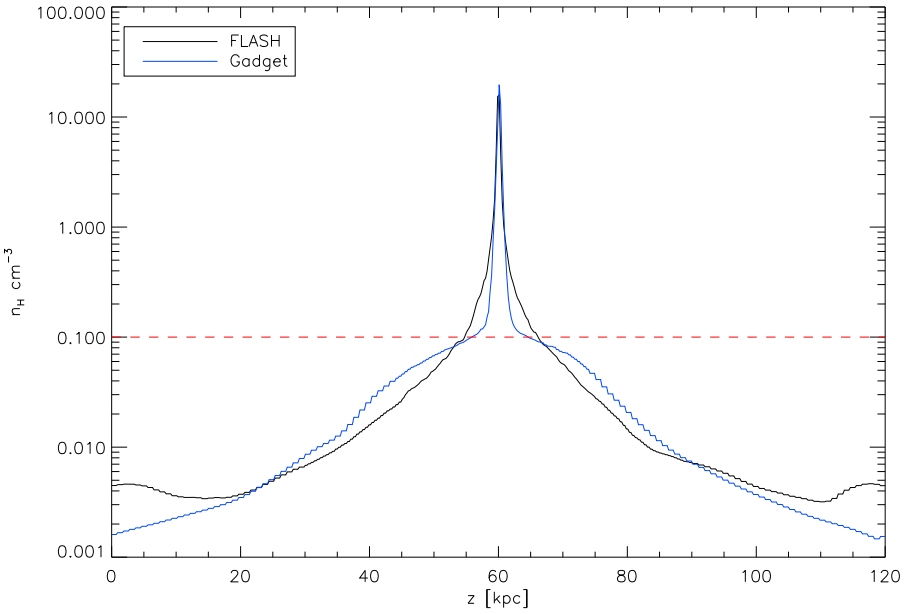

Figure 8.7: Mean density slice through the volume, orthogonal to the mid-plane of the disk in the 10^{12} $M_\odot$ galaxy. The default feedback simulations Fm12ml2 in *FLASH* (black) and Gm12ml2 in *GADGET-2* are plotted. Whereas the *GADGET-2* disk is well defined with a sharp density drop at the edges, the *FLASH* disk is puffed out to much greater radii resulting in a smoother transition between the disk and the halo. The star formation threshold of $n_H = 0.1$ cm^{-3} is marked as a red dashed line. Material above this is star forming.

gas on the effective equation of state above or below it. The collision then shock heats the gas off the effective equation of state and puffs up the outer surface of the disk. This leads to a smearing of the disk boundary and the formation of hot dense gas cells. As the cooling times for such cells are very small due to their high densities, they can then radiate this energy quickly over the space of a few million years. The cooled gas returns to the equation of state, resumes star formation and continues to present resistance to future outflows. This leaves a denser halo just around the disk that slowly puffs up as it is struck by further outflowing gas.

The SPH simulations clearly do not suffer from this effect and material on the effective equation of state does not appear to present the same resistance. Energy in the SPH implementation is clearly not dissipated as efficiently by particles as they are kicked out of the galaxy, suggesting that they act in a more ballistic fashion rather than coupling strongly to the hydrodynamics like in the AMR code.

8.7.2 Morphological Effects of Feedback on the 10^{10} $M_\odot$ Galaxy

Figure 8.8 displays edge-on (top) and face-on (bottom) logarithmic projected gas density for the 10^{10} $M_\odot$ galaxy simulations; Fm10ml2 (1^{st} column), Fm10nowind (2^{nd} second column), Gm10ml2 (3^{rd} column) and Gm10nowind (4^{th} column) at time $t = 250$ Myr. All images are 17.5 h^{-1} kpc on a side.

In the absence of feedback with only cooling and star formation (models Fm10nowind & Gm10nowind), the disks are almost indistinguishable apart from the marginally thinner disk in *FLASH* as was also observed for the $10^{12}M_\odot$ galaxy. This again shows firm agreement between the sub-grid polytropic equation of state and the star formation routines in *FLASH* and *GADGET-2*.

In the presence of feedback, both *FLASH* and *GADGET-2* experience extensive disruption of the disk. Despite more small scale structure being visible in the *GADGET-2* simulation due to the higher mass resolution, both codes produce very similar morphologies.

8.7.2.1 Outflow Geometry

The arrows in figure 8.8 correspond to the mean projected gas velocity and have peak lengths of 241 kms^{-1} in Fm10ml2 and 244 kms^{-1} in Gm10ml2, giving the two codes very comparable outflow velocities. Unlike for the larger mass galaxy, the winds are able to efficiently remove material from the disk in both codes and in particular a series of arcs from material that has been lifted off the disk are visible in the *GADGET-2* simulation. As the wind is more able to escape the gravitational potential of the galaxy there is substantially less buildup of a gaseous halo. Because of this the outflow in *FLASH* remains strongly bipolar with little wind leaving at very large opening angles. The *GADGET-2* simulation however, like for the 10^{12} $M_\odot$ galaxy shows a fairly isotropic outflow since it uses randomly orientated wind trajectories.

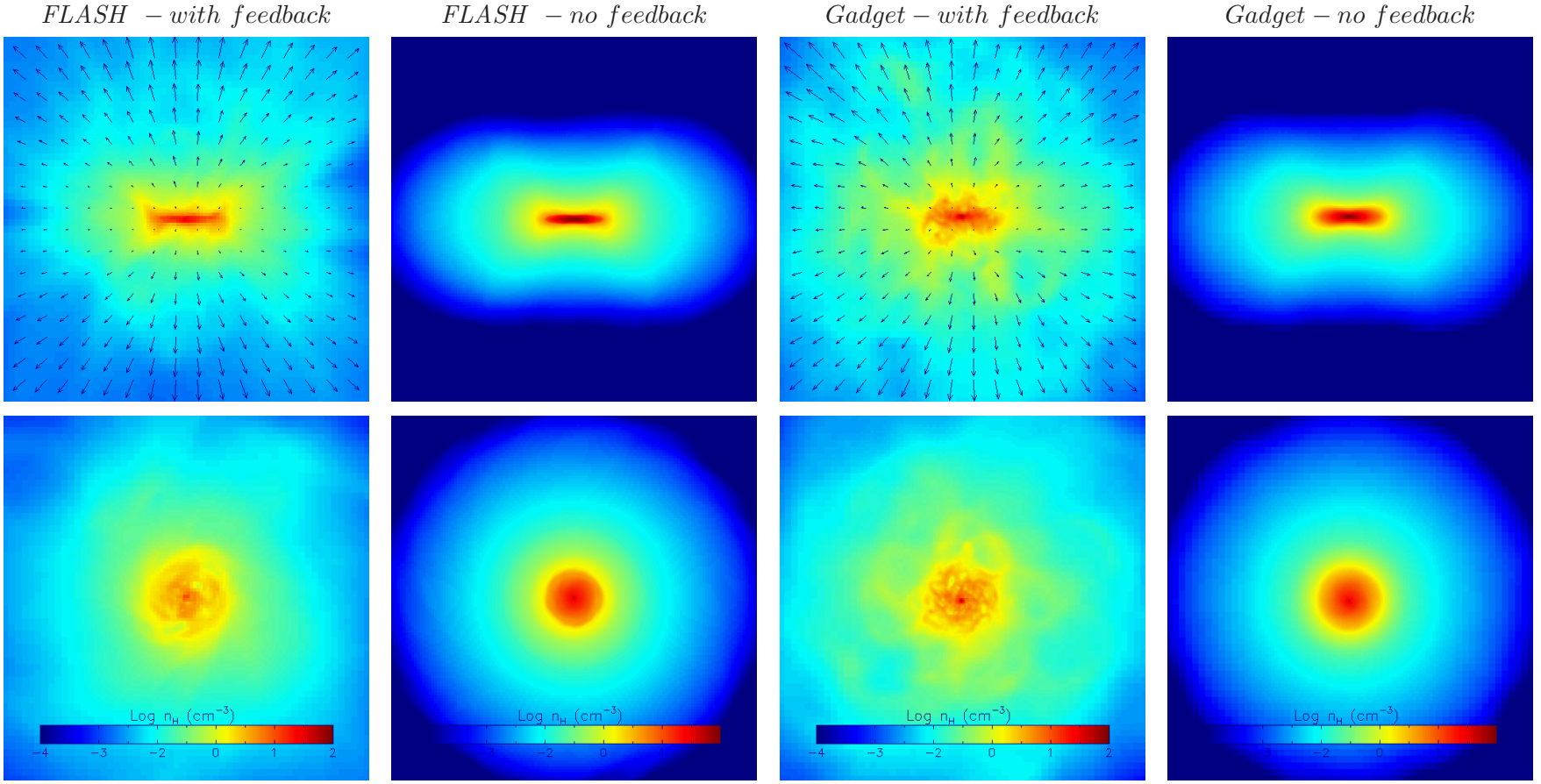

Figure 8.8: Edge-on (top) and face-on (bottom) logarithmic projected gas density for simulations Fm10ml2 (1^{st} column), Fm10nowind (2^{nd} column), Gm10ml2 (3^{rd} column) and Gm10nowind (4^{th} column) at time $t = 250$ Myr. All images use a common colour scheme for the density shown in the colour bars and are 17.5 h^{-1} kpc on a side. The arrows correspond to the mean projected gas velocity and have peak lengths of 241 kms^{-1} in Fm10ml2 and 244 kms^{-1} in Gm10ml2 with both codes showing comparable wind speeds for the same initial kick speed and mass loading.

8.7.2.2 Global Star Formation History

Figure 8.9 shows the global star formation rate for the 10^{10} h^{-1} $M_{\odot}$ dwarf galaxy in *GADGET-2* (left panel) and *FLASH* (right panel). All of the *GADGET-2* simulations which feature feedback have converged to give a single star formation history with no defining difference between the effects of increased mass loading or wind speed. *FLASH* on the other hand displays a heavy dependency upon the mass loading. As the mass loading increases, so too does the efficiency of the supernova feedback with a mass loading of two (Fm10ml2) actually achieving a good fit with the SPH data and only a marginally lower final star formation rate. Only two *FLASH* runs are shown, as when a mass loading of four is used, the disk is completely disrupted - a result which was too numerically unstable to run. This would suggest that in less dense structures, the feedback in *FLASH* is heavily dependent upon the mass loading, with greater mass loading resulting in greater mass loss. Evidently, unlike the SPH code, high wind velocities do not tend to drag up as much material and force the same amount of gas out of the galaxy as lower wind velocity runs with higher mass loading.

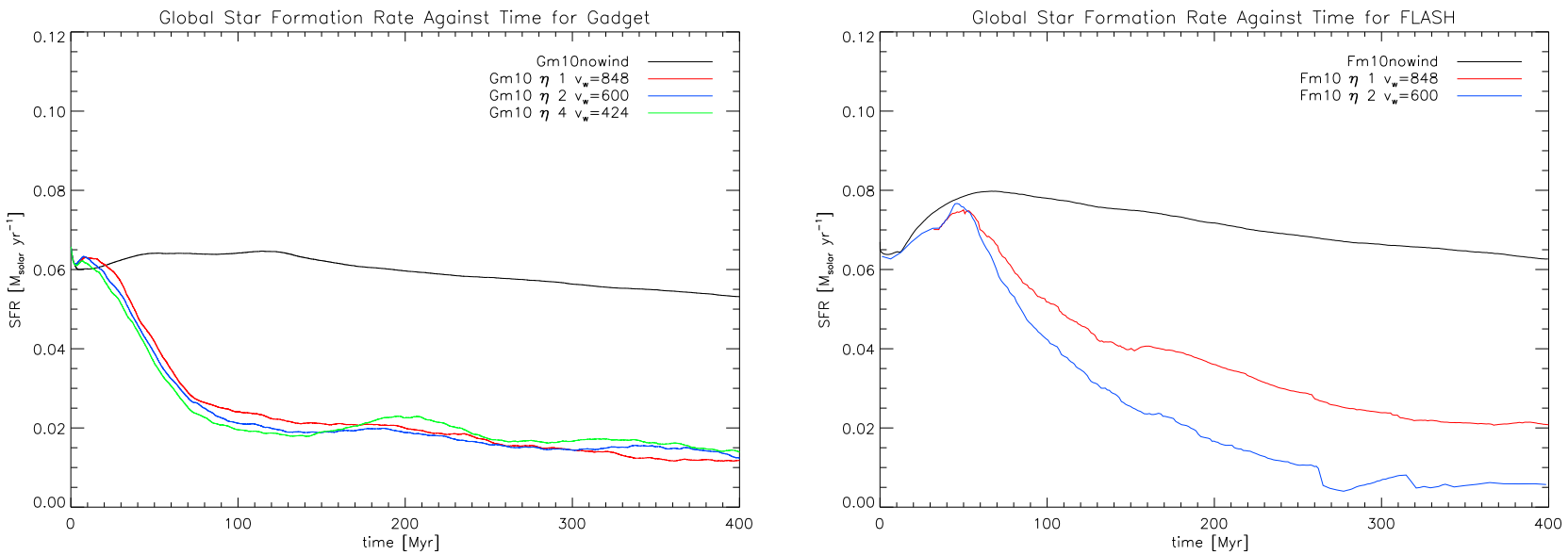

Figure 8.9: Global star formation rate as a function of time for the 10^{10} h^{-1} $M_{\odot}$ dwarf galaxy in *GADGET-2* (right panel) and *FLASH* (left panel). In the left panel, the simulations correspond to Gm10nowind (black), Gm10ml1 (red), Gm10ml2 (blue) and Gm10ml4 (green). In the right panel the simulations correspond to Fm10nowind (black), Fm10ml1 (red) and Fm10ml2 (blue). Whilst all *GADGET-2* simulations converge upon a single solution, as the mass loading is increased in the *FLASH* simulations, the efficiency of the feedback increases drastically. The *FLASH* simulation with $\eta = 4$ proved to completely disrupt the disk, the result of which was too numerically unstable to run. Clearly the *FLASH* feedback prescription is much more efficient at removing gas in less tightly bound structures.

8.7.2.3 Disk Density Cross Section

Figure 8.10 shows the mean gas density along a line running orthogonally through the mid-plane of the 10^{10} $M_{\odot}$ disk galaxy for the default feedback simulations; Fm10ml2 in *FLASH* (black) and Gm10ml2 in *GADGET-2* (blue) at $t = 125$ Myrs. Again both simulations reach the same peak density in the galactic core but unlike the higher mass galaxy, now *FLASH* possesses a thinner galactic disk with less evidence for the buildup of a dense gaseous halo. Instead the disk in *GADGET-2* is slightly more puffed up.

This time both codes have relatively identical amounts of material above the star forming threshold (red dashed line) yielding similar star formation rates. Clearly *FLASH* no longer has trouble expelling material from around the disk and the wind is free to travel out to larger radii, escaping the disk.

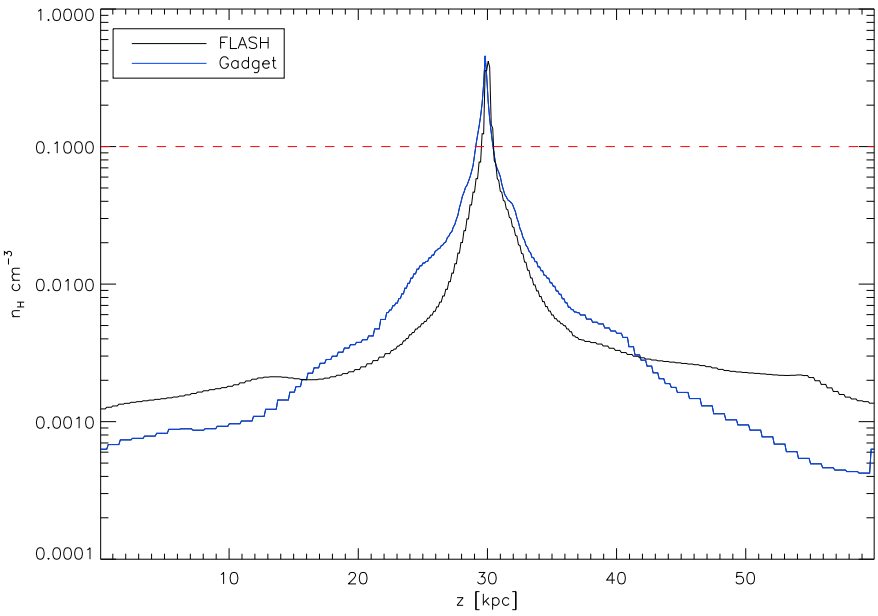

Figure 8.10: Mean density slice through the volume, orthogonal to the mid-plane of the disk in the 10^{10} $M_{\odot}$ galaxy. The default feedback simulations Fm10ml2 in *FLASH* (black) and Gm10ml2 in *GADGET-2* are plotted. Whereas the *GADGET-2* disk is well defined with a sharp density drop at the edges, the *FLASH* disk is puffed out to much greater radii resulting in a smoother transition between the disk and the halo. As some of this material is above the star formation threshold of $n_H = 0.1$ cm^{-3}, marked as a red dashed line, this leads to a higher global star formation rate in *FLASH*.

