Towards Efficient and Accurate Numerical Simulations of Galaxies using Task-based Parallelism and Application to Dwarf Galaxies

Thèse n. 1234 2021
présentée le 31 août 2021
à la Faculté des sciences de base
laboratoire LASTRO
programme doctoral en Physique
École polytechnique fédérale de Lausanne

pour l'obtention du grade de Docteur ès Sciences
par

Loic Hausammann

acceptée sur proposition du jury :

Prof. Ricci Paolo, président du jury
Dr. Revaz Yves, directeur de thèse
Prof. Kneib Jean-Paul, co-directeur de thèse
Prof. Deparis Simone, rapporteur
Prof. Teyssier Romain, rapporteur
Prof. Brooks Alyson, rapporteur

Lausanne, EPFL, 2021

Acknowledgements

> If more of us valued food and cheer and song above hoarded gold, it would be a merrier world.
> — J. R. R. Tolkien, The Hobbit

As in any good story, this PhD quest would not have been possible without many supporting characters completing various heroic deeds. Continuing with my analogy to J.R.R. Tolkien's universe a little longer, I dedicate this first part to my own Fellowship, who deserve all my gratitude for bringing me all the way to Mount Doom and back. Let me start by thanking Yves Revaz for taking me on as his first PhD student, spending his time teaching me about numerical astrophysics and showing me the correct path through my thesis. Thanks to his trust, I had the pleasure of growing from a young PhD student asking everyone many questions, to the old (and sometimes grumpy) PhD student answering the next generation's questions. While Yves gave me the position, it would not have been possible without the support of Jean-Paul Kneib.

I met the next members of my Fellowship in the `SWIFT` collaboration, that contains many remarkable scientists. Leading them, Matthieu Schaller has been a tremendous and reliable support with, amongst others, his advice, his thorough reviews of my code and his patience with some of my craziest ideas. As an unending source of knowledge and thanks to our many, many, and many discussions, he certainly had a very deep impact on the quality of my work. On the computational science side, I would like to thank Pedro Gonnet for sharing his knowledge and helping me improve my programming skills through our work on the Continuous Simulation Data Stream (CSDS). Due to the natural surface tension present between PhD students, I had the pleasure to get to know Josh Borrow as a passionate colleague whose knowledge of hydrodynamics and dedication for `SWIFT` merchandising I could always rely on. He certainly had a valuable influence on shaping the hydrodynamics presented in this thesis. Special thanks also goes out to Richard Bower for all the optimism and joy he brought during our meetings. Last but not least, as I think the hardest work is often unnoticed by regular users, I feel it is really important to thank Peter Draper for all his work on `SWIFT` that strongly facilitated my thesis.

As Yves' first PhD student, I started digging into codes and papers with only Yves' help to figure out what numerical astrophysics consist of. With the arrival of Mladen Ivkovic in LASTRO and in the `SWIFT` team, I had the pleasure to not only share my knowledge, but find a colleague to valiantly spar with me and thoroughly challenge my ideas, thanks to his attachment of understanding the details. Before I start with the next member of my fellowship, let me state that a group doing numerical simulations cannot be complete without a computer scientist. In our small group, Florian Cabot played this role and provided us (or maybe himself in light of how he speaks about the project) the amazing visual tool that is `VIRUP`. Through our collaboration on the integration of the CSDS into his own code, I had the pleasure to share long discussions on how Linux will rule the world. Without Florian and Mladen, my time spent at LASTRO would not have been the same and they both have my infinite gratitude for that.

In the last years of my thesis, I had the opportunity of collaborating with the `AGORA` group. Thanks to this experience, I extended my knowledge of galaxies to include the domain of Milky Way-like galaxies, as well as the adaptive mesh refinement technique. While I am grateful to all the members in this collaboration, I would like to give a special thanks to Santi Roca-Fàbrega for his close collaboration and help through out the project.

Drinking beers is rarely thought as an important part of a thesis, but through experimentation by many clearly unbiased PhD students, it is empirically proven to be a key source of disruptive ideas and should be fully acknowledged as a skill obtained with a PhD. While my partners in this continuous experiment have been numerous and will recognize themselves, I would like to especially thank Damien Spérone-Longin, Elodie Savary, Christoph Schäfer, Vivien Bonvin and Bert Vandenbroucke for their help at keeping up my motivation all the way to the finishing line.

While the objective of a thesis is about advancing the current knowledge of humankind (or at least tying), it definitely requires blindly diving into the abyssal sea that is the administrative world and I would like to thank Sophie Oblette for guiding me through it as Gollum guided Sam and Frodo to Mordor, but without the betrayal at the end.

Let me finish these acknowledgments by thanking my friends and familly. First, I would like to thank my parents, Françoise and Urs Hausammann, but also my stepfather Marc Wittwer for supporting me through out my education journey, all the way into physics and this PhD. If they had not fought for me to be placed in the academic track despite my teachers' focus on my bad language grades, I might not have written this thesis. Next, I would like to thank all my friends for their suppport through out this thesis, which took many forms from providing me with stress outlets and making sure I stayed sane, to actually reviewing my thesis. Here is a special call-out to Emil Rotilio, who took on that last challenge despite his busy lives (and my recommendations). Finally, I am very grateful to Christèle Zbinden, who probably got more than she signed up for by becoming my officemate during the coronavirus lockdowns in addition to being my loving girlfriend. Despite the long distance, the coronavirus-changed the circumstances and all the unsolicited (but, according to all the protests I ignored, deeply wanted) physics explanations, she still stuck by my side during this entire PhD to support me in my crazy quest, even pitching in to

correct my english here and there.

Lausanne, August 31, 2021 L. H.

Abstract

Numerical simulations are of a tremendous help to understand the growth of non-linear cosmological structures and how they lead to the formation of galaxies. In recent years, with the goal of improving their prediction power, new hydrodynamical techniques and physical models have been developed. Beside those improvements, the community must be ready for what could be a major paradigm change, the exascale supercomputer that will allow for the simulation of higher resolution and/or larger cosmological volume. Being ready to fully benefit from those upcoming facilities requires an important effort in adapting existing codes.

Among the domains that will benefit from the exascale computing are dwarf galaxies and their low mass end, the ultra faint dwarf galaxies (UFD). Since twenty years, dwarfs have become an important research topic in astrophysics due to recent observing facilities allowing a large number of unprecedented observations. Simulations of such galaxies require extremely high resolution down to the individual star level and, thus, require the adaptation of existing physical models.

In the first part of my thesis, I extended a previous work performed with the code `GEAR` on understanding the formation of dwarf galaxies in a cosmological context. I studied the ram pressure stripping mechanism due to the hot halo of the Milky Way and how it modifies the observed properties of dwarf galaxies. I demonstrated that the thermal pressure due to the hot halo on the dwarf's gas can strongly impact the effect of ram pressure.

To ensure the quality of our simulations, I have participated in the `AGORA` project that aims at comparing the predictions of different simulation codes and understanding the underlying differences. In the last `AGORA` paper, we have performed the first comparison of cosmological simulations using common models for the radiative cooling and star formation.

In the second part of this thesis, an important effort has been dedicated to migrate our physical models from our code `GEAR` towards `SWIFT`. Thanks to the improved performance of `SWIFT` (speedup of 7.65x) and the successful migration, our code is now fast enough to simulate UFDs at the resolution of individual stars. On the physics side, our star formation and stellar feedback methods need to be further improved due to the high resolution required in UFDs. To solve this issue, I designed and started the implementation of a new star formation scheme based on a new type of particles called "sink particles".

Going towards the exascale also requires a better design of the output system previously based on snapshots of the simulation saved at regular time intervals. In this purpose, I introduced the "Continuous Simulation Data Stream" (CSDS) specially designed for simulations where gravity is producing strong differences in timescale between particles or cells. The CSDS reduces the disk space used by a single simulation while improving the time resolution of the output and making the analysis easier and flexible.

Finally, while the simulation code and the associated physical models have a tremendous impact on the quality of the simulations, the initial conditions need to be carefully designed as they can deeply impact the results. Using a Bayesian approach, I have developed my own code that generates constrained initial conditions that reproduce the Local Group.

Keywords: Dwarf galaxies – Numerical Simulations – Ram Pressure Stripping – Thermal Pressure – Constrained Initial Conditions – Star Formation – Input/Output – Chemical Evolution – Galaxies Evolution

Résumé

Les simulations numériques sont d'une grande aide pour comprendre la croissance des structures cosmologiques dans leurs parties non linéaire et la façon dont les galaxies sont formées. Récemment, de nouvelles techniques ont été développées pour l'hydrodynamique ainsi que de nouveaux modèles physiques. En dehors de ces améliorations, la communauté doit être prête pour un changement majeur avec l'arrivée des super-calculateurs exaflopiques qui permettront des simulations avec une plus grande résolution et/ou un plus grand volume cosmologique. Etre prêt à bénéficier de ces prochains calculateurs demande de larges efforts pour adapter les codes existants.

Parmi les domaines qui vont en profiter, il y a les galaxies naines ainsi que la sous catégories des galaxies ultra pâles (UFD). Depuis vingts ans, les naines sont devenues un important sujet de recherche en astrophysique grâce aux récents télescopes qui permettent un large nombre de nouvelles observations. Les simulations de ces galaxies demandent des résolutions extrêmement élevée, au niveau des étoiles individuelles, ainsi que l'adaptation de nos modèles physiques.

Dans la première partie de ma thèse, j'ai étendu les travaux faits précédemment avec le code `GEAR` sur la compréhension de la formation de galaxies naines. Je me suis concentré sur le mécanisme d'effeuillage des galaxies au travers de la pression de bélier due au halo chaud de la Voie Lactée et comment il modifie les propriétés des galaxies naines observées. J'ai montré que la pression thermique du halo de la Voie Lactée sur les satellites peut fortement impacter les effets de la pression de bélier.

Pour s'assurer de la qualité de nos simulations, j'ai participé au projet `AGORA` qui vise à comparer les prédictions de différents codes et à comprendre les raisons des différences. Dans la dernière publication d'`AGORA`, on a produit la première comparaison d'une simulation cosmologique en utilisant des modèles communs pour le refroidissement radiatif et la formation stellaire.

Dans la deuxième partie, un effort important a été dédié à la migration de nos modèles physiques depuis le code `GEAR` vers le code `SWIFT`. Grâce à l'amélioration des performances qu'apporte `SWIFT` et la migration des modèles réussie, notre code peut maintenant simuler des UFDs à la résolution requise. Du côté de la physique, notre méthode pour la formation stellaire a besoin d'être améliorée pour ce type de simulation. J'ai donc conceptualisé et commencé l'implémentation d'une nouvelle méthode de formation stellaire basée sur un nouveau type de particules.

Pour atteindre l'échelle exaflopique, il faut aussi un système plus performant pour stocker les simulations. L'approche usuelle consiste à écrire l'état complet d'une simulation à intervalle régulier. Pour améliorer cela, j'ai développé le système de flux continu d'information contenu dans la simulation spécialement conçu pour des simulations où la gravité produit de fortes différences d'échelle de temps entre particules. Cette approche réduit la taille du disque utilisée en plus d'augmenter la résolution temporelle.

Finalement, bien que le code utilisé pour les simulations et les modèles associés ont un large impact sur la qualité des simulations, les conditions initiales ont elles aussi un fort impact sur les résultats. En utilisant une approche Bayesienne, j'ai développé mon propre code qui génère des conditions initiales contraintes qui reproduisent le Groupe local.

Mots-clés : Galaxies naines – Simulations numériques – Effeuillage par pression bélier – pression thermique – Conditions initiales contraintes – Formation stellaire – Entrée/Sortie (I/O) – Evolution chimique – Evolution des galaxies

Contents

V Conclusion 239

Introduction Part I

1 The Impact of Dwarf Galaxies

> Then darkness took me, and I strayed out of thought and time, and I wandered far on roads that I will not tell. Naked I was sent back – for a brief time, until my task is done.
> — J. R. R. Tolkien, The Two Towers

Dwarf galaxies are the smallest galaxies in our universe. They are born in the early universe from tiny initial fluctuations that grew through the ages thanks to gravity. Although the expansion of the universe diluted the matter, the gravitational forces within these fluctuations kept them compact and made them grow. Today dwarf galaxies can be observed all around the Milky Way and Andromeda with some powerful telescopes and some of them can even be seen without instrument (for example the Small and Large Magellanic Clouds). Thanks to numerical simulations, we are now able to make the link between the observed Cosmic Microwave Background representing the early universe and recent observations of the Local Universe. In this introduction, I will start by providing a quick summary of the main theoretical component of dwarf galaxies called dark matter along with the standard model of cosmology which has a strong impact on the dwarfs' evolution. Then, I will give an overview of our current knowledge on dwarf galaxies and why they can play an important role in our understanding of physics. Next, I will describe the current challenges in simulating galaxies. Finally I will discuss the topics of my thesis and their impact. A complete list of the abbreviations used in this thesis is available in table 1.1.

Before discussing the cold dark matter, a few words on the central concept of redshift are necessary. The light can be seen as a wave where the distance between peaks give the color. Due to the expansion of the universe, the light (or photon) that is not interacting with anything changes its color and become more and more red. Indeed, with the expansion, the distance between peaks is increased and thus shifts the color from blue (short wavelengths) to red (long wavelengths).

A photon traveling without any interaction for a long time will be far more redshifted than one traveling for a short time. Therefore from this redshift, the age of the photons can be measured but also the distance as photons have a constant speed. Therefore the concept of redshift represents both the distances and the time. A high redshift corresponds to large distances and early time of the universe while a low redshift corresponds to small distances and late time. This simple vision is in practice more complex as different sources of redshift exists. The dominant one is due to the peculiar velocities of the objects emitting the photons that Doppler shift the wavelength (either in blue or red direction).

1.1 Cold Dark Matter

The dark matter is one of the most intriguing problems of astrophysics. What is called dark matter is a collection of differences between the theory and observations that can be solved by adding a large amount of unknown and hidden matter sensitive to gravity in the universe. While the addition of unknown mass is the most commonly accepted solution to the dark matter problems, we still do not have any direct proof that it is the correct solution. Thus, some research is still done on other models such as MOND that includes changes to the gravity law in order to agree with observations. Here, I will consider only the cold dark matter model.

Early results on the dark matter comes from Fritz Zwicky in 1933. He studied the Coma cluster with the virial theorem applied to the velocity dispersion of 8 galaxies and discovered that the velocities were not compatible with the amount of visible matter (Zwicky, 2009). In the 70s, the search for dark matter really started with the measurement of rotation curves in disk galaxies at large radii. According to Newton's theory, the circular velocity of a system in equilibrium should be given by $v = \sqrt{GM(r)/r}$ where G is the gravitational constant and $M(r)$ is the gravitational mass contained inside a given radius r. Outside the radius of the visible matter distribution, the mass should stay constant if no dark matter existed and its velocity should decrease as $\sqrt{1/r}$. Observations have shown that the circular velocity instead increases at small radii and then remains constant around $100 - 300$ km/s for disk galaxies (Gaitskell, 2004; Persic et al., 1996; Battaner and Florido, 2000; Binney and Tremaine, 1987). In Figure 1.1, the rotation curve for M33 is shown along with the best fit from a model containing 3 components: dark matter, gas and stars. The rotation curve keeps increasing outside the radius of stars and thus strongly shows that a non-visible component is required (Corbelli and Salucci, 2000). At cluster scale, the gravitational potential becomes so deep that it starts to influence the trajectory of photons and produce some gravitational lensing. Through the shape of the lens, the matter distribution can be recovered (Newman et al., 2013; Zumalacárregui and Seljak, 2018), confirming the finding in the circular rotation curves.

Such studies already provide some insight on the density of dark matter in the universe. We can define the cosmological parameters $\Omega_x = \rho_x/\rho_c$ where x corresponds to the different parameters (dark matter, baryonic matter, radiation, curvature and dark energy in the standard model) and ρ_c

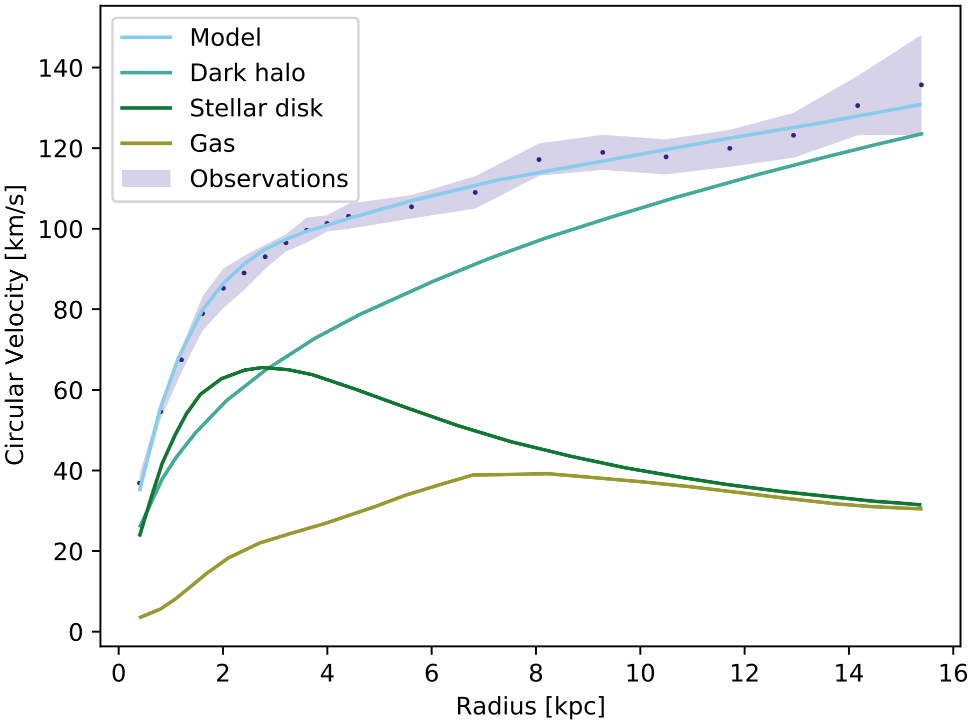


Figure 1.1 – Rotation curve for M33. The blue dots along with the grey area represent the observations. All the others lines represent the best fit of the model for the different components. Even if the impact of the visible components (gas and stars) is decreasing at large radii, the total circular velocity is still increasing due to the presence of dark matter (dark halo). The data have been extracted from Corbelli and Salucci, 2000.

is the critical density given by the cosmological model:

$$\rho_c = \frac{3H_0^2}{8\pi G}, \tag{1.1}$$

H_0 is the Hubble constant and G the gravitational constant. From the rotation curves, we can already have a lower bound for Ω_{dm}. It is possible to derive that 90% of the mass within galaxies is under the form of dark matter. The density of luminous matter is established to be given by $\Omega_{\text{lum}} \lesssim 0.01$, it means that $\Omega_{dm} > 0.1$ assuming that galaxies are an accurate representation of the complete universe (Gaitskell, 2004).

The Cosmic Microwave Background (CMB) provides an additional constraint on the abundance of dark matter. At the epoch of recombination ($z \approx 1100$), the protons and electrons combined to form hydrogen. Almost immediately after the recombination, photons started to decouple from the baryonic matter, and, for the first time, the universe became optically thin to radiation. The photons which decoupled at that time can still be observed today at much lower energy due to the expansion of the universe and produce the CMB (for more details see for example Padmanabhan, 2002). By studying its energy distribution, the baryonic matter distribution at high redshift can be recovered. The evolution of this matter distribution can be simulated thanks to theory and numerical methods. Then the model can be verified by comparing the result of simulations and observations of the Local Universe. Without any additional matter or modification to the law of gravity, the growth rate of the structures is far too low to explain today's universe. Thus, Jim Peebles proposed to add dark matter to increase the growth rate (Peebles, 1982). To match our observations of the universe, this matter needs to be cold (meaning a low velocity in comparison to the speed of light) to accurately reproduce the universe.

In other words, an acceptable model would require the existence of an hidden matter with the following properties (Gaitskell, 2004; Porter et al., 2011):

- weak or no interactions with electromagnetic radiations,
- represents a large fraction of the matter in the universe at all times,
- stable in term of decay on time scales comparable with the age of the universe,
- low velocity in comparison to the speed of light.

Theoretical physicists have been fairly imaginative and theorized a large quantity of exotic particles to explain this dark matter. Here, I will quickly present only three interesting dark matter candidates: the Weakly Interacting Massive Particles (WIMP), neutrinos and primordial black holes.

1.1.1 Weakly Interacting Massive Particles

The WIMPs are hypothetical particles with a mass between roughly 10 GeV to a few TeV [1] and have a cross-section comparable to the one of the weak force (Jungman et al., 1996). Thus, in the hot early universe, such particles and their anti-particles would have been formed by light particles and would annihilate when combined. Since then, the universe cooled down and the energy of the lighter particles became insufficient to form new WIMPs while the WIMPs kept annihilating each other. At current redshift, the WIMPs would still be present in large quantity due to their low annihilation probability. Therefore, in high density environment (e.g. centre of galaxies), they would still interact with each other and might produce some observable electromagnetic signals in the process (Gaitskell, 2004).

---

[1] Corresponding to masses of 10^{-23} to 10^{-21} g or 10 to 1'000 protons using the following conversion $\mathrm{eV}/c^2 = 1.78 \cdot 10^{-36}$ kg.

1.1.2 Neutrinos

The neutrinos are particles from the Standard Model of particle physics that are only interacting through gravity and the weak force, therefore they are logical candidates for dark matter. Classical neutrinos are in theory massless and thus could not be candidate, but recent experiments have shown that they do possess a mass (see for example the review from King, 2004). In the Big Bang theory, neutrinos are the second most abundant particles after the CMB's photons (Wong, 2011). Therefore, even a low mass would have a tremendous impact on the total mass of the universe and could explain the presence of dark matter. Unfortunately, the classical neutrinos have relativistic velocities and, in consequence, are considered as hot dark matter. As such, they cannot explain the cold dark matter, but some theories suppose the existence of sterile neutrinos that cannot interact through the weak force and could be far more massive. With their higher mass, they would have lower velocities compatible with the cold dark matter theory (Boyarsky et al., 2009).

1.1.3 Primordial Black Holes

The primordial black holes (PBH) are hypothetical black holes formed just after the Big Bang (less than 1s). The largest overdensities of matter could have collapse directly into black holes without requiring a star (not like recently born black holes) and thus could have masses far below the range of stars (Carr and Kühnel, 2020). Depending on the formation time, they can have masses of 10^{-5} g (at 10^{-43}s) to 10^{5} $M_\odot$ (at 1s). Stephen Hawking predicted that black holes are losing mass due to the formation of photon pairs at the event horizon such that one of them is able to escape (Hawking radiation). While for large black holes this radiation has almost no impact, the radiation can easily evaporate a low mass black hole. In the case of PBHs, it means that the black holes with a mass below 10^{15}g would have evaporated by now, assuming no accretion. Thus the dark matter presents in the recent universe would consist in massive objects. This type of dark matter can be detected through various methods such as gravitational waves, lensing or dynamical effects. Since none of these methods hinted the existence of this phenomenon, the probability of PBHs being the only type of dark matter is becoming lower and lower. While PBHs are currently not really successful to explain the dark matter, they can still play an important role as seeds of supermassive black holes found in galaxies (Carr and Kühnel, 2020).

1.1.4 Dark Matter Detection in Dwarf Galaxies

All the types of dark matter presented before can produce electromagnetic signals, but they will all be different. The Hawking radiation created by black holes is inversely proportional to the black hole's mass. Thus, just before the radiation evaporates a PBH a burst of gamma ray should be seen (Belyanin et al., 1996). In the case of WIMPs, the self-annihilation process is supposed to decompose into particles of the Standard Model including photons (Jungman et al., 1996). Finally, the sterile neutrino cannot self-annihilate as the WIMPs, but can decay into a neutrino and a photon (Boyarsky et al., 2009). These emitted photons have a single frequency $E_\gamma = m_{\mathrm{DM}}/2$, where m_{DM} is the mass of a particle of dark matter, as the neutrinos can be considered at rest for the decay process. From the number of observed satellites around the Milky Way, a lower limit for m_{DM} can be derived. This mass corresponds to photons in the x-ray / γ-ray band (Boyarsky et al., 2009). As all the previous processes are rare and the average luminosity is proportional to the density of dark matter, the center of halos is ideal for any observations. Due to the high contamination from other baryonic physics in more massive galaxies, dwarfs are preferred.

1.2 The Standard Cosmological Model

Here, I will present the concepts in cosmology that will be useful for the rest of the thesis. I will start this section with a presentation of General Relativity (GR). This theory opened the door to modern cosmology thanks to Friedmann equations. Once the equations are presented, I will finish with an overview of the standard model.

In 1915, Albert Einstein made his largest contribution to theoretical physics by publishing his theory of general relativity which links gravity to the geometry of space-time (Einstein, 1915b). One of the first achievement of GR was the explanation of the orbital procession of Mercury (first observed in 1859) which Newtonian gravity cannot fully explain (Einstein, 1915a). Since then, the theory's predictions have been tested during different occasions and are currently used, for example, in GPS.

One of the most interesting prediction of GR is the existence of gravitational waves (GW). They arise from the acceleration of masses (e.g. merger of black holes and neutron stars) which produces a space-time distortion. This distortion propagates at the speed of light in the form of GW. Even if the event that generates them releases a large amount of energy, the distortions weakly deform the space-time at large distances and are difficult to measure. The first detection was made with an indirect method on the Hulse-Taylor binary system, composed of a neutron star and a pulsar (Hulse and Taylor, 1975). Thanks to the precise timing of the pulses from the pulsar, Weisberg et al., 1981 were able to accurately measure the changes in orbital period of the binary due to the energy transformed into gravitational waves and confirmed their existence. Finally, in 2016, the LIGO and VIRGO's team opened officially a new field in the study of the universe with the first direct detection of a gravitational wave (LIGO Scientific Collaboration and VIRGO Collaboration, 2016).

From Einstein's equations, a model for the evolution of the universe can be derived. The full derivation was done thanks to the contribution of 4 different scientists. Initially, Alexander Friedmann managed it between 1922 and 1924, but unfortunately died in 1925 after publishing his results in German (Friedmann, 1924) which remained unnoticed. A few years later, Georges Lemaître independently obtained similar results and published them in a Belgian journal (Lemaître, 1927). Both results include the usage of the now called Friedmann-Lemaître-Robertson-Walker (FLRW) metric. This metric has been shown by Robertson and Walker to be the only one for space-time that respects the cosmological principle (meaning a universe that is spatially homogeneous and isotropic; Robertson, 1935; Robertson, 1936a; Robertson, 1936b; Walker, 1937). From their results, it is possible to compute the evolution of the size of the universe a (called scale factor), with the Friedmann equations:

$$H(a) = H_0 E(a), \tag{1.2}$$

$$E(a)^2 = \Omega_m a^{-3} + \Omega_r a^{-4} + \Omega_k a^{-2} + \Omega_\Lambda \exp(3\tilde{\omega}(a)), \tag{1.3}$$

$$\tilde{\omega}(a) = (a-1)\omega_a - (1 + \omega_0 + \omega_a)\log(a) \tag{1.4}$$

where $H(a) = \dot{a}/a$ and $H_0 = H(z = 0)$ are the Hubble-Lemaître parameter and constant, Ω_m, Ω_r, Ω_k, Ω_Λ are the matter, radiation, curvature, and dark energy density parameters, $\tilde{\omega}$ is the perturbation due to the dark energy equation of state ($\omega(a) = \omega_0 + \omega_a(1-a)$ where ω_0 and ω_a are free parameters). This dark energy is largely unknown and is necessary to reproduce recent observations of the universe expansion (Riess et al., 1998; Bonvin et al., 2017). The scale factor a is the relative size of the universe compared to today's value and directly relates to the redshift by $z = \frac{1}{a} - 1$. As the universe is only expanding, this scale factor can also be used as a time variable.

The standard cosmological model ΛCDM consists in the best match between observations and the previous equations. It describes a universe which has today an energy composed of dark energy (~70%), cold dark matter (~25%) and finally baryonic matter (~5%). In Figure 1.2, the evolution of the scale factor is shown along with its derivative with the parameters given in Collaboration et al., 2020. As it can be seen in the figure, the universe started in a singularity called the Big Bang. It quickly expanded from it and then started to slow down its expansion. Recently, the expansion is starting to accelerate again due to the dark energy.

The Friedmann equation can be approximated at $z = 0$ ($a = 1$ or today) in order to give a simple law linking the distance of the galaxies and their velocities. This law is known as the Hubble-Lemaître law. Starting with $H(a) = \dot{a}/a = H_0 E(a)$, the factor $E(a)$ can be removed as the sum of the cosmological parameters is 1 and $a \approx 1$. Then we can approximate the scale factor by $a/a_0 \approx r/r_0$. Putting everything together, we obtain the Hubble-Lemaître law:

$$\dot{r} = H_0 r \tag{1.5}$$

This equation is thus a local approximation of the cosmological model and represents the effect of cosmology at small scales. As H_0 is positive, it means that the expansion of the universe counteracts partially the gravity and the difference of velocities is far larger at larger distances. In Figure 1.3, the law is shown with the results from Hubble, 1929.

Finally, it is worth mentioning the so-called "little h". In the early days of cosmology, large uncertainties existed on the value of the Hubble constant ($50 - 100$ km s^{-1} Mpc^{-1}). Due to its importance in almost all the computations done by observers, they introduced a parameter h in

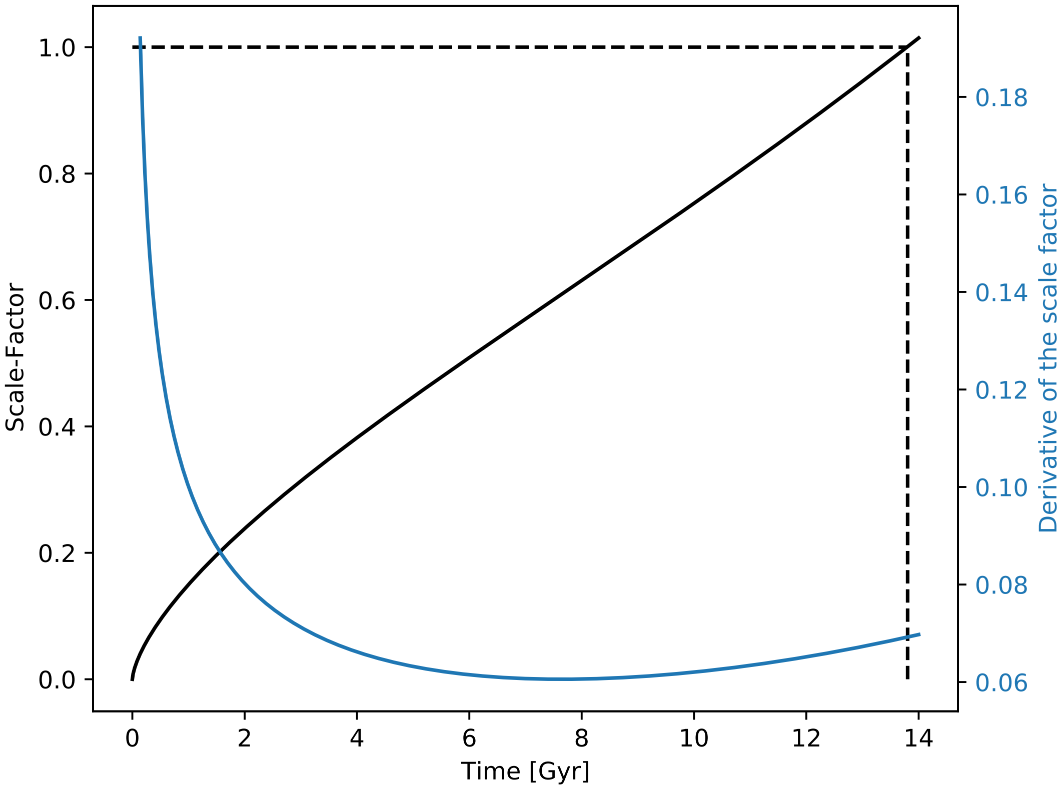


Figure 1.2 – Evolution of the scale factor and its derivative as a function of time. The dashed lines show today's universe. The constants are taken from the last Planck's results: $\Omega_m = 0.3111$, $\Omega_r = 0$, $\Omega_\Lambda = 0.6889$, $\omega_0 = -1$, $\omega_a = 0$ and $H_0 = 67.66$ km / (s Mpc) (Collaboration et al., 2020). The universe started with a singularity that quickly expanded and then slowly decreased its expansion rate. Around 7 Gyr, the universe started to accelerate its expansion due to the dark energy.

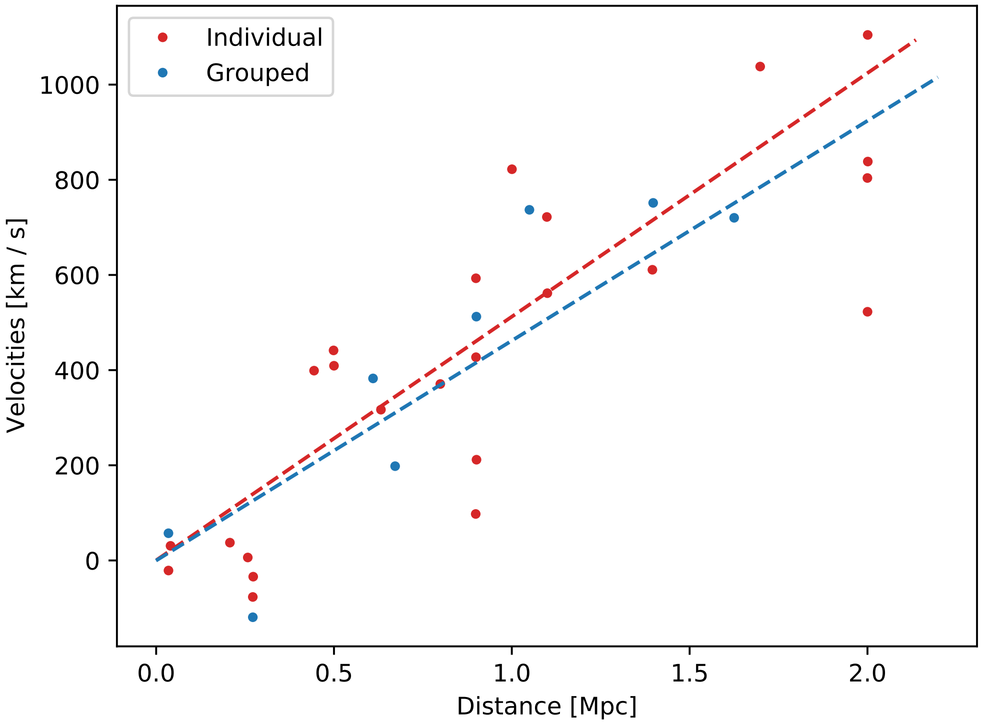


Figure 1.3 – The Hubble-Lemaître law linking the distance and velocity (or redshift) of the galaxies. This Figure is a reproduction of the one in Hubble, 1929. The red dots represents individual “nebulae” (today they would be called galaxies) and blue dots represents “nebulae” that are grouped together when belonging to the same cluster. The lines are the best fits of the Hubble-Lemaître law based on the corresponding dots.

the expression of the constant $100h$ km s^{-1} Mpc^{-1}. This parameter is then introduced within the units in order to try to make a result independent from the initial supposition of h (e.g. distances given in Mpc / h). For a long time, it allowed the results to be easily compared between different assumptions on the Hubble constant, but nowadays, with the low uncertainties in measurements, h is not as important as before.

1.3 Dwarf Galaxies

Before entering the topic of this thesis, it is important to define what is a dwarf galaxy in order to avoid any misunderstanding. Galaxies are gravitationally bound objects spanning a very large range of stellar masses, going from containing almost no stars ($M_\star \approx 10^2$ M$_\odot$) to $M_\star \approx 10^{12}$ M$_\odot$. As the name indicates it, dwarf galaxies set the lower bound in masses. We will consider 3 main arbitrary categories, but other denominations exist. The dwarfs can be classified as bright dwarfs ($M_\star \approx 10^7 - 10^9 M_\odot$), classical dwarfs ($M_\star \approx 10^5 - 10^7$) and ultra-faint dwarfs (UFDs; $M_\star \approx 10^2 - 10^5$). Actually, dwarfs' properties are not showing any discontinuity in the dwarf regime and therefore the different types of dwarfs are not distinct classes of objects, just an artificial denomination. Some people are also using the terms dwarf spheroidals and dwarf irregulars in order to differentiate dwarfs lacking gas and without star formation from dwarfs with both of them (Bullock and Boylan-Kolchin, 2017).

UFDs look like globular clusters in terms of stellar mass, but they still present striking difference. The most interesting difference is in the dark matter content. While dwarf galaxies are the most dark matter dominated objects, globular clusters have a negligible fraction of dark matter (Strigari et al., 2008; Simon and Geha, 2007; Kleyna et al., 2005). UFDs tend to be also far more spatially extended (McConnachie, 2012), have abundances of Fe and α elements[2] showing a longer star formation epoch (Martin et al., 2007; Simon and Geha, 2007) and a relation between their luminosity and their metallicity (McConnachie, 2012) [3].

In the next sections, I will summarize the current observational and theoretical knowledge of dwarf galaxies before moving to the numerical challenges. I will start with a quick reminder of the abundances and metallicities. Then, I will describe what observers are capable of measuring today. Next, I will present the evolutionary process of dwarf galaxies in order to describe some important topics currently investigated. The different topics presented here are: why dwarfs are relics from the early universe, how they can help to understand the r-process and lastly how they can probe the cosmological model. I will conclude this small review of the observational and theoretical knowledge with the next generation of telescopes and how they can impact our knowledge.

---

[2] The α elements are processed through the nuclear reactions contained within the α ladder. The stable α elements are C, O, Ne, Mg, Si and S. The list can be extended to Ar and Ca as they have been shown observationally to be sufficiently stable.