8.7.3 Mass Outflow Rates and Wind Velocity

Finally I conclude this chapter with a comparison of the input initial wind speed and effective mass loading to their actual values as realised in the *FLASH* and *GADGET-2* simulations. I do this for both the 10^{10} and 10^{12} h^{-1} $M_{\odot}$ galaxies (Fm12ml2, Fm10ml2, Gm12ml2, Gm10ml2).

Following the method of analysis used by Dalla Vecchia and Schaye (2008), they define the net mass outflow rate through a surface S as

$$\dot{M} = \int_S \rho \mathbf{v} \cdot \mathbf{dS} \tag{8.15}$$

where ρ is the gas density and $\mathbf{v}$ is the velocity relative to the centre of the galaxy. The outflow and inflow can be differentiated according to the sign of the product $\mathbf{v} \cdot \mathbf{r}$. The net mass outflow rate is then the sum of both components. Descretizing the above equation gives

$$\dot{M}(r, \Delta r) = \frac{1}{\Delta r} \sum_{i=1}^{N_{shell}} m_i \mathbf{v}_i \cdot \frac{\mathbf{r}_i}{r_i} \tag{8.16}$$

where Δr is the thickness of a spherical shell of radius r centred on the origin (in line with Dalla Vecchia and Schaye (2008) I use $\Delta r = r_{vir}/150$) and N_{shell}, m_i and $\mathbf{r_i}$ are the total number, mass and position of the particles/cells within that shell.

The average outflow velocity is the mass-weighted radial velocity where only particles moving away from the disk (i.e. $\mathbf{v} \cdot \mathbf{r} > 0$) are included:

$$\langle v \rangle = \frac{\sum_{i=1}^{N_{shell}} m_i \left(\mathbf{v}_i \cdot \frac{\mathbf{r}_i}{r_i} \right)_+}{\sum_{i=1}^{N_{shell}} m_i} \tag{8.17}$$

where the subscript $+$ is used to indicate that only outgoing particles are taken into account.

8.7.3.1 Mass Outflow Rates in the $10^{12}\,M_{\odot}$ Galaxy

The top left of figure 8.11 show the net mass outflow as a function of time measured through a spherical shell at radius $r = 0.2 r_{virial}$. The bottom left panel shows the mean outflow velocity as a function of time for the same spherical shell. The top right panel shows the net mass outflow as a function of radius at time $t = 400$ Myr, whilst the bottom right shows the mean outflow velocity as a function of radius. The red lines correspond to *FLASH* whilst the black lines are for *GADGET-2*.

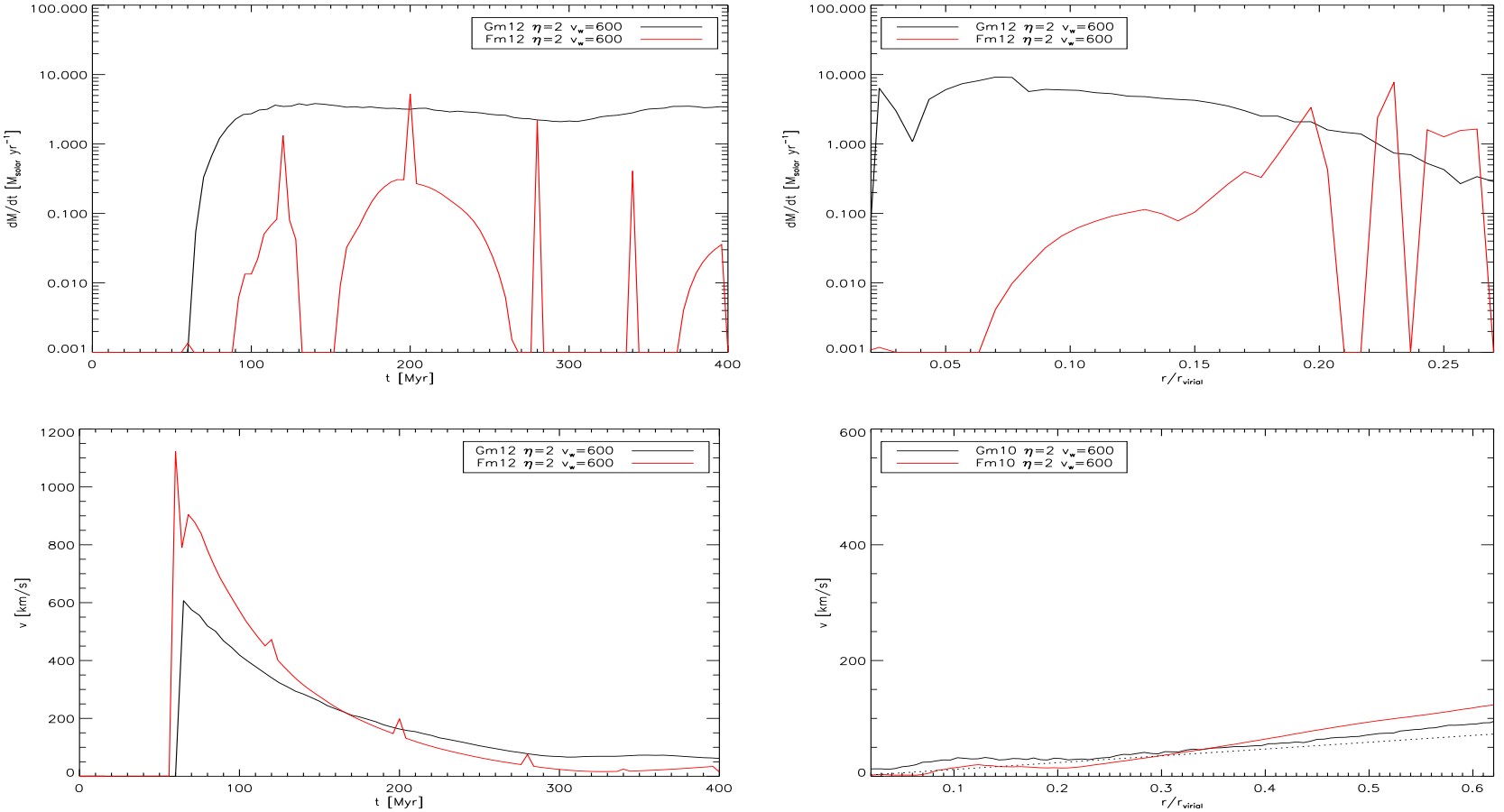

Figure 8.11: The left panels show the net outflow mass (top) and outflow velocity (bottom) as a function of time for the 10^{12} h^{-1} $M_{\odot}$ galaxy measured through a spherical shell at radius $r = 0.2r_{virial}$. The right hand panels show the net outflow mass (top) and outflow velocity (bottom) as a function of radius at time $t = 400$ Myr. The black lines correspond to *GADGET-2* and the red lines correspond to *FLASH*.

First impressions clearly show that the mass outflow is much lower in *FLASH* at all times and radii, than in *GADGET-2*. Whilst the net mass outflow in *GADGET-2* increases abruptly as feedback takes hold, plateauing at $\sim 3\,\mathrm{M}_{\odot}\,\mathrm{yr}^{-1}$ for the remainder of the simulation, *FLASH* shows a more variable net mass outflow. With an initial "blip" in the net mass outflow appearing at the same time as the *GADGET-2* outflow begins, there is no indication of any substantial net mass outflow until around 90 Myrs into the simulation after which the net outflow increases and falls repeatedly, each cycle separated by a space of no net mass outflow. These spikes in the *FLASH* net mass outflow result from the lumpy nature of the outflow, with *FLASH* tending to kick fewer cells that are generally more massive whilst the SPH approach kicks many lower mass particles more frequently. The drops in outflow show up in the top right panel as a series of sudden dips in the net mass outflow as a function of radius when the outflow of mass temporarily stopped. Presumably this delay in the onset of the outflow despite the kicking of material, is due to the need for the wind in *FLASH* to build up escape channels. A few sharp peaks exist for *FLASH* which may be the result of particularly dense cells near the edge of the disk being kicked, leading to a short lived burst in the outflow rate.

The outflow velocity in *GADGET-2* begins equal to the input wind speed as there is no gas present in the halo to resist it. It then declines over the duration of the simulation as the halo begins to build up, offering more resistance to outflowing material. The higher than input wind speed observed in *FLASH* at around 60 Myrs is due to the aforementioned fact that cells can occasionally be kicked twice (see § 8.5.4). I note that this effect quickly reduces as time progresses until the mean *FLASH* outflow velocity drops below that in *GADGET-2*, presumably as the halo in *FLASH* presents more resistance to outflowing gas than in *GADGET-2*. The substantial decline in the wind velocity over time in both codes highlights the transfer of kinetic energy as extra material is swept up in the wind, increasing the effective mass loading beyond the input parameters.

Critically, even though some cells are initially kicked around $t = 60$ Myrs with twice the input wind speed, the mass outflow in *FLASH* barely registers until 90 Myrs into the simulation indicating that the wind in *FLASH* is subject to greater resistance.

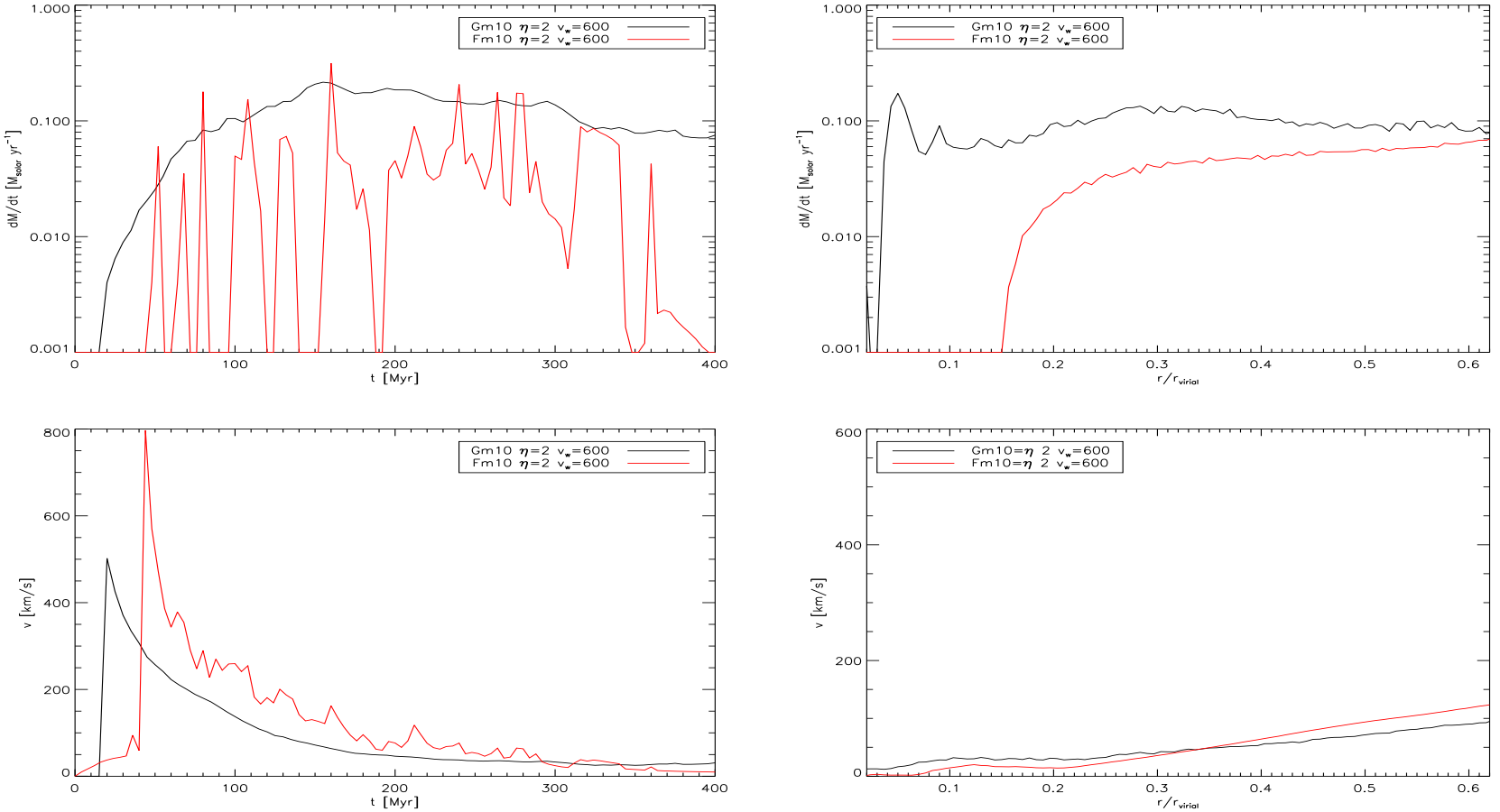

Figure 8.12: The left panels show the net outflow mass (top) and outflow velocity (bottom) as a function of time for the $10^{10}\,h^{-1}\,\mathrm{M}_{\odot}$ galaxy measured through a spherical shell at radius $r = 0.2r_{virial}$. The right hand panels show the net outflow mass (top) and outflow velocity (bottom) as a function of radius at time $t = 400$ Myr. The black lines correspond to *GADGET-2* and the red lines correspond to *FLASH*.

8.7.3.2 Mass Outflow Rates in the $10^{10}\,\mathrm{M}_{\odot}$ Galaxy

Figure 8.12 shows the equivalent set of plots for the $10^{10}\,h^{-1}\,\mathrm{M}_{\odot}$ galaxy. The top left panel shows the net mass outflow as a function of time measured through a spherical shell at radius $r = 0.2r_{virial}$. The bottom left panel shows the mean outflow velocity as a function of time for the same spherical shell. The top right panel shows the net mass outflow as a function of radius at time $t = 400$ Myr, whilst the bottom right shows the mean outflow velocity as a function of radius. Again the red lines correspond to *FLASH* whilst the black lines are for *GADGET-2*.

Like the higher mass galaxy, the net mass outflow in *FLASH* for the dwarf galaxy increases slower than in *GADGET-2* with a slight delay of around $40 - 50$ Myrs before feedback is able to initiate an outflow. From this point onwards the net outflow rate regularly rises and then falls. Despite the troughs in the outflow rate, the peaks are in good agreement between *FLASH* and *GADGET-2*. This is particularly clear when looking at the net outflow rate at large radii, where the outflow rates in the two codes converge (see top right panel). Temporary drops in the net mass outflow rate lead to a total outflow that is lower in *FLASH* than *GADGET-2*. However feedback in *FLASH* must still be sufficient to disturb the dense star forming regions within the disk in order to achieve the same reduction in star formation as achieved in *GADGET-2* (see figure 8.9).

At late times whilst the net mass outflow rate remains constant in *GADGET-2*, it plummets in *FLASH*. This can clearly be seen in both the top left panel and at small radii in the top right panel (since the most recently kicked gas will not have had time to travel far). This drop could correspond to the point at which the least dense gas around the galaxy has efficiently been stripped away, leaving just the dense core. Whilst *GADGET-2* is able to continue removing gas from this dense core, the feedback efficiency in *FLASH* is reduced due to the same mechanisms present in the higher mass galaxy.

Again the actual outflow velocity in *FLASH* is initially greater than the input wind speed due to the potential ability to kick cells twice. It continues to remain marginally higher than in *GADGET-2* ($\sim +25\,\mathrm{kms}^{-1}$) until late times. Both codes show a clear decline in the mean outflow velocity as the halo puffs up - presenting more resistance to outflowing material.

The results presented here show some interesting agreement with the original simulations of Dalla Vecchia and Schaye (2008). Clearly the build up of an outflow fuelled gaseous halo leads to a reduction in the actual outflow velocity even though cells receive

the same kicks. This clearly highlights how the mass loading of the wind increases above the input parameter with the wind sweeping up mass as it travels out through the disk.

At times both codes show similar peak net mass outflow rates showing good agreement with the general feedback implementation. However, unlike in *GADGET-2* where the outflow rate remains reasonably constant over time for both galaxies, in *FLASH* the net outflow periodically increases and then falls. This is presumably due to redistribution of mass by feedback which later presents renewed resistance to material trying to leave the disk.

The periods of net mass outflow in *FLASH* are generally shorter lived in the higher mass galaxy, but the expected mass loading of the wind was exactly in line with expectations (see § 8.6.1). This would therefore suggest that denser gas within the disk presents greater resistance in AMR codes than in SPH codes.

8.8 Discussion

The implications of the much lower feedback efficiencies in *FLASH* relative to *GADGET-2* are far reaching. Although the SPH *GIMIC* and *OWLS* simulations use a polytropic index of 4/3 whilst the AMR *Horizon-Mare Nostrum* simulation uses an index of 5/3, consistent with my setup, the feedback strengths and efficiencies may not be directly comparable.

Tentative signs from the wider community such as the work of Wada and Norman (2007), would appear to agree with my results. They find that their AMR code demonstrates star formation rates that are relatively unaffected by the onset of even very high levels of feedback. In order to drive efficient feedback some authors including Mori et al. (1997); Thacker and Couchman (2000); Brook et al. (2004); Stinson et al. (2006); Agertz et al. (2010) have opted to switch off cooling in regions with feedback in order to allow an adiabatic expansion wave to be driven. Given the efficiency with which *FLASH* was able to dissipate the injected feedback energy, this would provide a useful although not necessarily physical fix to combat this effect and produce a more efficient feedback capable of reproducing observations.

When resolving smaller non-cosmological volumes such as individual galaxies however, it becomes possible to use more sophisticated models for the ISM with Springel and Hernquist (2003) developing a multi-phase model in *GADGET-2*. By decoupling the cold dense molecular clouds from the hot tenuous ISM component, the hot gas can more easily pass out away from the galaxy without feeling the resistance presented by the molecular

cloud component that is felt in a single phase medium. Booth et al. (2007) and Giovalli (2009) have extended this work to allow for the creation of "sticky particles" which enable molecular clouds to grow through both coagulation during collisions, quiescent accretion and evaporate as a result of energy release from supernova. As such models remove the need for a polytropic equation of state they can provide for a more detailed understanding of the physics governing the ISM. Booth et al. (2007) has shown that for high resolution galaxy simulations their model accurately reproduce the observed Kennicutt-Schmidt law, molecular cloud mass function and stellar density profile. These models also allow for more complex behaviour such as the continued accretion of mass onto the galactic core whilst hot gas is expelled out away from the galaxy - a property which cannot be followed with a single phase medium which has only one direction of flow.

8.9 Conclusion

I have performed identical disk galaxy simulations with both a 10^{12} and $10^{10}\,h^{-1}\,\mathrm{M}_{\odot}$ galaxy using a near identical sub-grid model for supernova feedback to that used by Dalla Vecchia and Schaye (2008).

As part of my efforts I have developed and tested series of feedback models. The first incorporates only pure kinetic feedback in identical fashion to that used by Dalla Vecchia and Schaye (2008) in *GADGET-2*. When tested on the same $10^{12}\,\mathrm{M}_{\odot}$ galaxy as used in chapter 7, I find that the feedback has almost no effect upon the global star formation rate of the galaxy. This is due to the dissipation of the kinetic energy in the wind when it collides with the gas in other cells on the effective equation of state. As the effective equation of state does not explicitly allow gas to shock, the energy that would normally be thermalised in shocks is lost. This prevents supernova driven winds from blowing bubbles in the ISM and leads to ineffective feedback. To overcome this I allow gas on the effective equation of state to shock heat if its temperature exceeds its effective equation of state temperature. Although gas can now shock heat off of the polytropic equation of state, it still only achieves half the reduction in the global star formation rate achieved in *GADGET-2* with the pure kinetic feedback model.

When dense cells in the centre of the galactic disk are kicked, they collide with other dense neighbouring cells around them and shock heat, converting the kinetic energy into thermal energy. As these cells are so dense they have very short cooling timescales and typically radiate the thermal energy away over only ~ 24 timesteps. This prevents the

gas from having time to blow cavities in the dense regions of the disk and the gas can resume star formation.

In my third model, in order to maximise the efficiency of my feedback, I orientate all supernova kicks away from the mid-plane of the galactic disk. This ensures that gas leaves through the route of least resistance and reduces the likelihood of dense gas cells colliding. I find that such an approach is able to mimic the feedback efficiency observed in *GADGET-2*. Thus both the ability to effectively thermalise wind energy and an optimal escape trajectory are required to reproduce the SPH results.

I test the robustness of my feedback model by injecting four times as much energy by either quadrupling the mass loading or doubling the input wind speed. I find that despite a slightly sharper initial drop in the star formation rate when feedback first becomes active, the global star formation rate converges back on the same profile for the default feedback parameters. Clearly the model is extremely robust to changes in the model parameters as the dissipation of energy in dense gas is far too efficient.

I have ensured that the kicking of gas cells in the AMR approach is able to reproduce the expected mass loading of the wind to within less than a percent at all times, giving feedback levels that should be in good agreement with those in the SPH.

From analysis of the actual net mass outflow rates and the realised outflow velocity in both codes, using identical input parameters, it has become clear that the SPH code is able to remove more mass and achieve less deceleration of the wind than in *FLASH*.

The actual outflow velocities relative to the input wind speeds in both codes decrease with time as expelled gas builds up a dense halo. This halo then presents increased resistance to future mass ejecta. Due to the hydrodynamic coupling of the wind this leads to an effective mass loading which is higher than the input level as the wind sweeps up material.

I follow up my initial feedback simulations with an investigation of the effect of differing the galaxy mass, using two galaxies; a $10^{12}\,h^{-1}\,\mathrm{M}_{\odot}$ Milky Way like galaxy and a $10^{10}\,h^{-1}\,\mathrm{M}_{\odot}$ dwarf galaxy. The net mass outflow rates in *GADGET-2* remain fairly constant with time. Those in *FLASH* however are prone to rise and fall over time interspersed by periods of a few million years during which there is no net outflow of mass. These periods of net mass outflow in *FLASH* are shorter lived the in higher mass galaxy than in the dwarf, suggesting that increasing density of the disk gas leads to an increased level of resistance. This does not appear to be an issue in *GADGET-2* and the wind is able to effectively leave the disk regardless of the gas density in which it is situated.

The galactic disk in the *FLASH* Milky Way like simulations remains thinner than in the SPH simulations as dense gas in the core is unable to escape. The lower mass galaxy however is able to greatly puff the disk up and expel much more gas, in line with the *GADGET-2* simulations. The higher mass *FLASH* galaxy simulation demonstrates a much more clumpy halo with substantially more structure being visible. Even though I orientate supernova kicks tangentially away from the disk, the build up of a gaseous halo still causes the outflow to move out along directions of least resistance. This creates a more isotropic outflow, similar to in *GADGET-2* where the kicks are randomly orientated. In the lower mass galaxy, the halo presents less resistance to the outflow and thus a more bi-polar outflow develops instead whereas the SPH outflow remains isotropic. To ensure that the buildup of the clumpy halo in *FLASH* does not affect my results, I confirm that the gaseous halo does not offer resistance to the ram pressure of outflowing material. The smaller spiral arms in the higher mass *FLASH* galaxy are destroyed early on by the feedback leaving a double spiral armed galaxy whilst they remained intact in the *GADGET-2* simulations.

These effects show that the wind experiences greater hydrodynamic resistance in the AMR code compared to in SPH, especially in very dense central regions. Whilst SPH particles appear to be able to stream out more like ballistic particles which we confirm are actually coupled to the hydrodynamic, the wind in *FLASH* is quickly stalled. In the absence of feedback however the two codes agree remarkably well.

This clearly holds substantial consequences for current and future galaxy formation simulations. I was able to change the galaxy morphology from a multi-spiral armed galaxy in *GADGET-2* to a denser double armed spiral in *FLASH* simply by changing the hydrodynamic code adopted. This throws doubt upon whether observed differences between the results from different theory groups are as a result of the different physics they implement or through fundamental differences in the hydrodynamic implementation. The results I have presented here unfortunately do not allow us to determine which code gives a more physically reasonable description. On one hand, it is acknowledged that SPH can suppress effects such as the formation of hydrodynamic instabilities (Agertz et al., 2007), turbulent mixing (see, e.g., chapter 6, Dolag et al., 2005) and in the absence of sufficient resolution can smooth over shocks (see, e.g., chapter 4). This could imply that the higher hydrodynamic resistance in *FLASH* is more physical. It does seem reasonable that the wind should need to blow bubbles in the disk, forming cavities and channels from which gas can then escape before feedback will take hold. This however results in

a less efficient feedback mechanism as much of the feedback energy is lost puffing up the disk and thus like other AMR galaxy formation simulations, is unable to prevent over production of stars in massive galaxies (see also, e.g. Tonnesen et al., 2007).

The feedback in *GADGET-2* however despite experiencing less hydrodynamic resistance, does accomplish the initial goal of being able to efficiently reduce the global star formation rate of the galaxy. It also leads to puffing up of the disk and the formation of a gaseous halo. To obtain the same level of efficiency in an AMR code would require either physically unreasonable energy input to the ISM by increasing the input mass loading or wind velocity or the creation of a multi-phase ISM model where the hot and cold phases of the ISM are decoupled and evolved separately. As § 8.6.4 has shown that the resistance within the AMR code is extremely robust against such increases, such an approach is unlikely to produce more effective feedback.

The "wispy" halo formed in *FLASH* is undoubtedly one of the more visible differences between the two codes. As the fluid in an AMR code is not confined to discrete mass units as in an SPH code, it is free to disperse as it leaves the disk and mix freely. This provides a vital means for redistributing mass and metals should stellar enrichment be included (not present in these simulations). The *GADGET-2* halo instead is more sparse, with the wind mass being confined to wind particles which freely leave the vicinity of the disk. This may result in SPH simulations producing regions of metal over abundance as they are unable to properly disperse them.

Ultimately the best way to resolve such a problem lies either in being able to resolve the true multi-phase structure of the ISM and comparing the results between the two recipes or through direct comparison to observation. This could be accomplished by taking the SPH results in the *OWLS* simulations and re-simulating them in the *FLASH* AMR code. As the SPH results are already available this presents an more feasible approach until hardware and algorithms capable of dealing with the multi-phase ISM modelling approach become available.

Chapter 9 | *Conclusions*

9.1 Discussion

9.1.1 Towards Understanding Numerical Simulations of Galaxy Formation

Numerical simulations present a fundamental building block of our modern theoretical understanding of the Universe. Recent work by authors such as Frenk et al. (1999); Agertz et al. (2007) have highlighted fundamental differences between the two fundamentally different types of hydrodynamic codes; SPH and AMR. As small differences in the behaviour of these numerical implementations can lead to substantial deviations in cosmological simulations, I focus the work in this thesis on trying to highlight and understand the fundamental differences between these hydrodynamic schemes.

9.1.2 Optimisations to the FLASH Code

In order to make use of available resources and to enable large simulations to be run I have made a series of optimisations to the *FLASH* AMR code. These have included reducing the memory consumption and in collaboration with Dr. Tom Theuns, I have helped developed and rigorously test a new FFTW gravity solver. This new gravitational solver has been found to give up to a factor of twenty four speed up in the runtime execution of the code.

9.1.3 A Test Suite for Quantitative Comparison of Hydrodynamics Codes in Astrophysics

I have collaborated as part of a large code comparison project in which we subject four different popular hydrodynamics code to a series of demanding numerical test problems with known solutions. We determine a general criteria for contrasting results run with different codes and find that simulations that contain one SPH particle to every AMR grid cell produce comparable results. We also highlight specific instances in which a given

hydrodynamic implementation outperforms the others in order to best focus future use of the codes.

9.1.4 On the Origin of Cores in Simulated Galaxy Clusters

In the Santa Barbara code comparison project of Frenk et al. (1999), a significant discrepancy between the size of entropy cores in AMR and SPH codes was first established. Previous investigations have used complex cosmological simulations to investigate the problem and as such have not been able to conclusively determine the underlying mechanisms responsible. In this work we have determined the origin of the discrepancy using suites of model cluster mergers in the AMR code *FLASH* and the SPH code *GADGET-2*. As model mergers allow us to systematically explore the various proposed mechanisms for creating higher entropy cores in AMR codes, we can unambiguously isolate their true cause.

We explore the effects of gravitational resolution, Galilean non-Invariance, and preshocking in SPH codes and find that none of these mechanisms can explain the origin of the higher entropy cores in the AMR codes. In the end we find that turbulent mixing drives the mixing between the low entropy material within the cluster core with higher entropy material outside of the core. This leads to much hotter, higher entropy cores in the *FLASH* AMR code. The suppression in SPH codes of the hydrodynamic instabilities that lead to mixing causes them to develop much cooler, denser cores.

Current cosmological simulations in SPH codes find that the formation of cool core clusters is too hard to reverse and are unable to form the observed populations of non-cool core clusters without using excessive levels of feedback. Our results significantly show that it may be possible to produce the observed population of non-cool core clusters just by changing hydrodynamic implementation to the AMR method.

9.1.5 Sub-Grid Models for Star Formation in FLASH

Given the startling findings of our work on the origin of cores in simulated clusters, I explore the effect of the hydrodynamic implementations upon different sub-grid physical recipes.

I have implemented into the *FLASH* AMR code a near identical copy of the sub-grid recipe for star formation developed by Schaye and Dalla Vecchia (2008). This model, originally written for the SPH code *GADGET-2* is currently being used in the massive *OWLS* and *GIMIC* cosmological simulations. By systematically comparing the results

from runs in both the *FLASH* AMR code and *GADGET-2* SPH code I have been able to isolate distinct differences in the way in which the sub-grid pressure laws operate in the two different hydrodynamic codes.

I find that when using equivalent polytropic indices, the AMR code produces much denser and more compact structures. For the default polytropic index of 4/3 as used in Dalla Vecchia and Schaye (2008) I find that whilst the *GADGET-2* simulation produces smooth spiral arms and a well converged star formation history, the *FLASH* code creates very compact dense knots of gas which disrupt the spiral arms. These dense knots increase in density and reduce in size as the resolution is increased and are not found to converge.

I instead find excellent agreement between the two hydrodynamic codes when using different polytropic indices, the *FLASH* index of 5/3 higher than that in *GADGET-2*(4/3). The higher polytropic index stabilises the collapse of gas by offering more pressure support with increasing density. This leads to a Jeans length that increases with density, allowing perturbations to be smoothed out instead of collapsing. The resulting simulations show well converged global star formation rates and near identical morphological structure.

9.1.6 Sub-grid Models for Supernova Feedback in FLASH

I have also implemented the kinetic supernova feedback prescription of Dalla Vecchia and Schaye (2008). Since the two hydrodynamic codes differ so strongly I have developed and tested several different feedback models in the *FLASH* code and contrasted the results to those obtained using the *GADGET-2* code. The models successfully reproduce the expected mass loading to within less than a percent.

I find that without the ability to shock gas on the effective equation of state, the kinetic energy of gas in the wind is lost during collisions with other dense gas in the galactic disk. By incorporating the ability to shock gas off of the effective equation of state, I increase the feedback prescription's efficiency. However I still find that it only achieves half the reduction in the star formation rate that the SPH code is capable of. I determine that this arises when gas in the wind, collides with other dense gas, shock heats and leaves the equation of state. Although at very high temperature, due to its very high density it quickly cools over the space of a few timesteps. This prevents the gas from having time to adiabatically expand and as such it cannot blow bubbles in the ISM. Once the gas has cooled it returns to the effective equation of state and resumes star formation. I find that the model is extremely robust with ten times the energy injection still failing to alter the feedback efficiency.