[3] An updated database of McConnachie, 2012 is available here http://www.astro.uvic.ca/~alan/Nearby_Dwarf_Database.html

1.3.1 Abundances and Metallicities

Let us start by quickly defining the abundances and metallicities of stars that are terms often used in astrophysics. By observational tradition, a lot of quantities are normalized by the Sun's values. The element abundances are a good example and represent the ratio between two elements. For example, the abundance of magnesium with respect to iron in a star is defined in the following way:

$$[\mathrm{Mg/Fe}] = \log_{10}\left(\frac{M_{\mathrm{Mg}}}{M_{\mathrm{Fe}}}\right)_{\mathrm{star}} - \log_{10}\left(\frac{M_{\mathrm{Mg}}}{M_{\mathrm{Fe}}}\right)_{\mathrm{Sun}} \tag{1.6}$$

where M are the masses. In this way, a star with a larger (lower) ratio of magnesium than the Sun will have a positive (negative) abundance. A more meaningful quantity for simulations is the metallicity that represents the total fraction of metals:

$$Z = \sum_{i > \mathrm{He}} \frac{m_i}{M} \tag{1.7}$$

where m_i is the mass of the element i and M is the total mass of the star.

1.3.2 Current Status of Observations

The study of dwarf galaxies was for a long time focused only on the few bright ones around the Milky Way (MW) such has the Large and Small Magellanic Clouds (LMC and SMC). The first observation of the Magellanic Clouds certainly dates from prehistory as they are visible by the naked eye. With the beginning of digital surveys, the number of known dwarfs exploded from a dozen to about 60 MW's satellites, 35 Andromeda's satellites and 50 within the Local Group and the surrounding environment (online table from McConnachie, 2012 updated in 2019). Figure 1.4 provides the evolution of the number of dwarfs found. While the total number of dwarfs exploded, the lower limit on the luminosity of the observed dwarfs decreased and thus the first UFDs were discovered. This digitalization did not only produce a larger set of dwarf galaxies, but also provided us with improved measurement within individual galaxies in terms of quality and quantity.

Even with a larger dataset, the observation of dwarf galaxies (especially UFDs) remain complex due to the low number of stars they contain. Usually, a statistical approach is used to decrease the impact from any foreground contamination and binary stars on the observed properties of galaxies. In both cases it requires far more observed stars than available and thus large uncertainties are still present. Recent measurement have shown that such bias were present in previous studies (e.g. Simon, 2018; Venn et al., 2017; Ji et al., 2016; Moe et al., 2019). In the case of UFDs, for approximately a third of the galaxies no spectroscopic observations are available yet and thus we lack of knowledge on the velocity dispersion and metallicities (Simon, 2019). Anyway, the quality of the observations is still sufficient to produce some constraints.

For example, Color-Magnitude Diagrams (CMD) for stars belonging to a single dwarf are an

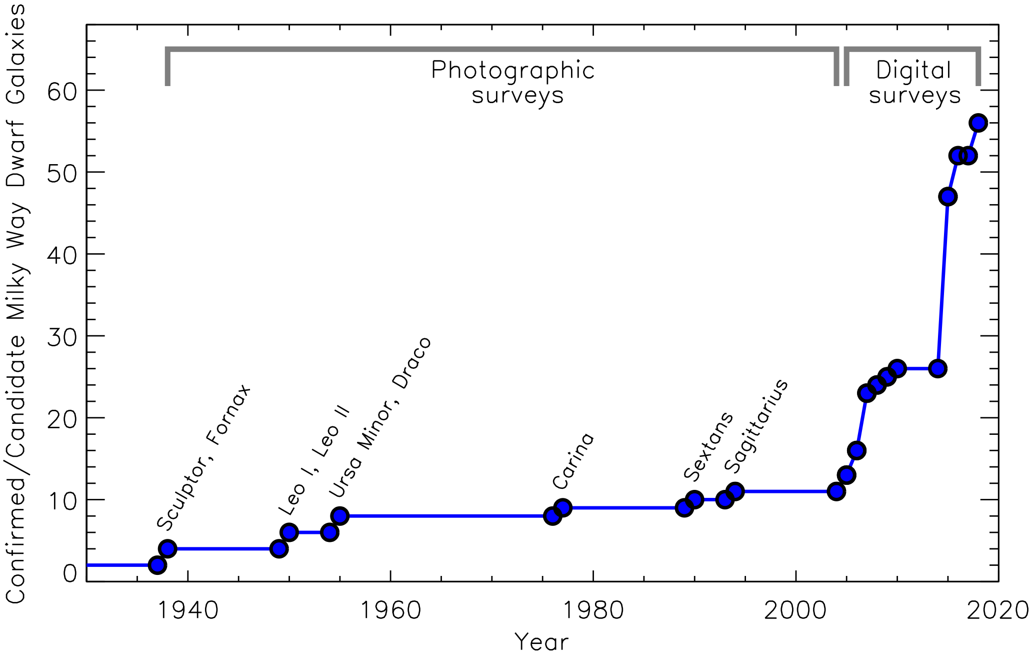


Figure 1.4 – Evolution of the number of dwarfs observed. Thanks to the beginning of digital surveys, the number of dwarf galaxies observed has exploded since 2000. Credit: Simon, 2019

important tool to understand the evolution of the galaxy. In particular, Star Formation Histories (SFH) can be derived from CMDs with the help of accurate stellar models. Notice that without any metallicity measurement, a degeneracy exists between two stars having the same position in the CMD. Indeed, CMDs cannot differentiate between ages, metallicities and masses without additional information (Tolstoy et al., 2009). Recent spectroscopic surveys of individual stars in dwarf galaxies can alleviate this degeneracy but require larger observational time than photometric surveys. Another issue with the CMD approach is that the complete star formation requires some assumptions on the most massive stars as their lifetime is too small to be seen today. By combining the number of observed stars born at a given time and a stellar initial mass function, it is possible to estimate the total number of stars born at a given time but includes large uncertainties. It means that the further in the past we look, the larger they become, since a larger fraction of stars will have already died (see for example Boer et al., 2012b for more details on this technique).

The abundances of stars also provide information on the SFH and here we will be specially interested into [Mg/Fe] as function of [Fe/H] (see McWilliam, 1997 for a deep review). The abundance [Fe/H] can be seen as a time indicator. Indeed, the iron is almost only increasing with time and the hydrogen is almost constant. It means that a high amount of iron indicates a late birth of a star. Both the iron and magnesium are mostly produced by supernovae. Two distinct types of supernovae are important for the previous elements. Supernovae of type Ia (SNIa) are the largest contributors of iron while supernovae of type II (SNII) of magnesium (as fraction of iron). The two types have large differences in time scales between the formation of stars and their supernovae. SNII have a relatively low time scale while SNIa have long time scales and, thus, they impact the abundances at different times. Early on, only the SNII contribute to the enrichment and create high fraction of magnesium. As time passes, SNIa start to explode and slowly bring down the ratio. The point where the magnesium ratio starts to decrease is called the knee.

The two most interesting things to observe in such graphs are when the knee appears and what is the average magnesium fraction at low metallicity. Indeed, as the delay between the first SNII and the first SNIa can be considered as independent from the dwarf's history, it means that in every galaxy the delay will be the same. If the knee starts at low metallicity, it means that a relatively low number of SNII exploded before the SNIa. Therefore, the abundances [Fe/H] of the knee indicates the efficiency of the star formation rate at high redshift. For the average value of the magnesium, it depends on the initial mass function that describes the distribution of stars according to their mass (see section 4.6.1 for more details). As the magnesium fraction released by SNII increases as function of the mass of the progenitor, it means that a high fraction of magnesium indicates a large fraction of massive stars.

As an examples, Figure 1.5 provides the abundances of [Mg/Fe] as function of [Fe/H] for the dwarf galaxies Carina (Shetrone et al., 2003), Draco (Shetrone et al., 2001; Cohen and Huang, 2009; Kirby et al., 2010; Tsujimoto et al., 2015), Fornax (Shetrone et al., 2003), Sculptor (Shetrone et al., 2003; Tafelmeyer et al., 2010; Tolstoy et al., 2009) and Sextans (Aoki et al., 2009; Shetrone et al., 2001; Tafelmeyer et al., 2010; Theler et al., 2020). For comparison, the background is obtained from the APOGEE survey that targets Milky Way's red giants (Jönsson et al., 2020). At low [Fe/H] (meaning early time), all the galaxies have an abundance of magnesium over iron of approximately 0.5 which indicates a universality of the initial mass function. All the galaxies have their knee at different amount of iron, thus they must have slighly different, but still comparable, relative star formation rate at early time. When comparing to the Milky Way (and assuming no bias in APOGEE), the knee starts at far larger [Fe/H] for the Milky Way and, as one can expect, the star formation should be strongly different.

The satellite GAIA was launched in 2013 with the aim of observing stars with a full sky coverage within a few years and then to retrace its steps in order to measure any displacement (Gaia Collaboration, 2016). By combining the apparent displacement with the earth's displacement, an accurate estimation of the distance to each star is computed along with the tangential velocity[4]. When using this information on stars belonging to a dwarf galaxy, the orbital properties of the dwarf can be recovered and provide insights on the impact of the environment on its evolution (Simon, 2018; Massari et al., 2018; Fritz et al., 2018). Almost all the closest observed UFDs are close to their pericenters (within 5 kpc). This seems to indicate that these dwarfs are close to the detection limits of the telescopes (Simon, 2018). Such bias results in a lack of diversity in the observed low mass dwarfs and reduces our ability to understand the impact of the environment.

---

[4] Tangential means here the velocity across the sky and not towards earth.

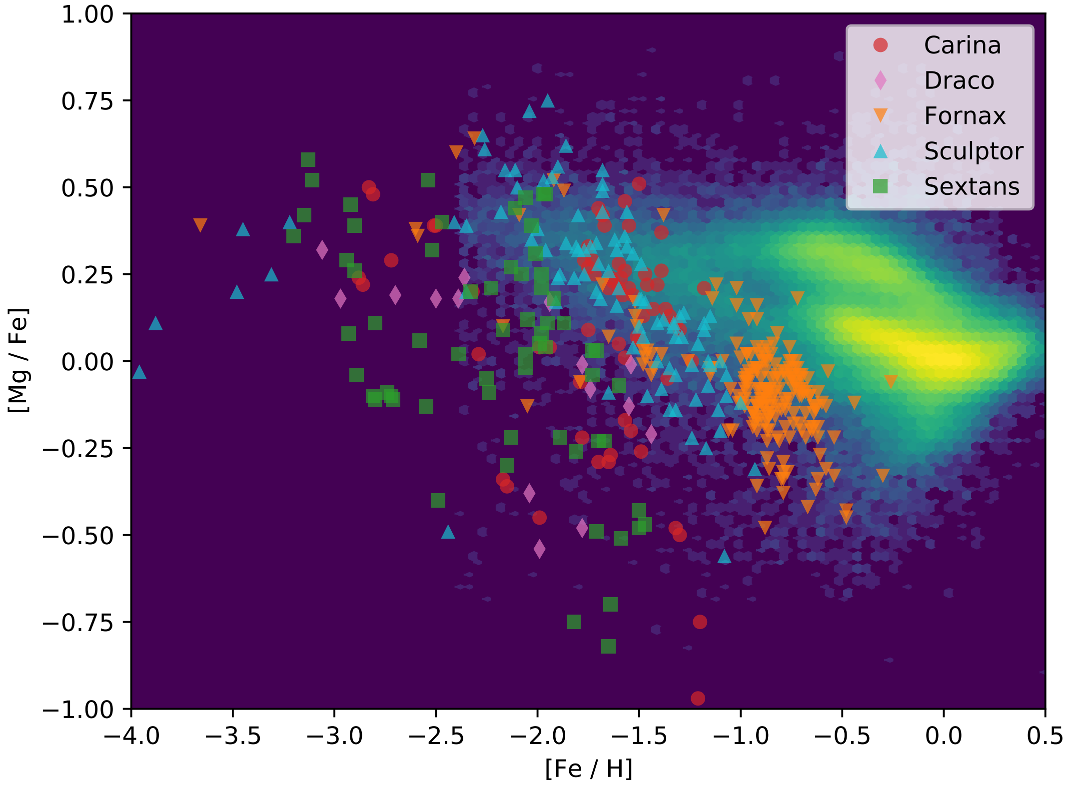


Figure 1.5 – Abundances of [Mg/Fe] as function of [Fe/H] in five dwarf galaxies. The background is given by Jönsson et al., 2020 that targets red giant stars in the Milky Way. Those stars have ages between around 10 Myr and 10 Gyr. All the dwarfs (and the Milky Way) show a trend moving from [Mg/Fe] ≈ 0.5 at low [Fe/H] towards a negative value at high [Fe/H].

Most of the observed dwarfs, and specially UFDs, are Milky Way's satellites, but some efforts are done to reach further away. The logical first step is to look around our closest neighbor: Andromeda. For example, the Pan-Andromeda Archaeological Survey (PAndAS) was able to find 16 new dwarf galaxies within a projected radius of 150 kpc of the host (Martin et al., 2016). In this survey, they were barely able to reach light fluxes corresponding to UFDs but still managed to observe 8 of them. The detection of UFDs further away from the MW will be impossible for a long time. However some bright dwarfs have been observed around different clusters (see for example Geha et al., 2017).

1.3.3 Evolution Through Cosmic Time

In the next paragraphs, I will quickly introduce the global theoretical ideas explaining the evolution of dwarf galaxies. Within the ΛCDM model, the universe was initially homogeneous. Some slight overdensities already existed and are the progenitors of all the dwarfs that we can observe today (Li et al., 2010). In this early universe and until the creation of the first stars (around redshift 20; Frebel and Bromm, 2012), the evolution of the proto-dwarfs is due to gravity, the expansion of universe and hydrodynamics. First some dark matter halos appear due to the lack of counteraction from hydrodynamics on the dark matter and the insufficient impact from the expansion of the universe. In such early universe, a large quantity of mergers happens and quickly grow the mass of halos until $z \approx 2$. With a relatively deep gravitational potential, the gas pressure is not sufficient to compensate gravity. Thus, the first galaxies appear and create the first stars that radically changed the situation.

During their lives, the stars reach extreme temperatures and thus both radiate photons and produce stellar winds that can heat the nearby gas (Schaye et al., 2015). Once they reach the end of their life, the most massive stars produce supernovae that almost instantaneously (in comparison to the galactic timescale) heat the surrounding gas (Stinson et al., 2006). Following this heating of the gas, the hydrodynamic forces become more important than gravity. Indeed, the increase of pressure and the relatively long cooling times prevent the gas from collapsing. In the most extreme cases, gravity is not strong enough to keep the gas inside dwarf galaxies (Bland-Hawthorn et al., 2015; Webster et al., 2014), and it marks the beginning of a non-star forming era in the history of the dwarf. In case of larger galaxies (and maybe high redshift dwarf galaxies (Koudmani et al., 2021)), Active Galactic Nucleus (AGN) are also playing the role of an energy source and can evaporate all the gas inside the largest galaxies (Thomas et al., 2021).

The impact of stars is not limited in their role of an energy source, they are also the primary sites of nuclear reactions and thus are the drivers of the chemical evolution. Through their evolution, the stars will process their gas and create heavy elements from light ones. A large fraction of the metals processed are then released into the surrounding by supernovae (Tsujimoto et al., 1995; Kobayashi et al., 2000). This enriched gas behaves differently than the pristine gas. Indeed, due to the large number of electron layers in metals, the heavy elements strongly regulate the temperature of the gas. By interacting with photons, metallic atoms (but also lighter elements in a lesser extend) can strongly heat or cool down the gas, even if they represent a few percent of the total composition of a galaxy (Smith et al., 2017; Arnett and Cameron, 1967). This process is usually called radiative cooling even if the gas might heat.

Starting from redshift $\sim$ 12 and peaking at redshift 2-3, the UV produced in galaxies is energetic enough to leave some galaxies and reionize the matter in the whole universe impacting the evolution of every galaxies (called UV background; Haardt and Madau, 2012). Thus, low mass dwarfs will usually quickly stop to evolve as their Star Formation Rate (SFR) decrease (Bullock et al., 2000). Indeed, the UVB heats the low density gas and evaporates it due to the optically thin regime that prevents any shielding against the UV. Without cold gas, the galaxy is forced to stop forming stars. For larger dwarf galaxies, the UVB has a far lower impact but usually still modify a bit the SFR (Revaz and Jablonka, 2018; Aubert and Teyssier, 2010). The moment where the universe is again fully ionized around redshift 10 is called the reionization and play an important role in observations as the universe was mainly optically thick until then. Due to their large number, dwarf galaxies are believed to be a dominant contributors to the reionization (Livermore et al., 2017). While the picture provided before is well accepted for the evolution of galaxies, many details are still under investigation such as how the different forms of feedback (e.g. radiation, stellar winds, supernovae and AGN) impact the galaxies, what is the impact of the first stars or how the AGNs are born and evolve.

As dwarf galaxies are the tiniest objects that have formed stars in isolation, they are ideal laboratories to understand the impact of different forms of heating. While in larger galaxies only the strongest heating can influence the evolution of the galaxy, the dwarfs, due to their low gravitational potential, will be strongly impacted by lower form of heating. For example, in the case of dwarfs heavier than $M_{\text{vir}} \approx 10^9 M_\odot$, the reionization is not strong enough to quench them (Bovill and Ricotti, 2009; Benson et al., 2002), and, in halos smaller than $M_{\text{vir}} \approx 10^7 M_\odot$, a single supernova is sufficient to temporarily remove all the gas (Bland-Hawthorn et al., 2015; Webster et al., 2014).

Impact of the Environment

Depending on their environment, this picture is enough to understand the basics of any galaxy. However, in the case of satellites, the evolution of the host, along with the orbit of the dwarf, can play an important role (Emerick et al., 2016; Simpson et al., 2018; Williamson and Martel, 2018). For the lowest pericenters, the dwarf galaxies will be totally destroyed by the tidal forces. The Sagittarius Dwarf Spheroidal is a perfect example of a tidally destroyed dwarf galaxy (Vasiliev and Belokurov, 2020). On more eccentric orbits, the tidal forces become less important and the interaction between the host's hot halo and dwarf's gas, called ram pressure stripping, becomes one of the main cause of the evolution. In this case, the dwarf is moving at a velocity of the order of 100 km / s with respect to the hot halo gas and, thus, frictions and collisions are produced with the dwarf's gas. The dwarf's gas can be (partially) lost due to this ram pressure (Emerick et al., 2016; Mayer et al., 2006; Hausammann et al., 2019). Thus, the dwarf galaxy can either be directly quenched by the removal of all the gas or will starve until all the remaining cold gas is converted into stars (Fillingham et al., 2016; Simpson et al., 2018; Bahé et al., 2012). Some other processes could be playing a role such as galactic winds or radiations from the host (Hausammann et al., 2019; Dashyan et al., 2019).

To understand more the impact of the environment, one can look at the correlation between [Fe/H] and the luminosity (see Figure 1.6). As the iron fraction is constantly increasing with time, it traces the history of the star formation. The luminosity is more a tracer of the current content of stars as high mass stars (with the lowest lifetime) dominate it. A strong stripping of the stars would impact the luminosity but not really the metallicity. Thus, the tight correlation argues in favor of an inefficient tidal stripping of the stellar component. This stripping can still remove large quantities of dark matter without perturbing the stars, as the dark matter is far more extended than the stellar component (Kirby et al., 2013; Simon, 2019).

Even if the previous correlation and some others such as the relation between the baryonic mass and the circular velocity (McGaugh et al., 2016), are tight, a large diversity of dwarfs exists as it can be seen with the diversity of shapes for the rotation curves shown in Creasey et al., 2017. Such variety cannot be explained from dark matter simulations within a ΛCDM universe and requires further investigations.

In UFDs, no neutral gas has been observed yet except for Leo T (Irwin et al., 2007; Ryan-Weber et al., 2008). The absence of gas is not unexpected, as a low number of supernovae is sufficient to remove it, but as the only observed UFDs are within the influence of a massive galaxy (usually the Milky Way but also Andromeda for a few), the origin of this stripping is not clear yet and among the possibilities are supernovae feedback, reionization and ram pressure stripping. With the possible future observations of the first truly isolated dwarfs, the impact of the different processes will become clearer (Simon, 2019).

Finally, a large fraction of dwarfs end their lives by merging with larger galaxies. In consequence, dwarf galaxies in the early universe are considered as the building blocks of larger galaxies (Jiang et al., 2021; Valcke et al., 2008). Thus understanding dwarf galaxies is essential to understand the growth of Milky Way like galaxies.

1.3.4 Relics From the Early Universe

Simulations of UFDs show that the star formation rate dropped to zero soon after the reionization (Weisz et al., 2014; Brown et al., 2012) and, thus, UFDs can be considered as relics from the early universe. In the case of classical dwarfs, even if the star formation is not totally stopped, in many dwarfs, most of the stars were produced in the early universe (Boer et al., 2012b; Boer et al., 2014; Boer et al., 2012a; Revaz and Jablonka, 2018). With almost no new stars in recent epoch, such systems have been preserved from any internal chemical evolution and, in consequence, are pristine relics of the early universe and exhibit low metallicities. Some stars even show extremely low metallicities compatible with being enriched in elements only by the first stars (called Pop III)

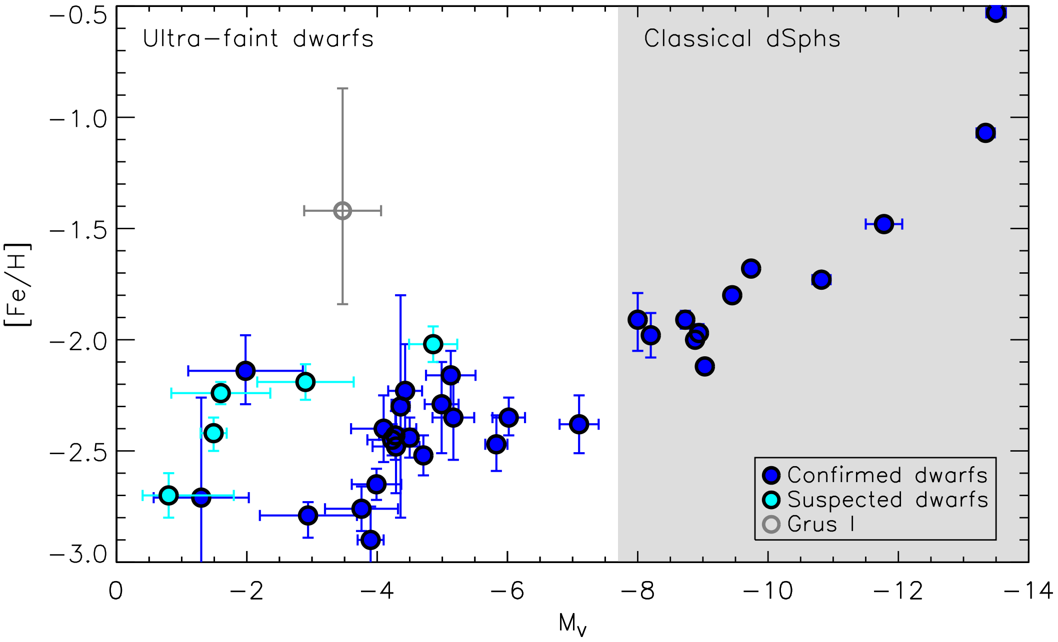


Figure 1.6 – The abundance [Fe/H] is given as function of the V-band magnitude of dwarf galaxies. This magnitude is simply a logarithm of the luminosity with a filter in the V-band. Even at low magnitude, the relation stays tight and is thus an indication of an universal evolution for a given magnitude. Also, this shows that the evolution of the stellar component is weakly influenced by the environment. Credit: Simon, 2019

and thus might provide insights on their chemical signature (Bonifacio et al., 2018; Yoon et al., 2019). Due to the absence of metals in the primordial gas and the corresponding low efficiency of the radiative cooling, such stars are expected to be far larger than their children (called Pop II and Pop I; Haemmerlé et al., 2020; Bromm et al., 2002). The main driver of stellar evolution is the mass, therefore Pop III stars with larger mass might have different supernovae yields than Pop II and I.

As more massive stars could have been produced during the epoch of the first stars, it would be interesting to obtain an idea of the initial mass function (IMF) that describes their mass distribution. As the most massive stars quickly died, only low mass stars are available, but still not observed, for such early time (less than $1M_\odot$ assuming comparable lifetime than Pop II and I (Raiteri et al., 1996)). Thus deriving the IMF on the high mass end requires heavy assumptions. While the IMF is assumed to be universal, at least for the Pop II and Pop I stars, there are still some unknowns about the universality at low, but non-null metallicity, and high redshift (Barbosa et al., 2021; Kroupa, 2001). Thanks to their relatively large population of old stars, dwarf galaxies are the ideal galaxies where to probe the IMF.

R-Process

Stars produced most of the elements seen today and have done it through mainly two sets of nuclear reactions. The first one called α process (along with the help of the tri-α process) is responsible for the creation of the elements from helium to iron. While all the stars start the α process, only the heaviest ones will be able to process all the elements in it. The second set of nuclear reactions is composed of the s (slow neutron capture) and r (rapid) processes and is responsible for the creation of elements heavier than iron (Maeder, 2009). While the s-process is well understood and produced mainly by stars in the Asymptotic Giant Branch (AGB stars), the origin of the elements in the r-process is still under investigations. Currently, the possible origins considered are core-collapse supernovae (Nishimura et al., 2006; Woosley et al., 1994) and neutron star mergers (Freiburghaus et al., 1999). As dwarf galaxies contain a large fraction of old and metal poor stars, they are often targeted in searches for stars enriched by a low number of supernovae and including some elements from the r-process (Starkenburg et al., 2017). With the observation of the first neutron star mergers through gravitational waves (Abbott et al., 2017) and the related additional observational constraints, a possible answer to the origin of the r-process could arrive soon.

1.3.5 Cosmological Probes

The ΛCDM has proved its exactness at large scale (meaning sizes larger than galaxies), but with recent progress at small scales (meaning dwarf galaxies) some differences between observations and theoretical predictions have been seeding some doubts. In this section, I will briefly discuss the problems called: plane of satellites, missing satellites, cusp vs core and the too-big-to-fail.

Plane of Satellites

Recent observations seems to indicate a preferential plane for the dwarfs around the Milky Way (MW), Andromeda and the Centaurus group (Pawlowski, 2018; Li et al., 2021; Müller et al., 2016; Lynden-Bell, 1976) while ΛCDM was supposed to produce isotropic spherical distributions. Until recently, the low number of dwarfs observed was not sufficient to fully acknowledge the existence of this disk. With the large number of newly discovered dwarfs, not only the disk of the Milky Way has been confirmed, but also the Andromeda and Centaurus group ones. The Andromeda plane is shown in Figure 1.7.

Even if isotropic distributions were expected from ΛCDM, recent simulations have shown that non-rotationally supported disk distributions are possible (Bahl and Baumgardt, 2014; Libeskind et al., 2015; Gillet et al., 2015). In the case of the MW, It might be possible that a large quantity of dwarfs were satellites to the Magellanic clouds before accretion and thus are still on comparable orbits. Therefore, the remaining question is how likely disks of satellites are and are they rotationally supported. For the first question, surveys have started to perform similar studies of dwarfs outside the Local Group, and for the second one, proper 3D motions of Andromeda's dwarfs should give a proper answer (Pawlowski, 2018).

Missing Satellites Problem

The second issue but oldest one is the missing satellites problem and consist in the lack of consistency between the number of dwarfs around the MW in ΛCDM simulations and observations. While the simulations are predicting thousands of dark matter halos with masses compatible with dwarfs, the observations are counting only ~ 60 dwarfs (Simon and Geha, 2007). Recent simulations have shown that the missing satellites problem can be solved with an accurate treatment of the baryonic physics. At such low masses, the presence of a baryonic disk enables a more efficient tidal stripping and decreases the amount of observable galaxies. Thus, forcing to match directly the dark matter halos masses with the observed dwarfs produces an inconsistency (Garrison-Kimmel et al., 2017; Brooks et al., 2013; Sawala et al., 2016; Revaz and Jablonka, 2018).

Cusp vs Core

Studies of the inner profile of dwarf galaxies have shown that the NFW profile (see Navarro et al., 1996 for details on the profile), expected from Dark Matter Only (DMO) simulations, was not correctly describing the observations (cusp profile; Navarro et al., 2010). Indeed, the profile in many dwarfs is showing the presence of a constant density core as seen in the velocity profiles of two dwarf irregulars in Figure 1.8 (McGaugh et al., 2016; Marchesini et al., 2002; Simon et al., 2005; Blok et al., 2008). Many simulations have shown that stellar feedback can modify the central distribution (Read et al., 2016; Oñorbe et al., 2015; Mashchenko et al., 2008; Governato et al., 2012). In case of average star formation, the feedback is sufficiently powerful to

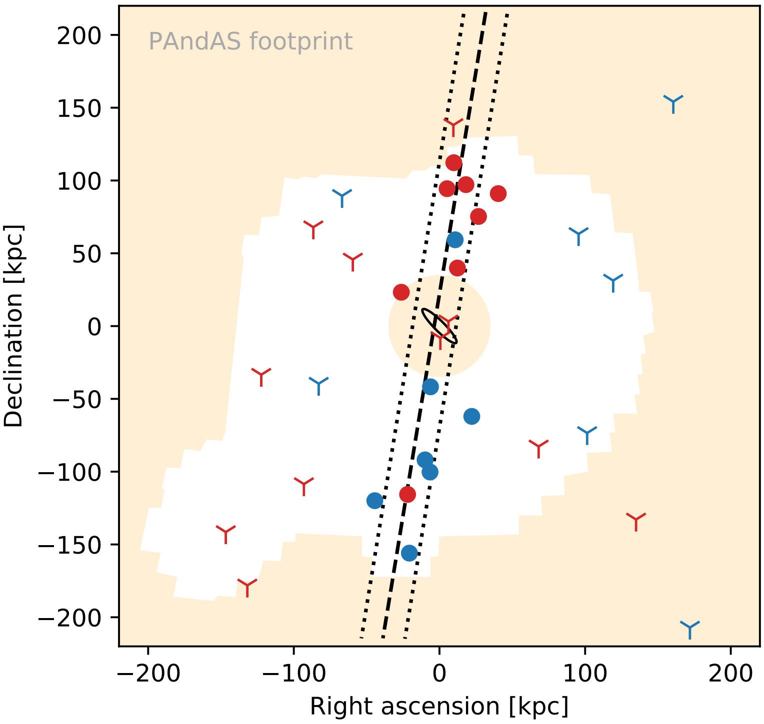


Figure 1.7 – Dwarf galaxies distribution around the Andromeda galaxy (ellipse at the center). In red (blue), the galaxies that are moving away from (towards) us are shown. A plane of satellites can be seen from the distribution and is indicated with the three black lines. The dwarfs belonging to the planes are indicated with dots. The data have been extracted from Pawlowski, 2018.

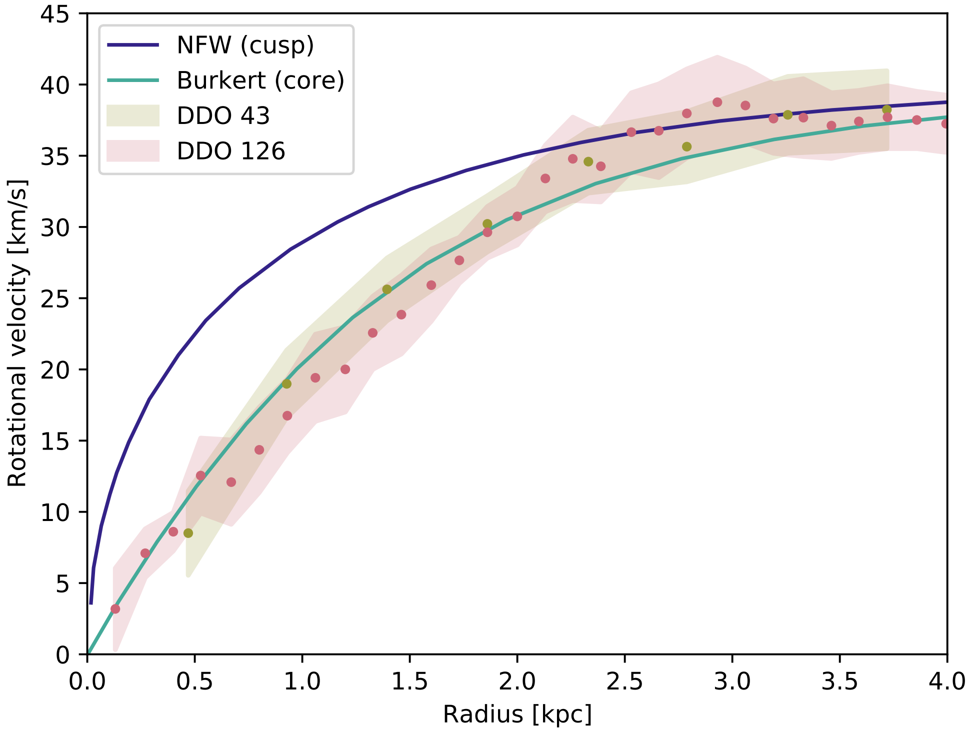


Figure 1.8 – Rotational velocity as function of the radius of two galaxies with the same maximal velocity (DDO 43 and 126) along with their fit (Burkert in cyan). The observed values (given in red and green) can be compared to theoretical predictions from dark matter only simulations (NFW). The discrepancy observed here is the cusp/core problem, where observations are showing a constant density core while the simulations a cusp. The data have been extracted from Bullock and Boylan-Kolchin, 2017

redistribute the baryonic mass and, indirectly the non-baryonic mass, within the inner part of the galaxy. If too much gas is accumulated at the center (and converted into stars), this additional mass can drag the dark matter back and recreate a cusp (Di Cintio et al., 2014). Unfortunately, this process depends on many non directly physically motivated parameters such as the stellar feedback. Due to the low amount of constrains on such parameters, a large range of value can be used and depending on the value, this process might not be observed as it is the case in Sawala et al., 2016. Therefore it cannot be stated with certainty yet that the stellar feedback is solving the problem.

Too-Big-To-Fail Problem

Finally, the too-big-to-fail problem can be seen either as a subset of the missing satellites or related to the cusp/core problem. In DMO simulations, dark matter halos more dense than the brightest satellites are present around Milky Way like galaxies. Such big halos should not fail at producing stars, as their gravitational potential is deep enough to keep the gas, and should be easily observable. So either some dwarfs are really missing or something is happening to those massive halos in order to decrease their mass (such as the presence of a core) and to give masses corresponding to the bright dwarfs (Del Popolo and Le Delliou, 2017; Bullock and Boylan-Kolchin, 2017).

Unified Solutions

Some solutions to the different individual problems seem to give promising results, but no unified solution are widely accepted yet (Del Popolo and Le Delliou, 2017). For example, a solution to the cusp/core problem resulting in the flatting of the inner profile could help to increase the impact of any form of feedback. Thus, it could be responsible at the same time for the disappearance of dwarfs in the missing satellites and also the decrease in mass in the Too-Big-To-Fail problem such as presented in Garrison-Kimmel et al., 2017.

Current surveys are reaching the limit of dark matter halos with only hundreds of stars within and, according to ΛCDM, smaller halos should exist in large quantity. If such halos were detected, it would provide strong constrains on cosmological models, but also some indications to the solution of the previous issues. In stellar streams with low velocity dispersion (lower than disrupted dwarfs) around the Milky Way, a halo crossing them would create some perturbations and could generate some gaps and clumps of stars (Carlberg, 2009; Ibata et al., 2002). Through the distribution of the gaps and clumps, it could be possible to have limits on the number of low mass dark halos.

1.3.6 What to Expect in the Future From Observations

The next generation of telescopes (e.g. 4MOST, WEAVE, JWST, Vera C. Rubin Observatory, SKA, eLISA, ELT, TMT) are currently in construction or starting to release their data and will provide important results to deepen our understanding of astrophysics. A particular interest has been given to telescopes including spectroscopy. Two such examples are 4MOST and WEAVE that will do the follow-up of GAIA's observations, each in its own hemisphere, and will provide crucial spectrum to dwarf galaxies and thus more accurate star formation histories and metallicities will be available.

Finding More Dwarfs

According to Newton et al., 2018, we are still lacking a detection for at least half of the MW's dwarf satellites. Detecting them could improve our models thanks to additional constraints. One of the missions of the Vera C. Rubin Telescope[5] will be to detect them, but also mapping the Sagittarius stream that will help to increase our understanding of tidal stripping along with discovering the presence of dark halos. It will also be an essential tool in the detection of gravitational waves as its large field of view will easily allow to see the electromagnetic counterpart of the waves (LSST Science Collaboration, 2009). JWST and SKA will also contribute to finding more dwarfs, but they will focus on the high redshift universe. On one hand, JWST, with its infrared camera, will focus on observing the redshifted visible light and will mainly provide information about stars. It means that we could detect dwarfs composed mainly of Pop III stars or supernovae of Pop III. It will also be possible to follow the evolution of the star formation rate along with the metallicity as function of the redshift. Once enough statistics obtained, the luminosity function will be computed and provide additional information on the role of the dwarfs in the reionization. Finally, an ultra deep field image is scheduled and could in theory capture the light from a $10^6 M_\odot$ Pop III star at $z \sim 20$ (Gardner et al., 2006). On the other hand, SKA is a radio telescope and will focus on the gas. It will provide the first full sky map of the 21 cm line (HI) and greatly improve our understanding of the gas in dwarf galaxies. Through the Faraday rotation of background radio emission, SKA can measure the magnetic field in galaxies and provide constraints on the role of the magnetic field in the evolution of galaxies (Lobanov, 2012).

Relics From the Early Universe

Searches for low metallicity stars in dwarf galaxies as done by the Pristine survey might reveal the signature of Pop III stars or r-process. In the second case, some insights on the origin could be derived from such signatures. Additionally, Pristine is able to clearly identify the stars belonging to dwarf galaxies or being foreground contamination and specially at large radii. This should largely increase our trust into the observations and provide stronger constraints to the models (Starkenburg et al., 2017).