I find that the feedback in *FLASH* is only as effective as that in *GADGET-2* when using the thermal component and orientating supernova kicks tangentially away from the disk. This helps minimise collisions between neighbouring cells and therefore optimises the feedback efficiency.

By comparing the SPH and AMR feedback implementations in both a dwarf and Milky Way like galaxy I am able to determine that in less dense regions such as those found in dwarf galaxies, the feedback in *FLASH* is much more efficient. The feedback in *FLASH* leads to dramatically different galaxy morphologies compared to the *GADGET-2* simulations. Distinct features in the AMR simulations include a much more lumpy gaseous halo and the complete disruption of all but the densest spiral arms. The SPH simulations however show a much smoother gaseous halo, preserve the many spiral arms of the galaxy and eject more gas in the supernova driven outflows.

9.2 Proposed Future Work

The results obtained from chapter 8 highlight the need to adopt a more sophisticated multi-phase model in the *FLASH* code. Instead of using a single fluid and a polytropic equation of state to model the ISM, a better approach would be to explicitly split the warm/hot and cold molecular cloud phases. By tracking the two phases separately, the tenuous hot gas can be blown out of the galaxy by supernova without experiencing as much resistance as in the single phase model. In this way the cold molecular phase can be modelled more accurately as a collisionless fluid, coupled to the hot gaseous phase through a small drag term. The benefits of such a model would include being able to gain a more in depth understanding of the processes which mediate star formation and feedback cycles without hiding the physics within all encompassing sub-grid models. This is currently an active area of research for me.

In addition to the production of a multi-phase ISM model, further comparison of data from the *OWLS* and *GIMIC* simulations with *FLASH* AMR simulations will allow us to continue to develop our understanding of different numerical techniques. Most importantly such comparisons are vital to determine whether deviations between theory and observation are due to the lack of some fundamental physical processes, or whether they arise due to a numerical limitation or artefact.

Appendix 10

COSMA - The Cosmology Machine

10.1 System Architecture

The following constitutes a list of present and past hardware which the author has used for the development of the *FLASH* code and for running production level simulations. The first systems I had access to included Quintor and Titania. The latest installment termed "OCTOR" comprises of Oberon, Cordelia, Miranda and Rosalind. All are SUN machines.

10.1.1 Titania

Titania was a shared-memory machine consisting of 24 UltraSPARC-III 750 Mhz processors with a total of 48 Gigabytes of memory. The system was predominantly used for code development with support for the DDT debugger along with the ability to run large memory serial jobs or small parallel runs interactively. In addition it was vital for the analysis of my large datasets and provided disk serving facilities to the rest of COSMA until it was retired shortly after OCTOR was commissioned in April 2006.

10.1.2 Quintor

Quintor comprises of 259 dual core SUNFire V210s nodes providing a total of 518 1002 MHz Ultrasparc IIIi cores. 227 of these nodes have 2 Gigabytes of memory giving 1 Gigabyte of memory per core whilst 32 special nodes exist with 4 Gigabytes of memory offering twice the memory of a standard node. Quintor was used for some of the earliest work presented as part of this thesis but due to the high latency in the interconnect between nodes on

different racks, *FLASH* runs were extremely limited in scope due to the exceptionally high run times.

10.1.3 Oberon

The head node and disk server for the OCTOR systems, Oberon is a V40z shared memory machine with eight 2.2 GHz AMD Opteron 875 cores and a total of 32 Gigabytes of shared memory. The system provides an excellent development framework with support for debuggers and the ability to run interative jobs. The system can be used to run large memory serial jobs and given that it is a shared memory system, it is ideal for running parallel codes without incurring large penalties that arise from communication between different racks either via myrinet or ethernet.

10.1.4 Cordelia

Cordelia is a 264 X2100 node cluster with a total of 528 2.2 GHz AMD Opteron 175 cores with a total of 1056 Gigabytes of RAM, providing 2 Gigabytes of RAM per processor core. They are connected via dual Gigabit over Nortel 5510 Switches with two dual core processors per rack. Jobs which fit onto a single rack allow for all communication to be peformed across the motherboard and therefore are not subject to the much higher latencies when communicating across racks via ethernet or myrinet systems.

10.1.5 Miranda

Miranda is a 64 X4100 node cluster with a total of 256 2.6 GHz AMD Opteron 285 cores with a combined total of 640 Gigabytes of RAM. This is split between 64 large memory nodes with 4 Gigabytes per core and 192 standard nodes with 2 Gigabytes of RAM. The system performs system service communications via dual gigabit ethernet connection and high performance communication is performed via the Myrinet 2000F cards. This system is currently the best suited to communication and memory intensive HPC applications such as *FLASH* and is therefore used for the majority of the large simulations presented within this thesis.

10.1.6 Rosalind

Rosalind is a V890 shared-memory machine with eight 1.5 GHZ UltraSPARC VI+ cores and a total of 64 Gigabytes of RAM. It is predominantly used for large serial applications

requiring huge amounts of memory such as initial conditions generation. It also has access to IDL for analysis.

Bibliography

S. J. Aarseth. Direct n-body calculations. In J. Goodman & P. Hut, editor, *Dynamics of Star Clusters*, volume 113 of *IAU Symposium*, pages 251–258, 1985.

M. G. Abadi, R. G. Bower, and J. F. Navarro. Self-similar shocked accretion of collisional gas with radiative cooling. MNRAS, 314:759–767, June 2000. doi: 10.1046/j.1365-8711.2000.03349.x.

K. Abazajian, Z. Zheng, I. Zehavi, D. H. Weinberg, J. A. Frieman, A. A. Berlind, M. R. Blanton, N. A. Bahcall, J. Brinkmann, D. P. Schneider, and M. Tegmark. Cosmology and the Halo Occupation Distribution from Small-Scale Galaxy Clustering in the Sloan Digital Sky Survey. ApJ, 625:613–620, June 2005. doi: 10.1086/429685.

G. O. Abell. Clustering of Galaxies. ARA&A, 3:1–+, 1965. doi: 10.1146/annurev.aa.03.090165.000245.

R. G. Abraham, P. Nair, P. J. McCarthy, K. Glazebrook, E. Mentuch, H. Yan, S. Savaglio, D. Crampton, R. Murowinski, S. Juneau, D. Le Borgne, R. G. Carlberg, I. Jørgensen, K. Roth, H.-W. Chen, and R. O. Marzke. The Gemini Deep Deep Survey. VIII. When Did Early-Type Galaxies Form? ApJ, 669:184–201, November 2007. doi: 10.1086/521138.

O. Agertz, B. Moore, J. Stadel, D. Potter, F. Miniati, J. Read, L. Mayer, A. Gawryszczak, A. Kravtsov, Å. Nordlund, F. Pearce, V. Quilis, D. Rudd, V. Springel, J. Stone, E. Tasker, R. Teyssier, J. Wadsley, and R. Walder. Fundamental differences between SPH and grid methods. MNRAS, 380:963–978, September 2007. doi: 10.1111/j.1365-2966.2007.12183.x.

O. Agertz, R. Teyssier, and B. Moore. The formation of disk galaxies in a LCDM universe. *ArXiv e-prints*, March 2010.

A. Aguirre, L. Hernquist, J. Schaye, N. Katz, D. H. Weinberg, and J. Gardner. Metal Enrichment of the Intergalactic Medium in Cosmological Simulations. ApJ, 561:521–549, November 2001. doi: 10.1086/323370.

S. Andreon. Is There a Deficit of S0 Galaxies at Intermediate Redshift? ApJ, 501:533–+, July 1998. doi: 10.1086/305853.

S. Arnouts, C. J. Walcher, O. Le Fèvre, G. Zamorani, O. Ilbert, V. Le Brun, L. Pozzetti, S. Bardelli, L. Tresse, E. Zucca, S. Charlot, F. Lamareille, H. J. McCracken, M. Bolzonella, A. Iovino, C. Lonsdale, M. Polletta, J. Surace, D. Bottini, B. Garilli, D. Maccagni, J. P. Picat, R. Scaramella, M. Scodeggio, G. Vettolani, A. Zanichelli, C. Adami, A. Cappi, P. Ciliegi, T. Contini, S. de la Torre, S. Foucaud, P. Franzetti, I. Gavignaud, L. Guzzo, B. Marano, C. Marinoni, A. Mazure, B. Meneux, R. Merighi, S. Paltani, R. Pellò, A. Pollo, M. Radovich, S. Temporin, and D. Vergani. The SWIRE-VVDS-CFHTLS surveys: stellar mass assembly over the last 10 Gyr. Evidence for a major build up of the red sequence between z = 2 and z = 1. A&A, 476:137–150, December 2007. doi: 10.1051/0004-6361:20077632.

Y. Ascasibar and M. Markevitch. The Origin of Cold Fronts in the Cores of Relaxed Galaxy Clusters. ApJ, 650:102–127, October 2006. doi: 10.1086/506508.

Y. Ascasibar, G. Yepes, V. Müller, and S. Gottlöber. The radial structure of galaxy groups and clusters. MNRAS, 346:731–745, December 2003. doi: 10.1111/j.1365-2966.2003.07116.x.

I. K. Baldry, M. L. Balogh, R. G. Bower, K. Glazebrook, R. C. Nichol, S. P. Bamford, and T. Budavari. Galaxy bimodality versus stellar mass and environment. MNRAS, 373:469–483, December 2006. doi: 10.1111/j.1365-2966.2006.11081.x.

M. L. Balogh, F. R. Pearce, R. G. Bower, and S. T. Kay. Revisiting the cosmic cooling crisis. MNRAS, 326:1228–1234, October 2001. doi: 10.1111/j.1365-2966.2001.04667.x.

M. L. Balogh, I. G. McCarthy, R. G. Bower, and V. R. Eke. Testing cold dark matter with the hierarchical build-up of stellar light. MNRAS, 385:1003–1014, April 2008. doi: 10.1111/j.1365-2966.2008.12903.x.

J. Barnes and P. Hut. A hierarchical O(N log N) force-calculation algorithm. Nature, 324:446–449, December 1986. doi: 10.1038/324446a0.

C. M. Baugh. A primer on hierarchical galaxy formation: the semi-analytical approach. *Reports on Progress in Physics*, 69:3101–3156, December 2006. doi: 10.1088/0034-4885/69/12/R02.

V. Belokurov, M. G. Walker, N. W. Evans, G. Gilmore, M. J. Irwin, D. Just, S. Koposov, M. Mateo, E. Olszewski, L. Watkins, and L. Wyrzykowski. Big fish, small fish: Two New Ultra-Faint Satellites of the Milky Way. *ArXiv e-prints*, February 2010.

C. L. Bennett, M. Halpern, G. Hinshaw, N. Jarosik, A. Kogut, M. Limon, S. S. Meyer, L. Page, D. N. Spergel, G. S. Tucker, E. Wollack, E. L. Wright, C. Barnes, M. R. Greason, R. S. Hill, E. Komatsu, M. R. Nolta, N. Odegard, H. V. Peiris, L. Verde, and J. L. Weiland. First-Year Wilkinson Microwave Anisotropy Probe (WMAP) Observations: Preliminary Maps and Basic Results. ApJS, 148:1–27, September 2003. doi: 10.1086/377253.

A. J. Benson. Orbital parameters of infalling dark matter substructures. MNRAS, 358: 551–562, April 2005. doi: 10.1111/j.1365-2966.2005.08788.x.

A. J. Benson, R. G. Bower, C. S. Frenk, C. G. Lacey, C. M. Baugh, and S. Cole. What Shapes the Luminosity Function of Galaxies? ApJ, 599:38–49, December 2003. doi: 10.1086/379160.

E. Bertschinger. Self-similar secondary infall and accretion in an Einstein-de Sitter universe. ApJS, 58:39–65, May 1985. doi: 10.1086/191028.

E. Bertschinger. The evolution of cooling flows - Self-similar cooling waves. ApJ, 340: 666–678, May 1989. doi: 10.1086/167428.

J. Binney and G. Tabor. Evolving Cooling Flows. MNRAS, 276:663–+, September 1995.

J. Binney and S. Tremaine. *Galactic dynamics*. 1987.

C. M. Booth, T. Theuns, and T. Okamoto. Molecular cloud regulated star formation in galaxies. MNRAS, 376:1588–1610, April 2007. doi: 10.1111/j.1365-2966.2007.11570.x.

S. Borgani, A. Finoguenov, S. T. Kay, T. J. Ponman, V. Springel, P. Tozzi, and G. M. Voit. Entropy amplification from energy feedback in simulated galaxy groups and clusters. MNRAS, 361:233–243, July 2005. doi: 10.1111/j.1365-2966.2005.09158.x.

S. Borgani, K. Dolag, G. Murante, L.-M. Cheng, V. Springel, A. Diaferio, L. Moscardini, G. Tormen, L. Tornatore, and P. Tozzi. Hot and cooled baryons in smoothed particle

hydrodynamic simulations of galaxy clusters: physics and numerics. MNRAS, 367: 1641–1654, April 2006. doi: 10.1111/j.1365-2966.2006.10075.x.

A. P. Boss. Collapse and Fragmentation of Molecular Cloud Cores. VII. Magnetic Fields and Multiple Protostar Formation. ApJ, 568:743–753, April 2002. doi: 10.1086/339040.

A. P. Boss. Collapse and Fragmentation of Molecular Cloud Cores. X. Magnetic Braking of Prolate and Oblate Cores. ApJ, 697:1940–1945, June 2009. doi: 10.1088/0004-637X/697/2/1940.

F. Bournaud and F. Combes. Gas accretion on spiral galaxies: Bar formation and renewal. A&A, 392:83–102, September 2002. doi: 10.1051/0004-6361:20020920.

R. G. Bower, A. J. Benson, R. Malbon, J. C. Helly, C. S. Frenk, C. M. Baugh, S. Cole, and C. G. Lacey. Breaking the hierarchy of galaxy formation. MNRAS, 370:645–655, August 2006. doi: 10.1111/j.1365-2966.2006.10519.x.

R. G. Bower, I. G. McCarthy, and A. J. Benson. The flip side of galaxy formation: a combined model of galaxy formation and cluster heating. MNRAS, 390:1399–1410, November 2008. doi: 10.1111/j.1365-2966.2008.13869.x.

C. B. Brook, D. Kawata, B. K. Gibson, and C. Flynn. Stellar halo constraints on simulated late-type galaxies. MNRAS, 349:52–56, March 2004. doi: 10.1111/j.1365-2966.2004.07464.x.

M. Brueggen and E. Scannapieco. Self-Regulation of AGN in Galaxy Clusters. *ArXiv e-prints*, May 2009.

G. L. Bryan. Explaining the Entropy Excess in Clusters and Groups of Galaxies without Additional Heating. ApJ, 544:L1–L5, November 2000. doi: 10.1086/317289.

G. L. Bryan and M. L. Norman. A Hybrid AMR Application for Cosmology and Astrophysics. *ArXiv Astrophysics e-prints*, October 1997.

G. L. Bryan, M. L. Norman, J. M. Stone, R. Cen, and J. P. Ostriker. A piecewise parabolic method for cosmological hydrodynamics. *Computer Physics Communications*, 89:149–168, August 1995. doi: 10.1016/0010-4655(94)00191-4.

J. O. Burns, E. J. Hallman, B. Gantner, P. M. Motl, and M. L. Norman. Why Do Only Some Galaxy Clusters Have Cool Cores? ApJ, 675:1125–1140, March 2008. doi: 10.1086/526514.

D. Calzetti. The effects of dust on the spectral energy distribution of star-forming galaxies. *New Astronomy Review*, 45:601–607, October 2001. doi: 10.1016/S1387-6473(01)00144-0.

D. Calzetti, A. L. Kinney, and T. Storchi-Bergmann. Dust extinction of the stellar continua in starburst galaxies: The ultraviolet and optical extinction law. ApJ, 429: 582–601, July 1994. doi: 10.1086/174346.

G. Chabrier. Galactic Stellar and Substellar Initial Mass Function. PASP, 115:763–795, July 2003. doi: 10.1086/376392.

S. Chandrasekhar. The highly collapsed configurations of a stellar mass. MNRAS, 91: 456–466, March 1931.

S. Chandrasekhar. The highly collapsed configurations of a stellar mass (Second paper). MNRAS, 95:207–225, January 1935.

Y. Chen, T. H. Reiprich, H. Böhringer, Y. Ikebe, and Y.-Y. Zhang. Statistics of X-ray observables for the cooling-core and non-cooling core galaxy clusters. A&A, 466: 805–812, May 2007. doi: 10.1051/0004-6361:20066471.