Probing the Dark Matter

On the dark matter side, we can hope to detect an electromagnetic signal from the dark matter in dwarf galaxies with, for example, the telescopes Fermi Gamma-ray Space Telescope and the Cherenkov Telescope array. If the detection is made, we will be able to reduce the currently large number of dark matter models to only a few. Even without any detection, an upper limit on the emission of photons will be available and could slightly constrain the models. The dark matter profiles' quality will be impacted by the next generation telescopes such as the ELT, TMT, Vera C. Rubin Observatory and even eLISA. They should improve our understanding of the cusp/core

---

[5] Previously known as the Large Synoptic Survey Telescope

problem. Both the ELT and TMT will provide 3D proper motions of stars in dwarf galaxies which will provide a better description of the mass profile. Thanks to their large mirrors, they will also be able to probe stars of lower luminosity which have longer lifetime and thus could be remnant of the Pop III stars. The ELT also aims at understanding the evolution of dwarf galaxies at intermediate redshift (from $z \sim 1$ to ~ 3) through their chemo-dynamical properties. This will provide information about the origin and the mass assembly of dwarf galaxies (Evans et al., 2015). In the case of the TMT, a search for intermediate-mass BHs (IMBHs) in dwarf galaxies is scheduled in order to provide a larger sample than currently available. Currently, they are still considered as having a low (or even negligible) impact on the global evolution of dwarfs, but this could be revised depending on the results. As they are supposed to be the progenitor of supermassive BHs, they can also provide information on their mass assembly. For dwarf galaxies, it will be possible to observe them at larger redshift and thus provide scaling relations such as the luminosity-metallicity relation but also obtain the stellar chemistry of individual stars in the Local Group. Both types of observation will improve our knowledge of the evolution of dwarfs. A last important mission of TMT for dwarf galaxies is the search of massive stars with low metallicity in order to understand their evolution. With a better knowledge of this type of stars, it will improve our models of chemical evolution of dwarfs (Skidmore et al., 2015). Finally, the gravitational wave detectors eLISA will be able to detect the merging of IMBHs. If the dwarf galaxies really have a cusp, the merger rate will be strongly increased in comparison to a core, thus eLISA will provide some limits on the shape of the inner profile of dwarf galaxies (Amaro-Seoane et al., 2013).

The next generation of telescopes will provide large quantities of data in the Local Group (e.g. Pristine, TMT and 4MOST/WEAVE), intermediate redshift (e.g. TMT and ELT) and high redshift (e.g. SKA and JWST). Among the new data are spectroscopy of individual stars (in the Local Group) that will strongly constrain the star formation histories and chemical abundances, new constraints on the dark matter through indirect detections or the inner profile of dwarf galaxies and global trends on the evolution process through the global properties of dwarfs. With this, numerical simulations of dwarf galaxies might be able to overcome their current limitations and provide more accurate predictions.

1.4 The Challenges of Dwarf Galaxy Simulations

In this section, I will provide a list of some of the issues found in simulations of galaxies and a few specific to dwarf galaxies. I will start with the issues related to the subgrid models introduced in such simulations (section 1.4.1). Then, as hydrodynamics play an important role in the evolution of galaxies, a large fraction of this section will be dedicated to the problems related to it (section 1.4.2). Next, the numerical issues are presented with the high performance computing issues (section 1.4.3) and the chaos (section 1.4.4). I will conclude with issues more related to dwarf galaxies. First the ones related to stars (section 1.4.5) and, then, the issue related to our bias from the Local Environment (section 1.4.6).

1.4.1 Subgrid Models

Numerical simulations in astrophysics, and specially for galaxy formation, require a large quantity of models that spans numerous scales (from atomic physics to cosmology), and different physics. Today's numerical simulations are able to integrate a large quantity of meaningful physics such as gravity, cosmology, hydrodynamics (Springel, 2005; Teyssier, 2002; Springel, 2010; Borrow et al., 2018; Schaye et al., 2015; Tremmel et al., 2017), radiative cooling (Smith et al., 2017; Wiersma et al., 2009a), star formation (Katz et al., 1996), supernovae explosions (Simpson et al., 2018; Revaz and Jablonka, 2018; Schaye et al., 2015; Springel et al., 2018), stellar winds (Oñorbe et al., 2015), AGNs (Schaye et al., 2015; Springel et al., 2018; Tremmel et al., 2017), chemical evolution (Wiersma et al., 2009b; Revaz and Jablonka, 2018), dust (Aoyama et al., 2018; Davé et al., 2019; McKinnon et al., 2017), radiations (Rosdahl et al., 2013; Hopkins and Grudić, 2019; Vandenbroucke and Wood, 2019)), cosmic rays (Farber et al., 2018; Chan et al., 2019), magnetic fields (Marinacci et al., 2018; Tricco, 2015; Fromang et al., 2006) and exotic physics (e.g. Chan et al., 2018; Harvey et al., 2018). It would be impossible to resolve the details of each model presented previously within a simulation, thus, subgrid models have been introduced to significantly simplify them. For example in the case of star formation (see section 4.5 for more details), we need to reach densities comparable to the Sun's central density (100 g cm^{-3}) in order to follow the formation. Such densities are 10^{22} times larger than typical densities currently reached in cosmological simulations. As increasing the resolution is not feasible, the gas is assumed to instantaneously transform into stars when some conditions are met (usually collapsing cloud, low temperature and high densities) with a probability proportional to the density (Katz et al., 1996; Revaz and Jablonka, 2012; Schaye et al., 2015; Oñorbe et al., 2015). While the subgrid models had a remarkable success, they are highly uncertain due to the lack of direct constraints.

A particular issue is related to the feedback from supernovae. During the explosion, a large quantity of energy is converted into different forms. While most of the energy is transported by a burst of neutrinos, a fraction is transported in the ejected gas and photons and, together, they heat and push the surrounding medium producing effects observable at the galactic scale. The details of this interaction are observable only with a high spatial resolution, thus theoretical models of supernovae are constrained mainly by a single event (SN 1987a; Arnett et al., 1989). For galaxy simulations, it means that we have a large degree of freedom in the choice of the feedback efficiency, and it is often used through the calibration process of simulations raising questions about if we are really predicting properties of galaxies or simply fitting them. Even worst, the coupling with the radiative cooling renders the heating from supernovae inefficient as gas with high density and temperature has a strong cooling rate, thus creating the cooling catastrophe (Katz et al., 1996). To solve this issue, an additional subgrid model, the delayed cooling, was invented (Abadi et al., 2003; Stinson et al., 2006). It consists in simply turning off locally the radiative cooling for a given amount of time after a supernova (usually around 5 Myr). It can be argued that this cooling catastrophe is due to some missing physics such as discussed in Teyssier et al., 2013, but the amount of time is still picked through a calibration process and not based on direct physical constraints. Other groups have favored to deposit the energy in form of mechanical energy and not thermal energy (Lupi, 2019; Whalen et al., 2010; Dalla Vecchia and Schaye, 2008; Springel and Hernquist, 2003). This form is indeed taking more time to dissipate and avoids the usage of such non-physical subgrid models, but the remaining question is: what fraction of the energy should be deposited in form of mechanical energy? A possible way to answer this question is to study dwarf galaxies as they are the most easily perturbed galaxies in the universe. The end of their star formation provides an indirect constraint on the different form of feedback present in galaxies such as the supernovae feedback, UV radiations, stellar winds or hypothetical feedback from dark matter.

1.4.2 Hydrodynamics

In the case of hydrodynamics simulations, the two main issues are shocks and the Kolmogorov cascade. While shocks can be solved with most numerical scheme, it can be only done with numerical schemes of first order and requires high resolution to be accurately resolved (Hesthaven, 2018), the Kolmogorov cascade cannot be fully recovered without extremely high resolution. The principle is to transfer the energy from large scales towards small scales through the turbulence. The idea developed by Kolmogorov is that the largest eddies are unstable and thus will split into smaller eddies containing the energy of the previous one. This process is done until reaching a length scale where the viscous forces dominate, destroy the eddies and thus transform the kinetic energy of the eddies into internal energy (Dubrulle, 2019). In the case with magnetic field, this cascade becomes even more important as it can amplify the magnetic field at large scale through the transformation of the kinetic energy of the smallest eddies into magnetic energy. In such case, the viscous length for the interstellar medium is typically of 10^{-5} pc (Schober et al., 2013). Unfortunately, this resolution will not be obtained before a long time as cosmological simulations of dwarf galaxies currently achieve resolutions of the order of 1 pc.

Mainly two different approaches exist for hydrodynamics in astrophysics that are driven by a choice on the focus between either gravity or hydrodynamics. The two methods have both advantages and drawbacks. On one hand, Smoothed Particles Hydrodynamics (SPH) is a Lagrangian and particle based approach that is particularly good at dealing with gravity and automatically increases its resolution in dense environment. As the equations are derived from a time independent Lagrangian (see section 4.3), the momentum, angular momentum and entropy are well conserved by SPH, but shocks are not well handled. Even if the conservation is an essential feature for simulations, an artificial viscosity is added in order to suppress the entropy conservation in shocks as they are naturally producing entropy (Balsara, 1995; Springel, 2005; Borrow et al., 2020). Another issue comes from the contact discontinuities between a low and high density region where the pressures are not well computed due to the strongly non-homogeneous distribution of particles. Even if modern formulations can partially solve this issue (Hopkins, 2013), they are not fully consistent when coupled with more advanced physics. An appealing solution seems to come from the addition of an artificial diffusion that reduces discontinuities within a particle's kernel (Borrow et al., 2020). In general, it is also complicated to generate boundary conditions with SPH, but in the case of astrophysics, the boundaries have trivial geometries (e.g. periodic or isolated).

On the other hand, Adaptive Mesh Refinement (AMR) is particularly well suited for hydrodynamics, and specially for shocks, thanks to the Godunov method and the Riemann solver (Hesthaven, 2018; Teyssier, 2015). Due to the multi scale physics present in cosmological simulations, the mesh requires to be dynamically refined depending on the local properties of the medium (Teyssier, 2002). The refinement criterion needs to be defined with extra care as, for example, a simulated object with high velocity might leave a high resolution cell before the new cells were refined enough (Springel, 2010). AMR simulations suffer from numerical diffusion that is proportional to the spatial discretization and the velocity of the medium. Thus, AMR is usually considered as Galilean non-invariant as if a constant and homogeneous velocity is added to the initial conditions, it will result in a different numerical diffusion and solution (Robertson et al., 2010). Pontzen et al., 2021 propose a solution to mitigate partially this issue by setting to zero the average velocity of the targeted object in the initial conditions, but such a solution cannot be applied globally. Finally, at high redshift, the initial mesh is roughly homogeneous and cannot properly resolve the growth of the smallest structures. The adaptive mesh refinement tries to anticipate this but the refinement by a factor of 8 produces a discontinuity in the evolution. At low redshift, this suppression results in a lack of small halos (Springel, 2010).

Two hybrid methods, that are the moving mesh method (`AREPO`; Springel, 2010) and the meshless method (`GIZMO`; Hopkins, 2015), have been implemented in cosmological codes. Both methods can be constructed on SPH codes with changes to the equations in order to make them compatible with the Godunov method used in AMR. As for SPH, the quality of the solution depends on the distribution of particles, but in the case of hybrid methods, a non-homogeneous distribution can lead to catastrophic results. To solve this issue, `AREPO` contains a regularization procedure that moves the particles around without necessarily following the flow and `GIZMO` reverts to SPH for the gradient estimators in such conditions. While the hybrid methods are designed to resolve more accurately shocks than SPH, they are not able to manage flows strongly dominated by the kinetic energy and include a switch between solving two different set of equation. In `GIZMO`, they evolve the internal energy along with the usual total energy and use it to set the pressure and temperature in supersonic flows. In `AREPO`, they evolve the entropy along with the total energy and drop the total energy conservation in such flows in favor of entropy conservation (Hopkins, 2015). In summary, we have a large variety of methods for the hydrodynamics that works well on the classical tests (Springel, 2010; Hopkins, 2015; Teyssier, 2002; Borrow et al., 2020), but large differences call still be observed in cosmological simulations without clear answer on the correct solution (Roca-Fàbrega et al., 2021).

1.4.3 High Performance Computing

The telescope Euclid should be launched in 2022 and will start to observe a third of the sky up to redshift 2 for quiescent galaxies of stellar mass larger than $4 \cdot 10^{10} M_\odot$ (Euclid Collaboration, 2011). If we wish to simulate the same volume with a comparable resolution, it would require a simulation cube of 8.5 Gpc containing at least 10^{15} particles (assuming a resolution of 1000 particles for the lowest stellar mass given previously). At the opposite scale for galaxy simulations, TMT (among other telescopes) will look at individual stars in dwarf galaxies, thus requiring simulation with resolution better than $1 M_{\odot}$ or about 10^7 times better than the large scale simulations given previously. As seen before, dwarfs are particularly sensitive to their environment and a direct comparison with observations of Milky Way's satellites should be done only with simulations in a similar environment. Such system requires a cube of roughly 50 Mpc (Garrison-Kimmel et al., 2019). Simulating such systems, without using the zoom technique, would require more or less the same number of particles than large scale simulations given previously. With such large differences of resolution between small and large scales, it is impossible to find universal methods for the simulation of galaxies and thus a large variety of methods exists (e.g. see models in Revaz and Jablonka, 2018; Wheeler et al., 2019; Libeskind et al., 2020; Sawala et al., 2016; Cloet-Osselaer et al., 2014; Emerick et al., 2016; Lupi, 2019; Macciò et al., 2017; Teyssier and Commerçon, 2019). Current hardware and software are not able to handle such large number of particles. In order to move towards higher resolution and/or larger scale, simulation codes need to be significantly improved in terms of performances. This is particularly true if additional physics such as radiative transfer or magnetic fields are included. An additional issue usually largely disregarded is the disk space required for storing the output of the simulations. Most of the disk space used by simulations is for writting "snapshots" that contains the state of the simulation at a given time. It means that larger simulations will need either more disk space or to decrease the time accuracy of the output by reducing the number of snapshots.

To reduce the computation time and output size, the zoom technique can be applied in order to focus on a single region (e.g. a Milky Way like galaxy and its satellites). It consists, for particle based codes, to adapt the mass of the particles in the initial conditions depending on the position in the simulation volume (see section 9.2 or Hahn and Abel, 2011 for more details). A similar approach is possible for AMR. If the high resolution area is properly defined, no particle with large mass (background particle) will be found within the selected galaxies and the galaxies will behave in the same way as if they were simulated in a gravitational potential corresponding to a full volume simulation. For the hydrodynamics and SPH, the gas is only added in the high resolution area and thus cannot be influenced by baryonic physics outside the region of interest. While this method allows us to obtain extremely high resolution with small amount of computational time, it comes with the disadvantages of having a low quantity of galaxies for statistics, and therefore is not a replacement to low resolution and full volume simulations.

1.4.4 Chaos in Numerical Simulations

The reproducibility of galaxy simulations is another important issue for the quality of science and is rendered complex due to the chaotic behavior of galaxies. The linear perturbation theory describes particularly well the behavior of the protogalaxies at high redshift and the large scales seen at z=0, but this linear theory breaks in galaxies and non-linear effects become dominants.

As gravity is a purely attractive force, any perturbation can be quickly amplified and produces large differences in the final state of a system. The hydrodynamics also contribute to the chaotic evolution of galaxies with turbulence. Therefore, galaxies are chaotic objects and as such are highly sensitive to the initial conditions and/or perturbations (Keller et al., 2019). In simulations, this chaotic behavior deeply impacts the reproducibility of simulations at the individual galaxy level. Depending on the hardware and compiler, the rounding errors might be different and seed the growth of instabilities leading to different solutions. While both can be kept under control, advanced parallel computing techniques (such as task based parallelism) can change the order of the operations (e.g. $(a + b) + c$ vs $a + (b + c)$) in different runs and introduce different rounding errors depending on which thread is doing its computations first. Another source of noise comes from the subgrid models that sometimes directly include random behavior such as the star formation or stellar feedback.

1.4.5 Issues Related to the Stars

In high resolution simulations of dwarf galaxies, a correct sampling of the stellar initial mass function (IMF) is critical as low mass dwarfs can be destroyed by a single supernova. For example, in low resolution simulations, a stellar particle will explode hundreds of supernovae per time step due to both the large steps and the large number of stars contained in each stellar particle. At current resolution for dwarf galaxies ($\sim 10^3 M_\odot$ per particle), this is not the case any more and, in average, less than 1 supernova explodes in some time steps and requires a proper discrete treatment otherwise a continuous heating is produced (Revaz et al., 2016). In the same paper, two other sampling methods were analyzed and provide accurate results for resolutions of $10^4 M_\odot$. When moving at higher resolution with the classical stellar particles representing a population of stars, all methods will start to have issues with their sampling of the IMF as the hypothesis of a representative population of stars cannot hold any more. Cosmological simulations of dwarf galaxies are, therefore, at the dawn of simulations at the individual star level (Grudić et al., 2021; Wall et al., 2019).

Observations of the mean metallicity and luminosity relation of galaxies show a strong correlation. In the ultra faint dwarf domain, this correlation quickly changes and the metallicity becomes far less dependent on the luminosity. In such system, the star formation stopped early on and thus the population is strongly dominated by old stars such that their metals come primarily from Pop III stars. Due to our lack of understanding in Pop III stars, current simulations tend to under-predict the metallicity of such systems (e.g. Figure 10 in Applebaum et al., 2021).

Even worst, ultra faint dwarf galaxies require an accurate description of metal diffusion through the gas. While some methods exists for the diffusion in SPH (Greif et al., 2009; Shen et al., 2010), they are still suffering some issues (e.g. extremely low time steps or bad behavior with individual time steps). At lower resolution, the diffusion is usually absent and replaced by the smoothed metallicity (Wiersma et al., 2009b). In this scheme, the metallicity used in all computations is averaged over the true metallicity of the neighbors and, therefore, the smoothed metallicity is not perfectly conserved but oscillates around the expected value. As this smoothed metallicity is applied only within the particles' radius, this is strongly dependent on the resolution and is unable to "diffuse" metals at large scale in high resolution simulations. In consequence, the abundances in stars will show a scatter too large in comparison to observations at such resolution without diffusion. A proper treatment of the metal diffusion should emerge in the upcoming years and will enable to accurately follow the chemo-dynamical evolution of ultra faint dwarf galaxies.

1.4.6 Local Environment Bias

In observations, the resolution of galaxies is always proportional to the distance from the Milky Way, therefore a strong bias is present due to our relatively good knowledge of our own environment. Until recently, all the simulations comparing observations of the Local Group were run without any geometrical constraints on the initial conditions assuming that the Milky Way is a typical galaxy in a typical environment (e.g. as done in Revaz and Jablonka, 2018; Garrison-Kimmel et al., 2019; see Hahn and Abel, 2011 for details on the technique). While this assumption is at the center of modern physics since the scientific revolution of Copernicus, some recent works seems to indicate that we might not be in the most standard galaxy (e.g. lack of star formation in the satellites (Geha et al., 2017), the late reionization (Aubert et al., 2018) or plane of satellites (Gillet et al., 2015; Buck et al., 2016; Bahl and Baumgardt, 2014)). No matter if it is really the case or not, it becomes important to improve the quality of our initial conditions when comparing with observations of the Local Group. A solution seems to emerge through the constrained simulations (Zaroubi et al., 1999) and the usage of distance and velocity measurement of the Local Group's galaxies. The idea consists in constraining at the same time the initial conditions through the power spectrum (as usual) and the geometry of the Local Group obtained by the distance-velocity measurement. The main issues currently faced are the proper treatment of the measurement uncertainties and how to properly move backward in time the Local Group's geometry (Doumler, 2012; Graziani et al., 2019; Libeskind et al., 2020; Sorce, 2015; Sorce et al., 2014).

1.5 Impact of my Work

The aim of my thesis is to tackle some of the previous challenges. In this section, I will provide my different project in the order of importance, In the rest of the thesis, they will be given in a more chronological way. The main ones have been the study of the ram pressure stripping on dwarf galaxies and the development of `SWIFT` through the implementation of `GEAR`'s models and the Continuous Simulation Data Stream (CSDS). When I started my PhD, our code `GEAR` (based on `GADGET-2`) has been under constant development during roughly 10 years and was able to accurately simulate dwarf galaxies as seen in Revaz and Jablonka, 2018 in a cosmological but low density environment. A question arising directly from this publication was "What is the impact of the environment on the galaxies produced by `GEAR`?".

1.5.1 Ram Pressure Stripping with `GEAR`

In my first paper (Hausammann et al., 2019), I tried to answer this question by running isolated simulations with initial conditions extracted from cosmological simulations at high redshift. Using the moving box technique (wind tunnel simulation including an evolving central potential and accurate orbits; see chapter 2 for more details and chapter 5 for details on `GEAR`'s models), I have discovered that the thermal pressure from the host's hot halo strongly impacts our description of the ram pressure. To evaluate the relative impact of both the thermal pressure and ram pressure, I have introduced the ratio of the two pressures: $\beta_{\mathrm{RP}} \propto T_W/v_w^2$, where T_w and v_w are the wind's temperature and velocity. In case of high thermal pressure ($\beta_{\mathrm{RP}} > 2.5$ in the galaxy studied and in an environment compatible with our own at high redshift), this additional, and usually neglected, pressure can compress the cold gas (below $T = 1000K$) still present at high redshift. It can even totally protect it from any evaporation due to the supernovae or UV background. Indeed, the additional pressure increases the density of the gas such that it becomes self-shielded against the UV due to H. For the hot gas, the difference of thermal pressure is not sufficient to protect it and, thus, is quickly stripped. This preservation of the cold gas enhances the star formation and then a slow starvation is observed due to the lack of a hot gas reservoir. With my results, the infall time becomes far more important as a galaxy that already lost all its gas would not be impacted by the thermal pressure.

The dwarfs produced with the moving box technique are far more luminous and have abundances patterns unseen around the Milky Way (MW). It means that the thermal pressure (and the ram pressure) is unlikely to have played a dominant role in the formation of the MW's satellites. The survey SAGA focused on MW like galaxies' satellites and found a large quantity of bright satellites (Geha et al., 2017) that might be an indication of the thermal pressure's role in these galaxies. Possible reconciliation between the MW's observations and simulations could come from the inclusion of UV-fluxes produced by the MW that would heat the cold gas and help the ram pressure stripping or simply a late accretion of our satellites.

1.5.2 Moving our models into `SWIFT`

From my results, we have seen, as expected, that including the environment is necessary to produce correct predictions and to have a fully consistent simulation. Unfortunately, `GEAR` is not fit for modern usage of HPC servers and to simulate both the MW and its satellites at the required resolution for dwarfs. It requires a large upgrade of its performances. Therefore, we decided to move our models to the exascale code `SWIFT` that was still under heavy development at this moment and only managed to simulate basic radiative cooling, hydrodynamics and gravity (without any cosmology) [6]. `SWIFT` has shown a speed-up of 36 against `GADGET`-2 thanks to its task based approach and graph based repartition of the work (Borrow et al., 2018). This should be largely sufficient for our future needs.

---

[6] See for example state at commit 43d8be2da6eda1bffeca001d450868d75b692fda

My main contributions to `SWIFT` were through the development of the missing tasks related to the stellar physics, the implementation of `GEAR`'s physics and a visual tool for the task dependencies (see graphs in chapter 5). This project was successfully completed and now `SWIFT` reproduces the dwarf galaxies presented in Revaz and Jablonka, 2018 with a speed-up of 7.65 times (see Figure 6.13) and could be even better on larger simulations. As `SWIFT` contains multiple models (e.g. `EAGLE`, planetary simulations, ...), the name `SWIFT-GEAR` will be used to describe our own implementation.

1.5.3 Continuous Simulation Data Stream (CSDS)

My last main project was the development of the Continuous Simulation Data Stream (CSDS) in `SWIFT` that consists in a new output system more efficient in terms of storage than the snapshots. As cosmological simulations have a large variety of time scales, any efficient output should take this into account to maximize the information written on the disk. In the CSDS, we were able to reduce the disk space by one order of magnitude for the same time accuracy (or almost two orders of magnitude in accuracy for the same storage) by moving from a vision where we write the state of the simulation at a given time towards a particle based approach that writes the particles individually only when needed (see chapter 7 for more details).

1.5.4 The `AGORA` collaboration

My first side project and the most compelling among them for the community is the `AGORA` project. This collaboration between some of the main groups in numerical astrophysics aims at comparing results from common simulations and simplified physics with the aim of improving the reproducibility of simulations and trying to understand the origin of the differences. Such projects are of tremendous importance in a domain where reproducibility is often neglected and codes are kept private or not fully open source.

1.5.5 New Model for the Star Formation

Lately, I have been working on my two last side projects to prepare our group for the next generation of simulations with the new star formation strategy and the generation of constrained Initial Conditions (ICs). Thanks to the two projects, we will be able to have simulations in an environment comparable to the Local Group and to achieve larger resolution.

Achieving high resolutions requires to overcome the limits given by the single stellar population (SSP) representation of stars. In the SSP, each stellar particle is massive enough to contain at least thousands of stars and thus a statistical approach can be applied based on the stellar initial mass function (IMF). Due to the low number of stars in ultra faint dwarf galaxies, it would mean that all the stars would be contained into a low quantity of particles and it is clearly not sufficient to have an accurate evolution of the dwarfs. Moving towards higher resolution requires to drop this

SSP approach and move towards an individual star representation. While this is usually easily done for the feedback, the star formation process becomes a lot more complicated due to the insufficient mass contained in a single gas particle for massive stars. To solve this issue, I have implemented the basics for the sink particles applied to stars in `SWIFT`. The sink particles will be spawned in the unresolved and collapsing areas of galaxies and will immediately start to accrete the surrounding gas. Once they reach a sufficient mass to produce a star, they will simply spawn it and let it evolve totally independently of the sink. This project still require some work, but the hardest part is done and ready to use without any knowledge of the internal structure of `SWIFT`.

1.5.6 Constrained Initial Conditions

Finally, my last side project on the constrained ICs was based on Graziani et al., 2019. While they were only interested into reproducing the large scale structure of the Local Universe as seen today, their method can be applied to the generation of initial conditions. Following their work, Sorce et al., 2014 and with the help of the code `MUSIC`, I have implemented the first fully open source code [7] on GPU containing the complete pipeline for the generation of constrained ICs.

The structure of my thesis is done in a chronological manner moving from projects done with `GEAR`: the study of the ram pressure (chapter 2) and the `AGORA` project (chapter 3). Then the current state of `SWIFT` is given: first the description of our models (chapter 4), followed by the implementation details (chapter 5), some verification and scaling tests (chapter 6). The end of my thesis focuses on projects that serves as foundation for future research within our group: the CSDS (chapter 7), the new physics with the sink particles (chapter 8) and finishing with the constrained initial conditions (chapter 9). In the appendix, I am describing how to generate a cooling table for `GRACKLE`, the examples that I have implemented in `SWIFT` to test the code and the computation of the coefficient for the initial mass function.

---

[7] https://gitlab.com/loikki/initialconditions

Table 1.1 – List of the abbreviations used in this document.

ΛCDM	Λ Cold Dark Matter	MCMC	Markov Chain Monte Carlo
AGB	Asymptotic Giant Branch	MPI	Message Passing Interface
AGN	Active Galactic Nucleus	MW	Milky Way
AMR	Adaptive Mesh Refinement	OS	Operating System
BAO	Baryonic Acoustic Oscillations	PDF	Probability Distribution Function
BH	Black Hole	POP I	Third generation of stars
CF	CosmicFlows	POP II	Second generation of stars
CIC	Cloud In Cell	POP III	First generation of stars
CMB	Cosmic Microwave Background	RAM	Random Access Memory
CMD	Color-Magnitude Diagram	RP	Ram Pressure
CPU	Central Processing Unit	RPS	Ram Pressure Stripping
CR	Constrained Realization	RR	Random Realization
CSDS	Continuous Simulation Data Stream	RT	Radiative Transfer
DMO	Dark Matter Only	RZA	Reverse Zel'dovich Approximation
FFT	Fast Fourier Transform	SF	Star Formation
FMM	Fast Multipole Method	SFH	Star Formation History
FoF	Friend of Friend	SMAH	Stellar Mass Assembly History
GPU	Graphics Processing Unit	SMC	Small Magellanic Cloud
GR	General Relativity	SN	Supernova
GW	Gravitational Waves	SNII	Supernova of type II
HPC	High Performance Computing	SNIa	Supernova of type Ia
IC	Initial Condition	SP	Stellar Population
ICM	Intracluster Medium	SPH	Smoothed Particles Hydrodynamics
IGM	Intergalactic Medium	SS	Single Star
IMBH	Intermediate-Mass Black Hole	TP	Thermal Pressure
IMF	Initial Mass Function	UVB	Ultraviolet Background
ISM	Interstellar Medium	WF	Wiener Filter
LMC	Large Magellanic Cloud	WIMP	Weakly Interacting Massive Particles
LoS	Line of Sights		

GEAR Part II

2 Satellite Dwarf Galaxies: Stripped but not Quenched

> Hobbits always so polite, yes! O nice hobbits! Smeagol brings them up secret ways that nobody else could find. Tired he is, thirsty he is, yes thirsty; and he guides them and he searches for paths, and they saw sneak, sneak. Very nice friends, O yes my precious, very nice.
>
> — J. R. R. Tolkien, The Two Towers

In this project, I have studied the impact of the ram pressure on dwarf galaxies resulting from a cosmological context using the code `GEAR` (see section 9.1 for details on the generation of initial conditions) . This work resulted in a publication where I have shown that the thermal pressure can have a tremendous impact on the expected stripping from the ram pressure. While I will not provide the details of `GEAR` here, the models used are the same as for `SWIFT-GEAR`. It means that looking at the details provided in chapter 4 is enough to understand how `GEAR` is behaving. As this work resulted in a publication, I will only introduce it with some context along with a small summary and then provide the full article.

Dwarf galaxies have become today an important tool to test cosmological models at low scales thanks to, for example, the missing satellites problem presented in the introduction. In the last 20 years, both observations and simulations have been improved and are slowly converging to the same picture where the missing satellites are simply destroyed by the environment, the UV background (UVB) and the stellar feedback. On the simulation side, new numerical techniques such as zoom simulation (Hahn and Abel, 2011), individual time steps (Springel, 2005), synchronization of particles (Durier and Dalla Vecchia, 2012) or modern SPH (Hopkins, 2013) have enabled us to largely increase the resolution and improve the physical models with, for example, the smoothed metallicity (Wiersma et al., 2009b), the H self shielding (Aubert and Teyssier, 2010), the UVB (Haardt and Madau, 2012) or an accurate sampling of the IMF

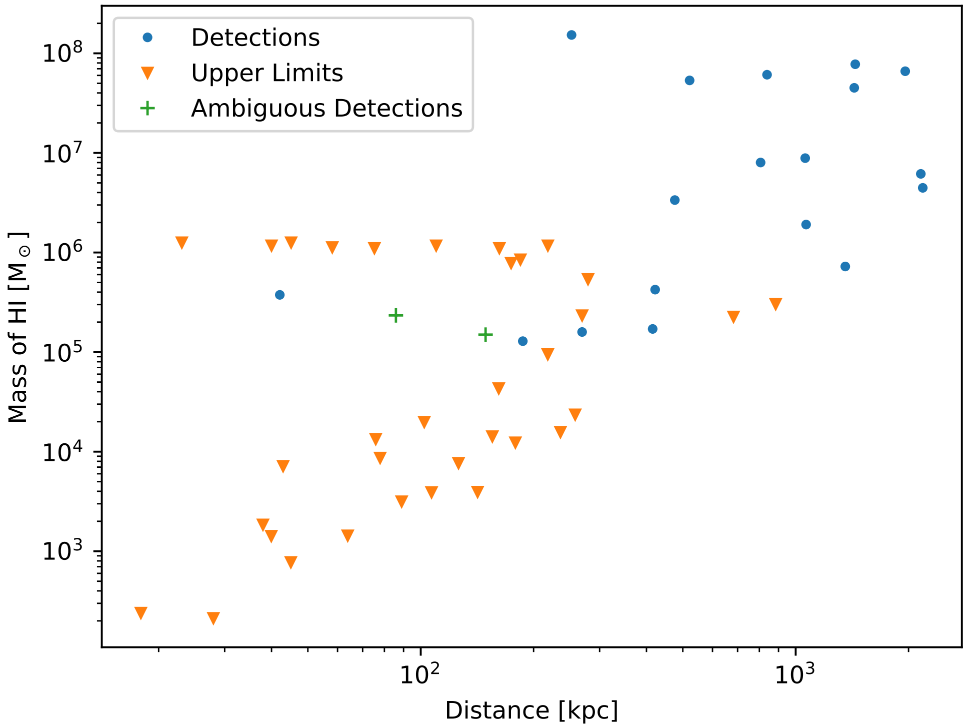


Figure 2.1 – Morphology-density relation for dwarf galaxies shown as the mass of HI as function of the distance from the host. In this graph, it can be seen that the closer a dwarf is from its host, the less gas it contains and thus it provides a good indication that the environment must be playing a role. Data extracted from Grcevich and Putman, 2009.

(Revaz et al., 2016) giving us today realistic galaxies (Valcke et al., 2008; Revaz et al., 2009; Sawala et al., 2010; Schroyen et al., 2011; Revaz and Jablonka, 2012; Cloet-Osselaer et al., 2014; Sawala et al., 2016; Wetzel et al., 2016; Fitts et al., 2017; Macciò et al., 2017; Escala et al., 2018; Revaz and Jablonka, 2018; Applebaum et al., 2021; Brooks and Zolotov, 2014). In the physical models, a particular point of interests has been the impact of the environment on the evolution of the galaxies. Observations of the morphology-density relation in the Local Group suggest that they are deeply connected to the environment as gas rich dwarfs are preferentially found at large distance from any host (e.g. Milky Way or Andromeda) and gas poor dwarfs close to their host (McConnachie, 2012; Grcevich and Putman, 2009; Einasto et al., 1974). In Figure 2.1, this relation is shown. Therefore any process that can remove the gas close to a host has been studied and particularly the tidal and ram pressure stripping. If the gas is removed quickly enough, it is expected that the luminosity at $z = 0$ and the star formation are reduced too. Thus, it could explain why, we are not able to see more visible dwarf galaxies around the Milky Way (Del Popolo and Le Delliou, 2017; Arraki et al., 2014).

Initially, studies of the environment were performed in dark matter only simulations and were only able to take into account the gravitational impact through the tidal stripping. Even if this stripping is able to totally destroy a galaxy, it tends to be really efficient only for orbits with low pericenter and does not seem to be able to explain the morphology-density relation alone (Peñarrubia et al., 2008; Nichols et al., 2014; Simpson et al., 2018). With the first simulations including baryonic physics, the studies started to focus more and more on ram pressure stripping that consists, as mentioned in the introduction, in the friction of the satellite gas with the gas from the hot halo of the host and is usually described by the following relation:

$$P_{\mathrm{RP}} = \rho_w v_w^2 \tag{2.1}$$

where ρ_w is the local density of hot halo and v_w the wind's velocity (or in the case of dwarf galaxies, the dwarf's velocity). Such friction starts by removing the external layers and slowly move towards the inner part thanks to the size reduction of the dwarf and the decrease of pressure at the outer layers. When the outer layer pressure decreases, the internal gas is decompressed and thus reduces its density such as it becomes more impacted by the ram pressure. This can be repeated extremely quickly and remove all the gas from a galaxy in less than a Gyr. While the basic view is easily understood, the interplay between the different physics (e.g. star formation, hydrodynamics, UV background, radiative cooling, stellar feedback) in the baryonic matter complicates deeply the situation. For example, supernovae reduce the local density and create large scale outflows that will accelerate the gas stripping. In recent years, studies have shown that the ram pressure is efficient at removing the gas and followed by a fast dampening of the star formation (Mayer et al., 2006; Simpson et al., 2018; Fillingham et al., 2016; Emerick et al., 2016; Yozin and Bekki, 2015) but also other groups have reported the exact opposite with an enhancement of the star formation (Wright et al., 2019; Henderson and Bekki, 2016; Nichols et al., 2015; Kapferer et al., 2009; Kronberger et al., 2008). This indicates that efficiency of the ram pressure stripping is dependent on the hydrodynamics method and the subgrid models.

Recently, simulations started to be able to resolve the cold gas and to include the H self shielding (Revaz and Jablonka, 2018; Wheeler et al., 2019). In our paper (Hausammann et al., 2019), we have shown that this cold gas dramatically changes the picture presented before. We used the simulations of dwarf galaxies presented in Revaz and Jablonka, 2018 that can be considered as done in isolation from any host due to the relatively small size of the simulation volume and the lack of massive galaxies. We extracted the galaxies from their cosmological volume at $z = 2.4$ and injected them into our improved wind tunnel simulation called "Moving box technique". This technique consists in a wind tunnel setup where the box is in a non-inertial frame that allows to move it around a host galaxy and adapt the wind according to the orbital parameters and the evolution of the host. Even if we are perturbing the galaxy with the extraction, it allows us to have dwarfs with realistic histories and not necessarily at equilibrium as done with analytical models such as in Mayer et al., 2006; Nichols et al., 2015; Emerick et al., 2016.

In the publication, we have shown that due to the large difference of temperature between the hot halo and the cold gas of the satellite, the thermal pressure starts to play an important role. This additional pressure compacts the dwarf galaxy and makes it more resistant to external effect such as the ram pressure. Another example is the case of the UV background (UVB) that heats the dwarf's gas and efficiently evaporates it in the low mass isolated dwarf galaxies. In the presence of the additional thermal pressure from the hot halo, such galaxy increases its density and resist more strongly against the UV background as it can reach the H self shielding. This shielding makes the gas optically thick and prevents the UV background from reaching the inner part of the galaxy, thus reducing the evaporation speed. It is worth to mention that due to this interaction between the ram pressure and the UVB, the infall time of a dwarf galaxy plays an important role. If it happens before the peak of the UVB, the galaxy will be protected against it and tends to form numerous stars but, if it happens after the peak, the gas will be, at least partially, evaporated and thus a lower number of stars will be produced.

Finally, we concluded that in the case of the Milky Way's satellites without any additional physics, it would mean that the satellites arrived very early (before the end of the galaxy mass assembly) or very late (after the star formation quenching by the UVB). Another possibility would be that we underestimate the impact of the Milky Way's own radiation in our treatment of the UVB and thus would evaporate the gas of the dwarf galaxies earlier and faster. The complete article is presented in the next few pages.

Astronomy & Astrophysics manuscript no. main
August 26, 2021

Satellite dwarf galaxies: Stripped but not quenched.

Loic Hausammann[1], Yves Revaz[1], and Pascale Jablonka[1, 2]

[1] Laboratoire d'Astrophysique, Ecole Polytechnique Fédérale de Lausanne (EPFL), 1290 Sauvergny, Switzerland
e-mail: `loic.hausammann@epfl.ch`
[2] CNRS UMR 8111, GEPI, Observatoire de Paris, Université PSL, 92125 Meudon Cedex, France

Received September XXX; accepted XXX

ABSTRACT

In the Local Group, quenched gas-poor dwarfs galaxies are most often found close to the Milky Way and Andromeda, while star forming gas-rich ones are located at greater distances. This so-called morphology-density relation is often interpreted as the consequence of the ram pressure stripping of the satellites during their interaction with the Milky Way hot halo gas. While this process has been often investigated, self-consistent high resolution simulations were still missing. In this study, we have analysed the impact of both the ram pressure and tidal forces induced by a host galaxy on dwarf models as realistic as possible emerging from cosmological simulations. These models were re-simulated using both a wind tunnel and a moving box technique. The secular mass growth of the central host galaxy, as well as the gas density and temperature profiles of its hot halo have been taken into account. We show that while ram pressure is very efficient at stripping the hot and diffuse gas of the dwarf galaxies, it can remove their cold gas ($T < 10^3$ [K]) only in very specific conditions. Depending on the infall time of the satellites relatively to the build-up stage of the massive host, star formation can thus be prolonged instead of being quenched. This is the direct consequence of the clumpy nature of the cold gas and the thermal pressure the hot gas exerts onto it. We discuss the possibility that the variety in satellite populations among Milky Way-like galaxies reflects their accretion histories.

Key words. methods: numerical – galaxies: dwarfs – galaxies: interactions

1. Introduction

Dwarf galaxies are the faintest galaxies found in the Universe. In a hierarchical ΛCDM framework, they are the most common systems and, in their early evolution phase, they can serve as building blocks of larger galaxies. Suggestions are made that dwarfs could have played a substantial role during the epoch of reionization (Atek et al. 2015; Robertson et al. 2015; Bouwens et al. 2015). Understanding their role in this context requires a detailed picture of their formation and evolution.

Noteworthily, dwarf galaxies have challenged ΛCDM on a number of questions, such as the missing satellites (Moore et al. 1999; Klypin et al. 1999), the too-big-to-fail (Boylan-Kolchin et al. 2011, 2012) or the core-cusp (Navarro et al. 1996, 1997; Moore 1994) problems (see Bullock & Boylan-Kolchin 2017, for a complete review). These issues were originally highlighted for dark matter only cosmological simulations. However, since these pioneering simulations, major improvements have been achieved, in particular thanks to the inclusion of the evolution of the baryons in the simulations, but also thanks to very significant progresses in numerical methods (Springel 2005; Wiersma et al. 2009; Aubert & Teyssier 2010; Hahn & Abel 2011; Durier & Vecchia 2012; Haardt & Madau 2012; Hopkins 2013; Revaz et al. 2016). As a consequence, when baryonic physics is properly included, the numerical simulations are now able to reproduce a large variety of observed properties (Valcke et al. 2008; Revaz et al. 2009; Sawala et al. 2010; Schroyen et al. 2011; Revaz & Jablonka 2012; Cloet-Osselaer et al. 2012; Sawala et al. 2012; Cloet-Osselaer et al. 2014; Sawala et al. 2016; Wetzel et al. 2016; Fitts et al. 2017; Macciò et al. 2017; Escala et al. 2018; Revaz & Jablonka 2018). High resolution cosmological hydro-dynamical simulations of the Local Group such as APOSTLE (Sawala et al. 2016) or Latte (Wetzel et al. 2016; Garrison-Kimmel et al. 2018) also lead to solving the cosmological problems previously mentioned. However a global consensus on whether or not those problems are definitely solved is still missing. See Bullock & Boylan-Kolchin (2017) for a review.

While a proper treatment of the intrinsic evolution of the dwarf galaxies is mandatory, the possible impact of the environment of these systems ought to be understood as well. Observations have indeed highlighted a morphology-density relation in the Local Group (Einasto et al. 1974; McConnachie 2012). Gas-deficient galaxies are preferentially found close to either the Milky Way or M31, while gas-rich dwarfs are found at larger galacto-centric distances. This relation could result from the interaction between satellite systems and their massive host, through both tidal and ram pressure stripping. While tidal stripping is a pure gravitational process, ram pressure stripping is an hydrodynamical one, resulting from the interaction between the interstellar medium (ISM) of the dwarf and the hot virialized diffused gas of its host galaxy, that can reach temperature up to $\sim 10^6$ K, for a Milky Way analogue. The stripping of the dwarf galaxy results from a momentum exchange between the two gas components.

Ram pressure, with or without the help of tidal stripping has also been mentioned to possibly solve the missing satellites problem (Del Popolo & Le Delliou 2017; Arraki et al. 2014). Indeed, the quick removal of the ISM of the dwarf makes its luminosity drop down to the point of hampering its detection. The dynamics of the dwarf is also modified, impacting its mass distribution, eventually turning a cuspy profile into a cored one. While Mayer et al. (2006) and Simpson et al. (2018) found that ram pressure and tidal stripping are efficient at removing the gas of the dwarf galaxies and at quenching their star formation, others, such as Emerick et al. (2016) and Wright et al. (2019) found it far less so and sometimes even able to slightly enhance star formation. While most of those studies reproduce the relation between the dwarf neutral gas (HI) fraction and their distance to the host galaxy (Grcevich & Putman 2010), some are not run in a cosmological context and the treatment of the baryonic physics is generally incomplete. For example, hydrogen self-shielding against UV-ionizing photons, that let the gas efficiently cool below 10^4 K is missing. This hampers the capturing of the multiphase structure of the dense star forming gas.

The present work is based on the high resolution zoom-in cosmological simulations of Revaz & Jablonka (2018). A volume of $(3.4\,\mathrm{Mpc/h})^3$ has served the analysis of dwarf galaxies outside the influence of a massive Milky-Way like galaxy. It was shown that, when baryonic physics and UV-background is included, in vast majority, the observed variety of galaxy properties, star formation histories, metallicity distribution, stellar chemical abundance ratios, kinematics, and gas content, was reproduced in detail as a natural consequence of the ΛCDM hierarchical formation sequence. Some systems though could not be adequately reproduced, such as the Fornax dwarf spheroidal galaxy (dSph), which is dominated by an intermediate stellar population (Boer et al. 2012), or the Carina dSph (de Boer et al. 2014), which exhibits very distinct peaks of star formation. Others such Leo P or Leo T (McQuinn et al. 2015; Weisz et al. 2012) have more extended star formation histories than can be predicted as the result of their low halo mass and the impact of the UV-background heating.

The question of when and how the Milky-Way, or similar central host galaxy, can impact the evolution of its satellites is at the heart of this study. This can also shed light on the origin of the above mentioned Local Group dSphs, which stand as exceptions of a general framework. To this end, we extracted a series of models from Revaz & Jablonka (2018) and re-simulated them by taking into account a Milky Way-like environment. Two sets of simulations are presented in the following: a wind tunnel, which investigates the impact of the ram pressure alone and a moving box, which includes the tidal forces as well.

The structure of this paper is the following. In Section 2, we present our numerical tools, the code `GEAR`, the wind tunnel and the moving box techniques. In Section 3 we describe the initial conditions of our dwarf models as well as their orbits. The different Milky Way models are also presented. In Section 4 the sets of runs for our two different simulation techniques are detailed. Our results are presented in Section 5 and a discussion is proposed in Section 6, followed by a short conclusion in Section 7.

2. Numerical tools

Our simulations involve two galaxies: the satellite, a dwarf galaxy and its host, a Milky Way-like galaxy. The dwarf galaxy is self-consistently simulated as an N-body system using the code `GEAR`. To capture the ram pressure induced by the hot host halo, we used a wind tunnel method where gas particles are injected and interact with the dwarf galaxy. The effect of tidal forces is included by extending the wind tunnel simulation with a moving box technique. There, the gravity of the host galaxy is modelled by a potential that may evolve with time. Those different techniques are succinctly presented in this section.

2.1. *GEAR*

`GEAR` is a chemo-dynamical Tree/SPH code based on `GADGET-2` (Springel 2005). Its original version was described in Revaz & Jablonka (2012) with some improvements discussed in Revaz et al. (2016) and Revaz & Jablonka (2018). Gas radiative cooling and UV-background heating are computed through the `GRACKLE` library (Smith et al. 2017), using its equilibrium mode. In this mode, the cooling due to the primordial elements are precomputed following the assumption of ionization equilibrium under the presence of a photoionizing UV-background (Haardt & Madau 2012). Cooling from metals is included using a simple method where predictions for a solar-metalicity gas computed from the `CLOUDY` code (Ferland et al. 2017) are scaled according to the gas metallicity (see Smith et al. 2017, for the details of the method). The cooling due to the H_2 molecule is not included. Hydrogen self-shielding is included by suppressing the UV-background heating for densities above 0.007 cm^{-3} (Aubert & Teyssier 2010). A lower temperature limit of 10 [K] is imposed.

Star formation is performed using a modified version of the Jeans pressure (Hopkins et al. 2011) and an efficiency $c_\star = 0.01$. The chemical evolution scheme includes Type Ia and II supernova with yields from Kobayashi et al. (2000) and Tsujimoto et al. (1995) respectively. Exploding supernovae are computed stochastically using a random discrete IMF sampling (RIMFS) scheme (Revaz et al. 2016). An energy of 10^{50}erg is released per supernova into the ISM, following the thermal blastwave-like feedback scheme (Stinson et al. 2006). We used the smooth metalicity scheme (Okamoto et al. 2005; Tornatore et al. 2007; Wiersma et al. 2009) to further mix the polluted gas. Stellar V-band luminosities are computed using Vazdekis et al. (1996) relations and our initial mass function (IMF) is the revised IMF of Kroupa (2001). `GEAR` includes individual and adaptive time steps (Durier & Vecchia 2012) and the pressure-entropy SPH formulation (Hopkins 2013) which ensures the correct treatment of fluid mixing instabilities, essential in the RPS simulations.

In the present study, the physical models and its parameters are identical to the one used in Revaz & Jablonka (2018), where the properties of a few Local Group's dwarf galaxy such as NGC 6622, Andromeda II, Sculptor and Sextans have been reproduced in great details.

2.2. *Wind tunnel*

In order to study RP stripping, we supplement `GEAR` with a wind tunnel setup. A wind tunnel simulation consists in an object (an isolated galaxy in our case), placed in a box in which gas particles, called hereafter wind particles, are injected from one side (the front) and removed from the opposite one (the back). In-between wind particles may interact with the object and in particular with its gaseous component. In our implementation, the behaviour of particles at the box side, meaning, the six box faces different from the front and back ones differ according to their origin. If particles are gas from the wind, we apply periodic boundaries. On the contrary, if particles where gas, initially belonging to the satellite, they are removed. Finally, we remove all type of particles that cross the front side with negative velocities, that is moving against the wind.

The details of the parameters explored through those wind tunnel simulations will be presented in Section 3. While being the perfect tool to study RP and in particular the effect of a variation of the wind density, temperature and velocity, wind tunnels simulations do no include any tidal effect and its dependence along the satellite orbit.

2.3. Moving box

We complemented the wind tunnels simulations with moving box simulations. This simulation technique introduced by Nichols et al. (2015) allows to add the tidal stripping a satellite may suffer along its orbit, while ensuring simulations to run with the same very high resolution. Hereafter, we present a brief summary of this methods, including minor updates.

The moving box consists in a wind tunnel simulation supplemented with the gravitational forces between the host (a fixed potential) and a satellite moving along its orbit. Instead of launching a satellite in an orbit around a host potential, the satellite is placed inside a non inertial box corresponding to a frame in motion around the host potential. In addition to its motion along the orbit, we supplement the box with a rotation motion in order to keep the particles injection on the same front side. The latter is simulated by implementing fictitious forces induced by both the rotation and orbital motion of the box. This method is a CPU-economic way of simulating what a galaxy would experiment while orbiting around its host without the necessity to include the entire hot gas halo that would requires important memory and CPU resources.

Stars and dark matter are not sensitive to the hydrodynamical forces. However, they are indirectly affected by the RP through the gravitational restoring force the RP stripped gas will exerts on both of them (see the parachute effect described in Nichols et al. (2015)). This indirect interaction is responsible for a continuous drift of the satellite with respect to the box centre, which, in extreme case could make it leave the box. To avoid this, we apply an ad hoc correcting force which depends on the centre of the dwarf, defined as the centre of mass of the 64 star and dark matter particles of the dwarf having the lowest total specific energy. This definition is sensitively optimized compared to the one performed by Nichols et al. (2015), where only the potential energy was used, leading to the impossible differentiation between bounded particle and particles passing through at high velocity. Once the dwarf centre is defined, an harmonic force is apply to all particles, where the magnitude of the force scales with the distance between its centre and the centre of the box. The impact of this procedure on the satellite orbit is small. Only a slight reduction of the apocentre (about 15%) as well as of the velocity at pericentre (about 10%) after 10 Gyr is observed, with respect to the expected theoretical orbit where a satellite is considered as a point mass. One restriction of the method is the ill defined behaviour of the wind particles creation when the host centre lie inside the simulation box. Indeed, in the case where the host centre would enter the box, there is no way to clearly define a front face where we could inject the wind particles. Therefore we restrained the orbits to radius larger than the box size. The details of the orbits as well as the set of simulations performed are described in Section 3.

3. Models

3.1. Dwarf models

All our dwarf models have been extracted from the cosmological zoom-in simulations published in Revaz & Jablonka (2018). We refer to this paper regarding the name of dwarf models. 27 dwarfs have been simulated from $z_{\rm init} = 70$ until $z = 0$, assuming Planck Collaboration et al. (2015) cosmological parameters, with a gravitational softening of 10 and 50 pc/h for the gas and dark matter respectively and a mass resolution of $1'024$ $M_\odot/h$ for the stellar, $4'096$ $M_\odot$/h for the gas and $22'462$ $M_\odot$/h for the dark matter. Despite having still an important gas component at the injection redshift, none of the simulated dwarf show a disky structure. This is due to the lack of angular momentum accretion as well as the strong stellar feedback that continuously heats gas, maintaining it in a spherical structure around the dwarf.

In a first step, in order to test the ram pressure under a large number of parameters at low computational cost, we mainly focused on model `h159` in our wind tunnel simulations. This model is a quenched galaxy dominated by an old stellar population with a final V-band luminosity of $0.42 \cdot 10^6\,L_\odot$, a virial mass of $M_{200} = 5.41 \cdot 10^8\,M_\odot$ (See Table 1. of Revaz & Jablonka (2018)). Because of its low stellar mass and quenched star formation history this model is quickly simulated over one Hubble time. While results presented in Sec. 5.2 only rely on this galaxy, it is worth noting that similar results have been obtained with six more massive galaxies (see Table A.1).

In a second step, in our moving box simulations, seven galaxies have been selected according to their star formation history, spanning a total halo mass in the range $M_{200} = 5.4$ to 26.2×10^8 $M_\odot$ (see Table A.2). In Sec. 5.3, we focus on the two most representative cases, `h070` and `h159`. Model `h070` is brighter than model `h159` with an extended star formation history. It perfectly reproduces the observed properties of the Sculptor dSph.

Each selected dwarf model has been extracted from the cosmological simulation at $z_{\rm ext} = 2.4$ and converted from comoving coordinates to physical ones. The extraction radius is taken as the virial radius R_{200}, where R_{200} is the radius of a sphere that contains a mean mass density equal to 200 times the critical density of the Universe. For a dwarf spheroidal galaxy in a ΛCDM Universe, R_{200} is of the order of 30 kpc, much larger than the stellar component ($\sim$ 1 kpc). Using R_{200} has the advantage of being large enough to minimize perturbation due to the extraction and small enough to keep a reasonable box size. We tested our extraction method and how it can perturb the evolution of the dwarf by comparing the cumulative number of stars formed between the initial cosmological simulation and the extracted one at $z = 0$. The perturbation has been found to be negligible, of the order of a perturbation induced by changing the random number seed. Simulating the late stage of dwarf galaxies out of a full cosmological context is justified by their merger history (Revaz & Jablonka 2012; Fitts et al. 2018; Cloet-Osselaer et al. 2014) that finish early enough ($z \approx 5$ in our simulations) to be almost isolated for most of its life.

We chose the extraction redshift $z_{\rm ext}$ on the following basis. Due to the mergers at high redshift, $z_{\rm ext}$ must be low enough to avoid a perturbation from a major merger (mass ratio of 0.1 in Fitts et al. (2018)). It must be high enough to ensure the quenched dwarfs to be still star forming ($t \lesssim 2,$Gyr for the faintest models like `h159`) in order to study the MW perturbation on its star formation history. We therefore choose $z_{\rm ext} = 2.4$. This choice corresponds to a satellite infall time of about 9 Gyr ago, considered as an early infall time according to (Wetzel et al. 2015). A rather high fraction of present satellite galaxies, 15.8%, have approximately this first infall time (Simpson et al. 2018).

3.1.1. Milky Way models at $z = 0$

The Milky Way mass model at $z = 0$ is composed of two Plummer profiles representing a bulge and a disk, and an NFW profile representing its dark halo. The adopted parameters for these

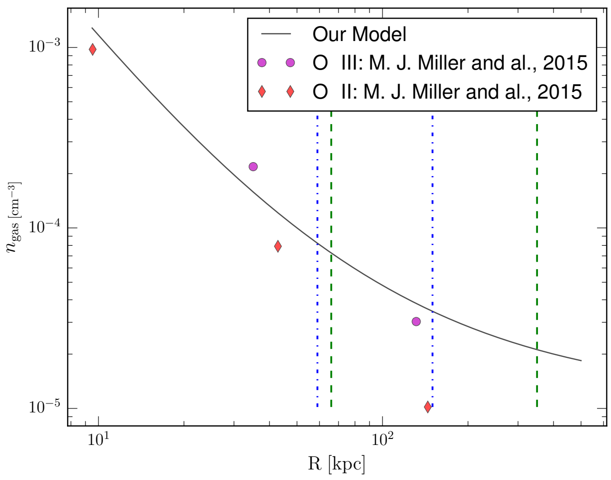


Fig. 1: Gas density model of the MilkyWay's hot halo (black line) compared to observational data from Miller & Bregman (2015). The two vertical lines correspond to the minimal pericentre and maximal apocentre of the satellite orbits explored in this work (blue static potential and green evolving potential).

three components are given in Tab. 1 and are similar to the ones used in Nichols et al. (2015).

The gas density of the hot halo is computed by assuming the hydrostatic equilibrium of an ideal isothermal gas of hydrogen and helium. Formally the total gas density profile $\rho(r)$ or equivalently the electron density profile n_{e} is obtained by solving :

$$\frac{n_{\mathrm{e}}(r)}{n_{\mathrm{e},0}} = \frac{\rho(r)}{\rho_0} = \exp\left(-\frac{\mu m_{\mathrm{p}}}{k_{\mathrm{B}} T}\left[\phi(r) - \phi_0\right]\right), \tag{1}$$

where, T is the constant gas temperature, ϕ the total potential, μ the mean molecular weight, m_{p} the proton mass and k_{B} the Boltzmann constant. $n_{\mathrm{e},0}$, ρ_0 and ϕ_0 are respectively the electron density, total gas density and potential at the centre of the galaxy. Following Nichols et al. (2015), we fixed $n_{\mathrm{e},0}$ to $2 \cdot 10^{-4}$ cm^{-3} at 50 kpc. The resulting density profile is displayed in Fig. 1 and compared to the data of Miller & Bregman (2015). The observed density and temperature intervals are $\rho \in [10^{-5}, 10^{-2}]$ atom/cm^3 and $T \in [1.5 \cdot 10^6, 3 \cdot 10^6]$ K at radii smaller than 100 kpc and are consistent with our MW model. At large radii our model slightly over-predicts the density. This is however unimportant as in any case, the ram pressure will not be negligible at those large radius compared to smaller ones. The two vertical lines shown on Fig. 1 indicate the minimal pericentre and maximal apocentre of the satellite orbits explored in this work and give and idea of the density studied in this work.

3.1.2. Time evolution of the models

All along a Hubble time, a Milky Way-like galaxy see its mass growing through a succession of merger and accretion events. This mass grows and subsequently the increase of its gas halo and in particular its temperature through thermalisation has potentially a strong impact on the ram pressure and tidal tripping of its dwarf satellites. For this purpose, we considered the mass evolution of the MW model by defining three different evolution modes (EM). In all of them, the MW ends up with the same properties at $z = 0$:

- Static (EM-{}): The MW does not evolve: Its potential remains fixed, equal to the one defined at $z = 0$. Similarly, the density and temperature of the halo gas stay constant.
- Dynamic with a constant temperature (EM-$\{\rho\}$): The MW potential evolves through an increase of its total mass and size, together with the density of the hot component. The temperature of the gas is however kept fixed.
- Dynamic with a dynamic temperature (EM-$\{\rho, T\}$): In addition to the second mode the gas temperature evolves too.

3.1.3. Mass and size evolution

Figure 2 displays the time evolution of the mass and size of our MW model used in the evolution mode EM-$\{\rho\}$ and EM-$\{\rho, T\}$. Those curves are computed from the model Louise of the ELVIS simulations (Garrison-Kimmel et al. 2014), where the mass growth of several simulated galaxies is studied. The mass and size of the Louise galaxy is scaled in order to match exactly our non-evolving Milky Way model at $z = 0$. As in Garrison-Kimmel et al. (2014) the Louise galaxy was fitted using an NFW profile, we use the scale radius ($R_s = R_{\mathrm{vir}}/c$) as a scaling for the Plummer softening parameter a.

3.1.4. Temperature evolution

In the evolution mode EM-$\{\rho, T\}$, in addition to the density, we evolve the temperature T as show in Fig. 2. T is computed assuming a virial equilibrium of the halo gas at any time, using the following equation:

$$T = \frac{2}{5}\frac{m_{\mathrm{p}} G M_{\mathrm{vir}}(t)}{R_{\mathrm{vir}}(t) k_{\mathrm{b}}}, \tag{2}$$

where $M_{\mathrm{vir}}(t)$ and $R_{\mathrm{vir}}(t)$ are respectively the time-evolving virial mass and radius, m_{p} the proton mass, G the gravitational constant, and k_{b} is the Boltzmann constant. The Plummer softening parameter a is chosen in order to match the scale radius $R_s = R_{\mathrm{vir}}/c$.

3.2. Satellite orbits

For the moving box simulations, we used only one generic orbit for the satellites galaxies. A deeper analysis of the influence of the orbital parameters on the dwarfs has been previously done with `GEAR` in Nichols et al. (2014).

According to recent proper motions and orbital parameters determination of dwarf galaxies based on the Gaia DR2 (Fritz et al. 2018), confirming earlier studies (Piatek et al. 2003, 2007), classical dwarfs such as Carina, Sextans and Sculptor have orbits with perigalacticon between 40 and 120 kpc and apogalacticon between 90 and 270 kpc (Fritz et al. 2018). It is worth noting that those measurements allow a fairly large interval of the orbital parameters. Therefore we decided to use a generic orbit with a pericentre of 60 kpc and an apocentre of 150 kpc, together with a current position of the dwarf at a distance of 85 kpc, with a negative velocity along the radial axis.

We emphasize here that wind tunnel simulations are very complementary to the moving box approach. Indeed they allow to explore a much larger parameter space of the hot gas temperature and density and infalling velocity of the satellites, than could be efficiently done with the moving boxes. In that respect, one does not need to sample a very large sets of orbits, as those would duplicate the parameters investigated by the wind tunnels.

Table 1: Milky Way model parameters used at $z = 0$ (Nichols et al. 2015). The analytic potential of each component is provided in the first column ($\phi(R)$) along with its parameters in the second column. The last column provides the corresponding references.

	$\phi(R)$	Parameters	Reference
Bulge	$-GM/\sqrt{R^2 + a^2}$	$M = 1.3 \cdot 10^{10} M_\odot$ $a = 0.5\text{kpc}$	Xue et al. (2008)
Disk	$-GM/\sqrt{R^2 + a^2}$	$M = 5.8 \cdot 10^{10} M_\odot$ $a = 5\text{kpc}$	Xue et al. (2008)
Halo	$-GM_{\text{vir}} \ln(1 + cR/R_{\text{vir}})/[\ln(1 + c) - c/(1 + c)]$	$M_{\text{vir}} = 8 \cdot 10^{11} M_\odot$ $c = 21$ $R_{\text{vir}} = 240\text{kpc}$	Kafle et al. (2014)

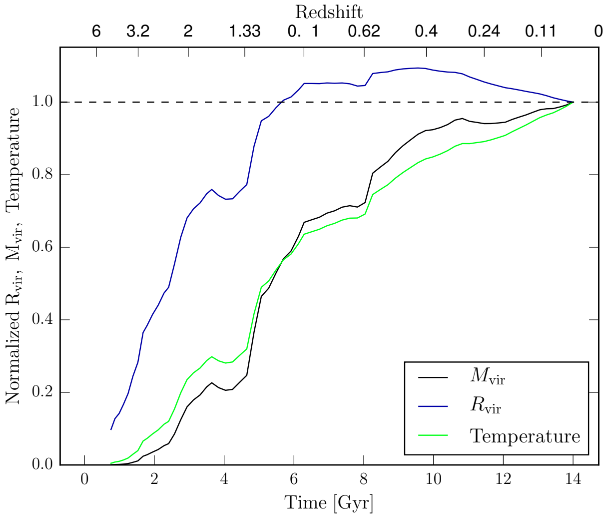


Fig. 2: Time evolution of the Milky Way parameters taken from the Louise galaxy ($M_{\text{vir}} = 10^{12}$ $\text{M}_\odot$ and $R_{\text{vir}} = 261.3$ kpc at $z = 0$) in the ELVIS(Garrison-Kimmel et al. 2014) simulations. The temperature is computed assuming a virial equilibrium at all time following equation 2.

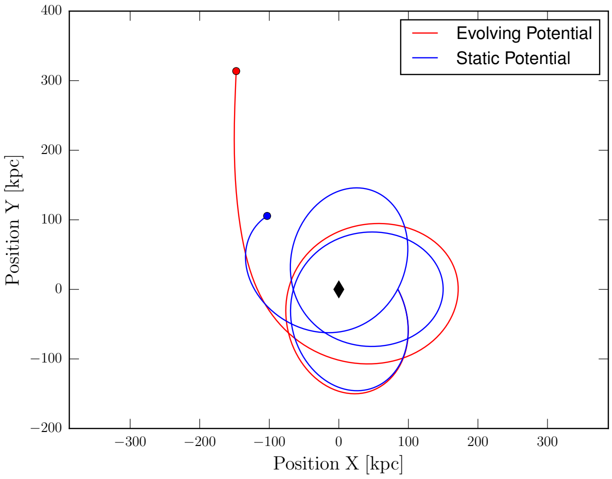


Fig. 3: Orbit used for the static (EM-{} mode) and evolving (EM-{ρ} and EM-{ρ, T} modes) MW potentials. The black diamond indicates the potential centre. The two points show the initial position at $z = z_{\text{ext}} = 2.4$. The final position is the same for both potentials and is situated at the coordinate $[0, 85]$ kpc.

We also recall that due to the constraints imposed by the moving box method (See Section 2.3), we are unable to use orbits with a pericentre smaller than 30 kpc, as the latter must be larger than half of our box size. To get the initial position of the extracted satellite at the infall time, $z = z_{\text{ext}}$, the orbit of a point mass is backward time-integrated in both the static (EM-{} mode) and evolving (EM-{ρ} and EM-{ρ, T} modes) MW potential, using a Runge-Kutta algorithm. The two orbits obtained are compared in Fig. 3. In the static case, the satellite will perform two and a half orbit around the MW, while only one and a half in the evolving case.

4. Simulations

4.1. Wind tunnel simulations

Those simulations explore the effect of the wind parameters on the evolution of the dwarf. Precisely, we explored its velocity relative to the dwarf v_{w}, its temperature T_{w} and density ρ_{w}. We preformed in total 96 simulations corresponding to each combination of the wind parameters as presented in Tab. 2. Each parameter is varied in a range of almost one dex around a fiducial value. They are set in order to match the observed MW constraints, either the gas density (Miller & Bregman 2015, 2013) or the satellites velocities constraints by their proper motions (Piatek et al. 2003, 2007).

The fiducial parameters of the wind are chosen to match our static Milky Way model at injection position. They are set to a density $\rho_{\text{w}} = 1.66 \cdot 10^{-5}$ atom/cm^3, a velocity $v_{\text{w}} = 100$ km/s and a temperature $T_{\text{w}} = 2 \cdot 10^6\text{K}$). The bottom line of Tab. 2 indicates the ratio of the parameters with respect to the fiducial ones.

As presented in Sec. 3.1, we exposed the dwarf model `h159` to the wind. This galaxy presents a rather shallow gravity potential, therefore the RP is efficient at stripping the gas and makes it sensitive to the wind parameters. Its initial cold ($T \leq 1000\,\text{K}$) and hot ($T < 1000\,\text{K}$) gas mass at infall time, $z = z_{\text{ext}}$ is respectively 4.58 and $27.0 \cdot 10^6$ $\text{M}_\odot$. 6 more massive galaxies have been also simulated (see Tab. A.1) confirming results obtained by model `h159`.

4.2. Moving box simulations

Those simulations explore the impact of the MW on the evolution of dwarf galaxies through a most complete interaction model which takes into account the orbits of the dwarf satellite through a time-variation of the wind parameters, but also the

Table 2: Wind parameters used in the wind tunnel simulations. The bottom line indicates the ratio of each parameter with respect to its corresponding fiducial one. The fiducial parameters are given in the fourth column.

v_w [km/s]	-	76.9	-	100	130	169
T_w [10^6K]	1.30	1.54	1.76	2.0	2.60	3.39
ρ_w [10^{-5} atom/cm^3]	-	1.28	-	1.66	2.16	2.81
ratio to the fiducial parameter	0.65	0.77	0.88	1.	1.3	1.69

Table 3: Description of the realistic simulations. The dwarf model names come from Revaz & Jablonka (2018) supplemented by the MW model as described in 3.1.1. If no MW model are given, it means that the simulation was done in isolation and therefore do not contain a host. `h159` displays a quenched star formation history while the one of `h070` is extended.

Name	Dwarf Model	MW Model
`h159_iso`	`h159`	-
`h159_sta`	`h159`	EM-{}
`h159_rho`	`h159`	EM-{ρ}
`h159_tem`	`h159`	EM-{ρ, T}
`h070_iso`	`h070`	-
`h070_sta`	`h070`	EM-{}
`h070_rho`	`h070`	EM-{ρ}
`h070_tem`	`h070`	EM-{ρ, T}

gravitational tidal effects together with the mass growth of the MW over a Hubble time.

We studied the evolution of 7 dwarfs, with total halo masses from $M_{200} = 5.4$ to $26.2 \cdot 10^8$ M$_\odot$. In the following, we will only focus on two representative models, the quenched model `h159` dominated by old stellar populations and the Sculptor-like model `h070` which has an extended star formation history. Other models, including more massive ones characterized by a sustained star formation rates give similar results. See Tab. A.2 for the list of additional models simulated. In a first step, each of these two dwarfs have been simulated in isolation. In a second step, they have been simulated in the three modes including the Milky Way interaction, EM-{}, EM-{ρ} and EM-{ρ, T}.

In Table 3, the parameters of each moving box and isolated fiducial simulations are given.

5. Results

5.1. Analysis

5.1.1. Pressure ratio

During the infall of a dwarf galaxy towards its host, the hot halo gas of the latter not only exerts a ram pressure against the ISM of the former, but also an almost uniform thermal pressure (TP) all around it. A key point to understand how the dwarf galaxy evolution is impacted upon infall, is to measure the individual effect of both the RP and TP, as they both have an opposite effect. While the RP removes the gas from the galaxy by momentum transfer, the TP tends to protect it by applying an additional force all around it, which prevents its removal due to RP, SNe feedback or UV-background heating resulting from the UV-photons emitted by active nuclei and star-forming galaxies. As presented by Sarazin (1986), the ram pressure is given by

$$P_{\mathrm{RP}} = \rho_{\mathrm{w}} v_{\mathrm{w}}^2,$$

where ρ_w is the wind density and v_w its velocity. The thermal pressure is given by the ideal gas law

$$P_{\mathrm{TP}} = n k_{\mathrm{B}} T_{\mathrm{w}},$$

where n is the particle number density, k_B the Boltzmann constant and T_w the wind temperature. Consequently, ratio of TP and RP which defines a unitless coefficient is written as

$$\beta_{\mathrm{RP}} = \frac{k_{\mathrm{B}}}{\mu m_{\mathrm{P}}} \frac{T_{\mathrm{w}}}{v_{\mathrm{w}}^2}, \tag{3}$$

where μ the mean molecular mass and m_P the proton mass.

We will see that this ratio will play a crucial role in the analysis and understanding of our simulations. It is worth noting that in Eq. (3), the density disappears and therefore the RP stripping is independent of it at first order.

5.1.2. Gas fraction computation

In order to estimate the effect of RP stripping, we compute the gas fraction of our dwarf galaxies with time. It is performed by computing the mass of the hot gas in a constant radius taken as the initial virial radius $R_{200}(z_{\mathrm{init}})$. As contrary to the hot gas, the cold gas is concentrated around the dwarf centre, we computed the cold gas mass in a radius R_{cg} equal to 10% of $R_{200}(z_{\mathrm{init}})$.

5.2. Wind tunnel simulations

Our wind tunnel simulations confirm the strong effect the hot halo gas has on the dwarf ISM through RP. However they also reveal the importance of the satellite's ISM multiphase structure. In a first step, we therefore split our analysis according to the gas temperature. In a second step, we will explore the effect on the star formation and study the impact of the wind parameters. A short summary will be given at the end of the section.

5.2.1. Stripping of the hot gas

Figure 4 shows the evolution of model `h159` exposed to a wind of temperature equal to $3.39 \cdot 10^6$ K, a density of $1.28 \cdot 10^{-5}$ atom/cm^3 and a velocity of 76.9 km/s. This time sequence shows four different important steps. The first frame shows the gas at $t = 2.1$ Gyr, before any hydrodynamic interaction between the wind and the dwarf. The second one shows the first contact, the third one shows the state of the dwarf about one Gyr after the first contact. The last one corresponds to the steady state reached after the RP stripping. As expected, soon after the first contact, the large hot halo gas of the dwarf is strongly distorted ($t = 2.6$ Gyr) and quickly stripped, forming a trailing tail beyond the dwarf ($t = 3.4$ Gyr). At later time, only a small hot halo gas remains around the dwarf. The latter was not initially part of the dwarf halo gas. It results from the permanent heating of the cold gas by both UV-background heating and supernovae feedback. The efficient stripping of the hot gas is confirmed by the left panel of Fig. 5 where the time evolution of the hot gas fraction is shown for all of our 96 wind tunnel simulations. The colour of each line corresponds to the parameter $\beta_{\rm RP}$, the ratio between the thermal and ram pressure (Eq. (3)). All simulations show a quick drop of their hot gas fraction, indicating the efficient stripping of the dwarf hot halo. This demonstrates that the ram pressure stripping is captured in our simulations. The left panel of Fig. 5 also reveals a weak dependency on $\beta_{\rm RP}$. Winds characterized by a smaller $\beta_{\rm RP}$ are more efficient to ram pressure strip the hot dwarf gas. Finally, we see that the isolated case traced by the green curve retains more hot gas after 4 Gyr as the latter do not suffer any ram pressure stripping. However, at later time the warm gas fraction decreases. This reveals the secular evaporation of the hot gas due to the continuous UV-background heating, until complete evaporation at $t \cong 9$ Gyr. The remaining of hot gas in the wind tunnel simulations after that time compared to the isolated model will be discussed below.

5.2.2. Stripping of the cold gas

Contrary to the hot dwarf gas, the cold one is much more difficult to strip. This is well observed on the last panel of Fig. 4 at $t = 5.1$ Gyr, where even 3 Gyr after the first contact, cold gas is still present in the dwarf. The right panel of Fig. 5 shows in more detail, the time-evolution of the cold gas fraction for all our wind tunnel simulations. We split our models in two categories according to their $\beta_{\rm RP}$ value: (i) thermal pressure-dominated models : $\beta_{\rm RP} \geq \beta_{\rm t}$, red colours, (ii) ram pressure-dominated models : $\beta_{\rm RP} < \beta_{\rm t}$, blue colours, where $\beta_{\rm t}$ is defined as the value at the transition and is about 3 for this galaxy.

Understanding the evolution in these different regimes first requires comprehension of the cold gas evolution in the isolated case. On the right panel of Fig. 5 the corresponding cold gas fraction is traced by the green curve. It is striking to see that the latter is dropping quickly, in less than 4 Gyr, faster than any other wind tunnel model. As for the hot gas, the origin of this drop is due to the UV-background ionizing photons which heat the gas. The potential well of this dwarf model being shallow, the latter evaporates (Efstathiou 1992; Quinn et al. 1996; Bullock et al. 2000; Noh & McQuinn 2014) resulting in the star formation quenching of the galaxy (Revaz & Jablonka 2018).