A. Chung, J. H. van Gorkom, J. D. P. Kenney, and B. Vollmer. Virgo Galaxies with Long One-sided H I Tails. ApJ, 659:L115–L119, April 2007. doi: 10.1086/518034.

E. Churazov, W. Forman, C. Jones, and H. Böhringer. XMM-Newton Observations of the Perseus Cluster. I. The Temperature and Surface Brightness Structure. ApJ, 590: 225–237, June 2003. doi: 10.1086/374923.

E. Churazov, W. Forman, C. Jones, R. Sunyaev, and H. Böhringer. XMM-Newton observations of the Perseus cluster - II. Evidence for gas motions in the core. MNRAS, 347: 29–35, January 2004. doi: 10.1111/j.1365-2966.2004.07201.x.

A. Cimatti, P. Cassata, L. Pozzetti, J. Kurk, M. Mignoli, A. Renzini, E. Daddi, M. Bolzonella, M. Brusa, G. Rodighiero, M. Dickinson, A. Franceschini, G. Zamorani, S. Berta, P. Rosati, and C. Halliday. GMASS ultradeep spectroscopy of galaxies at z ˜ 2. II. Superdense passive galaxies: how did they form and evolve? A&A, 482:21–42, April 2008. doi: 10.1051/0004-6361:20078739.

L. Ciotti. Galaxy formation: Anatomy of elliptical galaxies. Nature, 460:333–334, July 2009. doi: 10.1038/460333a.

P. Colella and H. M. Glaz. Efficient solution algorithms for the Riemann problem for real gases. *Journal of Computational Physics*, 59:264–289, June 1985. doi: 10.1016/0021-9991(85)90146-9.

P. Colella and P. R. Woodward. The Piecewise Parabolic Method (PPM) for Gas-Dynamical Simulations. *Journal of Computational Physics*, 54:174–201, September 1984. doi: 10.1016/0021-9991(84)90143-8.

M. Colless. First results from the 2dF Galaxy Redshift Survey. In G. Efstathiou & et al., editor, *Large-Scale Structure in the Universe*, pages 105–+, 1999.

H. M. P. Couchman, P. A. Thomas, and F. R. Pearce. Hydra: an Adaptive-Mesh Implementation of P 3M-SPH. ApJ, 452:797–+, October 1995. doi: 10.1086/176348.

L. L. Cowie, A. Songaila, E. M. Hu, and J. G. Cohen. New Insight on Galaxy Formation and Evolution From Keck Spectroscopy of the Hawaii Deep Fields. AJ, 112:839–+, September 1996. doi: 10.1086/118058.

R. A. Crain, T. Theuns, C. Dalla Vecchia, V. R. Eke, C. S. Frenk, A. Jenkins, S. T. Kay, J. A. Peacock, F. R. Pearce, J. Schaye, V. Springel, P. A. Thomas, S. D. M. White, and R. P. C. Wiersma. Galaxies-Intergalactic Medium Interaction Calculation –I. Galaxy formation as a function of large-scale environment. *ArXiv e-prints*, June 2009.

D. J. Croton, V. Springel, S. D. M. White, G. De Lucia, C. S. Frenk, L. Gao, A. Jenkins, G. Kauffmann, J. F. Navarro, and N. Yoshida. The many lives of active galactic nuclei: cooling flows, black holes and the luminosities and colours of galaxies. MNRAS, 365: 11–28, January 2006. doi: 10.1111/j.1365-2966.2005.09675.x.

E. Daddi, A. Renzini, N. Pirzkal, A. Cimatti, S. Malhotra, M. Stiavelli, C. Xu, A. Pasquali, J. E. Rhoads, M. Brusa, S. di Serego Alighieri, H. C. Ferguson, A. M. Koekemoer, L. A. Moustakas, N. Panagia, and R. A. Windhorst. Passively Evolving Early-Type Galaxies at 1.4 ¡˜ z ¡˜ 2.5 in the Hubble Ultra Deep Field. ApJ, 626:680–697, June 2005. doi: 10.1086/430104.

C. Dalla Vecchia and J. Schaye. Simulating galactic outflows with kinetic supernova feedback. MNRAS, 387:1431–1444, July 2008. doi: 10.1111/j.1365-2966.2008.13322.x.

C. Dalla Vecchia, R. G. Bower, T. Theuns, M. L. Balogh, P. Mazzotta, and C. S. Frenk. Quenching cluster cooling flows with recurrent hot plasma bubbles. MNRAS, 355: 995–1004, December 2004. doi: 10.1111/j.1365-2966.2004.08381.x.

M. Davis, G. Efstathiou, C. S. Frenk, and S. D. M. White. The evolution of large-scale structure in a universe dominated by cold dark matter. ApJ, 292:371–394, May 1985. doi: 10.1086/163168.

W. J. G. de Blok, F. Walter, E. Brinks, C. Trachternach, S.-H. Oh, and R. C. Kennicutt. High-Resolution Rotation Curves and Galaxy Mass Models from THINGS. AJ, 136: 2648–2719, December 2008. doi: 10.1088/0004-6256/136/6/2648.

S. De Grandi and S. Molendi. Metallicity Gradients in X-Ray Clusters of Galaxies. ApJ, 551:153–159, April 2001. doi: 10.1086/320098.

S. De Grandi, S. Ettori, M. Longhetti, and S. Molendi. On the iron content in rich nearby clusters of galaxies. A&A, 419:7–18, May 2004. doi: 10.1051/0004-6361:20034228.

A. Dekel and Y. Birnboim. Gravitational quenching in massive galaxies and clusters by clumpy accretion. MNRAS, 383:119–138, January 2008. doi: 10.1111/j.1365-2966.2007.12569.x.

S. Diehl and T. S. Statler. The Hot Interstellar Medium of Normal Elliptical Galaxies. II. Morphological Evidence for Active Galactic Nucleus Feedback. ApJ, 680:897–910, June 2008. doi: 10.1086/587481.

K. Dolag, F. Vazza, G. Brunetti, and G. Tormen. Turbulent gas motions in galaxy cluster simulations: the role of smoothed particle hydrodynamics viscosity. MNRAS, 364:753–772, December 2005. doi: 10.1111/j.1365-2966.2005.09630.x.

A. Doroshkevich, D. L. Tucker, S. Allam, and M. J. Way. Large scale structure in the SDSS galaxy survey. A&A, 418:7–23, April 2004. doi: 10.1051/0004-6361:20031780.

A. Dressler. Galaxy morphology in rich clusters - Implications for the formation and evolution of galaxies. ApJ, 236:351–365, March 1980. doi: 10.1086/157753.

Y. Dubois and R. Teyssier. On the onset of galactic winds in quiescent star forming galaxies. A&A, 477:79–94, January 2008. doi: 10.1051/0004-6361:20078326.

Y. Dubois, J. Devriendt, A. Slyz, and J. Silk. Influence of AGN jets on the magnetized ICM. MNRAS, 399:L49–L53, October 2009. doi: 10.1111/j.1745-3933.2009.00721.x.

R. J. H. Dunn and A. C. Fabian. Investigating heating and cooling in the BCS and B55 cluster samples. MNRAS, 385:757–768, April 2008. doi: 10.1111/j.1365-2966.2008.12898.x.

P. Dutta, A. Begum, S. Bharadwaj, and J. N. Chengalur. HI power spectrum of the spiral galaxy NGC628. MNRAS, 384:L34–L37, February 2008. doi: 10.1111/j.1745-3933.2007.00417.x.

P. Dutta, A. Begum, S. Bharadwaj, and J. N. Chengalur. The scaleheight of NGC 1058 measured from its HI power spectrum. MNRAS, 397:L60–L63, July 2009a. doi: 10.1111/j.1745-3933.2009.00684.x.

P. Dutta, A. Begum, S. Bharadwaj, and J. N. Chengalur. A study of interstellar medium of dwarf galaxies using HI power spectrum analysis. MNRAS, 398:887–897, September 2009b. doi: 10.1111/j.1365-2966.2009.15105.x.

A. C. Edge. The detection of molecular gas in the central galaxies of cooling flow clusters. MNRAS, 328:762–782, December 2001. doi: 10.1046/j.1365-8711.2001.04802.x.

G. Efstathiou, M. Davis, S. D. M. White, and C. S. Frenk. Numerical techniques for large cosmological N-body simulations. ApJS, 57:241–260, February 1985. doi: 10.1086/191003.

V. R. Eke, J. F. Navarro, and M. Steinmetz. The Power Spectrum Dependence of Dark Matter Halo Concentrations. ApJ, 554:114–125, June 2001. doi: 10.1086/321345.

B. G. Elmegreen. Star Formation from Galaxies to Globules. ApJ, 577:206–220, September 2002. doi: 10.1086/342177.

B. G. Elmegreen and Y. N. Efremov. A Universal Formation Mechanism for Open and Globular Clusters in Turbulent Gas. ApJ, 480:235–+, May 1997. doi: 10.1086/303966.

A. E. Evrard and J. P. Henry. Expectations for X-ray cluster observations by the ROSAT satellite. ApJ, 383:95–103, December 1991. doi: 10.1086/170767.

A. C. Fabian. Cooling Flows in Clusters of Galaxies. ARA&A, 32:277–318, 1994. doi: 10.1146/annurev.aa.32.090194.001425.

A. C. Fabian and P. E. J. Nulsen. Subsonic accretion of cooling gas in clusters of galaxies. MNRAS, 180:479–484, August 1977.

A. C. Fabian, J. S. Sanders, S. W. Allen, C. S. Crawford, K. Iwasawa, R. M. Johnstone, R. W. Schmidt, and G. B. Taylor. A deep Chandra observation of the Perseus cluster: shocks and ripples. MNRAS, 344:L43–L47, September 2003a. doi: 10.1046/j.1365-8711.2003.06902.x.

A. C. Fabian, J. S. Sanders, C. S. Crawford, C. J. Conselice, J. S. Gallagher, and R. F. G. Wyse. The relationship between the optical Hα filaments and the X-ray emission in the core of the Perseus cluster. MNRAS, 344:L48–L52, September 2003b. doi: 10.1046/j.1365-8711.2003.06856.x.

A. C. Fabian, C. S. Reynolds, G. B. Taylor, and R. J. H. Dunn. On viscosity, conduction and sound waves in the intracluster medium. MNRAS, 363:891–896, November 2005. doi: 10.1111/j.1365-2966.2005.09484.x.

E. E. Falco, M. J. Kurtz, M. J. Geller, J. P. Huchra, J. Peters, P. Berlind, D. J. Mink, S. P. Tokarz, and B. Elwell. The Updated Zwicky Catalog (UZC). PASP, 111:438–452, April 1999. doi: 10.1086/316343.

L.-L. Feng, C.-W. Shu, and M. Zhang. A Hybrid Cosmological Hydrodynamic/N-Body Code Based on a Weighted Essentially Nonoscillatory Scheme. ApJ, 612:1–13, September 2004. doi: 10.1086/422513.

G. J. Ferland, K. T. Korista, D. A. Verner, J. W. Ferguson, J. B. Kingdon, and E. M. Verner. CLOUDY 90: Numerical Simulation of Plasmas and Their Spectra. PASP, 110:761–778, July 1998. doi: 10.1086/316190.

L. Ferrarese and D. Merritt. A Fundamental Relation between Supermassive Black Holes and Their Host Galaxies. ApJ, 539:L9–L12, August 2000. doi: 10.1086/312838.

K. M. Ferrière. The interstellar environment of our galaxy. *Reviews of Modern Physics*, 73:1031–1066, October 2001. doi: 10.1103/RevModPhys.73.1031.

T. Ferris. The World Treasury of Physics Astronomy and Mathematics. *Science*, 252: 865–+, May 1991.

A. S. Font, R. G. Bower, I. G. McCarthy, A. J. Benson, C. S. Frenk, J. C. Helly, C. G. Lacey, C. M. Baugh, and S. Cole. The colours of satellite galaxies in groups and clusters. MNRAS, 389:1619–1629, October 2008. doi: 10.1111/j.1365-2966.2008.13698.x.

C. S. Frenk, S. D. M. White, P. Bode, J. R. Bond, G. L. Bryan, R. Cen, H. M. P. Couchman, A. E. Evrard, N. Gnedin, A. Jenkins, A. M. Khokhlov, A. Klypin, J. F. Navarro, M. L. Norman, J. P. Ostriker, J. M. Owen, F. R. Pearce, U.-L. Pen, M. Steinmetz, P. A. Thomas, J. V. Villumsen, J. W. Wadsley, M. S. Warren, G. Xu, and G. Yepes. The Santa Barbara Cluster Comparison Project: A Comparison of Cosmological Hydrodynamics Solutions. ApJ, 525:554–582, November 1999. doi: 10.1086/307908.

B. Fryxell, E. Müller, and D. Arnett. Computation of multi-dimensional flows with non-uniform composition. In W. Hillebrandt and E. Müller, editors, *Nuclear Astrophysics*, pages 100–102, April 1989.

B. Fryxell, K. Olson, P. Ricker, F. X. Timmes, M. Zingale, D. Q. Lamb, P. MacNeice, R. Rosner, J. W. Truran, and H. Tufo. FLASH: An Adaptive Mesh Hydrodynamics Code for Modeling Astrophysical Thermonuclear Flashes. ApJS, 131:273–334, November 2000. doi: 10.1086/317361.

Y. Fukazawa, K. Makishima, T. Tamura, K. Nakazawa, H. Ezawa, Y. Ikebe, K. Kikuchi, and T. Ohashi. Statistical properties of metal abundances of the intracluster medium in the central region of clusters. MNRAS, 313:21–31, March 2000. doi: 10.1046/j.1365-8711.2000.03204.x.

K. Gebhardt, R. Bender, G. Bower, A. Dressler, S. M. Faber, A. V. Filippenko, R. Green, C. Grillmair, L. C. Ho, J. Kormendy, T. R. Lauer, J. Magorrian, J. Pinkney, D. Richstone, and S. Tremaine. A Relationship between Nuclear Black Hole Mass and Galaxy Velocity Dispersion. ApJ, 539:L13–L16, August 2000. doi: 10.1086/312840.

M. Geha, R. P. van der Marel, P. Guhathakurta, K. M. Gilbert, J. Kalirai, and E. N. Kirby. Local Group Dwarf Elliptical Galaxies. II. Stellar Kinematics to Large Radii in NGC 147 and NGC 185. ApJ, 711:361–373, March 2010. doi: 10.1088/0004-637X/711/1/361.

A. M. Ghez, B. L. Klein, M. Morris, and E. E. Becklin. High Proper-Motion Stars in the Vicinity of Sagittarius A*: Evidence for a Supermassive Black Hole at the Center of Our Galaxy. ApJ, 509:678–686, December 1998. doi: 10.1086/306528.

A. M. Ghez, S. D. Hornstein, J. R. Lu, A. Bouchez, D. Le Mignant, M. A. van Dam, P. Wizinowich, K. Matthews, M. Morris, E. E. Becklin, R. D. Campbell, J. C. Y. Chin, S. K. Hartman, E. M. Johansson, R. E. Lafon, P. J. Stomski, and D. M. Summers. The First Laser Guide Star Adaptive Optics Observations of the Galactic Center: Sgr A*'s Infrared Color and the Extended Red Emission in its Vicinity. ApJ, 635:1087–1094, December 2005. doi: 10.1086/497576.

A. M. Ghez, S. Salim, N. N. Weinberg, J. R. Lu, T. Do, J. K. Dunn, K. Matthews, M. R. Morris, S. Yelda, E. E. Becklin, T. Kremenek, M. Milosavljevic, and J. Naiman.

Measuring Distance and Properties of the Milky Way's Central Supermassive Black Hole with Stellar Orbits. ApJ, 689:1044–1062, December 2008. doi: 10.1086/592738.

R. A. Gingold and J. J. Monaghan. Smoothed particle hydrodynamics - Theory and application to non-spherical stars. MNRAS, 181:375–389, November 1977.