Thermal pressure-dominated models ($\beta_{\rm RP} > \beta_{\rm t}$)

When the thermal pressure dominates over the ram pressure, the high pressurized wind compress the cold gas, protect it against ram pressure and act against its UV-background heating driven evaporation observed in the isolated case. This protection leads to keep up to 50% of cold gas, even after a Hubble time. The regular decrease of the mass fraction observed in this regime results from the conversion of the cold gas in to stars resulting from a continuous star formation rate. This point will be discussed further below.

Low wind velocity models show an important drop of the cold gas fraction followed by a strong rise between 2 and 4 Gyr. The drop results from some gas particles being pushed by the wind, leaving the cut off radius, where the cold gas is measured. However, those particles do not acquire enough kinetic energy to leave the galaxy and are thus slowly re-accreted by gravity, explaining the subsequent increase of the mass fraction.

The oscillations observed in nearly all models result from the continuously pulsation of the ISM induced by the numerous supernovae explosion which cause the gas to be ejected outwards R_{cg} (the radius used to compute the cold gas) before being slowly re-accreated.

Ram pressure-dominated models ($\beta_{\rm RP} < \beta_{\rm t}$)

When the ram pressure dominates over the thermal pressure, the pressure protection is much weaker and the cold gas evolution becomes similar to the one of the isolated case. While just below the transition $\beta_{\rm t}$, cold gas may still survive up to $z = 0$, for very low $\beta_{\rm RP}$, it is lost. Those cases correspond to a fast moving dwarf with a speed larger than 150 km/s entering the halo of its host with a temperature of at most 1.3×10^6 K. However, in any case when the ram pressure is present, the cold gas fraction remains larger than the isolated case. This indicates that the UV-background heating always dominates over the RP. The final loss is due to a supernovae that ejects almost all the gas further than the stripping radius which is then removed from the galaxy as a single cloud.

5.2.3. Impact on star formation

Together with an important change of the cold gas mass fraction with respect to the isolated model, our wind tunnel simulations strongly impact the star formation rate and subsequently the amount of stars formed. Fig. 6 displays the cumulative number of stars formed with time. Compared to the star formation history, this plot has the advantage of being much less noisy.

All wind tunnel models form stars more efficiently compared to the isolated case as a consequence of the remaining large reservoir of cold gas. For the extreme thermal pressure-dominated models, the final stellar mass is up to four times larger than the isolated galaxy model while the ram pressure-dominated models with very low $\beta_{\rm RP}$ remains similar. It is worth nothing that pressure-dominated models with very low $\beta_{\rm RP}$, the ones that lost all their cold gas before $z = 0$, still display truncated star formation histories, however, much more extended than the isolated case, up to 9 Gyr in the most extreme case

For the sake of clarity, we note a small difference between all models, in the amount of stars formed before the injection time at $z = z_{\rm ext}$, indicated by a vertical dashed line. Indeed, in order to computed the evolution of the stellar mass, we extracted all stellar particles in the dwarf at $z = 0$ and used their age to deduce the stellar mass present at any comic time. Consequently, star particles formed in the dwarf but leaving the galaxy at later time are no longer accounted for, which may induce a small bias and the scatter observed between the different models. This approach, contrary to others where the stellar mass is computed at any time during the evolution, is much more representative to what an observer would have obtained relying on stellar ages deduced from a colour-magnitude diagram at present time.

5.2.4. Effect of the wind parameters

Figure 7 shows the final cold gas fraction of our 96 wind tunnel simulations, as a function of the wind parameters, more precisely, its velocity ($v_{\rm w}$), density ($\rho_{\rm w}$) and temperature ($T_{\rm w}$). According to Eq. (3), for a fixed temperature, the parameter $\beta_{\rm RP}$ only depends on $v_{\rm w}$, to the inverse of its square. We thus supplement the velocity-axis (y-axis) with its corresponding $\beta_{\rm RP}$-value on the right of each plot.

As expected in the theoretical formulas, $v_{\rm w}$ and $T_{\rm w}$ are the two most sensitive parameters. For any temperature bin, increasing $v_{\rm w}$ from 80 to 160 km/s move from a regime where the gas is

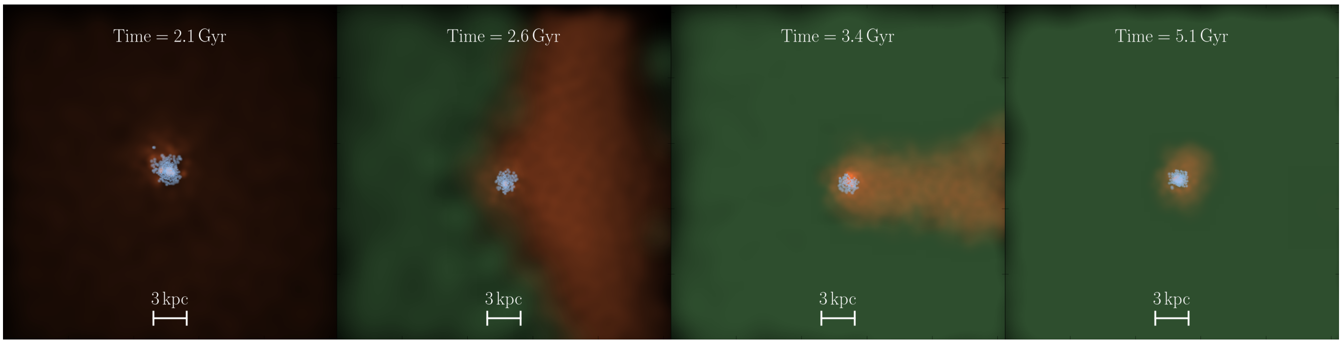
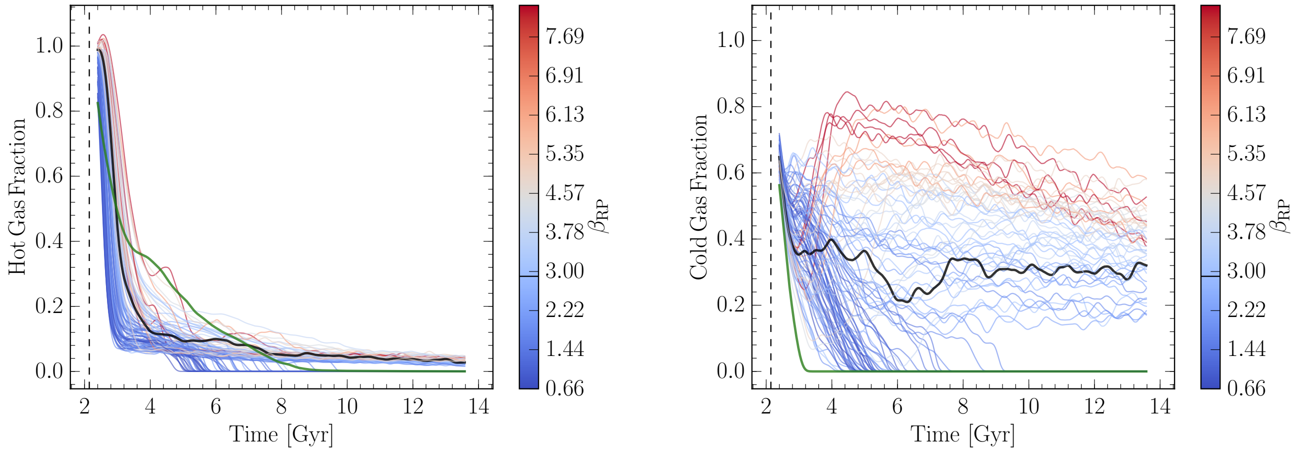


Fig. 4: Evolution of the cold, hot and wind gas during the first contact between the dwarf galaxy and the hot halo in a wind tunnel simulation with a wind temperature of 3.39 K, a density of $1.28 \cdot 10^{-5}$ atom/cm^3 and a velocity of 76.9 km/s. The hot gas of the dwarf ($T_{\mathrm{w}} > 10^3$ K) is shown in red, its cold gas ($T_{\mathrm{w}} < 10^3$ K) in blue. The green colours trace the gas of the wind. This sequence shows how the hot dwarf gas is quickly stripped while its cold gas remains.



(a) hot gas fraction

(b) cold gas fraction

Fig. 5: Time evolution of the gas fraction of the dwarf galaxy `h159` evolving through a wind tunnel simulation (see Table 2 for the list of parameters). left panel : the hot gas fraction contained in one virial radius. right panel : the cold gas fraction contained in 0.1 virial radius. The colour of each line reflect the corresponding β_{t}. In both panels, the green and black lines correspond to the isolated and fiducial wind tunnel model respectively. A moving average has been applied with a gaussian kernel (standard deviation of $\sim$ 100 Myr in a window of -500 to 500 Myr) to reduce the noise. Due to this filter, the earliest times are removed and the different curves start at different fraction.

protected, ending with an important cold gas mass fraction (between 0.3 to 0.5%) to a regime where all the gas is evaporated and the galaxy is quenched. Similarly, increasing T_{w} increases the thermal pressure which protect the dwarf gas. In strongly thermal pressure-dominated regimes ($\beta_{\mathrm{RP}} > \beta_{\mathrm{t}}$), a dwarf galaxy is thus able to protect its gas reservoir from stripping, up to 50%. On the contrary, from Eq. (3), the wind density has a limited impact on the galaxy gas fraction. However, it has a threshold effect. Indeed, for temperature between 1.3 to 1.76×10^6 K, below a density of about 1.2 to 1.4×10^{-5} atom/cm^3, the ram pressure stripping is enhanced, for a fixed v_{w} and T_{w}. Extrapolating Fig. 7 to lower temperature, we can predict that a quenched dwarf, like our `h159` model, orbiting in a halo with a temperature $T_{\mathrm{w}} < 1.3 \cdot 10^6$ K and a density $\rho_{\mathrm{w}} < 2 \cdot 10^{-5}$ atom/cm^3 will loose all its gas.

5.3. *Moving box simulations*

In this section, we go one step further by supplementing our wind tunnel simulations with tidal stripping induced by a realistic Milky Way model environment. We also study the time-variation of the wind parameters all along the dwarf orbit which reflects the inhomogeneous hot halo of the Milky Way but also its growth with time. A summary of the final properties of the six simulations performed are given in Tab. 4.

5.3.1. Effect of the Milky Way model

Figure 8 displays the star formation rate and time evolution of the cumulative stellar mass for each of the four cases studied for the two dwarf models `h159` and `h070`, namely, isolated, EM-{}, EM-{ρ} and EM-{ρ, T}. For the three last cases that include the ram pressure stripping, the bottom panel of Fig. 9 shows the cor-

Table 4: Properties at $z = 0$ of the dwarf models evolved in the moving box simulations. The model names fit the one of the corresponding dwarf in Revaz & Jablonka (2018). R_{200} and M_{200} corresponds to the virial radius and mass respectively. L_V is the final V-band luminosity. The cold gas is defined as the gas with a temperature lower than 1000 K while the hot one with a temperature above.

Name	R_{200} [kpc]	M_{200} [10^8 M$_\odot$]	$M_\star$ [10^6 M$_\odot$]	L_V [10^6 L$_\odot$]	Cold Gas [10^6 M$_\odot$]	Hot Gas [10^6 M$_\odot$]
`h159_iso`	19.4	5.37	1.08	0.43	0	9.05
`h159_sta`	11.3	1.06	2.47	1.04	0	0.
`h159_rho`	15.2	2.56	2.61	1.78	1.34	1.00
`h159_tem`	14.1	2.05	1.00	0.40	0	0.
`h070_iso`	26.3	13.3	5.72	2.04	0	20.6
`h070_sta`	14.5	2.22	18.5	10.9	6.32	4.61
`h070_rho`	20.3	6.09	23.8	14.9	6.23	6.93
`h070_tem`	17.3	3.76	5.42	1.92	0	0.

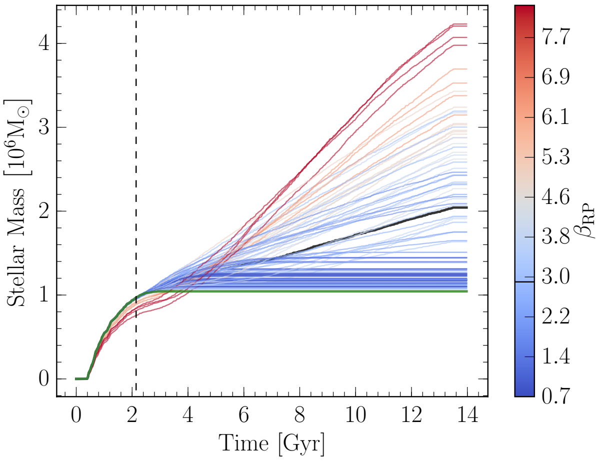


Fig. 6: Stellar mass as a function of time for the wind tunnel simulations. The colour is defined by the coefficient β_{RP} in equation 3. The black line corresponds to our fiducial wind parameters. The green line corresponds to the isolated case. The blue dashed line represents the injection time.

responding evolution of the coefficient β_{RP}, while the top panel shows the distance of the dwarf with respect to its host galaxy (top panel).

As discussed in Revaz & Jablonka (2018), when evolved in isolation, both models exhibit a star formation quenched after respectively $\sim$ 3 and $\sim$ 6 Gyr (black curves). However, when the dwarfs enter a static Milky Way halo (EM-{}, blue curve) at $z = z_{\rm ext}$, the star formation is no longer quenched but becomes continuous. As a consequence, the resulting final stellar mass is up to four times the one of the isolated case. This increase of the star formation is due to the high β_{RP} which pressurize the gas of the dwarf. As seen in Fig. 9, β_{RP} oscillates between 3 and 0.5 reflecting the dwarf orbit. Maximal values of 3, similar to our fiducial wind tunnel simulation, are reached during the apocentre passage, when the dwarf has the lowest velocity. On the contrary, at the pericentre passage, at a distance of 50 kpc, higher velocities increase the ram pressure with respect to the thermal one and β_{RP} drop down to 0.5. It is important to notice that even after four passages at the pericentre, the tidal force has not being strong enough to destroy the dwarf. This point is illustrated by the dark matter and stellar density profiles further discussed in Fig. 10.

The green curve (EM-{ρ}) corresponds to the case where the Milky Way increases its mass and density but keep a constant hot gas temperature. The Milky Way mass growth directly impacts on the dwarf orbit which experiments only three passages at the pericentre. It also impact on the β_{RP} parameter which starts with slightly higher values reflecting an initial larger distance ($\sim$ 350 kpc) and lower orbital velocity. As the density is initially much lower, a factor of about 30 compared to EM-{}, the dwarf cold gas is slightly less confined by the hot Milky Way halo (density threshold effect as shown in Fig. 7) and can evaporates. For model `h159`, this leads to the decrease of the averaged star formation rate with respect to the static model (EM-{}). This effect is however not seen in model `h070` for which the star formation rate of the EM-{ρ} model exceeds the one with model EM-{}. While a deeper analysis would be needed here, we interpret this difference by the deeper potential well of model `h070` compared to model `h159` at $z = z_{\rm ext}$. In this case, the gravitational confinement of the gas dominates over the pressure one. Finally, when the hot gas temperature scales with respect to the gas density (EM-{ρ, T}, red curves), at the infall time, the thermal pressure of the hot gas is no longer present to confine the dwarf gas, as shown by its very low β_{RP} in Fig. 9. Despite its increase at later time ($t > 6$ Gyr), the ram pressure stripping no longer impact the dwarf, as its cold gas already evaporated. In this model, both dwarf exhibit a star formation rate comparable to the isolated case sharing the same final stellar mass. As discussed in Sec. 5.2.3, the small differences at time $t < 2$ Gyr is due to the method used to compute the stellar content of the dwarf.

5.3.2. Impact on the final dwarf properties

Figure 10 and 11 present the final properties of the three interacting models of the dwarf `h159` and `h070` with the Milky Way, compared to their reference model in isolation.

In those two figures, the first row displays the stellar density profile (dashed line) along with the total density profile including the dark halo (continuous line). While none of the the interacting models are destroyed, they all show clear sign of stripping at radius larger than about 1 kpc, where the total density profiles drop compared to the isolated case. With four passages at the pericentre, `h159_sta` and `h070_sta` (EM-{}) are the most affected ones. They also see their total density profiles reduced up to 30% in the inner regions. However, due to their extended star formation rates, both models `_sta` and `_rho` exhibit a denser stellar density profile. On the contrary, with its quenched star

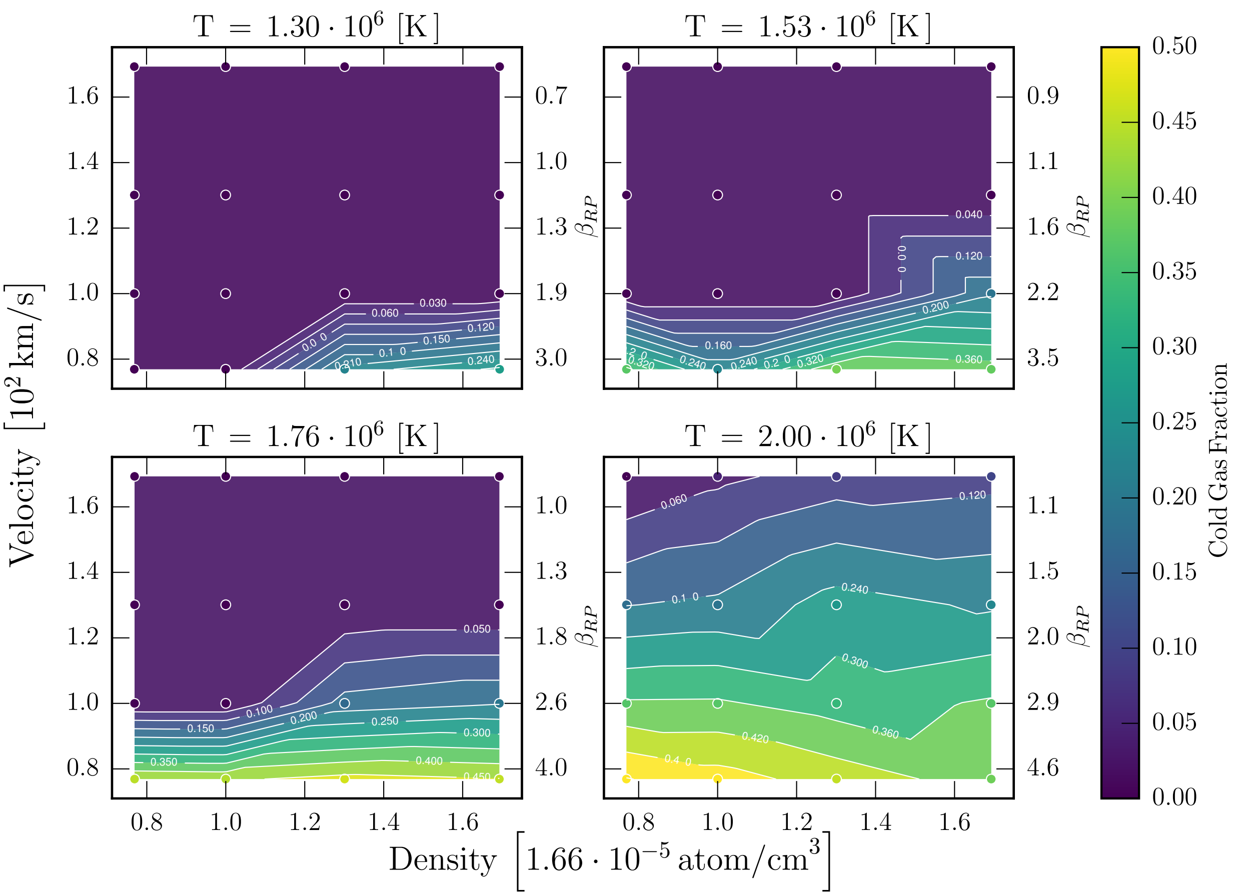
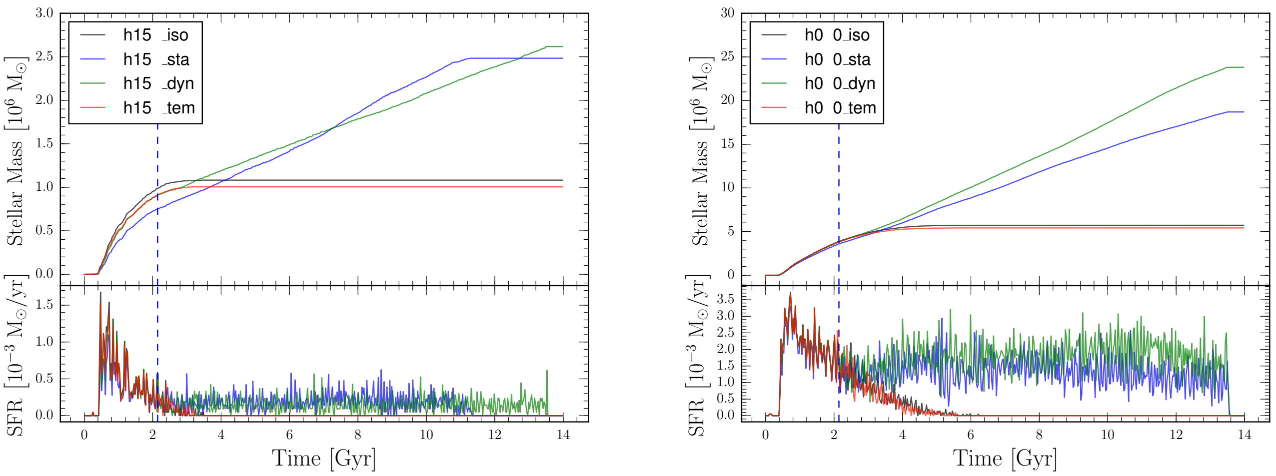


Fig. 7: Each plot represents the cold gas fraction at $z = 0$ of `h159` as a function of the hot halo density and satellite velocity for different halo temperatures. The white circles are the simulations done. β_{RP} is given on the right axis of each graph. Usually, the RPS is described only through density and velocity, but here the temperature dependency is shown to have an important impact.



(a) `h159`

(b) `h070`

Fig. 8: Time evolution of the cumulative stellar mass (top) and star formation rate (bottom) of model `h159` (left) and `h070` (right), in the four models: isolated (black), EM-$\{\}$ (blue), EM-$\{\rho\}$ (green) and EM-$\{\rho, T\}$ (red). The vertical dashed line indicates the injection time for models EM-$\{\}$, EM-$\{\rho\}$ and EM-$\{\rho, T\}$.

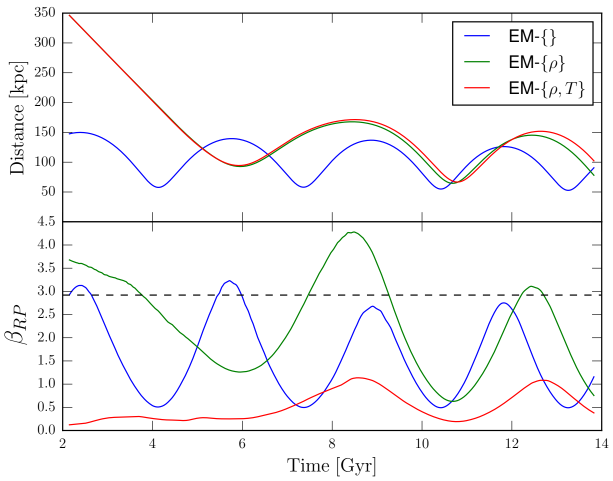


Fig. 9: top panel: Time evolution of the distance of the dwarfs for both the models `h159` and `h070` with respect to their host galaxy centre. bottom panel: The corresponding evolution of the β_{RP} parameter all along the dwarf orbit. The blue, green and red curves correspond respectively to the EM-{}, EM-{ρ} and EM-{ρ, T}. The black line represents our fiducial wind tunnel simulation (black line in Figure 5).

formation history, the stellar profile of `h159_tem` is similar to the isolated case.

The tidal stripping also impacts the stellar line of sight velocity dispersion profile showed in the second row. In five of the six interacting models, the velocity dispersion is lower than in the isolated case, up to 5 km/s for the `h159_sta` model. This decrease reflects the adiabatic decompression after the removal of the outer dark halo, also responsible of the reduction of the circular velocity. It is worth noting that this stripping could help reproduce the low velocity dispersion (down to 5 km/s) observed in six Andromeda galaxies and difficult to reproduce in isolated models (Revaz & Jablonka 2018). Only model `h070_dyn` sees its velocity dispersion and circular velocity increase in the central regions. This reflects its larger stellar content owing to its higher star formation rate.

The third and fourth rows of Fig. 10 and 11 compare the final chemical properties of the simulated dwarfs. The third row displays the abundance ratio of α-elements, traced here by the magnesium as a function of [Fe/H]. Because the dwarf galaxies enter their host halo at $t \cong 2$ Gyr, the old metal poor stellar population ([Fe/H] $\lesssim -1.5$) is not affected by the interaction. The stellar [Mg/Fe] vs [Fe/H] distribution is characterized by a plateau at very low metallicity ([Fe/H] $\lesssim -2.5$) followed by a decrease of [Mg/Fe], corresponding to the period where SNeIa yields dominates overs the SNeII, due to the drop of the star formation rate. In both the `_sta` and `_rho` models, the interaction with the hot gas halo leads to the extension of the star formation period. Therefore a new set of SNeII explode at a continuous rate and produce a constant injection of α-elements, quickly locked into new formed stars. This results into the formation of a plateau in [Mg/Fe] extending from [Fe/H] $\cong -1.5$ to [Fe/H] $\cong -0.5$ for model `h159` and [Fe/H] $\cong -1.4$ to [Fe/H] $\cong -0.2$ for model `h070`. The large amount of stars formed at those metallicities are responsible of a peak in the metallicity distribution function shown in the fourth row. This peak is strongly shifted towards higher metallicities compared to the isolated case. While [Mg/Fe] plateau have been observed for metal rich ([Fe/H] $\gtrsim$ -0.6) stellar population in Sagitarius (Hasselquist et al. 2017; Carlin et al. 2018), Fornax and LMC (Van der Swaelmen et al. 2013), and at a lower lever for Sculptor (see Fig. 11 of Tolstoy et al. 2009), they are found at solar or sub-solar [Mg/Fe], much lower than the one obtained here. A similar plateau may be obtained, to a somewhat shorter extension, for the brightest dwarf models of (Revaz & Jablonka 2018). While a dedicated study will be necessary, we claim that such plateau could also be obtained if `h070` would have entered its host halo at about 4−5 Gyr, the time needed to decrease [Mg/Fe] down to solar values, as shown in Fig. 11.

Finally, as their star formation history are similar to the isolated case, model `h159_tem` and `h070_tem` (EM-{ρ, T}) do not display any significant difference in their final chemical properties.

6. Discussion

Contrary to the widespread idea that local group dwarf spheroidal galaxies are easily quenched and devoid of gas due to the ram pressure stripping induced by its hot host halo, our simulations reveal a more complex picture. Both our wind tunnel and moving box simulations show that, while the hot gas of the dwarf is quickly ram pressure stripped, its cold and clumpy gas is not. On the contrary, due to the confinement of this gas by the thermal pressure of the hot halo gas which hamper the evaporation of the dwarf gas, the mass fraction of this cold phase can stay much above the one observed in the isolated case. Consequently depending on the orbital parameters of the dwarf, its infall time and the temperature of the host galaxy hot halo, the star formation of the dwarf may be extended over several Gyr or even heavily sustained up to the present time.

6.1. Comparison with other simulations

Numerous publications have been dedicated to the study of ram pressure stripping of galaxies, including our own.

Some of them concluded to the efficient stripping of gas implying the truncation or dampening of star formation (Mayer et al. 2006; Yozin & Bekki 2015; Fillingham et al. 2016; Emerick et al. 2016; Steinhauser et al. 2016). On the contrary, others concluded to the enhancement or reignition of the star formation (Bekki & Couch 2003; Kronberger et al. 2008; Kapferer et al. 2009; Nichols et al. 2015; Salem et al. 2015; Henderson & Bekki 2016; Wright et al. 2019). While a bunch of studies concluded that ram pressure may lead to both effects(Bahe et al. 2012; Bekki 2014) or no major effect (Williamson & Martel 2018). Those differences suggest that conclusions reached could strongly depend on the numerical methods used as well as the way the baryonic physics is implemented. Indeed, in those studies, a variety of hydrodynamical methods have been used. Those simulations relie on lagrangian SPH methods, with or without modern pressure-entropy formulation, eulerian methods with or without adaptive mesh refinement, or hybrid ones like moving-mesh methods. They differ by specific implementations of the ISM treatment, like radiative gas cooling below 10^4 k, external UV-background heating, hydrogen self-shielding against the UV-ionizing photons or magnetic field. Finally, they covers a large resolution range.

We discuss hereafter differences in our approach compared to other works that may lead to discrepancies, but also review

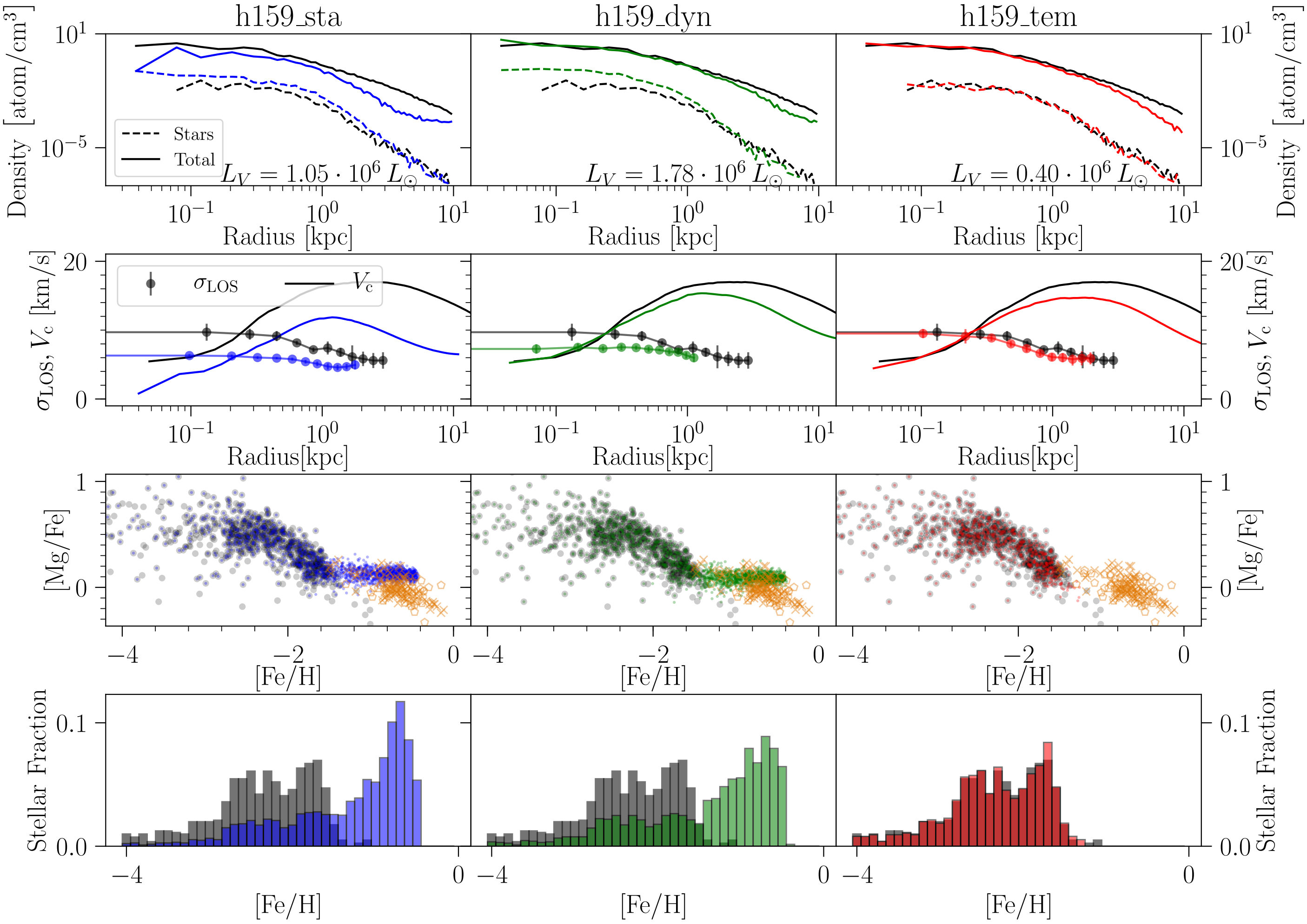


Fig. 10: Properties of the different simulations in a moving box. From left to right, the simulations are in blue `h159_sta`, `h159_dyn` and `h159_tem` and in red `h159_iso`. In the first line, the density profile is shown for the total mass (straight lines) and the stellar mass (dashed lines). In the second line, the circular velocity and line of sight velocity dispersion are shown. In the third line, the stellar [Mg/Fe] vs [Fe/H] distribution is shown. The orange crosses (pentagons) are observations of the LMC bar (inner disc) (Van der Swaelmen et al. 2013). In the last line, the metallicity distribution is shown.

works in different contexts that support our conclusions. Finally, will discuss our results in an observational context.

6.1.1. Stripping in dwarf galaxies

In a seminal paper, Mayer et al. (2006) showed that ram pressure stripping was efficient at completely removing the dwarf satellite ISM as long as they have a sufficiently low pericentre. However in their approach, the gas is not allowed to radiatively cool below 10^4 K. Under those conditions, the gas stays in a warm-hot and diffuse phase which is indeed easy to strip, as we demonstrated in Section 5.2.1. When the gas is allowed to cool down to lower temperature, it becomes clumpy (see Fig. 4 of Revaz & Jablonka (2018)) and exposes a smaller surface to the wind hampering an efficient momentum transfer between the wind and the cold gas. Efficient satellite ram pressure stripping have also been recently mentioned by Simpson et al. (2018), where the quenching of satellite star formation in 30 cosmological zoom simulations of Milky Way-like galaxies have been studied. In these simulations, up to 90% of satellites with stellar mass equal to about 10^6 $M_\odot$ are quenched, with ram pressure stripping being identified to be the dominant acting mechanism. This is nicely illustrated in their Fig. 8. However, those simulations also reveal a lack of any cold and clumpy phase which would be difficult to strip. In addition to a slightly lower resolution compared to ours, the absence of cold phase is the result of the stiff equation of state used, that represents a two-phase medium in pressure equilibrium (Springel & Hernquist 2003). However, it prevents the gas to cool down to low temperatures.

Recently, in high resolution simulation, Emerick et al. (2016) studied the ram pressure stripping of Leo T-like galaxies, including the effect of supernovae, and the presence of cold gas. However, they do not include a fully self-consistent star formation method, supernova being exploded at a location determined by a randomly sampled exponentially decreasing probability distribution centred on the galaxy. While concluding that the RP is unable to completely quench these type of galaxies in less than 2 Gyr, they show a clear decrease of the cold gas, contradicting our results. While being cooler and denser that the gas considered in Mayer et al. (2006) due to a temperature floor of $6 \cdot 10^3$ K,

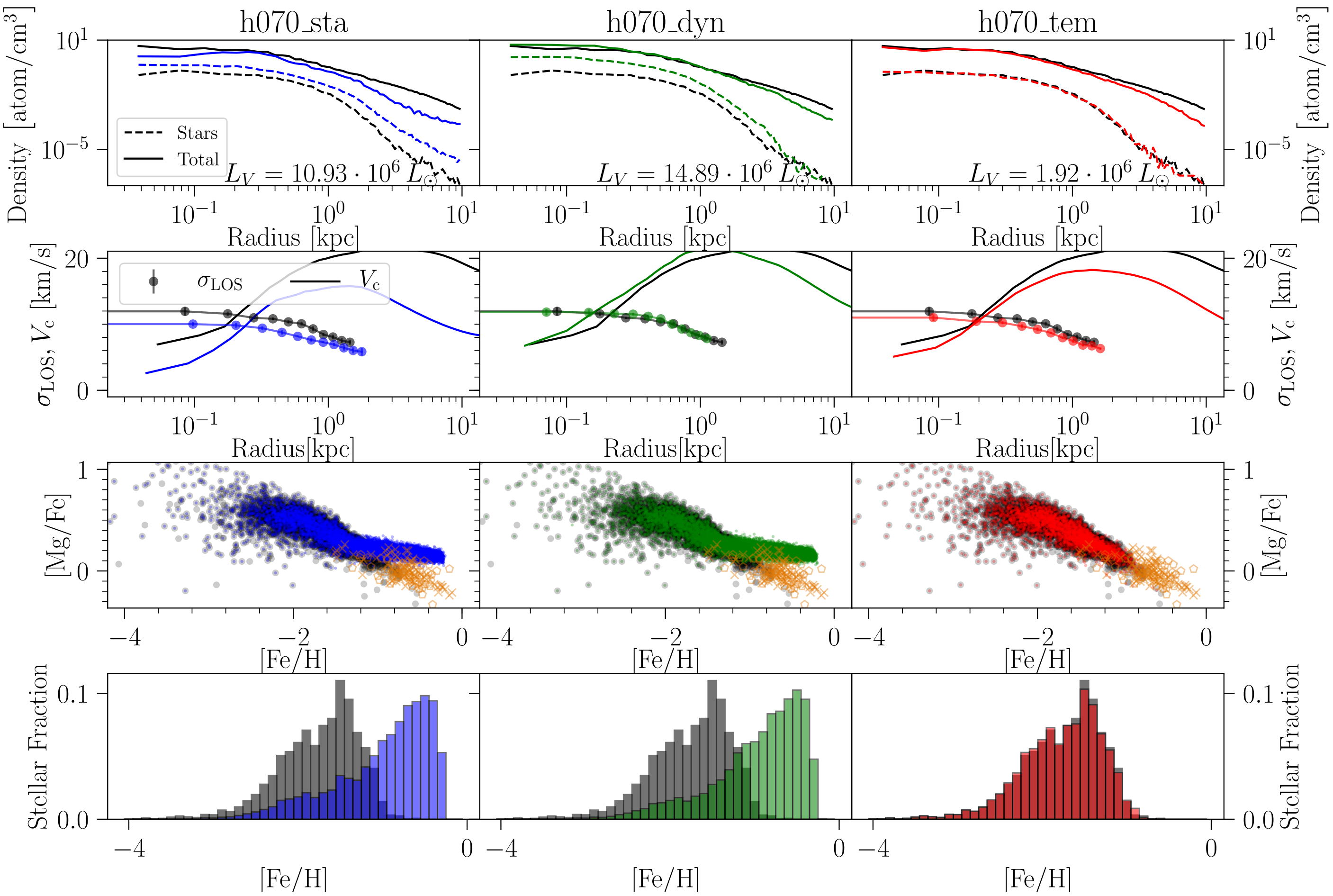


Fig. 11: Properties of the different simulations in a moving box. From left to right, the simulations correspond to `h070_sta`(blue), `h070_dyn`(green) and `h070_tem`(red) and are compared to the isolated model `h070_iso` in grey. In the first line, the density profile is shown for the total mass (straight lines) and the stellar mass (dashed lines). In the second line, the circular velocity and line of sight velocity dispersion are shown. In the third line, the stellar [Mg/Fe] vs [Fe/H] distribution is shown. In the last line, the metallicity distribution is shown.

as illustrated by their Fig. 3, it is nevertheless not as clumpy as the one considered in our work.

6.1.2. Thermal pressure confinement

One of the key effect that prevent the cold gas to evaporate and help sustain the star formation in our simulations is the thermal pressure confinement of hot ambient gas. We show hereafter that this effect is not only specific to our simulations but has been observed in other contexts.

Relying on SPH N-body simulations, Bekki & Couch (2003) studied the hydrodynamical effects of the hot ICM on a self-gravitating molecular gas in a spiral galaxy. They concluded that the high pressure of the Intra Cluster Medium (ICM) can trigger the collapse of molecular clouds leading to a burst of star formation. Along the same line, Kronberger et al. (2008) mentioned that in their models, the star formation rate is significantly enhanced by the ram-pressure effect (up to a factor of 3) when a disk galaxy move through an idealized ICM. Similarly, ram pressure can favour H_2 formation (Henderson & Bekki 2016), indirectly boosting the formation of stars.

Mulchaey & Jeltema (2010) observed the hot halo surrounding galaxies and found a deficit of X-ray in comparison to the K-band luminosity for field galaxies, when compared to the galaxies in groups or cluster. They interpreted this results as the possibility that, contrary to field galaxies than can loose gas by supernova-driven winds, galaxies in groups or clusters see this outflowing material being pressure confined, preventing it to leave the galaxy halo.

In a more quantitative way, using the GIMIC simulations, Bahe et al. (2012) explored the pressure confinement by studying its effect on normal galaxies falling in groups or cluster, directly computing the coefficient $\beta_{\rm RP}$. In their simulations, they found 16% of their galaxies to be dominated by thermal pressure.

Sign of star formation increase due to confinement pressure have been also mentioned for simulation at a dwarf scale. This effect has been described by Nichols et al. (2015) in their dwarf spheroidal simulations with however a star formation boost lower than the ones obtain in the present paper. While this study also relied on the moving box technique, the physical prescriptions used where not comparable to the one used in the present study. The simulations where run out of any cosmological context and neither UV-background nor hydrogen self-shielding where considered.

Williamson & Martel (2018) used a technique similar to our moving box and observed a thermal confinement. While the ram pressure has a negligible impact on the star formation, they showed that the outflows are confined and slightly increase the metallicity of the dwarf.

Wright et al. (2019) observed that in their cosmological simulations, dwarf galaxies can re-ignite star formation, following the complete quenching of the galaxy due to UV-background heating. This re-ignition results from the compression of remaining hot gas in the dwarf halo, following an interaction with streams of gas in the Inter Galactic Medium (IGM). Those gas streams being either due to cosmic filaments or resulting from nearby galaxy mergers. This mechanism is particularly efficient when the ram pressure is low compared to the thermal pressure (high β_{RP}), which corroborates with our own results.

6.2. Comparison with observations

From the observational point of view, the idea that satellites galaxies have been ram pressured stripped is mainly supported by the morphology-density relation observed in the Local Group (Einasto et al. 1974; van den Bergh 1994; Grcevich & Putman 2010). Quenched gas-poor spheroidals galaxies are found in the vicinity of their host galaxy ($R \lesssim 300\,\mathrm{kpc}$) while star forming gas-rich dwarf irregulars are found at larger distances. At the exception of Leo I, Fornax and Carina that show a very recent quenching time(see for example Skillman et al. 2017) the majority of dSphs have been quenched at least 5 Gyr ago.

We point out that recent observational facts suggest that the morphology-density relation may not be universal. Indeed, spectroscopic observations of satellites galaxies around the NGC 4258 group showed that the majority of the 16 detected probable and possible satellites, lying within a 250 kpc, with a V-band magnitude down to -12, appears to be blue star-forming irregular galaxies in the SDSS image (Spencer et al. 2014). This is in strong contrast with the observations of the Local Group.

More recently, the SAGA survey (Geha et al. 2017) observed satellites companions around eight Milky Ways analogues, with luminosities down to the one of Leo I ($M_{\mathrm{r}} < -12.3$), equivalent to about $M_{\star} = 10^{6}$ M$_{\odot}$ for star forming galaxies and $M_{\star} = 10^{7}$ M$_{\odot}$ for quenched galaxies. They found that among the 27 dwarf detected, the majority, 26 galaxies are star forming. This results points towards a less efficient quenching in those galaxies, compared the Milky Way.

The star formation rate of our models is strongly dependent on the infall time of the satellite relatively to the time when the galaxy halo is sufficiently hot and dense. As shown in Sect. 5.3.1 when the secular increase of the density and its temperature are taken into account, the evolution of the dwarfs entering the halo before a redshift of 2.4 are hardly different from those of their isolated counterparts. In that case the thermal pressure is unable to confine the gas of the satellites. This possibly could reflect that the different satellite population observed between the Milky Way and M31 and the ones of the SAGA survey could simply reflect a difference in the assembly history of the host galaxies.

6.3. Additional potential heating/cooling sources

As mentioned in section 2.1, our current cooling implementation does not include H_2. Adding this efficient coolant will increase the fragmentation of the gas, making it even more clumpy, strengthening our results.

It is worth mentioning that increasing the heating of the dwarf ISM could obviously help in quenching the star formation by ejecting more gas. Boosting the stellar feedback is not a viable solution as it would fail to reproduce the chemical observed properties of dwarf galaxies (Revaz & Jablonka 2018).

Another possible heating source is the thermal conduction between the MW's hot halo and the dwarf's cold gas. Cowie & McKee (1977) and McKee & Cowie (1977) developed an analytical model for the evaporation of an isolated spherical cloud in a hot gas. They considered both classical (electrons' mean free path smaller than the cloud size) and saturated thermal conduction (electrons' mean free path comparable to the cloud). Their analytical model shows that our dwarfs do not enter any saturated regime and are only marginally dominated by radiation loss. While detailed numerical simulations would be necessary to provide a conclusive answer, this first approximation predicts an evaporation over several Gyr.

Finally, considering the high UV-flux emitted by the proto-host Galaxy (van den Bergh 1994) or the potential strong impact of an AGN could help in removing the remaining confined gas.

7. Conclusions

We have presented high resolution `GEAR`-simulations of the interaction of dwarf spheroidal galaxies formed in a cosmological ΛCDM context with a Milky Way-like galaxy. We first ran a large set of wind tunnel simulations focusing on the hydrodynamical interaction between the dwarf system and the MW hot halo gas. We varied the wind parameters, which describe the velocity at which the dwarf enters the hot halo and orbits around the central galaxy, as well as the density and the temperature of the host halo gas. This allowed us to investigate how the ISM of the dwarf satellite was modified and to infer how its cold and hot gas phases could be ram pressure stripped. In a second step, we performed a set of moving box simulations that added the gravitational tidal interactions to the hydrodynamical ones. We also included the variation of the density and temperature of the hot halo all along the dwarf orbit as well as their increase due to the secular growth of the Milky Way.

The conclusions we reach are significantly different from those of previous works. Indeed, it turns out that including the hydrogen-self shielding that allows the gas to cool much below 10^{4} K, leading to a multiphase ISM, absent in most of the previous studies, is essential to capture the effect of the ram pressure stripping and its impact on the dwarf star formation history.

Our results can be summarized as follows:

- While the hot and diffuse gas phase of the dwarf ($T > 1000\,\mathrm{K}$) is efficiently and quickly stripped by the ram pressure induced by the gas of its host halo, the cold, star forming and clumpy gas phase ($T < 1000\,\mathrm{K}$) is not necessarily. The efficiency of the stripping of this cold gas depends on the ratio between the thermal pressure and the ram pressure both exerted on the dwarf by the hot halo gas. When the thermal pressure is high, the cold gas is confined and its stripping is slowed down.
- As a consequence of the above, the infall time of a dwarf galaxy plays a decisive role in the evolution of the dwarf satellites. If the interaction between the host galaxy and its satellite begins when the thermal pressure is low, that is the host halo is not sufficiently dense or hot, then, the evolution of the dwarf will be essentially the same as in isolation. The

cold ISM will evaporate due to the UV-background heating and star formation will be quenched. On the contrary, the cold ISM is confined and remains attached to the dwarf.

- The confinement of the cold gas in the dwarf satellite leads to an extension of its star formation history. While the same dwarf galaxy would see its star formation quenched due to the evaporation of the residual gas, its interaction with the Milky Way keeps the star formation rate roughly at the level it had when the dwarf entered the host halo. This translates into a higher final mean metallicity, by up to 1 dex in the examples presented in this study. Because our model dwarf spheroidals enter the Milky-Way like galaxy at ~ 2 Gyr, their star formation rates have already significantly decreased, therefore the ejecta of the SNeIa explosion contribute significantly to the dwarf's ISM enrichment. Hence, both our details models display an extended low, although still supersolar, [α/Fe] tail. A solar or sub-solar plateau similar to the Fornax or Sagittarius dwarf galaxy could be obtained if the dwarf enters the hot halo of its host galaxy at later time, where the [α/Fe] decreased to lower values. Firm conclusion on this point would require a dedicated and thorough investigation.

Ram-pressure and tidal interactions do not seem sufficient to explain by themselves the morphology-density relation observed in the Local Group. It would require very specific conditions, either a very late entry of the closest dSphs in the halo of the Milky Way, or a very early accretion before the end of the Galaxy mass assembly. Other processes might play a role, such as the heating by the UV-flux of the Milky Way itself.

Star forming satellites have been found around other Milky Way analogues or in groups (Spencer et al. 2014; Geha et al. 2017). As the effect on the hot host halo strongly depends on the infall time of the satellite galaxy, the different satellite populations observed between the Milky Way and M31 (dominance of quenched gas-poor galaxies) and the ones of the SAGA survey (star forming galaxies) could potentially reflect a difference in the assembly history of the host galaxies.

Acknowledgements. We are indebted to the International Space Science Institute (ISSI), Bern, Switzerland, for supporting and funding the international team 'First stars in dwarf galaxies'. We are grateful to Matthew Nichols to help us with the moving box technique. We enjoyed discussions with Matthieu Schaller, Romain Teyssier, Françoise Combes, Allessandro Lupi, Andrew Emerick. This work was supported by the Swiss Federal Institute of Technology in Lausanne (EPFL) through the use of the facilities of its Scientific IT and Application Support Center (SCITAS). The simulations presented here were run on the Deneb clusters. The data reduction and galaxy maps have been performed using the parallelized Python `pNbody` package (`http://lastro.epfl.ch/projects/pNbody/`). We are grateful to the `Numpy` (Oliphant 2015), `Matplotlib` (Caswell et al. 2018) `SciPy` (Jones et al. 2001) and IPython (Perez & Granger 2007) teams for providing the scientific community with essential python tools.

References

Arraki, K. S., Klypin, A., More, S., & Trujillo-Gomez, S. 2014, MNRAS, 438, 1466
Atek, H., Richard, J., Jauzac, M., et al. 2015, The Astrophysical Journal, 814, 69
Aubert, D. & Teyssier, R. 2010, The Astrophysical Journal, 724, 244
Bahe, Y. M., McCarthy, I. G., Crain, R. A., & Theuns, T. 2012, MNRAS, 424, 1179
Bekki, K. 2014, MNRAS, 438, 444
Bekki, K. & Couch, W. J. 2003, The Astrophysical Journal Letters, 596, L13
Boer, T. J. L. d., Tolstoy, E., Hill, V., et al. 2012, Astronomy & Astrophysics, 544, A73
Bouwens, R. J., Illingworth, G. D., Oesch, P. A., et al. 2015, The Astrophysical Journal, 803, 34
Boylan-Kolchin, M., Bullock, J. S., & Kaplinghat, M. 2011, MNRAS: Letters, 415, L40
Boylan-Kolchin, M., Bullock, J. S., & Kaplinghat, M. 2012, MNRAS, 422, 1203
Bullock, J. S. & Boylan-Kolchin, M. 2017, Annual Review of Astronomy and Astrophysics, 55
Bullock, J. S., Kravtsov, A. V., & Weinberg, D. H. 2000, The Astrophysical Journal, 539, 517
Carlin, J. L., Sheffield, A. A., Cunha, K., & Smith, V. V. 2018, The Astrophysical Journal Letters, 859, L10
Caswell, T. A., Droettboom, M., Hunter, J., et al. 2018, matplotlib/matplotlib v3.0.2
Cloet-Osselaer, A., De Rijcke, S., Schroyen, J., & Dury, V. 2012, MNRAS, 423, 735
Cloet-Osselaer, A., De Rijcke, S., Vandenbroucke, B., et al. 2014, MNRAS, 442, 2909
Cowie, L. L. & McKee, C. F. 1977, The Astrophysical Journal, 211, 135
de Boer, T. J. L., Tolstoy, E., Lemasle, B., et al. 2014, Astronomy & Astrophysics, 572, A10
Del Popolo, A. & Le Delliou, M. 2017, Galaxies, 5, 17
Durier, F. & Vecchia, C. D. 2012, MNRAS, 419, 465
Efstathiou, G. 1992, MNRAS, 256, 43P
Einasto, J., Saar, E., Kaasik, A., & Chernin, A. D. 1974, Nature, 252, 111
Emerick, A., Mac Low, M.-M., Grcevich, J., & Gatto, A. 2016, The Astrophysical Journal, 826, 148
Escala, I., Wetzel, A., Kirby, E. N., et al. 2018, MNRAS, 474, 2194
Ferland, G. J., Chatzikos, M., Guzmán, F., et al. 2017, Revista Mexicana de Astronomia y Astrofisica, 53, 385
Fillingham, S. P., Cooper, M. C., Pace, A. B., et al. 2016, MNRAS, 463, 1916
Fitts, A., Boylan-Kolchin, M., Bullock, J. S., et al. 2018, MNRAS, 479, 319
Fitts, A., Boylan-Kolchin, M., Elbert, O. D., et al. 2017, MNRAS, 471, 3547
Fritz, T. K., Battaglia, G., Pawlowski, M. S., et al. 2018, Astronomy and Astrophysics, 619, A103
Garrison-Kimmel, S., Boylan-Kolchin, M., Bullock, J. S., & Lee, K. 2014, MNRAS, 438, 2578
Garrison-Kimmel, S., Hopkins, P. F., Wetzel, A., et al. 2018, arXiv e-prints, 1806, arXiv:1806.04143
Geha, M., Wechsler, R. H., Mao, Y.-Y., et al. 2017, The Astrophysical Journal, 847, 4
Grcevich, J. & Putman, M. E. 2010, The Astrophysical Journal, 721, 922
Haardt, F. & Madau, P. 2012, The Astrophysical Journal, 746, 125
Hahn, O. & Abel, T. 2011, MNRAS, 415, 2101
Hasselquist, S., Shetrone, M., Smith, V., et al. 2017, The Astrophysical Journal, 845, 162
Henderson, B. & Bekki, K. 2016, The Astrophysical Journal Letters, 822, L33
Hopkins, P. F. 2013, MNRAS, 428, 2840
Hopkins, P. F., Quataert, E., & Murray, N. 2011, MNRAS, 417, 950
Jones, E., Oliphant, T., Peterson, P., & others. 2001, SciPy: Open source scientific tools for Python
Kafle, P. R., Sharma, S., Lewis, G. F., & Bland-Hawthorn, J. 2014, The Astrophysical Journal, 794, 59
Kapferer, W., Sluka, C., Schindler, S., Ferrari, C., & Ziegler, B. 2009, Astronomy and Astrophysics, 499, 87
Klypin, A., Kravtsov, A. V., Valenzuela, O., & Prada, F. 1999, The Astrophysical Journal, 522, 82
Kobayashi, C., Tsujimoto, T., & Nomoto, K. 2000, The Astrophysical Journal, 539, 26
Kronberger, T., Kapferer, W., Ferrari, C., Unterguggenberger, S., & Schindler, S. 2008, Astronomy and Astrophysics, 481, 337
Kroupa, P. 2001, MNRAS, 322, 231
Macciò, A. V., Frings, J., Buck, T., et al. 2017, MNRAS, 472, 2356
Mayer, L., Mastropietro, C., Wadsley, J., Stadel, J., & Moore, B. 2006, MNRAS, 369, 1021
McConnachie, A. W. 2012, The Astronomical Journal, 144, 4
McKee, C. F. & Cowie, L. L. 1977, The Astrophysical Journal, 215, 213
McQuinn, K. B. W., Skillman, E. D., Dolphin, A., et al. 2015, The Astrophysical Journal, 812, 158
Miller, M. J. & Bregman, J. N. 2013, The Astrophysical Journal, 770, 118
Miller, M. J. & Bregman, J. N. 2015, The Astrophysical Journal, 800, 14
Moore, B. 1994, Nature, 370, 629
Moore, B., Ghigna, S., Governato, F., et al. 1999, The Astrophysical Journal Letters, 524, L19
Mulchaey, J. S. & Jeltema, T. E. 2010, The Astrophysical Journal, 715, L1
Navarro, J. F., Frenk, C. S., & White, S. D. M. 1996, The Astrophysical Journal, 462, 563
Navarro, J. F., Frenk, C. S., & White, S. D. M. 1997, The Astrophysical Journal, 490, 493
Nichols, M., Revaz, Y., & Jablonka, P. 2014, Astronomy and Astrophysics, 564, A112
Nichols, M., Revaz, Y., & Jablonka, P. 2015, Astronomy and Astrophysics, 582, A23

Noh, Y. & McQuinn, M. 2014, MNRAS, 444, 503
Okamoto, T., Eke, V. R., Frenk, C. S., & Jenkins, A. 2005, MNRAS, 363, 1299
Oliphant, T. E. 2015, Guide to NumPy, 2nd edn. (USA: CreateSpace Independent Publishing Platform)
Perez, F. & Granger, B. E. 2007, Computing in Science Engineering, 9, 21
Piatek, S., Pryor, C., Bristow, P., et al. 2007, The Astronomical Journal, 133, 818
Piatek, S., Pryor, C., Olszewski, E. W., et al. 2003, The Astronomical Journal, 126, 2346
Planck Collaboration, Ade, P. A. R., Aghanim, N., et al. 2015, Astronomy and Astrophysics, 580, A22
Quinn, T., Katz, N., & Efstathiou, G. 1996, MNRAS, 278, L49
Revaz, Y., Arnaudon, A., Nichols, M., Bonvin, V., & Jablonka, P. 2016, Astronomy and Astrophysics, 588, A21
Revaz, Y. & Jablonka, P. 2012, Astronomy & Astrophysics, 538, A82
Revaz, Y. & Jablonka, P. 2018, Astronomy and Astrophysics, 616, A96
Revaz, Y., Jablonka, P., Sawala, T., et al. 2009, Astronomy and Astrophysics, 501, 189
Robertson, B. E., Ellis, R. S., Furlanetto, S. R., & Dunlop, J. S. 2015, The Astrophysical Journal Letters, 802, L19
Salem, M., Besla, G., Bryan, G., et al. 2015, The Astrophysical Journal, 815, 77
Sarazin, C. L. 1986, Reviews of Modern Physics, 58, 1
Sawala, T., Frenk, C. S., Fattahi, A., et al. 2016, MNRAS, 457, 1931
Sawala, T., Scannapieco, C., Maio, U., & White, S. 2010, MNRAS, 402, 1599
Sawala, T., Scannapieco, C., & White, S. 2012, MNRAS, 420, 1714
Schroyen, J., de Rijcke, S., Valcke, S., Cloet-Osselaer, A., & Dejonghe, H. 2011, MNRAS, 416, 601
Simpson, C. M., Grand, R. J. J., Gómez, F. A., et al. 2018, MNRAS, 478, 548
Skillman, E. D., Monelli, M., Weisz, D. R., et al. 2017, The Astrophysical Journal, 837, 102
Smith, B. D., Bryan, G. L., Glover, S. C. O., et al. 2017, MNRAS, 466, 2217
Spencer, M., Loebman, S., & Yoachim, P. 2014, The Astrophysical Journal, 788, 146
Springel, V. 2005, MNRAS, 364, 1105
Springel, V. & Hernquist, L. 2003, MNRAS, 339, 289
Steinhauser, D., Schindler, S., & Springel, V. 2016, Astronomy and Astrophysics, 591, A51
Stinson, G., Seth, A., Katz, N., et al. 2006, MNRAS, 373, 1074
Tolstoy, E., Hill, V., & Tosi, M. 2009, Annual Review of Astronomy and Astrophysics, 47, 371
Tornatore, L., Borgani, S., Dolag, K., & Matteucci, F. 2007, MNRAS, 382, 1050
Tsujimoto, T., Nomoto, K., Yoshii, Y., et al. 1995, MNRAS, 277, 945
Valcke, S., de Rijcke, S., & Dejonghe, H. 2008, MNRAS, 389, 1111
van den Bergh, S. 1994, The Astronomical Journal, 107, 1328
Van der Swaelmen, M., Hill, V., Primas, F., & Cole, A. A. 2013, Astronomy and Astrophysics, 560, A44
Vazdekis, A., Casuso, E., Peletier, R. F., & Beckman, J. E. 1996, The Astrophysical Journal Supplement Series, 106, 307
Weisz, D. R., Zucker, D. B., Dolphin, A. E., et al. 2012, The Astrophysical Journal, 748, 88
Wetzel, A. R., Deason, A. J., & Garrison-Kimmel, S. 2015, The Astrophysical Journal, 807, 49
Wetzel, A. R., Hopkins, P. F., Kim, J.-h., et al. 2016, The Astrophysical Journal Letters, 827, L23
Wiersma, R. P. C., Schaye, J., Theuns, T., Vecchia, C. D., & Tornatore, L. 2009, MNRAS, 399, 574
Williamson, D. & Martel, H. 2018, The Astrophysical Journal, 867, 72
Wright, A. C., Brooks, A. M., Weisz, D. R., & Christensen, C. R. 2019, MNRAS, 482, 1176
Xue, X. X., Rix, H. W., Zhao, G., et al. 2008, The Astrophysical Journal, 684, 1143
Yozin, C. & Bekki, K. 2015, MNRAS, 453, 14

Appendix A: Additional wind tunnel and moving box simulations

Table A.1: List of additional wind tunnel simulations performed. The first three parameters are for the wind. The four last columns are the properties (total mass, luminosity, cold gas mass and hot gas mass) of the galaxies at the end of the simulation.

Model	ρ_w [10^{-5} atom/cm^3]	u_w [km/s]	T_w [10^6K]	M_{200} [10^8 M$_\odot$]	L_V [10^6 L$_\odot$]	Cold Gas [10^6 M$_\odot$]	Hot Gas [10^6 M$_\odot$]
`h050`	1.277	77	2.00	21.8	59.10	42.42	74.3
`h050`	1.277	100	2.00	21.5	52.85	34.14	67.9
`h050`	1.277	130	2.00	21.0	44.49	27.14	64.5
`h050`	1.277	169	2.00	20.3	34.40	22.96	56.6
`h050`	2.160	77	2.00	22.0	57.06	38.58	93.1
`h050`	2.160	100	2.00	21.6	49.54	39.39	85.9
`h050`	2.160	130	2.00	20.9	39.83	26.27	83.9
`h050`	2.160	169	2.00	20.3	29.24	13.27	85.5
`h050`	2.810	77	2.00	22.2	55.03	42.15	116.0
`h050`	2.810	100	2.00	21.7	47.64	37.86	106.1
`h050`	2.810	130	2.00	20.8	35.38	24.37	99.2
`h050`	2.810	169	2.00	20.3	25.98	15.47	99.3
`h050`	1.660	77	2.00	21.8	56.74	35.09	88.2
`h050`	1.660	100	2.00	21.6	51.61	31.76	81.9
`h050`	1.660	130	2.00	21.1	43.20	26.47	73.8
`h050`	1.660	169	2.00	20.3	31.82	19.97	66.3
`h050`	1.277	169	1.30	20.0	27.80	22.31	54.1
`h070`	1.277	77	2.00	13.0	19.53	16.70	34.7
`h070`	1.277	130	2.00	12.6	13.30	11.35	28.4
`h070`	1.277	169	2.00	12.5	11.69	9.17	28.7
`h070`	1.277	100	2.00	12.8	15.78	13.31	30.4
`h070`	2.160	77	2.00	13.1	17.80	15.20	46.9
`h070`	2.160	130	2.00	12.8	12.95	11.32	43.1
`h070`	2.160	169	2.00	12.7	11.32	8.15	45.2
`h070`	2.160	100	2.00	12.9	14.78	12.03	45.4
`h070`	2.810	77	2.00	13.2	17.55	15.57	60.1
`h070`	2.810	130	2.00	12.8	12.52	10.09	55.2
`h070`	2.810	169	2.00	12.7	10.82	9.11	55.9
`h070`	2.810	100	2.00	13.0	14.79	8.43	60.5
`h070`	1.660	77	2.00	13.0	18.87	8.43	48.9
`h070`	1.660	130	2.00	12.7	12.86	10.86	35.0
`h070`	1.660	169	2.00	12.6	11.72	8.64	35.5
`h070`	1.660	100	2.00	12.8	14.90	14.56	34.4
`h070`	1.277	169	0.40	12.1	5.21	1.93	23.6
`h070`	1.277	169	0.60	12.2	6.57	5.18	23.9
`h070`	1.277	169	1.30	12.5	10.42	8.98	27.5
`h070`	1.277	200	0.40	12.1	4.93	1.07	22.9
`h070`	1.277	200	0.60	12.2	6.00	3.05	23.8
`h070`	1.660	220	0.20	12.1	4.25	0.00	28.5
`h070`	1.660	220	0.30	12.1	4.66	0.00	28.7
`h070`	1.660	250	0.20	12.1	4.36	0.00	28.6
`h070`	1.660	250	0.30	12.1	4.51	0.00	28.4
`h070`	1.660	300	0.30	12.1	4.72	0.00	28.6
`h070`	1.660	400	0.30	12.1	4.89	0.00	28.6
`h123`	2.160	30	0.20	6.7	0.14	0.00	19.7
`h123`	2.160	30	0.40	6.7	0.14	0.00	22.2
`h123`	2.160	30	0.60	6.9	0.33	2.81	37.9
`h123`	2.160	30	1.30	7.0	1.44	4.74	43.4
`h123`	2.160	30	1.54	7.1	2.33	6.10	47.1
`h123`	2.160	30	2.00	7.1	2.35	5.26	44.6
`h123`	2.160	30	3.39	7.2	4.34	7.34	50.0
`h132`	1.277	169	1.30	14.1	10.64	10.58	29.7
`h168`	2.810	30	3.39	10.1	7.70	8.28	85.8
`h074`	1.277	169	1.30	5.8	0.73	0.00	10.6

Table A.2: List of additional moving box simulations performed. The Milky Way model is given with the model name following the same convention than in table 3. The four last columns are the properties (total mass, luminosity, cold gas mass and hot gas mass) of the galaxies at the end of the simulation.

Model	M_{200} [10^8 M$_\odot$]	$M_\star$ [10^6 M$_\odot$]	L_V [10^6 L$_\odot$]	Cold Gas [10^6 M$_\odot$]	Hot Gas [10^6 M$_\odot$]
`h050_sta`	8.8	117.34	17.63	14.10	103.2
`h050_iso`	11.4	36.09	7.90	14.67	21.4
`h123_tem`	3.4	2.78	0.14	0.00	2.8
`h123_iso`	3.7	1.02	0.13	0.00	1.0
`h132_rho`	6.1	77.33	17.85	12.44	64.9
`h132_sta`	2.8	60.77	13.40	9.25	51.5
`h132_tem`	5.0	24.52	9.08	11.24	13.3
`h132_iso`	16.0	0.01	2.88	0.00	0.0
`h168_tem`	8.5	4.54	1.09	0.00	4.5
`h168_iso`	6.1	2.98	1.02	0.00	3.0
`h074_rho`	2.6	34.01	2.57	3.84	30.2
`h074_sta`	1.2	26.15	2.62	3.81	22.3
`h074_tem`	2.4	9.41	2.34	3.64	5.8
`h074_iso`	6.2	0.00	0.65	0.00	0.0

3 Cosmological Simulations in the Framework of AGORA

> It is not despair, for despair is only for those who see the end beyond all doubt. We do not.
> — J. R. R. Tolkien, The Fellowship of the Ring

In this chapter, I will describe the work done within the `AGORA` collaboration during the last 2.5 years in order to obtain the cosmological simulations required for further analysis. Through 5 different steps (only 4 presented in the publication), we have been able to converge towards a common cosmological model and comparable results for all the 7 codes (3 using Smoothed Particle Hydrodynamics, 3 using Adaptive Mesh Refinement and 1 Meshless method) and will be the source of multiple papers within the collaboration. As the previous chapter, all my simulations were carried out with `GEAR` (see chapter 4 for more details on the models), and, as it resulted in a publication, only the context and a summary are provided before the full publication.

A key ingredient in scientific results is the reproducibility of the experiments that allows anyone to verify the results produced by someone else. While numerical experiments have become one of the essential tool to develop new theories of galaxy formation, the reproducibility, unfortunately, has been hard to obtain due to the lack of transparency of some codes and the complexity of the systems solved.

The `AGORA` project[1] is an answer to this issue from the community and aims at improving the quality of the predictions from simulations and making them more realistic through comparison and analysis of simulations run with different codes (Kim et al., 2014). It is also worth it to mention that it also deeply impacts the quality of current codes and future ones by providing additional tests in order to find unexpected bugs. As mentioned before, the complexity of the

---

[1] https://sites.google.com/site/santacruzcomparisonproject/

systems solved makes the reproducibility harder to achieve. This is why, the `AGORA` project started with a simple dark matter only model (Kim et al., 2014) and then slowly increased the complexity by running an isolated disk with a simplified and common baryonic physics (Kim et al., 2016). The simplified baryonic physics consisted in a common cooling rate and UV background given by the `GRACKLE` library, pressure floor, star formation, stellar feedback and yields. While both comparisons projects produce excellent results at large scales between the different codes, some differences were already seen at lower scales and especially between Smoothed Particles Hydrodynamics (SPH) and Adaptive Mesh Refinement (AMR).

In the last project (Roca-Fàbrega et al., 2021 provided afterwards), we decided to move towards a more complicated model by using a cosmological simulation and allowing each group to use its own implementation of the feedback and yields. As such simulations start from high redshift where the universe was almost perfectly homogeneous, the structure growth is very sensitive to noise and any perturbation can easily produce strongly different situations at z=0. Our initial conditions were designed to simulate a galaxy evolving in a halo with a virial mass of about $10^{12} M_\odot$ and a relatively calm merger history after z=2. In order to achieve a resolution of $5.65 \cdot 10^4 M_\odot$ for the gas particles that enables us to resolve the internal structure of the target galaxy, we used the zoom technique described in section 9.2. It resulted in a total of roughly 35 millions particles.

Our calibration consisted in first running again the simulations presented in Kim et al., 2016 with the updated codes and the slight changes in the feedback. The calibration was based on the solar mass expected from the previous paper (Cal-0)[2]. Then we moved towards the cosmological simulation and ran it without any radiative cooling and stellar physics (Cal-1). It allowed us to ensure that the gravity and hydro solvers were behaving correctly before moving to testing with the addition of the radiative cooling (Cal-2). Once all the main solvers were tested, we started to calibrate our simulations by adding the star formation (but no feedback; Cal-3) and finally the feedback that requires far more calibration than the other processes (Cal-4). For the final calibration step, we all aimed at reaching the same stellar mass (between 10^9 and $5 \cdot 10^9 M_\odot$) in the main galaxy at z=4.

In our last publication, we presented our simulations and our calibration process along with a first comparison of the simulations. We were able to assert that the codes converge on critical parameters, but are also showing differences in the coldest and hottest phases of the gas due to the feedback prescription and also the hydrodynamics solver (AMR vs SPH). Finally, we showed that the spatial distribution of metals in the CircumGalactic Medium (CGM) could strongly constrain our different feedback techniques. My contribution in this publication was to produce `GEAR`'s simulations for the analysis through all the 5 different steps of the calibration procedure.

---

[2] This step is not described in the publication.

DRAFT VERSION JUNE 21, 2021
Typeset using LATEX **twocolumn** style in AASTeX62

The *AGORA* High-resolution Galaxy Simulations Comparison Project. III: Cosmological Zoom-in Simulation of A Milky Way-mass Halo

SANTI ROCA-FÀBREGA,[1,*] JI-HOON KIM,[2,3,*] LOIC HAUSAMMANN,[4,*] KENTARO NAGAMINE,[5,6,7,*] ALESSANDRO LUPI,[8,*]
JOHNNY W. POWELL,[9,*] IKKOH SHIMIZU,[10,*] DANIEL CEVERINO,[11,12] JOEL R. PRIMACK,[13] THOMAS R. QUINN,[14] YVES REVAZ,[4]
HÉCTOR VELÁZQUEZ,[15] TOM ABEL,[16,17,18] MICHAEL BUEHLMANN,[19] AVISHAI DEKEL,[20] BILI DONG,[21] OLIVER HAHN,[19]
CAMERON HUMMELS,[22] KI-WON KIM,[2] BRITTON D. SMITH,[23] CLAYTON STRAWN,[13] ROMAIN TEYSSIER,[24] MATTHEW J. TURK,[25]
AND THE *AGORA* COLLABORATION[26,27]

[1]*Departamento de Física de la Tierra y Astrofísica, Facultad de Ciencias Físicas, Plaza Ciencias, 1, 28040 Madrid, Spain*
[2]*Center for Theoretical Physics, Department of Physics and Astronomy, Seoul National University, Seoul 08826, Korea*
[3]*Seoul National University Astronomy Research Center, Seoul 08826, Korea*
[4]*Institute of Physics, Laboratoire d'Astrophysique, École Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Switzerland*
[5]*Department of Earth and Space Science, Graduate School of Science, Osaka University, Toyonaka, Osaka, 560-0043, Japan*
[6]*Kavli IPMU (WPI), University of Tokyo, 5-1-5 Kashiwanoha, Kashiwa, Chiba, 277-8583, Japan*
[7]*Department of Physics & Astronomy, University of Nevada Las Vegas, Las Vegas, NV 89154, USA*
[8]*Dipartimento di Fisica "G. Occhialini", Università degli Studi di Milano-Bicocca, I-20126 Milano, Italy*
[9]*Department of Physics, Reed College, Portland, OR 97202, USA*
[10]*Shikoku Gakuin University, 3-2-1 Bunkyocho, Zentsuji, Kagawa, 765-8505, Japan*
[11]*Departamento de Física Teórica, Facultad de Ciencias, Universidad Autónoma de Madrid, Cantoblanco, 28049 Madrid, Spain*
[12]*CIAFF, Facultad de Ciencias, Universidad Autónoma de Madrid, 28049 Madrid, Spain*
[13]*Department of Physics, University of California at Santa Cruz, Santa Cruz, CA 95064, USA*
[14]*Department of Astronomy, University of Washington, Seattle, WA 98195, USA*
[15]*Instituto de Astronomía, Universidad Nacional Autónoma de México, A.P. 70-264, 04510, Mexico, D.F., Mexico*
[16]*Kavli Institute for Particle Astrophysics and Cosmology, Stanford University, Stanford, CA 94305, USA*
[17]*Department of Physics, Stanford University, Stanford, CA 94305, USA*
[18]*SLAC National Accelerator Laboratory, Menlo Park, CA 94025, USA*
[19]*Laboratoire Lagrange, Université Côte d'Azur, Observatoire de la Côte d'Azur, CNRS, Blvd de l'Observatoire, CS 34229, 06304 Nice, France*
[20]*Center for Astrophysics and Planetary Science, Racah Institute of Physics, The Hebrew University, Jerusalem 91904, Israel*
[21]*Department of Physics, Center for Astrophysics and Space Sciences, University of California at San Diego, La Jolla, CA 92093, USA*
[22]*TAPIR, Department of Astronomy, California Institute of Technology, Pasadena, CA 91125, USA*
[23]*Institute for Astronomy, University of Edinburgh, Royal Observatory, Edinburgh EH9 3HJ, United Kingdom*
[24]*Centre for Theoretical Astrophysics and Cosmology, Institute for Computational Science, University of Zurich, Zurich, 8057, Switzerland*
[25]*School of Information Sciences and Department of Astronomy, University of Illinois, Urbana-Champaign, IL 61820, USA*
[26]http://www.AGORAsimulations.org
[27]The authors marked with * as code leaders contributed to the article by leading the effort within each code group to perform and analyze simulations.