M. Giovalli. Numerical approaches to star formation and SuperNovae energy feedback in simulations of galaxy clusters. *ArXiv e-prints*, April 2009.

G. R. Gisler. The fate of gas in elliptical galaxies and the density evolution of radio sources. A&A, 51:137–150, August 1976.

S. K. Godunov. A Difference Scheme for Numerical Solution of Discontinuous Solution of Hydrodynamic Equations. *Math. Sbornik*, 47:271–306, 1959.

P. L. Gómez, C. Loken, K. Roettiger, and J. O. Burns. Do Cooling Flows Survive Cluster Mergers? ApJ, 569:122–133, April 2002. doi: 10.1086/339280.

F. Governato, C. Brook, L. Mayer, A. Brooks, G. Rhee, J. Wadsley, P. Jonsson, B. Willman, G. Stinson, T. Quinn, and P. Madau. Bulgeless dwarf galaxies and dark matter cores from supernova-driven outflows. Nature, 463:203–206, January 2010. doi: 10.1038/nature08640.

B. R. Granett, M. C. Neyrinck, and I. Szapudi. An Imprint of Superstructures on the Microwave Background due to the Integrated Sachs-Wolfe Effect. ApJ, 683:L99–L102, August 2008. doi: 10.1086/591670.

J. E. Gunn and J. R. Gott, III. On the Infall of Matter Into Clusters of Galaxies and Some Effects on Their Evolution. ApJ, 176:1–+, August 1972. doi: 10.1086/151605.

F. Haardt and P. Madau. Modelling the UV/X-ray cosmic background with CUBA. In D. M. Neumann & J. T. V. Tran, editor, *Clusters of Galaxies and the High Redshift Universe Observed in X-rays*, 2001.

N. Häring and H.-W. Rix. On the Black Hole Mass-Bulge Mass Relation. ApJ, 604: L89–L92, April 2004. doi: 10.1086/383567.

E. Hayashi, J. F. Navarro, J. E. Taylor, J. Stadel, and T. Quinn. The Structural Evolution of Substructure. ApJ, 584:541–558, February 2003. doi: 10.1086/345788.

T. M. Heckman, M. D. Lehnert, D. K. Strickland, and L. Armus. Absorption-Line Probes of Gas and Dust in Galactic Superwinds. ApJS, 129:493–516, August 2000. doi: 10.1086/313421.

A. Heger, C. L. Fryer, S. E. Woosley, N. Langer, and D. H. Hartmann. How Massive Single Stars End Their Life. ApJ, 591:288–300, July 2003. doi: 10.1086/375341.

C. Heiles and T. H. Troland. The Millennium Arecibo 21 Centimeter Absorption-Line Survey. IV. Statistics of Magnetic Field, Column Density, and Turbulence. ApJ, 624: 773–793, May 2005. doi: 10.1086/428896.

P. Hennebelle and G. Chabrier. Analytical Theory for the Initial Mass Function: CO Clumps and Prestellar Cores. ApJ, 684:395–410, September 2008. doi: 10.1086/589916.

L. Hernquist. An analytical model for spherical galaxies and bulges. ApJ, 356:359–364, June 1990. doi: 10.1086/168845.

S. J. Higdon, J. L. Higdon, and J. Marshall. First Detection of PAHs and Warm Molecular Hydrogen in Tidal Dwarf Galaxies. ApJ, 640:768–783, April 2006. doi: 10.1086/500254.

G. Hinshaw, J. L. Weiland, R. S. Hill, N. Odegard, D. Larson, C. L. Bennett, J. Dunkley, B. Gold, M. R. Greason, N. Jarosik, E. Komatsu, M. R. Nolta, L. Page, D. N. Spergel, E. Wollack, M. Halpern, A. Kogut, M. Limon, S. S. Meyer, G. S. Tucker, and E. L. Wright. Five-Year Wilkinson Microwave Anisotropy Probe Observations: Data Processing, Sky Maps, and Basic Results. ApJS, 180:225–245, February 2009. doi: 10.1088/0067-0049/180/2/225.

J. G. Hosking and A. P. Whitworth. Fragmentation of magnetized cloud cores. MNRAS, 347:1001–1010, January 2004. doi: 10.1111/j.1365-2966.2004.07274.x.

X. Y. Hu and N. A. Adams. A multi-phase SPH method for macroscopic and mesoscopic flows. *Journal of Computational Physics*, 213:844–861, April 2006. doi: 10.1016/j.jcp.2005.09.001.

L. Iapichino and J. C. Niemeyer. Hydrodynamical adaptive mesh refinement simulations of turbulent flows - II. Cosmological simulations of galaxy clusters. MNRAS, 388: 1089–1100, August 2008. doi: 10.1111/j.1365-2966.2008.13518.x.

S.-I. Inutsuka. Reformulation of Smoothed Particle Hydrodynamics with Riemann Solver. *Journal of Computational Physics*, 179:238–267, June 2002. doi: 10.1006/jcph.2002.7053.

M. J. Irwin, V. Belokurov, N. W. Evans, E. V. Ryan-Weber, J. T. A. de Jong, S. Koposov, D. B. Zucker, S. T. Hodgkin, G. Gilmore, P. Prema, L. Hebb, A. Begum, M. Fellhauer, P. C. Hewett, R. C. Kennicutt, Jr., M. I. Wilkinson, D. M. Bramich, S. Vidrih, H.-W. Rix, T. C. Beers, J. C. Barentine, H. Brewington, M. Harvanek, J. Krzesinski, D. Long, A. Nitta, and S. A. Snedden. Discovery of an Unusual Dwarf Galaxy in the Outskirts of the Milky Way. ApJ, 656:L13–L16, February 2007. doi: 10.1086/512183.

R. M. Johnstone, A. C. Fabian, and P. E. J. Nulsen. The optical spectra of central galaxies in southern clusters Evidence for star formation. MNRAS, 224:75–91, January 1987.

G. I. Jordan, R. Fisher, D. Townsley, A. Calder, C Graziani, S Asida, D. Lamb, and J. Truran. Three-Dimensional Simulations of the Deflagration Phase of the Gravitationally Confined Detonation Model of Type Ia Supernovae. *ArXiv Astrophysics e-prints*, March 2007.

D. W. Just, D. Zaritsky, D. J. Sand, V. Desai, and G. Rudnick. The Environmental Dependence of the Evolving S0 Fraction. ApJ, 711:192–200, March 2010. doi: 10.1088/0004-637X/711/1/192.

N. Kaiser. Evolution of clusters of galaxies. ApJ, 383:104–111, December 1991. doi: 10.1086/170768.

N. Katz, D. H. Weinberg, and L. Hernquist. Cosmological Simulations with TreeSPH. ApJS, 105:19–+, July 1996. doi: 10.1086/192305.

S. T. Kay, P. A. Thomas, A. Jenkins, and F. R. Pearce. Cosmological simulations of the intracluster medium. MNRAS, 355:1091–1104, December 2004. doi: 10.1111/j.1365-2966.2004.08383.x.

R. C. Kennicutt, C.-N. Hao, D. Calzetti, J. Moustakas, D. A. Dale, G. Bendo, C. W. Engelbracht, B. D. Johnson, and J. C. Lee. Dust-corrected Star Formation Rates of Galaxies. I. Combinations of Hα and Infrared Tracers. ApJ, 703:1672–1695, October 2009. doi: 10.1088/0004-637X/703/2/1672.

R. C. Kennicutt, Jr. The star formation law in galactic disks. ApJ, 344:685–703, September 1989. doi: 10.1086/167834.

R. C. Kennicutt, Jr. Star Formation in Galaxies Along the Hubble Sequence. ARA&A, 36:189–232, 1998a. doi: 10.1146/annurev.astro.36.1.189.

R. C. Kennicutt, Jr. The Global Schmidt Law in Star-forming Galaxies. ApJ, 498:541–+, May 1998b. doi: 10.1086/305588.

R. C. Kennicutt, Jr., D. Calzetti, F. Walter, G. Helou, D. J. Hollenbach, L. Armus, G. Bendo, D. A. Dale, B. T. Draine, C. W. Engelbracht, K. D. Gordon, M. K. M. Prescott, M. W. Regan, M. D. Thornley, C. Bot, E. Brinks, E. de Blok, D. de Mello, M. Meyer, J. Moustakas, E. J. Murphy, K. Sheth, and J. D. T. Smith. Star Formation in NGC 5194 (M51a). II. The Spatially Resolved Star Formation Law. ApJ, 671:333–348, December 2007. doi: 10.1086/522300.

D. Kereš, N. Katz, D. H. Weinberg, and R. Davé. How do galaxies get their gas? MNRAS, 363:2–28, October 2005. doi: 10.1111/j.1365-2966.2005.09451.x.

D. Kereš, N. Katz, M. Fardal, R. Davé, and D. H. Weinberg. Galaxies in a simulated ΛCDM Universe - I. Cold mode and hot cores. MNRAS, 395:160–179, May 2009. doi: 10.1111/j.1365-2966.2009.14541.x.

F. J. Kerr and D. Lynden-Bell. Review of galactic constants. MNRAS, 221:1023–1038, August 1986.

A. Khalatyan, A. Cattaneo, M. Schramm, S. Gottlöber, M. Steinmetz, and L. Wisotzki. Is AGN feedback necessary to form red elliptical galaxies? MNRAS, 387:13–30, June 2008. doi: 10.1111/j.1365-2966.2008.13093.x.

W.-T. Kim, A. A. El-Zant, and M. Kamionkowski. Dynamical Friction and Cooling Flows in Galaxy Clusters. ApJ, 632:157–168, October 2005. doi: 10.1086/432976.

I. R. King. The structure of star clusters. III. Some simple dynamical models. AJ, 71: 64–+, February 1966. doi: 10.1086/109857.

R. S. Klessen. The Formation of Stellar Clusters: Mass Spectra from Turbulent Molecular Cloud Fragmentation. ApJ, 556:837–846, August 2001. doi: 10.1086/321626.

A. Kolmogorov. The Local Structure of Turbulence in Incompressible Viscous Fluid for Very Large Reynolds' Numbers. *Akademiia Nauk SSSR Doklady*, 30:301–305, 1941.

R. A. Koopmann, R. Giovanelli, M. P. Haynes, B. R. Kent, T. J. Balonek, N. Brosch, J. L. Higdon, J. J. Salzer, and O. Spector. A 500 kpc H I Extension of the Virgo Pair

NGC 4532/DDO 137 Detected by the Arecibo Legacy Fast ALFA (ALFALFA) Survey. ApJ, 682:L85–L88, August 2008. doi: 10.1086/591124.

J. Kormendy and D. Richstone. Inward Bound—The Search For Supermassive Black Holes In Galactic Nuclei. ARA&A, 33:581–+, 1995. doi: 10.1146/annurev.aa.33.090195.003053.

J. Kormendy, D. B. Fisher, M. E. Cornell, and R. Bender. Structure and Formation of Elliptical and Spheroidal Galaxies. ApJS, 182:216–309, May 2009. doi: 10.1088/0067-0049/182/1/216.

A. V. Kravtsov, D. Nagai, and A. A. Vikhlinin. Effects of Cooling and Star Formation on the Baryon Fractions in Clusters. ApJ, 625:588–598, June 2005. doi: 10.1086/429796.

M. Kuhlen, J. Diemand, P. Madau, and M. Zemp. The Via Lactea INCITE simulation: galactic dark matter substructure at high resolution. *Journal of Physics Conference Series*, 125(1):012008–+, July 2008. doi: 10.1088/1742-6596/125/1/012008.

L. D. Landau and E. M. Lifshitz. *Fluid mechanics*. 1959.

M. G. Lee, H. S. Park, H. S. Hwang, N. Arimoto, N. Tamura, and M. Onodera. The Globular Cluster System of the Virgo Giant Elliptical Galaxy NGC 4636. II. Kinematics of the Globular Cluster System. ApJ, 709:1083–1099, February 2010. doi: 10.1088/0004-637X/709/2/1083.

H.-b. Li, C. D. Dowell, A. Goodman, R. Hildebrand, and G. Novak. Anchoring Magnetic Field in Turbulent Molecular Clouds. ApJ, 704:891–897, October 2009. doi: 10.1088/0004-637X/704/2/891.

S. Li and H. Li. Parallel AMR Code for Compressible MHD or HD Equations. Technical report, Los Alamos National Laboratory, 2004.

W. P. Lin, Y. P. Jing, S. Mao, L. Gao, and I. G. McCarthy. The Influence of Baryons on the Mass Distribution of Dark Matter Halos. ApJ, 651:636–642, November 2006. doi: 10.1086/508052.

R. Löhner. An adaptive finite element scheme for transient problems in cfd. *Comput. Methods Appl. Mech. Eng.*, 61(3):323–338, 1987. ISSN 0045-7825. doi: http://dx.doi.org/10.1016/0045-7825(87)90098-3.

M. López-Corredoira. Intrinsic Colors and Ages of Extremely Red Elliptical Galaxies at High Redshift. AJ, 139:540–544, February 2010. doi: 10.1088/0004-6256/139/2/540.

L. B. Lucy. A numerical approach to the testing of the fission hypothesis. AJ, 82: 1013–1024, December 1977. doi: 10.1086/112164.

A. Marconi and L. K. Hunt. The Relation between Black Hole Mass, Bulge Mass, and Near-Infrared Luminosity. ApJ, 589:L21–L24, May 2003. doi: 10.1086/375804.

C. L. Martin. Mapping Large-Scale Gaseous Outflows in Ultraluminous Galaxies with Keck II ESI Spectra: Variations in Outflow Velocity with Galactic Mass. ApJ, 621: 227–245, March 2005. doi: 10.1086/427277.

C. L. Martin and R. C. Kennicutt, Jr. Star Formation Thresholds in Galactic Disks. ApJ, 555:301–321, July 2001. doi: 10.1086/321452.

M. L. Mateo. Dwarf Galaxies of the Local Group. ARA&A, 36:435–506, 1998. doi: 10.1146/annurev.astro.36.1.435.

K. Matsushita, E. Belsole, A. Finoguenov, and H. Böhringer. XMM-Newton observation of M 87. I. Single-phase temperature structure of intracluster medium. A&A, 386: 77–96, April 2002. doi: 10.1051/0004-6361:20020087.

L. Mayer, C. Mastropietro, J. Wadsley, J. Stadel, and B. Moore. Simultaneous ram pressure and tidal stripping; how dwarf spheroidals lost their gas. MNRAS, 369:1021–1038, July 2006. doi: 10.1111/j.1365-2966.2006.10403.x.

P. A. Mazzali, F. K. Röpke, S. Benetti, and W. Hillebrandt. A Common Explosion Mechanism for Type Ia Supernovae. *Science*, 315:825–, February 2007. doi: 10.1126/science.1136259.

I. G. McCarthy, M. L. Balogh, A. Babul, G. B. Poole, and D. J. Horner. Models of the Intracluster Medium with Heating and Cooling: Explaining the Global and Structural X-Ray Properties of Clusters. ApJ, 613:811–830, October 2004. doi: 10.1086/423267.

I. G. McCarthy, R. G. Bower, M. L. Balogh, G. M. Voit, F. R. Pearce, T. Theuns, A. Babul, C. G. Lacey, and C. S. Frenk. Modelling shock heating in cluster mergers - I. Moving beyond the spherical accretion model. MNRAS, 376:497–522, April 2007. doi: 10.1111/j.1365-2966.2007.11465.x.

I. G. McCarthy, A. Babul, R. G. Bower, and M. L. Balogh. Towards a holistic view of the heating and cooling of the intracluster medium. MNRAS, 386:1309–1331, May 2008a. doi: 10.1111/j.1365-2966.2008.13141.x.

I. G. McCarthy, C. S. Frenk, A. S. Font, C. G. Lacey, R. G. Bower, N. L. Mitchell, M. L. Balogh, and T. Theuns. Ram pressure stripping the hot gaseous haloes of galaxies in groups and clusters. MNRAS, 383:593–605, January 2008b. doi: 10.1111/j.1365-2966.2007.12577.x.

S. S. McGaugh and W. J. G. de Blok. Gas Mass Fractions and the Evolution of Spiral Galaxies. ApJ, 481:689–+, May 1997. doi: 10.1086/304100.

C. F. McKee and E. C. Ostriker. Theory of Star Formation. ARA&A, 45:565–687, September 2007. doi: 10.1146/annurev.astro.45.051806.110602.