(Received March 30, 2021; Revised June 21, 2021; Accepted June XX, 2021)

ABSTRACT

We present a suite of high-resolution cosmological zoom-in simulations to $z = 4$ of a 10^{12} $M_\odot$ halo at $z = 0$, obtained using seven contemporary astrophysical simulation codes (ART-I, ENZO, RAMSES, CHANGA, GADGET-3, GEAR, and GIZMO) widely used in the numerical galaxy formation community. Physics prescriptions for gas cooling, heating and star formation are similar to the ones used in our previous *AGORA* disk comparison (Kim et al. 2016) but now account for the effects of cosmological processes such as the expansion of the Universe, intergalactic gas inflow, and the cosmic ultraviolet background radiation emitted by massive stars and quasars. In this work, we introduce the most careful comparison yet of galaxy formation simulations run by different code groups, together with a series of four calibration steps each of which is designed to reduce the number of tunable

Corresponding author: Santi Roca-Fàbrega, sroca01@ucm.es
Corresponding author: Ji-hoon Kim, mornkr@snu.ac.kr
Corresponding author: Loic Hausammann, loic.hausammann@epfl.ch
Corresponding author: Kentaro Nagamine, kn@astro-osaka.jp

simulation parameters adopted in the final run. In the first two steps, we methodically calibrate the gas physics such as cooling and heating, in simulations without star formation. In the third, we seek an agreement on the total stellar mass produced with the common star formation prescription used in the *AGORA* disk comparison, in stellar feedback-free simulations. In the last calibration step, we activate stellar feedback, where each code group is asked to set the feedback prescriptions to be as close to the most used one in each code community as possible, while aiming for convergence in the stellar mass at $z = 4$ to the values predicted by semi-empirical models. After all the participating code groups successfully completed the calibration steps, we reach a suite of cosmological simulations with similar mass assembly histories down to $z = 4$. With numerical accuracy that resolves the internal structure of a target halo ($\lesssim 100$ physical pc at $z = 4$), we find that the codes overall agree well with one another in e.g., gas and stellar properties, but also show differences in e.g., circumgalactic medium (CGM) properties. We argue that, if adequately tested in accordance with our proposed calibration steps and common parameters, the results of high-resolution cosmological zoom-in simulations can be robust and reproducible. New code groups are invited to join and enrich this comparison by generating equivalent models or to test the code's compatibility on their own, by adopting the common initial conditions, the common easy-to-implement physics package, and the proposed calibration steps. Further analyses of the zoom-in simulations presented here will be in forthcoming reports from the *AGORA* Collaboration, including studies of the CGM, simulations by additional codes, and results at lower redshift.

Keywords: cosmology: theory – galaxies: formation – galaxies: evolution – galaxies: kinematics and dynamics – galaxies: intergalactic medium – galaxies: ISM – methods: numerical – hydrodynamics

1. INTRODUCTION

Established in 2012, the *AGORA* High-resolution Galaxy Simulations Comparison Project (*Assembling Galaxies of Resolved Anatomy*) has since aimed at collectively raising the predictive power of contemporary numerical galaxy formation studies, by carefully comparing high-resolution galaxy simulations on multiple code platforms widely used in the field. The main goal of the *AGORA* initiative has been to ensure that physical assumptions are responsible for any success in the numerical studies, rather than manifestations of a particular numerical implementation. As of this writing, we have more than 160 individuals from over 60 different academic institutions worldwide who have agreed to the Project's philosophy and participated in its collaborative effort in varying degrees. The Collaboration has continued to provide a sustainable platform on which members could talk to and learn from others from different code communities, and discuss ambitious "multi-platform" collaborations. The Project indeed has become a great social experiment in itself — about the scientific community's collective willingness to assure the integrity and reproducibility of its experiments.[1]

The first paper of the Collaboration (Kim et al. 2014, hereafter Paper I) focused on introducing the Project to the community. It presented the first proof-of-concept simulations, dark matter-only but using cosmological zoom-in initial conditions. Results from comparing the cosmological simulations among nine flavors of the state-of-the-art numerical codes showed a robust convergence. In the second paper from the *AGORA* Collaboration (Kim et al. 2016, hereafter Paper II) we presented a comparison of idealized Milky Way-mass galaxies simulated in isolation, obtained from nine widely-used state-of-the-art gravito-hydrodynamics codes, which were recently made available to be freely used by the community (Roca-Fàbrega et al. 2020). The simulations in Paper II achieved an overall agreement with one another in many parameter spaces for both gaseous and stellar components. Yet, some discrepancies were expected and present, which were understood as systematic differences between codes, for example, between mesh-based and particle-based codes in low-density regions, and between more diffusive and less diffusive schemes in the high-density region. Such intrinsic differences were, however, found to be small in general compared to the variations in the implementations of common subgrid physics such as supernova (SN) feedback.

The *AGORA* Project has helped to establish a simulation infrastructure essential to achieve our thorough comparisons so far, and it will allow and foster future comparisons. It includes, among others, a common initial condition generator (MUSIC; Hahn & Abel 2011),[2] a common gas cooling and heating scheme (GRACKLE; Smith et al. 2017),[3] and a common analysis toolkit (`yt`; Turk et al. 2011),[4] all of which are publicly available software. In particular, all the figures and plots in this article and Papers I and II have been pro-

---

* Code leaders

[1] See the Project website at http://www.AGORAsimulations.org/ for more information about the *AGORA* Collaboration.

[2] The website is https://www-n.oca.eu/ohahn/MUSIC/.

[3] The website is http://grackle.readthedocs.io/.

[4] The website is http://yt-project.org/.

duced with the *AGORA* common analysis platform based on `yt`. It is also worth noting that several recent comparison and calibration studies have been motivated by the results presented in our previous reports. Examples include the study of changes on the star formation efficiency in molecular clouds (Grisdale et al. 2019), tests of new star formation and supernova feedback implementations, both in isolated (Shimizu et al. 2019) and cosmological contexts (Oh et al. 2020).

Building upon the past achievements, in this third paper of our continuing endeavor in *AGORA*, we follow a path similar to Paper II, but this time with cosmological "zoom-in" simulations. This type of comparison has never been properly carried out due to its complexity and time-consuming nature. However, it is now possible — though still challenging — thanks to the infrastructure the *AGORA* Collaboration has built and maintained. A reproducibility check like this is essential as the field relies increasingly on the numerical verification of galaxy formation theories in cosmological contexts. All code groups started their simulations from a common initial condition generated with MUSIC (Section 2). The physics prescriptions (e.g., gas cooling and heating, star formation parameters) are also common among all participating codes as in Paper II, although some changes were made in each code (Section 3). Only the decision concerning the stellar feedback prescription and metal production to be used is left to each code group, and code groups are asked to use a prescription close to to the most widely-used practice in each code community. Spatial resolution of $\lesssim 100$ physical pc at $z = 4$ is imposed to resolve the internal structure of a target halo, and to make our physics prescriptions less reliant on platform-specific models (Section 4). After a series of calibration steps for the adopted physical processes (Figure 1 and Section 5), we reach a suite of simulations illustrating how seven state-of-the-art codes reproduce the formation and evolution of a Milky Way-type galaxy in a cosmological context down to $z = 4$ with their favorite stellar feedback and metal production prescriptions (Section 6). As in the previous *AGORA* comparisons, we caution that we do not intend to identify a correct or incorrect code, but to focus on juxtaposing different codes for physical insights and learn how much scatter one should expect among modern simulations.

This paper is organized as follows. Section 2 describes the initial condition of our experiment. We discuss physics modules employed in our simulations in Section 3, and the runtime parameters in Section 4. Section 5 presents our calibration steps designed to prepare the ground for the final simulation entries. In Section 6 we compare the results of our final runs, focusing on the stellar and gas properties of the target halo, and its evolution in time. Finally, in Section 7 we conclude the article with remarks on how *AGORA*'s "multi-platform" approach can significantly enhance the scientific value of numerical galaxy formation studies.

2. INITIAL CONDITION

We use a set of parameters for MUSIC, an initial condition (IC) generator with an adaptive multi-grid Poisson solver (Hahn & Abel 2011), that depicts a halo evolving to a virial mass of $\sim 10^{12}$ $M_\odot$ at $z = 0$ with a relatively quiescent merger history between $z = 2$ and 0.[5] The IC, tagged *1e12q*, is identified and made publicly available by the *AGORA* Collaboration (Paper I).[6] We assume a flat ΛCDM cosmology consistent with *WMAP7/9+SNe+BAO*: $\Omega_{\rm m} = 0.272$, $\Omega_\Lambda = 0.728$, $\sigma_8 = 0.807$, $n_{\rm s} = 0.961$, and $H_0 = 70.2$ km s^{-1}Mpc^{-1} (Komatsu et al. 2011; Hinshaw et al. 2013). The initial metallicity is set to $10^{-4} Z_\odot$ everywhere.[7]

With a 128^3 root resolution in a $(60$ comoving h^{-1} Mpc$)^3$ box and a series of five nested higher-resolution regions, the equivalent unigrid resolution at the finest "zoom-in" region is 4096^3 (i.e., MUSIC parameters $[\ell_{\rm min}, \ell_{\rm max}] = [7, 12]$). The highest-resolution region in this IC is in an ellipsoidal shape that is large enough to enclose all the particles that eventually end up within $4R_{\rm vir}$ of the target halo at $z = 0$. Correspondingly, the target halo contains the highest-resolution particles of masses $m_{\rm DM,IC} = 2.8 \times 10^5$ $M_\odot$ and $m_{\rm gas,IC} = 5.65 \times 10^4$ $M_\odot$ each, the latter designed to approximately match the gas resolution in Paper II, $m_{\rm gas} = 8.6 \times 10^4$ $M_\odot$. For more information about this IC and other available *AGORA* ICs, we refer the interested readers to Section 2 of Paper I.

3. PHYSICS IN THE CODES

We briefly summarize the key physics and code-by-code differences for this particular comparison.

3.1. *Common, Code-independent Physics*

The common baryonic physics for our study is based on Papers I and II. To begin with, the cooling library GRACKLE determines the rate of radiative gas cooling based on the properties of gas parcels (Smith et al. 2017). The interface we built for Paper II is utilized by each participating code, in the *equilibrium* cooling mode of GRACKLE-V3.1.1. Here, GRACKLE looks up a pre-computed CLOUDY cooling table for primordial and $1 Z_\odot$ metallicities as functions of gas density and temperature (Ferland et al. 2013). To obtain the corresponding gas cooling and heating rates, the $1 Z_\odot$ rates are linearly scaled by the gas metallicity (Section 3.1 of Paper II), and the result is added to the values from primordial gas to get the combined rate. GRACKLE also includes redshift-dependent cosmic ultraviolet background radiation (UVB; Haardt & Madau 2012) with hydrogen self-shielding

---

[5] Here we use MUSIC's changeset ID `eb870ed`.

[6] See http://www.AGORAsimulations.org/ or http://sites.google.com/site/santacruzcomparisonproject/blogs/quicklinks/.

[7] $1 Z_\odot = 0.02041$ is used across all participating the codes in order to follow our choice in Paper II (see Section 2 of Paper II for details).

(i.e., input file `CloudyData_UVB=HM2012_shielded.h5`; see also Section 3.3 of Paper I). In addition, instead of using GRACKLE's own cosmic microwave background (CMB) temperature floor, each code is supplemented with a redshift-dependent, but density-independent CMB floor.[8]

Lastly, in order to prevent unphysical collapse or fragmentation due to limited resolution, in Calibration steps 3 and 4 we apply a nonthermal pressure floor $P_{\rm Jeans}$ that forces the local Jeans length to be resolved at a given numerical resolution at all times. Its value is $P_{\rm Jeans} = (\gamma\pi)^{-1}N_{\rm Jeans}^2 G\rho_{\rm gas}^2\Delta x^2$, where $\gamma = 5/3$ is the adiabatic index, $N_{\rm Jeans} = 4$ is the Jeans number, G is the gravitational constant, $\rho_{\rm gas}$ is the gas density, and Δx is the finest spatial resolution in physical units (finest cell size for mesh-based codes, or gravitational softening length for particle-based codes; see Section 4). This additional pressure term can be interpreted as the extra pressure source due to the unresolved interstellar medium (ISM) turbulence. For actual implementations of the pressure floor in each code, we refer the readers to Sections 3.1 and Appendix A of Paper II.

When the density of a gas parcel exceeds $n_{\rm H,thres} = 1\,{\rm cm}^{-3}$ (note the difference with $n_{\rm H,thres}$ used in Paper II), a star particle can be created at a rate of $d\rho_\star/dt = \varepsilon_\star\rho_{\rm gas}/t_{\rm ff}$, where $\varepsilon_\star = 0.01$ is the formation efficiency and $t_{\rm ff} = (3\pi/(32G\rho_{\rm gas}))^{1/2}$ is the local free-fall time. The only freedom that is left to each code group is to choose the stochastic or deterministic nature of this process. A single star particle depicts a collection of cluster-sized masses sharing the same age and metallicity, corresponding to a single stellar population. It is required to weigh more than $6.1\times10^4\,{\rm M}_\odot$ at creation for the mesh-based codes — a value approximately matching the gas resolution in the IC, $m_{\rm gas,IC} = 5.65\times10^4\,{\rm M}_\odot$ — or inherits the mass of its parent gas particle in particle-based codes. In Paper II, our stellar feedback formula implied one Type II supernova event per every 91 ${\rm M}_\odot$ stellar mass formed, each of which instantaneously releasing 10^{51} ergs of thermal energy, 14.8 ${\rm M}_\odot$ of gas, and 2.6 ${\rm M}_\odot$ of metals. In contrast, in this work, while the returned mass is equal to that in Paper II, the exact deposit scheme into the ISM, such as stellar winds or supernova events, and their associated energy and metal yields are left to each code group's discretion. We do ask the deposit scheme to be as close to the *most widely-used practice* in its community as possible (detailed in Section 3.2 and Table 1; see also Sections 3.2, 5 and Appendix B of Paper II). We also leave the choice of whether to implement an explicit metal diffusion scheme to each particle-based code group (see Sections 3.2.4 to 3.2.6 for details).

We note that our common physics models including subgrid physics (e.g., star formation) helped us in Paper II to produce similar stellar disks across all codes — comparable in terms of their morphologies, kinematics, star formation relations, to name a few (Sections 6.4 to 6.6 of Paper II). In the present comparison, however, we use a fully cosmological setup that is substantially more complex. Although the common subgrid physics models here are based on the ones in the idealized galaxy setup (Paper II), we have found a need to introduce changes to the fiducial parameters to reproduce a realistic galactic system at low redshift. The fiducial set of parameters has been modified in e.g., the star formation threshold density ($n_{\rm H,thres} = 1\,{\rm cm}^{-3}$ instead of $10\,{\rm cm}^{-3}$ in Paper II) and the stellar feedback scheme (instead of a common simple thermal deposit model in Paper II; see Section 3.2). These changes have been motivated by the deviation in $M_*/M_{\rm halo}$ from the observed value in Paper II, and to account for the potential redshift dependence of the adopted physics.

3.2. *Participating Codes and Code-dependent Physics*

Here we briefly explain the physics included in each code, focusing only on the part that is changed from Paper II, or is unique for each code. Hence, the interested readers are encouraged to see our previous work to grasp the full picture of how each code works — Paper I for gravitational dynamics and Paper II for hydrodynamics. In particular, Table 1 summarizes the key stellar feedback parameters and effective metal yield in each code, in which each code group is left with freedom to choose its own feedback scheme for energy and metals. It should be noted that the code groups involved in future *AGORA* studies are not limited to the seven codes listed in this section.

3.2.1. ART-I

The ART-I code (Kravtsov et al. 1997; Kravtsov 2003; Ceverino & Klypin 2009) used to obtain the cosmological simulation presented here is based on the one used in the previous comparison efforts (Papers I and II). Only a few minor modifications should be noted. Among them is a change in the adaptive mesh refinement (AMR) strategy to better follow the cosmic evolution of large scale structures. This change is in line with what has been commonly used in previous ART-I cosmological zoom-in simulations (e.g., Ceverino et al. 2010, 2014, 2017). We have also updated the gas cooling and heating scheme from ART-I's own machinery using the CLOUDY table in Paper II to the standard package GRACKLE-V3.1.1 in the current paper. The nonthermal pressure floor in ART-I is slightly different from the common prescription (Section 3.1); in other words, the Jeans length is resolved by at least seven resolution elements at all times (Ceverino et al. 2010).

ART-I uses a stochastic star formation subgrid model. Details on this star formation model can be found in Ceverino et al. (2014). We slightly change the stochasticity of star formation to ensure that we use the common star formation ef-

---

[8] This functionality is planned to be added to the latest GRACKLE.

Table 1. Stellar feedback implementation adopted by each code group†

Code	Stellar feedback	SN & metal production model	Effective metal yield	Runtime parameters
Art-I	T+K, RP	SN Type Ia/II, AGB stars*	0.033	$E_{\rm thermal} = 2\times10^{51}\,{\rm ergs/SN},\ p = 3.6\times10^{6}\,{\rm M_{\odot}\,km\,s^{-1}/SN}$
Enzo	T	SN Type II	0.032	$E_{\rm thermal} = 5\times10^{52}\,{\rm ergs/SN}$
Ramses	T, DC	SN Type II	0.033	$E_{\rm thermal} = 4\times10^{51}\,{\rm ergs/SN},\ \sigma_{\rm min} = 100\,{\rm km\,s^{-1}},\ T_{\rm delay} = 10\,{\rm Myr}$
Changa	T+S	SN Type Ia/II, AGB stars**	0.032	$E_{\rm thermal} = 5\times10^{51}\,{\rm ergs/SN}$
Gadget-3	T+K, RP, DC	SN Type Ia/II, AGB stars	0.025	$E_{\rm SN} = 4\times10^{49}\,{\rm ergs/M_{\odot}},\ T_{\rm delay} = t_{\rm hot}$ (see Section 3.2.5)
Gear	T, DC	SN Type Ia/II	0.024	$E_{\rm thermal} = 4.5\times10^{51}\,{\rm ergs/SN},\ T_{\rm delay} = 5\,{\rm Myr}$
Gizmo	T+K	SN Type II	0.033	$E_{\rm SN} = 5\times10^{51}\,{\rm ergs/SN}$

† T = thermal feedback, K = kinetic feedback, RP = radiation pressure, DC = delayed cooling, S = superbubble, * = only for energy production (not metal), ** = only for metal production (not energy). While the total returned mass via feedback is constrained across the code platforms (Section 3.1), the exact feedback scheme and the metal yield are left to each code group's discretion to be as close to the most widely-used practice in its community as possible. For more information on the items listed here, see Section 3.2. For more information on the "effective" metal yield by stellar feedback measured in the entire simulation box at $z = 4$ for the `CosmoRun` suite of simulations (fourth column), see Section 6.2.2.

ficiency value (Section 3.1). Art-I's prescription fits within the agreed *AGORA* parameter range. The treatment of stellar feedback is similar to the model in Ceverino et al. (2017), which includes thermal, kinetic and radiation pressure feedback. The code also includes the later effects of supernova Type Ia and stellar mass loss, and it follows the metal enrichment of the ISM. The convergence goal in the calibration step 4 (`Cal-4`; Section 5.4) is achieved by the widely used feedback model in the `VELA6` simulations (Ceverino et al, in prep.) but with four times more injection of momentum (see parameter p in Table 1). This increase tries to compensate for the differences in resolution. The default AGORA effective metal yield has been obtained by increasing the standard SNII and SNIa yields in ART-I by a factor of four.

3.2.2. Enzo

The Enzo code (Bryan et al. 2014; Brummel-Smith et al. 2019) for this work is from the `master` branch in the publicly available `enzo-dev` repository.[9] Star formation is implemented following the same approach as in Paper II, that is a fully deterministic scheme. To incorporate the stellar feedback model established in Paper II, files such as `star_maker4.F` and `Grid_StarParticleHandler.C` in the said repository need a minor modification. To reach the convergence in our calibration step 4 (`Cal-4`; Section 5.4), the stellar feedback efficiency parameter is increased from the value in Paper II, matching the findings in recent Enzo calibration studies against observations (e.g., Oh et al. 2020, see also Table 1). The model only accounts for effects by supernova Type II. Other adopted schemes such as the hydrodynamics solver are the same as in Paper II, and are largely in line with the recent numerical galaxy formation studies using Enzo (e.g., Kim et al. 2019; Shin et al. 2020).

In order to realize the ellipsoid-shaped IC in simulations (Section 2), Enzo identifies and tracks the ellipsoidal Lagrangian region using a special type of dark matter particles called `MustRefineParticle` that eventually constitute the target halo at a predetermined target redshift. Cells around these particles are always refined at least down to 20.9 comoving kpc — or 5 additional refinement levels for a 128^3 root resolution in a $(60\ {\rm comoving}\ h^{-1}\,{\rm Mpc})^3$ box — corresponding to the Music parameter $\ell_{\rm max} = 12$.

3.2.3. Ramses

The Ramses code (Teyssier 2002) used in this comparison is from the December 2019 `master` branch of the code repository.[10] Star formation is implemented following Paper II, but without using a temperature threshold. This temperature threshold was closely linked with the implementation of a temperature polytrope to avoid numerical fragmentation, and this approach is no longer in use in the present work. Thus, the implementation of the nonthermal pressure support to avoid artificial fragmentation takes a different approach from the one in Paper II, being now consistent with the common implementation presented in Section 3.1. With this implementation we ensure that the local Jeans length is resolved at least by four AMR cells at all times. The star formation approach is well described in the most recent works within the code community (e.g., Nuñez-Castiñeyra et al. 2020).

The treatment of stellar feedback here closely follows the so-called "delayed cooling thermal feedback model" formulated in Dubois et al. (2015), and only accounts for effects by supernova Type II. The Ramses simulation presented here includes modifications to the model, however, as described in Rosdahl et al. (2017, Section 3.3) and Nuñez-Castiñeyra et al. (2020, Section 2.1.3). Our choices of runtime parameters are listed in the Table 1. We note that, out of our tested feedback prescriptions available in Ramses, the one used here is what succeeded in producing the target stellar mass at $z = 4$ in our calibration step 4 (`Cal-4`; Section 5.4).

---

[9] The website is http://enzo-project.org/. Here we use Enzo's changeset ID `02c88172`.

[10] The website is https://bitbucket.org/rteyssie/ramses/.

3.2.4. CHANGA

CHANGA-V3.4 is a reimplementation of the smoothed particle hydrodynamics (SPH) code GASOLINE (Wadsley et al. 2017) in the CHARM++ (CHARMPPPPWCPP96) v6.9 runtime system.[11] The code used for the present paper is based on the one in the previous hydrodynamic comparison; therefore, we refer the interested readers to Section 5.5 of Paper II and here we note only a few points and changes. In CHANGA, the k-th nearest neighbor algorithm is used to find the $N_{\rm ngb} = 64$ nearest neighbors, then the Wendland C4 kernel (Dehnen & Aly 2012) is employed to determine hydrodynamic properties. Energy and metals are diffused using the scheme of Shen et al. (2010). We have implemented GRACKLE-V3.1.1 after careful scrutiny.[12]

The treatment of stellar feedback follows the "superbubble" strategy presented by Keller et al. (2014), different from Paper II. It includes thermal conduction inside resolved hot bubbles, which maintains uniform temperatures (see the characteristic bubble shapes in Figure 16). This method makes the amount of cold gas heated by feedback not a free parameter, but set by the thermal conduction. In the first few Myr of feedback heating, the mass contained within a hot bubble can be smaller than the simulation's gas mass resolution, which could result in strong overcooling. To prevent overcooling, the resolution elements briefly represent two components: (1) a hot interior (bubble) where the feedback energy is injected, and (2) a cold shell in pressure equilibrium with the hot interior. The particle returns to a single phase once all the cold gas is evaporated or the hot phase cools below 10^5 K. Thermal energy representing supernova Type Ia and Type II is deposited to the neighboring $N_{\rm ngb}$ particles. Supernova Type II rates are calculated from the Raiteri et al. (1996) fit to the Padova stellar models. Type Ia rates are computed from the evolution timescales of secondaries in binaries (Matteucci & Greggio 1986). To reach the convergence in our calibration step 4 (`Cal-4`; Section 5.4), the thermal energy is increased to 5×10^{51} ergs per supernova for the Kroupa initial mass function (IMF), from the typical value used in the community. Metals are released by supernovae and asymptotic giant branch (AGB) stars following Raiteri et al. (1996).

3.2.5. GADGET-3

GADGET-3-OSAKA is a modified version of GADGET-3 — which itself is an extended version of the SPH code GADGET-2 (Springel 2005). The code includes the common cooling and star formation model detailed in Papers I and II, and the treatment of stellar feedback presented in Aoyama et al. (2017, 2018) and Shimizu et al. (2019). It also includes important improvements such as the density-independent, pressure-entropy formulation of SPH (Hopkins 2013; Saitoh & Makino 2013), the time-step limiter (Saitoh & Makino 2009), quintic spline kernel (Morris 1996), and the number of neighbor particles for each SPH particle is set to 128 ± 8.

For stellar feedback, we distribute both thermal and kinetic energy to neighboring gas particles within a hot bubble, whose size is determined by the local gas density, ambient gas pressure, and feedback energy (see Eqs.(6)-(7) in Shimizu et al. 2019). We utilize the CELIB chemical evolution library (Saitoh 2017) which provides the chemical yield distribution as a function of time for a given IMF. We deposit metals and energy according to the CELIB output with certain time delays ($t_{\rm hot}$) that depend on the feedback energy, density, and ambient gas pressure, treating supernova Type Ia, Type II, and AGB star contributions separately.[13] The total injected energy is slightly boosted over the canonical CELIB output, to 4×10^{49} ergs per 1 $\mathrm{M}_\odot$ of star forming gas, corresponding to $E_{\rm SN} = 4 \times 10^{51}$ erg per supernova for the Chabrier IMF adopted in CELIB. For details, see Shimizu et al. (2019). The exact prescription used in this paper is similar to the fiducial model `K30T70` therein, except for the equal division of supernova energy into thermal (50%) and kinetic (50%) component to reach the target stellar mass (`Cal-4`; Section 5.4). Early stellar feedback is also adopted in the form of thermal energy injected before the first supernova explodes. Metal diffusion is not implemented as an explicit process, but metals are smoothed over the SPH kernel when computing the metallicity or cooling rates of each gas particle, mimicking the effect of metal diffusion (Okamoto et al. 2005; Tornatore et al. 2007; Wiersma et al. 2009).

3.2.6. GEAR

The GEAR code is a chemo-dynamical tree SPH code based on GADGET-2 (Springel 2005). Its original version was described in Revaz & Jablonka (2012) with some improvements discussed in Revaz et al. (2016) and Revaz & Jablonka (2018). For the difference between GEAR and the public version of GADGET-2, we refer the interested readers to Section 5.8 of Paper II. Cooling and star formation prescriptions adopted here are similar to the ones in Paper II.

In our feedback prescription, both energy and yields are deposited among the nearest gas particles so that each neigh-

---

[11] The websites are http://github.com/N-BodyShop/changa/ and http://charm.cs.uiuc.edu/.

[12] The CLOUDY table used in CHANGA differs slightly from the one in the other codes, containing a latest update by the GRACKLE developers. This update only affects an unlikely case of very dense gas at very high redshifts, so it does not change the conclusion of the present article.

[13] For example, oxygen production is always dominated by Type II supernova, carbon is dominated by AGB stars after a few hundred Myrs, and iron is dominated by Type Ia supernova after 10^8 years.

bor receives a fraction of energy weighted by the SPH kernel. $N_{\rm ngb}$ corresponds to a *weighted* number of neighbors and is set to 50. Thus, depending on the spatial distribution of gas particles more or less than 50 particles will receive stellar ejecta. The stellar feedback is tightly coupled to our adopted chemical evolution model, that includes both supernova Type Ia and II with yields from Kobayashi et al. (2000) and Tsujimoto et al. (1995), respectively. Exploding supernovae are computed stochastically using a continuous IMF sampling scheme (CIMFS; Revaz et al. 2016). Thus here, a thermal energy equivalent to 4.5×10^{51} ergs per supernova is released into the ISM, following a blast wave-like feedback scheme (Stinson et al. 2006) with a 5 Myr delayed cooling time. While Gear does not include artificial metal diffusion, we use the smooth metallicity scheme to mix the metal-enriched gas effectively (as in Gadget-3; see Sections 3.2.5).

3.2.7. Gizmo

Gizmo is a mesh-free hydrodynamics code (Hopkins 2015), a descendant of Gadget-3, in which a kernel-based partition scheme is used to discretize the domain in a set of unstructured "cells" that are allowed to move and reshape with time. The Riemann problem is solved across the effective faces shared by neighbouring cells, similarly to what is done in the grid-based codes. The version used for this work includes the common cooling and star formation models described in Paper II while stellar feedback is based on the mechanical feedback model described in Hopkins et al. (2018); i.e., both kinetic and thermal energy are distributed among gas cells lying within each star particle kernel according to the evolutionary stage of the supernova blast-wave (energy or momentum conserving). The supernova rate used in this work is described by a piecewise function, where we assume the decaying power-law fit in Lupi et al. (2020) for star particles older than 5.089 Myr, and a constant rate equal to the power-law maximum value for younger stars, aimed at modelling early feedback by massive stars. For consistency, the integrated number of supernova events is normalised to ensure 1 supernova per every 91 M$_\odot$, while the injected energy is set to 5×10^{51} ergs per supernova in order to reproduce the desired stellar mass at $z=4$.

4. COMMON RUNTIME PARAMETERS

We describe our choices of common runtime parameters such as numerical resolution. They are based on what we used in the dark matter-only cosmological test for a galaxy-sized halo (Section 5 of Paper I), and in the isolated disk test in a Milky Way-sized halo (Section 4 of Paper II).

For the particle-based codes Changa, Gadget-3 and Gear, a spline kernel is used to soften the gravity (e.g., Eq. (A1) of Hernquist & Katz 1989). The gravitational softening length $\varepsilon_{\rm grav}$ in the highest-resolution region is set to 800 comoving pc until $z=9$, and 80 proper pc afterward. While this resolution is better than what Eq. (15) of Power et al. (2003) proposes (~ 220 pc), it is used to match the resolution of Paper II at which our fiducial subgrid physics models were initially calibrated. For particles in the lower-resolution region at a corresponding Music level ℓ, the softening length is set at $80\times8^{(\ell_{\rm max}-\ell)/2}$ proper pc after $z=9$, as Power et al. (2003) suggests $\varepsilon_{{\rm grav},\ell}\propto N_{200}^{-1/2}\propto(m_{{\rm DM},\ell})^{1/2}$. For particle-based codes, we also require that the minimum hydrodynamical smoothing lengths for gas particles be $0.2\,\varepsilon_{\rm grav}$. The exact choice for a smoothing scheme is left to each code group's discretion (see Section 5 and Appendix C of Paper II).

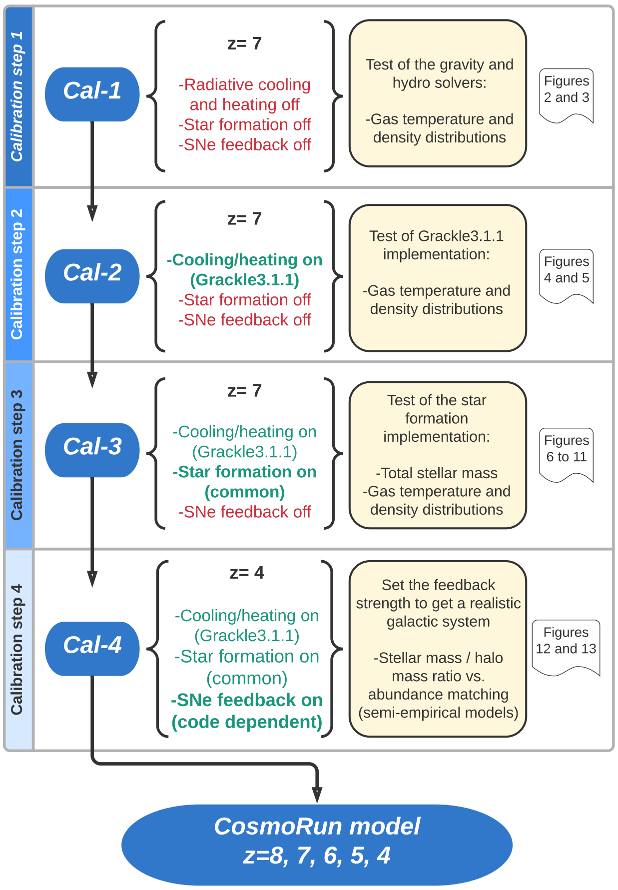


Figure 1. Summary of the physics calibration procedure. We indicate, from left to the right, the target redshift and the physics prescriptions in each step, the main objective and the used variables to test convergence, and the corresponding figures.

Meanwhile, the finest cell size of the mesh-based codes Art-I, Enzo and Ramses) is set to 163 comoving pc, or 12 additional refinement levels for a 128^3 root resolution in a $(60$ comoving $h^{-1}\,$Mpc$)^3$ box. A cell is adaptively refined into 8 child cells on particle or gas over-densities of 4. Given the differences in refinement algorithms among the codes,

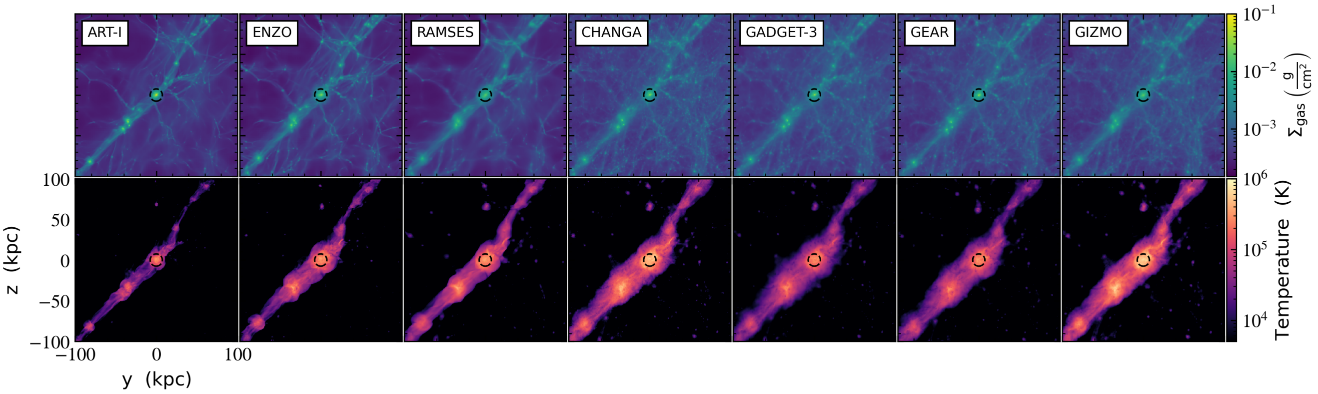


Figure 2. Gas density projection (*top*) and density-weighted temperature projection (*bottom*; each projected through a slab of thickness 200 kpc) at $z = 7$ from the first calibration step, `Cal-1` (adiabatic evolution test). We indicate the mean R_{200} among the codes ($\sim$ 7.5 kpc at $z = 7$) with a black dashed circle. Units are proper kpc. See Section 5.1 for more information on `Cal-1` and this figure, and Section 3.2 for descriptions of participating codes in this comparison. The full color version of this figure is available in the electronic edition. The high-resolution versions of this figure and article are available at the Project website, http://www.AGORAsimulations.org/. Simulations are performed by: Santi Roca-Fàbrega (ART-I, RAMSES), Ji-hoon Kim (ENZO), Johnny Powell and Héctor Velázquez (CHANGA), Kentaro Nagamine and Ikkoh Shimizu (GADGET-3), Loic Hausammann and Yves Revaz (GEAR), Alessandro Lupi and Bili Dong (GIZMO).

parameters that control the overall mesh structure and the aggressiveness of the refinement are left for each code group to decide (see Section 5 of Paper II). These differences can have an impact on the gas density and temperature distributions when without stellar feedback (as shown in Sections 5.1 and 5.2), but the impact becomes marginal once stellar feedback is activated (Section 5.4). Further analyses of such differences in the evolution of primordial gas at high z will be presented in future papers from the *AGORA* Collaboration.

Lastly, we recommend that each group stores simulation outputs at 200 epochs.[14] An explicit list of this *AGORA*-recommended output interval is publicly available, and can be used by anyone to compare their simulation with *AGORA*.

5. PHYSICS CALIBRATION STEPS

Before proceeding to generate the final cosmological simulations, all participating code groups have been asked to complete four rigorous calibration steps. The main objective of these calibrations is to reduce the number of free parameters and artifacts in each code that can have an impact on the evolution of simulated galaxies, that are not valid physical assumptions about the structure formation. By adding *one physical process at a time* into our cosmological zoom-in simulation, we seek a situation where all code groups converge to a final simulation with similar global properties (e.g., similar stellar mass) — and thus, any differences can only be attributed to the chosen stellar feedback prescriptions and intrinsic variations of the codes' numerics. We summarize the calibration procedure with a flowchart in Figure 1.

The first two calibration steps (hereafter `Cal-1` and `Cal-2`) are designed to first acquire qualitative convergence on the main gas properties, by calibrating the gas physics such as cooling and heating when star formation is not enabled. In the third calibration step (`Cal-3`), with star formation enabled, but the corresponding stellar feedback disabled, we look for agreement in the main gas properties and in the total stellar mass produced at $z = 7$. Finally, in the fourth step (`Cal-4`), we activate stellar feedback and aim to achieve convergence only in the stellar mass at $z = 4$ to the values predicted by semi-empirical models. Each code group is asked to set the feedback prescriptions to be as close to the most used one in each code community as possible. This last calibration step is a groundwork from which we can study how galactic properties depend on feedback prescriptions.

An important result of our set of calibrations is that the simulation parameters selected in an isolated disk test (Paper II) cannot be naively used in the cosmological simulations like the ones presented here. Gas properties (e.g., metallicity) and the external radiation field rapidly evolve with redshift, which has a strong impact on gas cooling and, thus, star formation. Furthermore, continuous acquisition of fresh gas from the intergalactic medium (IGM) and circumgalactic medium (CGM) makes the cosmological run substantially more complex than that of an isolated disk galaxy.