B. R. McNamara and P. E. J. Nulsen. Heating Hot Atmospheres with Active Galactic Nuclei. ARA&A, 45:117–175, September 2007. doi: 10.1146/annurev.astro.45.051806.110625.

N. L. Mitchell, I. G. McCarthy, R. G. Bower, T. Theuns, and R. A. Crain. On the origin of cores in simulated galaxy clusters. MNRAS, 395:180–196, May 2009. doi: 10.1111/j.1365-2966.2009.14550.x.

H. J. Mo, S. Mao, and S. D. M. White. The formation of galactic discs. MNRAS, 295: 319–336, April 1998. doi: 10.1046/j.1365-8711.1998.01227.x.

J. J. Monaghan. Smoothed particle hydrodynamics. ARA&A, 30:543–574, 1992. doi: 10.1146/annurev.aa.30.090192.002551.

J. J. Monaghan. SPH and Riemann Solvers. *Journal of Computational Physics*, 136: 298–307, September 1997. doi: 10.1006/jcph.1997.5732.

J. J. Monaghan. Smoothed Particle Hydrodynamics Code Basics. *Journal of Korean Astronomical Society*, 34:203–207, December 2001.

M. Mori and A. Burkert. Gas Stripping of Dwarf Galaxies in Clusters of Galaxies. ApJ, 538:559–568, August 2000. doi: 10.1086/309140.

M. Mori, Y. Yoshii, T. Tsujimoto, and K. Nomoto. The Evolution of Dwarf Galaxies with Star Formation in an Outward-propagating Supershell. ApJ, 478:L21+, March 1997. doi: 10.1086/310547.

P. M. Motl, J. O. Burns, C. Loken, M. L. Norman, and G. Bryan. Formation of Cool Cores in Galaxy Clusters via Hierarchical Mergers. ApJ, 606:635–653, May 2004. doi: 10.1086/381566.

S. D. Murray and D. N. C. Lin. Energy Dissipation in Multiphase Infalling Clouds in Galaxy Halos. ApJ, 615:586–594, November 2004. doi: 10.1086/424658.

S. D. Murray, S. D. M. White, J. M. Blondin, and D. N. C. Lin. Dynamical instabilities in two-phase media and the minimum masses of stellar systems. ApJ, 407:588–596, April 1993. doi: 10.1086/172540.

D. Nagai, A. V. Kravtsov, and A. Vikhlinin. Effects of Galaxy Formation on Thermodynamics of the Intracluster Medium. ApJ, 668:1–14, October 2007. doi: 10.1086/521328.

R. Narayan and M. V. Medvedev. Thermal Conduction in Clusters of Galaxies. ApJ, 562:L129–L132, December 2001. doi: 10.1086/338325.

J. F. Navarro, C. S. Frenk, and S. D. M. White. The Structure of Cold Dark Matter Halos. ApJ, 462:563–+, May 1996. doi: 10.1086/177173.

J. F. Navarro, C. S. Frenk, and S. D. M. White. A Universal Density Profile from Hierarchical Clustering. ApJ, 490:493–+, December 1997. doi: 10.1086/304888.

J. F. Navarro, L. V. Sales, J. Schaye, C. Dalla Vecchia, and V. Springel. Disk galaxies at z=2 in OWLS. *ArXiv e-prints*, May 2009.

A. F. Neto, L. Gao, P. Bett, S. Cole, J. F. Navarro, C. S. Frenk, S. D. M. White, V. Springel, and A. Jenkins. The statistics of Λ CDM halo concentrations. MNRAS, 381:1450–1462, November 2007. doi: 10.1111/j.1365-2966.2007.12381.x.

S.-M. Niemi, P. Heinämäki, P. Nurmi, and E. Saar. Formation, Evolution and Properties of Isolated Field Elliptical Galaxies. *ArXiv e-prints*, February 2010.

P. E. J. Nulsen. Transport processes and the stripping of cluster galaxies. MNRAS, 198: 1007–1016, March 1982.

R. W. O'Connell and B. R. McNamara. The Fate of Matter in Cooling Flows. AJ, 98: 180–+, July 1989. doi: 10.1086/115135.

P. Ocvirk, C. Pichon, and R. Teyssier. Bimodal gas accretion in the Horizon-MareNostrum galaxy formation simulation. MNRAS, 390:1326–1338, November 2008. doi: 10.1111/j.1365-2966.2008.13763.x.

S. S. R. Offner, R. I. Klein, and C. F. McKee. Driven and Decaying Turbulence Simulations of Low-Mass Star Formation: From Clumps to Cores to Protostars. ApJ, 686: 1174–1194, October 2008. doi: 10.1086/590238.

T. Okamoto and M. Nagashima. Environmental Effects on Evolution of Cluster Galaxies in a Λ-dominated Cold Dark Matter Universe. ApJ, 587:500–513, April 2003. doi: 10.1086/368251.

N. Oohama, S. Okamura, M. Fukugita, N. Yasuda, and O. Nakamura. Properties of Disks and Bulges of Spiral and Lenticular Galaxies in the Sloan Digital Sky Survey. ApJ, 705:245–254, November 2009. doi: 10.1088/0004-637X/705/1/245.

B. W. O'Shea, G. Bryan, J. Bordner, M. L. Norman, T. Abel, R. Harkness, and A. Kritsuk. Introducing Enzo, an AMR Cosmology Application. *ArXiv Astrophysics e-prints*, March 2004.

B. W. O'Shea, K. Nagamine, V. Springel, L. Hernquist, and M. L. Norman. Comparing AMR and SPH Cosmological Simulations. I. Dark Matter and Adiabatic Simulations. ApJS, 160:1–27, September 2005. doi: 10.1086/432645.

T. Padmanabhan. *Theoretical Astrophysics, Volume III: Galaxies and Cosmology*. October 2002.

F. R. Pearce and H. M. P. Couchman. Hydra: a parallel adaptive grid code. *New Astronomy*, 2:411–427, November 1997. doi: 10.1016/S1384-1076(97)00025-0.

S. Perlmutter, G. Aldering, G. Goldhaber, R. A. Knop, P. Nugent, P. G. Castro, S. Deustua, S. Fabbro, A. Goobar, D. E. Groom, I. M. Hook, A. G. Kim, M. Y. Kim, J. C. Lee, N. J. Nunes, R. Pain, C. R. Pennypacker, R. Quimby, C. Lidman, R. S. Ellis, M. Irwin, R. G. McMahon, P. Ruiz-Lapuente, N. Walton, B. Schaefer, B. J. Boyle, A. V. Filippenko, T. Matheson, A. S. Fruchter, N. Panagia, H. J. M. Newberg, W. J. Couch, and The Supernova Cosmology Project. Measurements of Omega and Lambda from 42 High-Redshift Supernovae. ApJ, 517:565–586, June 1999. doi: 10.1086/307221.

J. R. Peterson, S. M. Kahn, F. B. S. Paerels, J. S. Kaastra, T. Tamura, J. A. M. Bleeker, C. Ferrigno, and J. G. Jernigan. High-Resolution X-Ray Spectroscopic Constraints on Cooling-Flow Models for Clusters of Galaxies. ApJ, 590:207–224, June 2003. doi: 10.1086/374830.

G. B. Poole, M. A. Fardal, A. Babul, I. G. McCarthy, T. Quinn, and J. Wadsley. The impact of mergers on relaxed X-ray clusters - I. Dynamical evolution and emergent transient structures. MNRAS, 373:881–905, December 2006. doi: 10.1111/j.1365-2966.2006.10916.x.

G. B. Poole, A. Babul, I. G. McCarthy, M. A. Fardal, C. J. Bildfell, T. Quinn, and A. Mahdavi. The impact of mergers on relaxed X-ray clusters - II. Effects on global X-ray and Sunyaev-Zel'dovich properties and their scaling relations. MNRAS, 380: 437–454, September 2007. doi: 10.1111/j.1365-2966.2007.12107.x.

G. B. Poole, A. Babul, I. G. McCarthy, A. J. R. Sanderson, and M. A. Fardal. The impact of mergers on relaxed X-ray clusters - III. Effects on compact cool cores. MNRAS, 391: 1163–1175, December 2008. doi: 10.1111/j.1365-2966.2008.14003.x.

W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes in FORTRAN. The art of scientific computing.* 1992.

D. J. Price. Modelling discontinuities and Kelvin Helmholtz instabilities in SPH. *Journal of Computational Physics*, 227:10040–10057, December 2008. doi: 10.1016/j.jcp.2008.08.011.

V. Quilis, B. Moore, and R. Bower. Gone with the Wind: The Origin of S0 Galaxies in Clusters. *Science*, 288:1617–1620, June 2000. doi: 10.1126/science.288.5471.1617.

P. Rebusco, E. Churazov, H. Böhringer, and W. Forman. Impact of stochastic gas motions on galaxy cluster abundance profiles. MNRAS, 359:1041–1048, May 2005. doi: 10.1111/j.1365-2966.2005.08965.x.

P. Rebusco, E. Churazov, H. Böhringer, and W. Forman. Effect of turbulent diffusion on iron abundance profiles. MNRAS, 372:1840–1850, November 2006. doi: 10.1111/j.1365-2966.2006.10977.x.

P. M. Ricker and C. L. Sarazin. Off-Axis Cluster Mergers: Effects of a Strongly Peaked Dark Matter Profile. ApJ, 561:621–644, November 2001. doi: 10.1086/323365.

M. G. Rickes, M. G. Pastoriza, and C. Bonatto. Star formation, metallicity gradient and ionized gas: clues to the formation of the elliptical galaxies NGC6868 and 5903. MNRAS, 384:1427–1436, March 2008. doi: 10.1111/j.1365-2966.2007.12724.x.

A. G. Riess, A. V. Filippenko, P. Challis, A. Clocchiatti, A. Diercks, P. M. Garnavich, R. L. Gilliland, C. J. Hogan, S. Jha, R. P. Kirshner, B. Leibundgut, M. M. Phillips, D. Reiss, B. P. Schmidt, R. A. Schommer, R. C. Smith, J. Spyromilio, C. Stubbs, N. B. Suntzeff, and J. Tonry. Observational Evidence from Supernovae for an Accelerating Universe and a Cosmological Constant. AJ, 116:1009–1038, September 1998. doi: 10.1086/300499.

B. W. Ritchie and P. A. Thomas. Hydrodynamic simulations of merging clusters of galaxies. MNRAS, 329:675–688, January 2002. doi: 10.1046/j.1365-8711.2002.05027.x.

B. E. Robertson and A. V. Kravtsov. Molecular Hydrogen and Global Star Formation Relations in Galaxies. ApJ, 680:1083–1111, June 2008. doi: 10.1086/587796.

B. E. Robertson, A. V. Kravtsov, N. Y. Gnedin, T. Abel, and D. H. Rudd. Computational Eulerian hydrodynamics and Galilean invariance. MNRAS, 401:2463–2476, February 2010. doi: 10.1111/j.1365-2966.2009.15823.x.

E. Roediger and M. Brüggen. Ram pressure stripping of disc galaxies: the role of the inclination angle. MNRAS, 369:567–580, June 2006. doi: 10.1111/j.1365-2966.2006.10335.x.

E. Roediger and M. Brüggen. Ram pressure stripping of disc galaxies orbiting in clusters - I. Mass and radius of the remaining gas disc. MNRAS, 380:1399–1408, October 2007. doi: 10.1111/j.1365-2966.2007.12241.x.

E. Roediger and M. Brüggen. Ram-pressure stripping of disc galaxies orbiting in clusters - II. Galactic wakes. MNRAS, 388:465–486, August 2008. doi: 10.1111/j.1365-2966.2008.13415.x.

E. Roediger, M. Brüggen, P. Rebusco, H. Böhringer, and E. Churazov. Metal mixing by buoyant bubbles in galaxy clusters. MNRAS, 375:15–28, February 2007. doi: 10.1111/j.1365-2966.2006.11300.x.

N. Roy, L. Peedikakkandy, and J. N. Chengalur. Turbulence measurements from HI absorption spectra. MNRAS, 387:L18–L22, June 2008. doi: 10.1111/j.1745-3933.2008.00473.x.

V. C. Rubin. Dark matter in spiral galaxies. *Scientific American*, 248:96–106, June 1983.

V. C. Rubin, W. K. J. Ford, and N. . Thonnard. Rotational properties of 21 SC galaxies with a large range of luminosities and radii, from NGC 4605 /R = 4kpc/ to UGC 2885 /R = 122 kpc/. ApJ, 238:471–487, June 1980. doi: 10.1086/158003.

D. S. Rupke, S. Veilleux, and D. B. Sanders. Keck Absorption-Line Spectroscopy of Galactic Winds in Ultraluminous Infrared Galaxies. ApJ, 570:588–609, May 2002. doi: 10.1086/339789.

M. Ruszkowski, M. Brüggen, and M. C. Begelman. Cluster Heating by Viscous Dissipation of Sound Waves. ApJ, 611:158–163, August 2004. doi: 10.1086/422158.

D. Ryu, J. P. Ostriker, H. Kang, and R. Cen. A cosmological hydrodynamic code based on the total variation diminishing scheme. ApJ, 414:1–19, September 1993. doi: 10.1086/173051.

R. K. Sachs and A. M. Wolfe. Perturbations of a Cosmological Model and Angular Variations of the Microwave Background. ApJ, 147:73–+, January 1967. doi: 10.1086/148982.

E. E. Salpeter. The Luminosity Function and Stellar Evolution. ApJ, 121:161–+, January 1955. doi: 10.1086/145971.

C. L. Sarazin. *X-ray emission from clusters of galaxies*. 1988.

J. Schaye. Star Formation Thresholds and Galaxy Edges: Why and Where. ApJ, 609: 667–682, July 2004. doi: 10.1086/421232.

J. Schaye and C. Dalla Vecchia. On the relation between the Schmidt and Kennicutt-Schmidt star formation laws and its implications for numerical simulations. MNRAS, 383:1210–1222, January 2008. doi: 10.1111/j.1365-2966.2007.12639.x.

J. Schaye, C. Dalla Vecchia, C. M. Booth, R. P. C. Wiersma, T. Theuns, M. R. Haas, S. Bertone, A. R. Duffy, I. G. McCarthy, and F. van de Voort. The physics driving the cosmic star formation history. MNRAS, 402:1536–1560, March 2010. doi: 10.1111/j.1365-2966.2009.16029.x.

R. W. Schmidt, A. C. Fabian, and J. S. Sanders. Chandra temperature and metallicity maps of the Perseus cluster core. MNRAS, 337:71–78, November 2002. doi: 10.1046/j.1365-8711.2002.05804.x.

P. Schuecker, A. Finoguenov, F. Miniati, H. Böhringer, and U. G. Briel. Probing turbulence in the Coma galaxy cluster. A&A, 426:387–397, November 2004. doi: 10.1051/0004-6361:20041039.

L. I. Sedov. *Similarity and Dimensional Methods in Mechanics*. 1959.

P. R. Shapiro, H. Martel, J. V. Villumsen, and J. M. Owen. Adaptive Smoothed Particle Hydrodynamics, with Application to Cosmology: Methodology. ApJS, 103:269–+, April 1996. doi: 10.1086/192279.

A. E. Shapley, C. C. Steidel, M. Pettini, and K. L. Adelberger. Rest-Frame Ultraviolet Spectra of z˜3 Lyman Break Galaxies. ApJ, 588:65–89, May 2003. doi: 10.1086/373922.

H. Shapley. A Determination of the Distance to the Galactic Center. *Proceedings of the National Academy of Science*, 25:113–118, March 1939. doi: 10.1073/pnas.25.3.113.

D. Sijacki and V. Springel. Physical viscosity in smoothed particle hydrodynamics simulations of galaxy clusters. MNRAS, 371:1025–1046, September 2006. doi: 10.1111/j.1365-2966.2006.10752.x.

D. Sijacki, C. Pfrommer, V. Springel, and T. A. Enßlin. Simulations of cosmic-ray feedback by active galactic nuclei in galaxy clusters. MNRAS, 387:1403–1415, July 2008. doi: 10.1111/j.1365-2966.2008.13310.x.

J. D. Simon, A. D. Bolatto, A. Leroy, L. Blitz, and E. L. Gates. High-Resolution Measurements of the Halos of Four Dark Matter-Dominated Galaxies: Deviations from a Universal Density Profile. ApJ, 621:757–776, March 2005. doi: 10.1086/427684.

M. F. Skrutskie, R. M. Cutri, R. Stiening, M. D. Weinberg, S. Schneider, J. M. Carpenter, C. Beichman, R. Capps, T. Chester, J. Elias, J. Huchra, J. Liebert, C. Lonsdale, D. G. Monet, S. Price, P. Seitzer, T. Jarrett, J. D. Kirkpatrick, J. E. Gizis, E. Howard, T. Evans, J. Fowler, L. Fullmer, R. Hurt, R. Light, E. L. Kopan, K. A. Marsh, H. L. McCallon, R. Tam, S. Van Dyk, and S. Wheelock. The Two Micron All Sky Survey (2MASS). AJ, 131:1163–1183, February 2006. doi: 10.1086/498708.

G. F. Smoot, C. L. Bennett, A. Kogut, E. L. Wright, J. Aymon, N. W. Boggess, E. S. Cheng, G. de Amici, S. Gulkis, M. G. Hauser, G. Hinshaw, P. D. Jackson, M. Janssen, E. Kaita, T. Kelsall, P. Keegstra, C. Lineweaver, K. Loewenstein, P. Lubin, J. Mather, S. S. Meyer, S. H. Moseley, T. Murdock, L. Rokke, R. F. Silverberg, L. Tenorio,

R. Weiss, and D. T. Wilkinson. Structure in the COBE differential microwave radiometer first-year maps. ApJ, 396:L1–L5, September 1992. doi: 10.1086/186504.

G. A. Sod. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *Journal of Computational Physics*, 27:1–31, April 1978. doi: 10.1016/0021-9991(78)90023-2.

V. Springel. The cosmological simulation code GADGET-2. MNRAS, 364:1105–1134, December 2005. doi: 10.1111/j.1365-2966.2005.09655.x.

V. Springel. E pur si muove: Galilean-invariant cosmological hydrodynamical simulations on a moving mesh. MNRAS, 401:791–851, January 2010. doi: 10.1111/j.1365-2966.2009.15715.x.

V. Springel and L. Hernquist. Cosmological smoothed particle hydrodynamics simulations: the entropy equation. MNRAS, 333:649–664, July 2002. doi: 10.1046/j.1365-8711.2002.05445.x.

V. Springel and L. Hernquist. Cosmological smoothed particle hydrodynamics simulations: a hybrid multiphase model for star formation. MNRAS, 339:289–311, February 2003. doi: 10.1046/j.1365-8711.2003.06206.x.

V. Springel, N. Yoshida, and S. D. M. White. GADGET: a code for collisionless and gasdynamical cosmological simulations. *New Astronomy*, 6:79–117, April 2001. doi: 10.1016/S1384-1076(01)00042-2.

V. Springel, T. Di Matteo, and L. Hernquist. Modelling feedback from stars and black holes in galaxy mergers. MNRAS, 361:776–794, August 2005a. doi: 10.1111/j.1365-2966.2005.09238.x.

V. Springel, S. D. M. White, A. Jenkins, C. S. Frenk, N. Yoshida, L. Gao, J. Navarro, R. Thacker, D. Croton, J. Helly, J. A. Peacock, S. Cole, P. Thomas, H. Couchman, A. Evrard, J. Colberg, and F. Pearce. Simulations of the formation, evolution and clustering of galaxies and quasars. Nature, 435:629–636, June 2005b. doi: 10.1038/nature03597.

V. Springel, J. Wang, M. Vogelsberger, A. Ludlow, A. Jenkins, A. Helmi, J. F. Navarro, C. S. Frenk, and S. D. M. White. The Aquarius Project: the subhaloes of galactic haloes. MNRAS, 391:1685–1711, December 2008. doi: 10.1111/j.1365-2966.2008.14066.x.

M. Steinmetz and S. D. M. White. Two-body heating in numerical galaxy formation experiments. MNRAS, 288:545–550, July 1997.

G. Stinson, A. Seth, N. Katz, J. Wadsley, F. Governato, and T. Quinn. Star formation and feedback in smoothed particle hydrodynamic simulations - I. Isolated galaxies. MNRAS, 373:1074–1090, December 2006. doi: 10.1111/j.1365-2966.2006.11097.x.

J. M. Stone and M. L. Norman. ZEUS-2D: A radiation magnetohydrodynamics code for astrophysical flows in two space dimensions. I - The hydrodynamic algorithms and tests. ApJS, 80:753–790, June 1992. doi: 10.1086/191680.

J. M. Stone and E. G. Zweibel. MHD Stability of Interstellar Medium Phase Transition Layers. I. Magnetic Field Orthogonal to Front. ApJ, 696:233–240, May 2009. doi: 10.1088/0004-637X/696/1/233.

G. Strang. On the construction and comparison of difference schemes. *SIAM Journal on Numerical Analysis*, 5:506–517, September 1968.

M. Sun, C. Jones, W. Forman, A. Vikhlinin, M. Donahue, and M. Voit. X-Ray Thermal Coronae of Galaxies in Hot Clusters: Ubiquity of Embedded Mini-Cooling Cores. ApJ, 657:197–231, March 2007. doi: 10.1086/510895.

R. A. Swaters, M. A. W. Verheijen, M. A. Bershady, and D. R. Andersen. The Kinematics in the Core of the Low Surface Brightness Galaxy DDO 39. ApJ, 587:L19–L22, April 2003. doi: 10.1086/375045.

T. Tamura, J. S. Kaastra, J. W. A. den Herder, J. A. M. Bleeker, and J. R. Peterson. Elemental abundances in the intracluster medium as observed with XMM-Newton. A&A, 420:135–146, June 2004. doi: 10.1051/0004-6361:20040103.

E. J. Tasker and G. L. Bryan. Simulating Star Formation and Feedback in Galactic Disk Models. ApJ, 641:878–890, April 2006. doi: 10.1086/500567.

E. J. Tasker and G. L. Bryan. The Effect of the Interstellar Model on Star Formation Properties in Galactic Disks. ApJ, 673:810–831, February 2008. doi: 10.1086/523889.

E. J. Tasker, R. Brunino, N. L. Mitchell, D. Michielsen, S. Hopton, F. R. Pearce, G. L. Bryan, and T. Theuns. A test suite for quantitative comparison of hydrodynamic codes in astrophysics. MNRAS, 390:1267–1281, November 2008. doi: 10.1111/j.1365-2966.2008.13836.x.

R. J. Thacker and H. M. P. Couchman. Implementing Feedback in Simulations of Galaxy Formation: A Survey of Methods. ApJ, 545:728–752, December 2000. doi: 10.1086/317828.

R. J. Thacker, E. R. Tittley, F. R. Pearce, H. M. P. Couchman, and P. A. Thomas. Smoothed Particle Hydrodynamics in cosmology: a comparative study of implementations. MNRAS, 319:619–648, December 2000. doi: 10.1046/j.1365-8711.2000.03927.x.

S. Tonnesen, G. L. Bryan, and J. H. van Gorkom. Environmentally Driven Evolution of Simulated Cluster Galaxies. ApJ, 671:1434–1445, December 2007. doi: 10.1086/523034.

A. Toomre. On the gravitational stability of a disk of stars. ApJ, 139:1217–1238, May 1964. doi: 10.1086/147861.

G. Tormen, A. Diaferio, and D. Syer. Survival of substructure within dark matter haloes. MNRAS, 299:728–742, September 1998. doi: 10.1046/j.1365-8711.1998.01775.x.

H. Trac, A. Sills, and U.-L. Pen. A comparison of hydrodynamic techniques for modelling collisions between main-sequence stars. MNRAS, 377:997–1005, May 2007. doi: 10.1111/j.1365-2966.2007.11709.x.

J. H. van Gorkom. Interaction of Galaxies with the Intracluster Medium. In J. S. Mulchaey, A. Dressler, & A. Oemler, editor, *Clusters of Galaxies: Probes of Cosmological Structure and Galaxy Evolution*, pages 305–+, 2004.

S. Veilleux, G. Cecil, and J. Bland-Hawthorn. Galactic Winds. ARA&A, 43:769–826, September 2005. doi: 10.1146/annurev.astro.43.072103.150610.

G. M. Voit. Tracing cosmic evolution with clusters of galaxies. *Reviews of Modern Physics*, 77:207–258, April 2005. doi: 10.1103/RevModPhys.77.207.

G. M. Voit and M. Donahue. Problems with Cold Clouds and Cooling Flows. ApJ, 452: 164–+, October 1995. doi: 10.1086/176288.

G. M. Voit and M. Donahue. An Observationally Motivated Framework for AGN Heating of Cluster Cores. ApJ, 634:955–963, December 2005. doi: 10.1086/497063.

G. M. Voit and T. J. Ponman. Signatures of Galaxy Formation in the Intracluster Medium. ApJ, 594:L75–L78, September 2003. doi: 10.1086/378627.

G. M. Voit, G. L. Bryan, M. L. Balogh, and R. G. Bower. Modified Entropy Models for the Intracluster Medium. ApJ, 576:601–624, September 2002. doi: 10.1086/341864.

G. M. Voit, M. L. Balogh, R. G. Bower, C. G. Lacey, and G. L. Bryan. On the Origin of Intracluster Entropy. ApJ, 593:272–290, August 2003. doi: 10.1086/376499.

G. M. Voit, S. T. Kay, and G. L. Bryan. The baseline intracluster entropy profile from gravitational structure formation. MNRAS, 364:909–916, December 2005. doi: 10.1111/j.1365-2966.2005.09621.x.

K. Wada and C. A. Norman. Density Structure of the Interstellar Medium and the Star Formation Rate in Galactic Disks. ApJ, 660:276–287, May 2007. doi: 10.1086/513002.

J. W. Wadsley, G. Veeravalli, and H. M. P. Couchman. On the treatment of entropy mixing in numerical cosmology. MNRAS, 387:427–438, June 2008. doi: 10.1111/j.1365-2966.2008.13260.x.

Q. D. Wang, H. Dong, A. Cotera, S. Stolovy, M. Morris, C. C. Lang, M. P. Muno, G. Schneider, and D. Calzetti. HST/NICMOS Paschen-α Survey of the Galactic Centre: Overview. MNRAS, 402:895–902, February 2010. doi: 10.1111/j.1365-2966.2009.15973.x.

S. M. Weinmann, F. C. van den Bosch, X. Yang, H. J. Mo, D. J. Croton, and B. Moore. Properties of galaxy groups in the Sloan Digital Sky Survey - II. Active galactic nucleus feedback and star formation truncation. MNRAS, 372:1161–1174, November 2006. doi: 10.1111/j.1365-2966.2006.10932.x.

S. D. M. White and C. S. Frenk. Galaxy formation through hierarchical clustering. ApJ, 379:52–79, September 1991. doi: 10.1086/170483.

R. P. C. Wiersma, J. Schaye, and B. D. Smith. The effect of photoionization on the cooling rates of enriched, astrophysical plasmas. MNRAS, 393:99–107, February 2009. doi: 10.1111/j.1365-2966.2008.14191.x.

B. Willman, M. R. Blanton, A. A. West, J. J. Dalcanton, D. W. Hogg, D. P. Schneider, N. Wherry, B. Yanny, and J. Brinkmann. A New Milky Way Companion: Unusual Globular Cluster or Extreme Dwarf Satellite? AJ, 129:2692–2700, June 2005a. doi: 10.1086/430214.

B. Willman, J. J. Dalcanton, D. Martinez-Delgado, A. A. West, M. R. Blanton, D. W. Hogg, J. C. Barentine, H. J. Brewington, M. Harvanek, S. J. Kleinman, J. Krzesinski, D. Long, E. H. Neilsen, Jr., A. Nitta, and S. A. Snedden. A New Milky Way Dwarf Galaxy in Ursa Major. ApJ, 626:L85–L88, June 2005b. doi: 10.1086/431760.

C. J. Willott, P. Delorme, A. Omont, J. Bergeron, X. Delfosse, T. Forveille, L. Albert, C. Reylé, G. J. Hill, M. Gully-Santiago, P. Vinten, D. Crampton, J. B. Hutchings, D. Schade, L. Simard, M. Sawicki, A. Beelen, and P. Cox. Four Quasars above Redshift 6 Discovered by the Canada-France High-z Quasar Survey. AJ, 134:2435–2450, December 2007. doi: 10.1086/522962.

C. J. Willott, P. Delorme, C. Reylé, L. Albert, J. Bergeron, D. Crampton, X. Delfosse, T. Forveille, J. B. Hutchings, R. J. McLure, A. Omont, and D. Schade. Six More Quasars at Redshift 6 Discovered by the Canada-France High-z Quasar Survey. AJ, 137:3541–3547, March 2009. doi: 10.1088/0004-6256/137/3/3541.

C. J. Willott, P. Delorme, C. Reylé, L. Albert, J. Bergeron, D. Crampton, X. Delfosse, T. Forveille, J. B. Hutchings, R. J. McLure, A. Omont, and D. Schade. The Canada-France High-z Quasar Survey: Nine New Quasars and the Luminosity Function at Redshift 6. AJ, 139:906–918, March 2010. doi: 10.1088/0004-6256/139/3/906.

K. K. S. Wu, A. C. Fabian, and P. E. J. Nulsen. Non-gravitational heating in the hierarchical formation of X-ray clusters. MNRAS, 318:889–912, November 2000. doi: 10.1046/j.1365-8711.2000.03828.x.

G. Yepes, R. Kates, A. Khokhlov, and A. Klypin. Hydrodynamical simulations of galaxy formation: effects of supernova feedback. MNRAS, 284:235–256, January 1997.

J. D. Younger and G. L. Bryan. Cosmological Simulations of the Preheating Scenario for Galaxy Cluster Formation: Comparison to Analytic Models and Observations. ApJ, 666:647–657, September 2007. doi: 10.1086/520619.

M. Zeilik and S. Gregory. *Introductory Astronomy and Astrophysics*. Harcourt College Publishers, 1998.

W. Zhang, A. MacFadyen, and P. Wang. Three-Dimensional Relativistic Magnetohydrodynamic Simulations of the Kelvin-Helmholtz Instability: Magnetic Field Amplification by a Turbulent Dynamo. ApJ, 692:L40–L44, February 2009. doi: 10.1088/0004-637X/692/1/L40.

D. B. Zucker, V. Belokurov, N. W. Evans, J. T. Kleyna, M. J. Irwin, M. I. Wilkinson, M. Fellhauer, D. M. Bramich, G. Gilmore, H. J. Newberg, B. Yanny, J. A. Smith, P. C. Hewett, E. F. Bell, H.-W. Rix, O. Y. Gnedin, S. Vidrih, R. F. G. Wyse, B. Willman, E. K. Grebel, D. P. Schneider, T. C. Beers, A. Y. Kniazev, J. C. Barentine, H. Brewington, J. Brinkmann, M. Harvanek, S. J. Kleinman, J. Krzesinski, D. Long, A. Nitta, and S. A. Snedden. A Curious Milky Way Satellite in Ursa Major. ApJ, 650:L41–L44, October 2006a. doi: 10.1086/508628.

D. B. Zucker, V. Belokurov, N. W. Evans, M. I. Wilkinson, M. J. Irwin, T. Sivarani, S. Hodgkin, D. M. Bramich, J. M. Irwin, G. Gilmore, B. Willman, S. Vidrih, M. Fellhauer, P. C. Hewett, T. C. Beers, E. F. Bell, E. K. Grebel, D. P. Schneider, H. J. Newberg, R. F. G. Wyse, C. M. Rockosi, B. Yanny, R. Lupton, J. A. Smith, J. C. Barentine, H. Brewington, J. Brinkmann, M. Harvanek, S. J. Kleinman, J. Krzesinski, D. Long, A. Nitta, and S. A. Snedden. A New Milky Way Dwarf Satellite in Canes Venatici. ApJ, 643:L103–L106, June 2006b. doi: 10.1086/505216.

F. Zwicky. Die Rotverschiebung von extragalaktischen Nebeln. *Helvetica Physica Acta*, 6:110–127, 1933.

F. Zwicky. On the Masses of Nebulae and of Clusters of Nebulae. ApJ, 86:217–+, October 1937. doi: 10.1086/143864.