In this section, we carefully describe the four calibration steps one by one. We start each subsection by explaining its setup, and then go through the important findings and conclusions from each step. One could consider each of our calibration steps as a standalone comparison in itself. Nevertheless,

---

[14] 200 epochs starting from $a = 0.062$ ($z \sim 15$) to $a = 0.325$ ($z \sim 2$), equally spaced in log(a) with $\Delta\log(a) = |\log(384/2013)/200|$, plus a set of redshift snapshots at $z = 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2$. Downloadable at http://physics.snu.ac.kr/cosmo/agora/output_z_cosmorun.txt.

when successively executed and combined with other steps, our calibration procedure provides a solid ground on which advanced cosmological simulations could be performed and trusted. For example, new code groups may test their code's compatibility with the other contemporary codes, by following the common initial conditions, the common physics package, and the calibration steps proposed herein.

5.1. *Calibration Step One* (`Cal-1`)*: Adiabatic Evolution of Gas*

The first calibration step we undertake (`Cal-1`) is designed to detect inter-platform variations in the temperature and density of the accreted gas at $z = 7$ when no radiative process or subgrid physics is present. Each cosmological run has been performed without any radiative cooling processes or heating sources, or any subgrid models such as star formation or the pressure floor. Under such conditions, the system exchanges no energy with its surroundings, and is considered adiabatic. The system's entropy, however, is not necessarily constant as it may increase owing to the presence of shocks. If so, any variation between the codes is in principle caused only by the differences in hydrodynamics solvers — namely, how each code solves the conservation laws of fluid dynamics and how shocks, e.g., in the accreting gas, are captured and treated. Despite small differences described below, an overall convergence has been found among the seven participating simulation codes.

5.1.1. *Findings From* `Cal-1`

In Figure 2 we show the projected density (top row) and temperature (bottom row) from `Cal-1` at $z = 7$. The virial radius, defined as R_{200}, is approximately 7.5 kpc at $z = 7$ across all the codes (see Table 2), shown as black dashed circles. In Figure 2 and similar projection images hereafter, particle-based codes are smoothed using a spline kernel in `yt`.[15] However, these codes are not smoothed in other types of figures and analysis in this paper. Meanwhile, in Figure 3, we show the density-temperature probability distribution function (PDF) at the same epoch for all the gas within 100 kpc from the center of the main progenitor. Because the virial radius of the target halo at this redshift is ~ 7.5 kpc, we are showing a volume that includes gas not only in the galaxy, but also inside filaments, sheets, knots, and voids.

Overall, the large-scale density structures in all seven panels of Figure 2 are remarkably similar with one another, and multiphase density-temperature structures in Figure 3 are also comparable. Unsurprisingly, in both plots, the convergence is very good qualitatively for the particle-based codes,

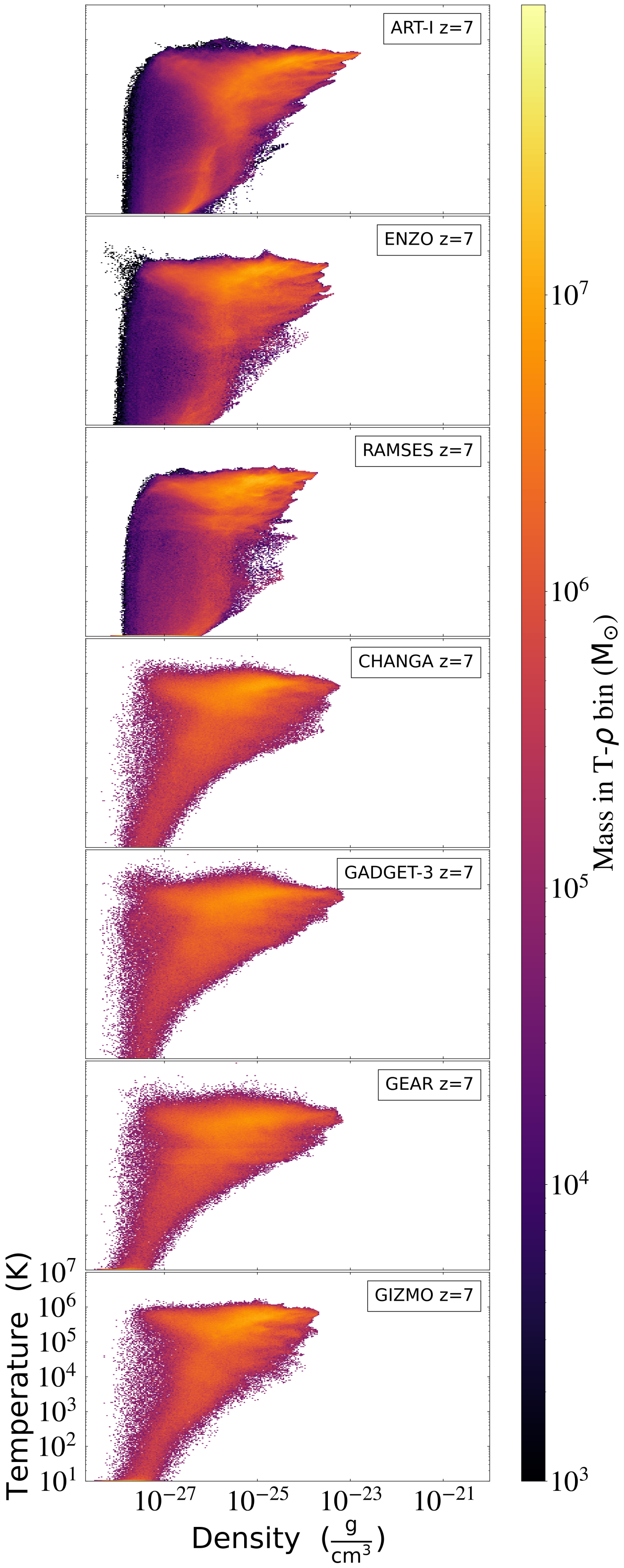


Figure 3. The $z = 7$ composite of 2-dimensional probability distribution function (PDF) of density and temperature for the gas within 100 kpc from the center of the main galactic system in the `Cal-1` runs. The 100 kpc-radius sphere encloses the main galaxy, the CGM, and the nearby IGM. Colors represent the total gas mass in each 2-dimensional bin. In all analyses for particle-based codes hereafter — except the graphical visualizations such as Figures 2 or 10 — raw particle fields are used, not the smoothed fields built by `yt`. See Section 5.1 for more information on `Cal-1` and this figure.

---

[15] We employ `yt-v4.0` which better handles the SPH particles, an improvement from `yt-v3.3` used in Paper II. See how `yt-v4.0`'s handling of SPH particles differs from that of its predecessors at https://matthewturk.github.io/yt4-gallery/.

CHANGA, GADGET-3, GEAR, and GIZMO as they share gravity solvers and take similar SPH approaches. The three mesh-based codes, ART-I, ENZO, and RAMSES, show minor differences but an overall agreement, too. Larger discrepancies are observed when comparing particle-based codes with mesh-based codes. In particular, the differences in the resolved structures in low-density regions at high redshift were discussed in the previous *AGORA* comparison with dark matter-only simulations. It is because particle-based codes achieve better resolution at early times than the mesh-based codes assuming little or no adaptive refinement for the mesh-based codes at high z (for detailed discussion, see Section 5.3.2 of Paper I). We also notice in Figure 3 that the highest densities that each code reaches are somewhat different, particularly among the mesh-based codes. This is due to differences in the refinement strategies adopted in each code, and we plan to study this issue further in future publications.

5.1.2. *Comments On The Differences In The Warm-Hot Intergalactic Medium In* `Cal-1`

From Figure 2, one however notices some discrepancies in the temperature maps. While all codes reproduce the virialized hot gas expected around massive haloes, with temperature between $10^5 - 10^6$ K, it is clear from Figure 3 that the extension of this hot component to lower densities — the warm gas that surrounds the main galactic systems — slightly differs. In particular, in ART-I, the intergalactic warm gas extends only up to the virial radius indicated in Figure 2 by the dashed black circles, while it extends beyond the virial radius and encompasses more mass in CHANGA, GADGET-3, GEAR, and GIZMO.

The effects that accretion shocks have over the warm gas around the main galactic systems could be different between codes, as they can be caused by small differences in numerical techniques. This phenomenon has been documented by many authors: *(1)* Gas could be overheated via collisional heating with dark matter particles due to differences in gravity solvers, integrators, timestepping strategies for force calculations, and refinement strategies (Springel 2010; Lukić et al. 2014; Jia et al. 2020). *(2)* Gas could be overheated also by the artificial viscosity in the sharp accretion shocks in particle-based codes (Scannapieco et al. 2012; Taylor & Miller 2012; Hosono et al. 2016). *(3)* Gas could be overcooled in the accretion shocks due to low resolution in the insufficiently refined CGM (Hubber et al. 2013). Although here we present the first analysis, this will be better characterized in a future paper from the Collaboration.

5.2. *Calibration Step Two* (`Cal-2`)*: Cooling and Heating of Gas By Common Physics Package*

The second calibration step (`Cal-2`) is designed to check if the common physics package (i.e., cooling, heating, UVB) by GRACKLE-V3.1.1 is properly interfaced in all the codes

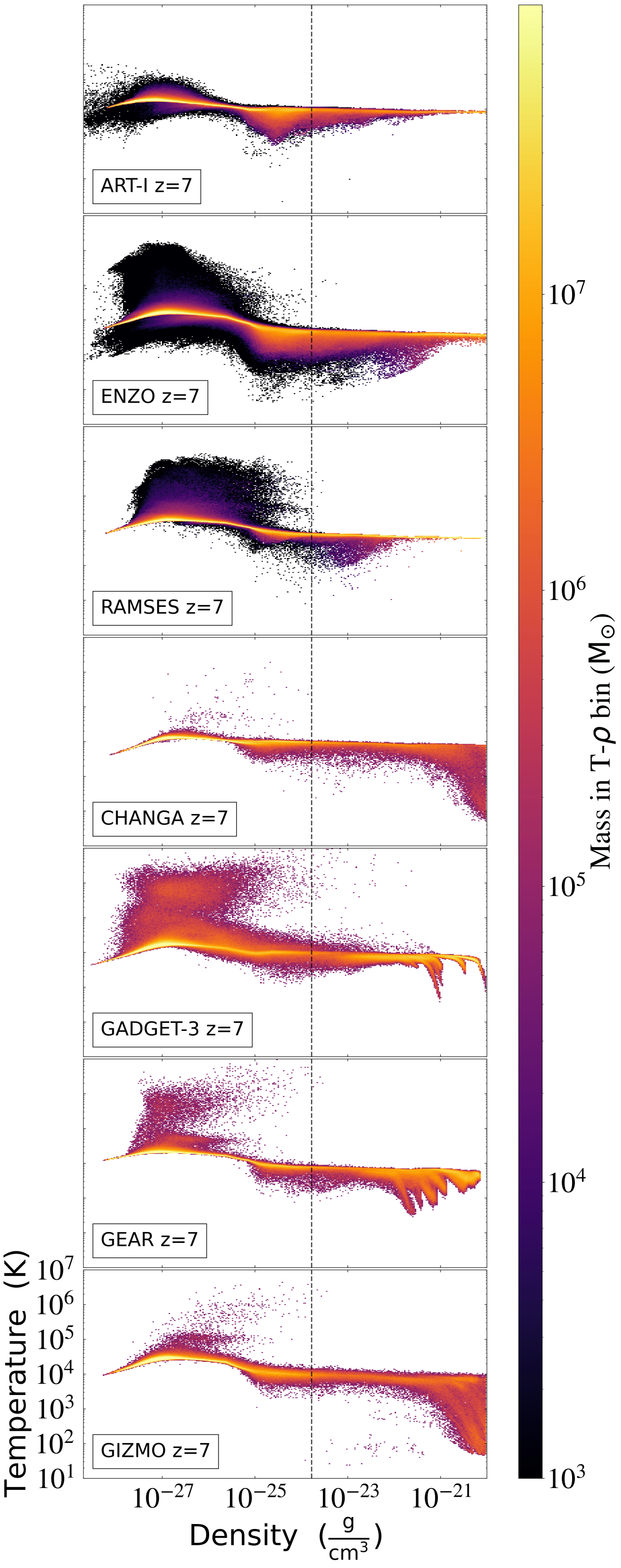


Figure 4. The $z = 7$ composite of 2-dimensional PDF of density and temperature for the gas within 100 kpc from the center of the main galactic system in the `Cal-2` runs (cooling and heating test). The 100 kpc-radius sphere encloses the main galaxy, the CGM, and the nearby IGM. Colors represent the total gas mass in each 2-dimensional bin. A black dashed vertical line is placed at the value of the star formation density threshold (Section 3.1) to be later adopted in the final simulations in Section 6. See Section 5.2 for more information on `Cal-2` and this figure.

for cosmological runs. Here, each run is performed with Grackle-v3.1.1 but *without* any subgrid models such as a pressure floor, star formation, or feedback. This approach allows us to check the agreement on the gas distribution in the density-temperature plane (expected when the radiative gas physics is treated via the common package Grackle-v3.1.1), and if all codes use the same initial metallicity.

5.2.1. *Findings From* `Cal-2`

`Cal-2` has turned out to be a critical calibration step during which the participant code groups found and fixed problems in their Grackle-v3.1.1 interface.[16] Note that an earlier version of Grackle was implemented and tested for an isolated galaxy disk simulation for all codes (see Section 3.1 of Paper II), but not for a fully cosmological zoom-in run with an expanding simulation volume.

The gas mass distribution from `Cal-2` in the density-temperature plane, is shown in Figure 4 at $z = 7$. Since the virial radius of the target progenitor at $z = 7$ is ~ 7.5 kpc (see Table 2), and we include all the gas inside a sphere of 100 kpc centered on the main halo, the plot includes not just the galactic gas, but most of the IGM inside the Lagrangian zoom-in region. Above $\sim 10^4$ K, the gas cools extremely efficiently owing to both hydrogen and helium recombination. Below $\sim 10^4$ K, however, the cooling of the low metallicity primordial gas (see Section 2) is very weak due to the absence of efficient cooling channels other than primordial molecules. On the other hand, the low-density gas is strongly heated by the UV background up to $\sim 10^4$ K, while at higher density above the UV self-shielding limit, it is heated by adiabatic compression. The combination of these effects leads to the bulk of the gas being found in a well-defined plateau at $\sim 10^4$ K, extending up to high densities ($10^{-20}\,\mathrm{g\,cm^{-3}}$).

Despite a general good agreement in reproducing this plateau, discrepancies between the participant codes have been noted. They reside primarily in the low-density, high-temperature gas in Figure 4. First, it is worth noting that mesh-based codes sample the low-density gas with a large number of bins with *small* mass per bin (blue bins) — which is hard to reproduce by particle-based codes with a (roughly) constant particle mass. In Figure 4, this leads to a large blue area at density $10^{-27} - 10^{-24}\,\mathrm{g\,cm^{-3}}$, above and below the 10^4 K plateau. This area is absent in the particle-based simulations. Second, a discrepancy exists in the prediction of the rarefied and shocked gas surrounding the halo and filaments. While particle-based codes predict the presence of the virialized hot gas at 10^{5-6} K (low-density, high-temperature gas around [$\sim 10^{-27}\,\mathrm{g\,cm^{-3}}$, $\sim 10^{5-6}$ K] in Figure 4, or a similar gas structure in Figure 6 or 10), it is almost absent in the mesh-based codes. We have carefully studied the behavior of this warm-hot gas, and found that the hot gas is outflowing, while the warm gas is inflowing, confirming that the warm gas surrounding the main galactic system contains shock-heated gas. While at this stage of our analysis, the exact origin of the temperature discrepancy between the codes remains unclear, we hypothesize that they result from the different hydrodynamic schemes adopted (differences in the hot virialized gas have already been mentioned in `Cal-1`), and in particular how the schemes treat shocks in strongly cooling gas phases.

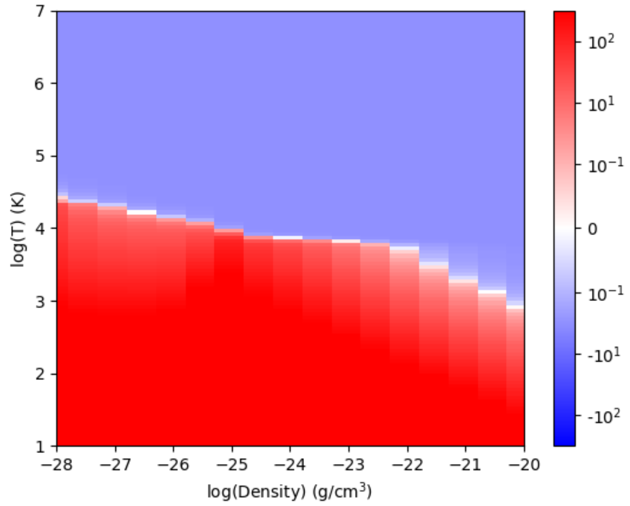


Figure 5. The density and temperature plane colored by the ratio of (heating rate − cooling rate)/(cooling rate) in each bin, obtained from the Cloudy table at $z \sim 7$ in Grackle-v3.1.1.

Finally, it is worth noting that although those discrepancies may look important, they typically disappear as soon as the stellar feedback is activated (Section 5.4). Since they have little impact on the star formation in our final `CosmoRun` simulations (Section 6), we have chosen to defer the detailed discussion to a future paper. The extensive studies on the differences in numerical approaches, and how they manifest themselves in the discrepancies in the warm gas surrounding the main galactic system will be in a forthcoming paper by the Collaboration (*AGORA* Collaboration et al., in prep.).

5.2.2. *Comments On The Cooling “Tails” At High Density In* `Cal-2`

In Figure 4, a repeating pattern of cooling “tails” appears at high density ($\gtrsim 10^{-22}\,\mathrm{g\,cm^{-3}}$), especially in the particle-based codes Gadget-3 and Gear — although we have confirmed that these features also exist in Changa, Gizmo, and the mesh-based codes (e.g., Ramses) at other epochs. After carefully checking the physics in each of the partici-

---

[16] During our comparison study using an earlier version of Grackle, we found that a small correction on the cooling and heating rates was needed in the Grackle/Cloudy tables, to ensure correct gas evolution at high redshift. This issue has been addressed in the Grackle-v3.1.1 release.

pant codes, we have found that such features are caused by the cooling and heating tables in our common physics package GRACKLE-V3.1.1. To illustrate our finding, in Figure 5 we show the tabulated rates of primordial cooling and heating at $z \sim 7$ from our adopted CLOUDY table (see Section 3.1). Here, it is easy to notice how the pre-computed table is binned in density and temperature. Readers may notice a larger bin size in the density axis, and that the discrete jumps at high density ($\gtrsim 10^{-22}\,\mathrm{g\,cm^{-3}}$) in the cooling and heating rates exactly coincide with the cooling "tails" in Figure 4. We therefore conclude that the observed cooling "tails" originate from the density binning in the pre-computed CLOUDY table; and, the differences among the participating codes are due to variations in how exactly each code's cooling and heating solver interfaces with GRACKLE-V3.1.1, and its interpolation scheme.

While the cooling "tails" are an interesting observation, we note that these artificial features have little impact on the final cosmological runs presented in Section 6, because they occur at densities much higher than the star formation threshold, $n_{\mathrm{H,thres}} = 1\,\mathrm{cm}^{-3}$, where, in addition, the pressure is dominated by the artificial pressure floor (Section 3.1). The features start to disappear once the dense gas is consumed by stars at later times (Section 5.3), and will completely vanish as soon as stellar feedback and the pressure floor are activated (Section 5.4).[17]

5.3. *Calibration Step Three* (`Cal-3`)*: Common Star Formation Physics*

The third calibration step (`Cal-3`) is designed to detect and study the impact of any discrepancies in the implementation of the common star formation prescription (see Section 3.2). Each simulation has been carried out with GRACKLE-V3.1.1, common star formation and pressure floor prescriptions, but without any stellar feedback. The main objective of `Cal-3` is to ensure that our final cosmological simulation entries in Section 6 is not dominated by variations (or errors) in how the common star formation physics is implemented in each code. At the end of `Cal-3`, each code group confirms that the feedback-free simulations converge within 0.5 dex in stellar masses at $z = 7$, and in stellar mass growth history down to that point.

5.3.1. *Findings From* `Cal-3`

In Figure 6, we plot the 2-dimensional density-temperature PDF at $z = 7$. It displays a good agreement on the general features in the density-temperature plane, such as the shape

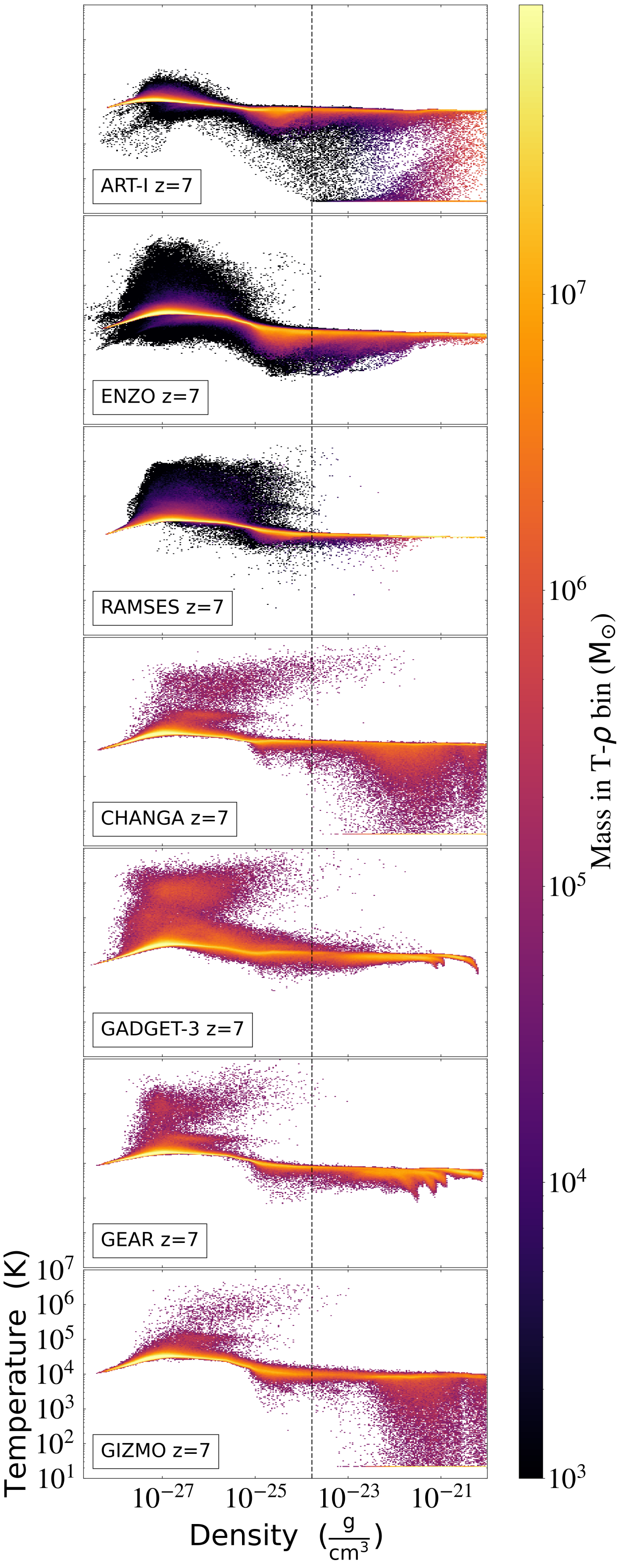


Figure 6. The $z = 7$ composite of 2-dimensional PDF of density and temperature for the gas within 100 kpc from the center of the main galactic system in the `Cal-3` runs (star formation test). The 100 kpc-radius sphere encloses the main galaxy, the CGM, and the nearby IGM. Colors represent the total gas mass in each 2-dimensional bin. A black dashed vertical line marks the density threshold for star formation. See Section 5.3 for more information on `Cal-3` and this figure.

---

[17] Although this feature does not affect the final simulations presented herein, we caution the GRACKLE users when they use the default CLOUDY tables provided with the package. A new table with smaller density bins and/or a careful interpolation scheme would be needed, if interested in studying the very dense gas when no star formation is present.

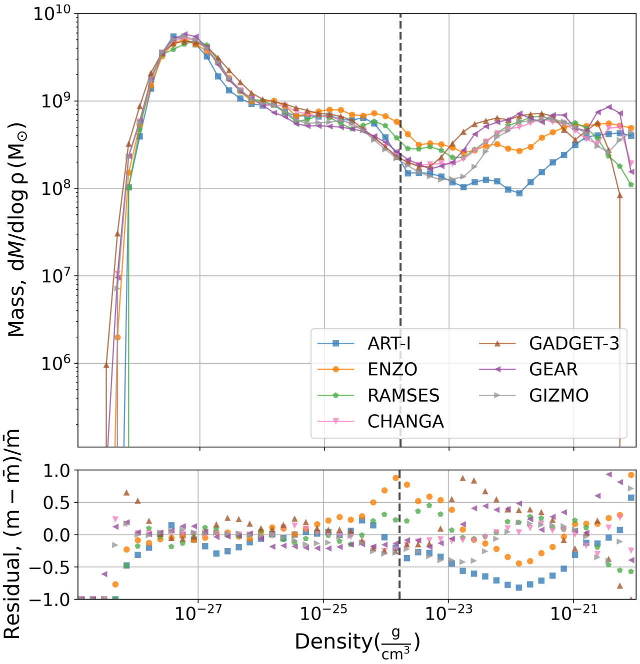


Figure 7. Distribution of gas mass as a function of gas density at $z = 7$ for all the gas inside the target progenitor's mean R_{200} ($\sim$ 7.5 kpc at $z = 7$) in `Cal-3`. The vertical dashed line denotes the star formation threshold, $n_{\rm H,\,thres} = 1\,{\rm cm}^{-3}$. Shown in the bottom panel is the fractional deviation from the mean of these profiles. See Section 5.3 for more information on `Cal-3` and this figure.

of the $\sim 10^4$ K cooling plateau where most of the gas mass resides. Nevertheless, there are differences, some of which were discussed in previous sections — e.g., a large number of bins with small mass in the low-density, high-temperature region (blue bins; Section 5.2.1), and the cooling "tails" at high density (Section 5.2.2). An interesting new discrepancy in Figure 6 is the presence of high-density, low-temperature gas found in ART-I, CHANGA and GIZMO, with its density near the star formation threshold and its temperature near the CMB floor. This artificial feature results from using a stochastic star formation recipe and a particular pressure floor implementation (see Sections 3.2.1 and 3.2.4);[18] however, the discrepancy becomes largely marginal once stellar feedback is turned on as we will discuss it in Section 5.4 and Figure 14.

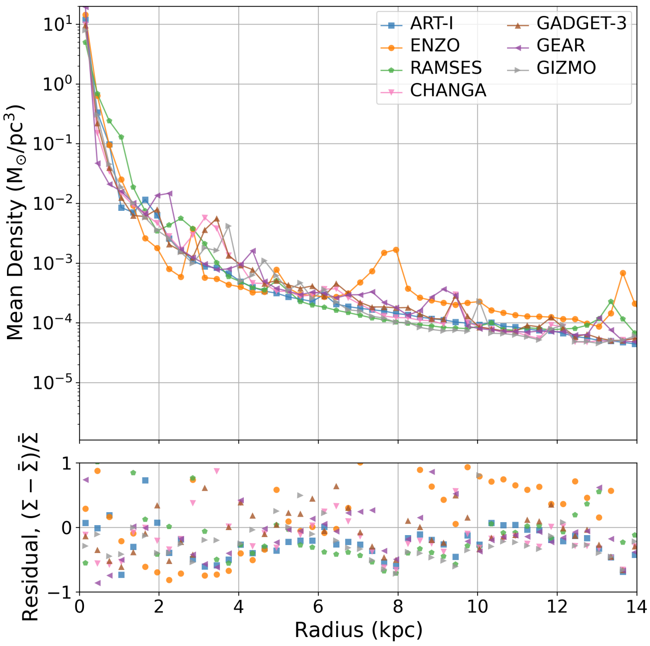


Figure 8. Spherically-averaged gas density profiles as functions of distance from the galactic center at $z = 7$ for the `Cal-3` runs. Shown in the bottom panel is the fractional deviation from the mean of these profiles. See Section 5.3 for more information about how the center of the system is selected, the `Cal-3` runs, and this figure.

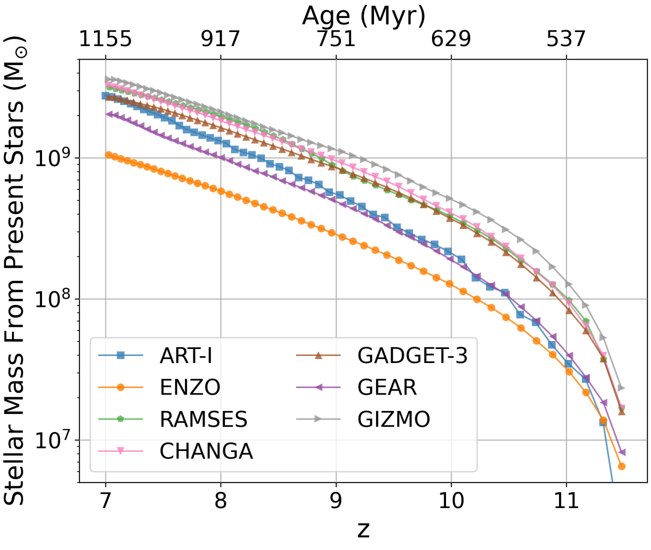


Figure 9. Stellar mass growth histories for the `Cal-3` runs in a 100 kpc sphere centered at the target progenitor. The curve is computed using the ages or creation times recorded in star particles at $z = 7$.

Nevertheless, on the whole, the `Cal-3` entries from the participating code groups exhibits robust overall convergence in the gas distribution around the target progenitor galaxy, as illustrated in Figures 7 to 9. In Figure 7, we display the gas mass distribution as a function of its density, including all

---

[18] The ART-I code, for example, uses stochastic star formation along with a treatment to avoid complete gas depletion in a star-forming gas cell (see Section 3.2.1). Hence, after a cell spawns a star particle, a fraction of gas is still left in the cell with the same temperature as before but with a significantly lowered density. Due to the imposed pressure floor, the equilibrium with the surrounding cells can only be achieved through rapid cooling, and a slightly increase on the density. This process results in a build-up of the observed cold gas near the CMB floor. Similar features have been reproduced in other codes (e.g., RAMSES) when stochastic star formation is employed.

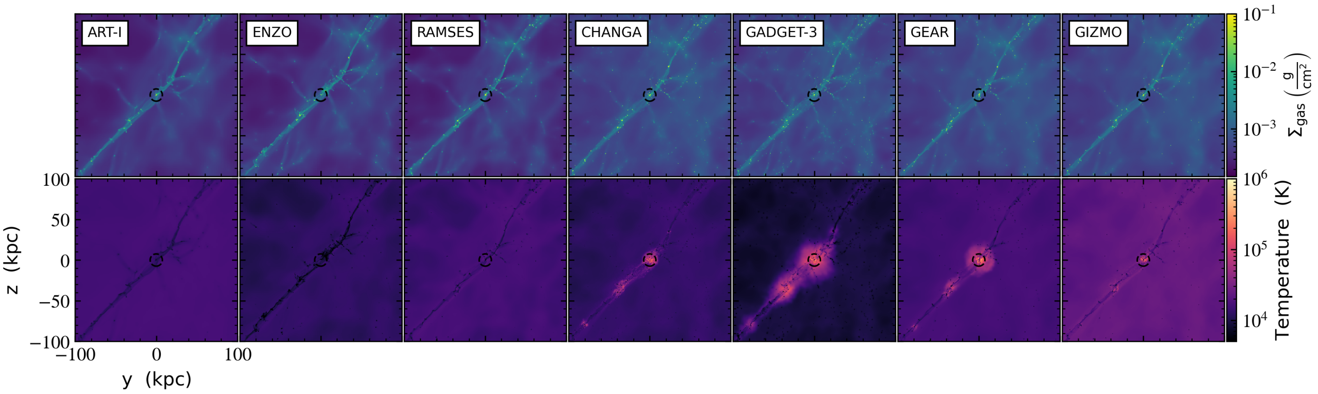


Figure 10. Gas density projection (*top*) and density-weighted temperature projection (*bottom*) at $z = 7$ from the third calibration step, `Cal-3` (star formation test). We indicate the mean R_{200} among the codes (~ 7.5 kpc) with a black dashed circle. Units are proper kpc. The projections along the other axes are available as digital supplements to this article. See Section 5.3 for more information on `Cal-3` and this figure.

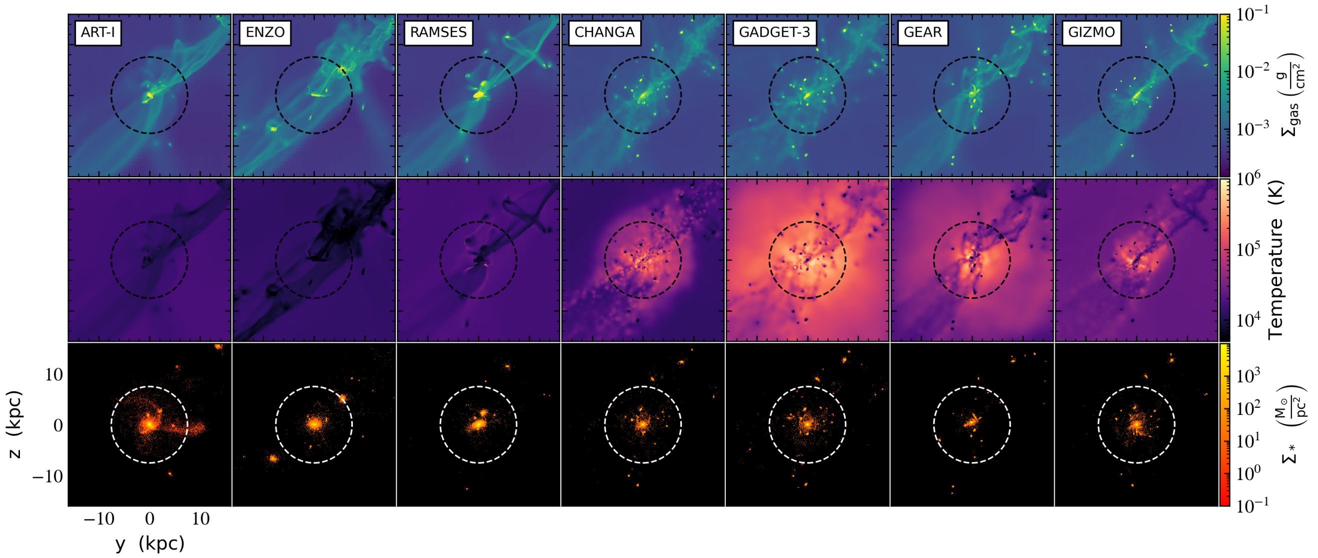


Figure 11. Similar to Figure 10, but now in zoomed-in regions. Gas density projection (*top*), density-weighted temperature projection (*middle*), and stellar surface density (*bottom*) at $z = 7$ from the third calibration step, `Cal-3`. The width of each panel is $4R_{200} = 30$ kpc. The mean R_{200} among the codes (~ 7.5 kpc) is indicated with a black/white dashed circle. See Section 5.3 for more information on `Cal-3` and this figure.

the gas inside the virial radius R_{200} (~ 7.5 kpc at $z = 7$). We find that all participant codes produce a very similar gas density probability distribution inside R_{200}. Note that the convergence is better than in our disk comparison (Figure 18 of Paper II) in which, by design, gaseous halos — low-density tails towards the left side of this plot — existed only in mesh-based codes, but not in particle-based codes. In Figure 8, we show the spherically-averaged gas density as a function of radius, again demonstrating solid convergence aside from small variations due to the halo substructures and clumps.[19] In both Figures 7 and 8 we include the fractional deviation from the mean of these profiles to better illustrate the convergence among the codes.

The most relevant result from `Cal-3` is, however, the convergence in the stellar mass $M_\star$ evolution (in a 100 kpc sphere centered at the target progenitor) in Figure 9. Though small variations exist, all codes follow similar stellar mass growth histories, within half a dex from one another at all times. Differences among codes are due to variations in how the common star formation prescription is implemented (e.g., stochastic in ART-I, CHANGA, GADGET-3, GEAR vs. deterministic in ENZO and RAMSES; see Section 3.1), refinement

---

[19] The profile center is set to be the location of maximum stellar density within a successively shrinking distance from the dark matter center of mass.

strategy (Section 4), and/or numerical accuracies of hydrodynamics solvers (Section 5 of Paper II).[20]

5.3.2. *Comments On The Differences In Galactic Morphology In* `Cal-3`

Finally, a detailed comparison of the gas and stellar distribution in real space is shown in Figure 10 and 11. In Figure 10 we show the projected gas density (top row) and temperature (bottom row) of all the gas inside the $(200\,\mathrm{kpc})^3$ volume (compare with Figure 2 in `Cal-1`). The mean virial radius R_{200} among the codes is shown as a black dashed circle. In the gas density map, the large-scale structures are nearly identical across all participant codes, although the aforementioned differences in the low-density region (discussed in Section 5.1.1 and 5.2.1 with `Cal-1` and `Cal-2`, respectively) still exist between the mesh-based and particle-based approaches. Figure 11 demonstrates this more dramatically, in which we show the projected gas density (top row), temperature (middle row), and stellar surface density (bottom row) at $z = 7$ inside a $(4R_{200})^3$ volume. Notable is that, in the stellar surface density map, the particle-based codes harbor more satellites (clumps of star particles) than the mesh-based codes. This discrepancy is caused by the same effect that leads particle-based codes to preserve more substructures in the low-density region. It has been well documented that due to the lack of force resolution at high z, mesh-based codes tend to suppress the low-mass end of the halo mass function (see Section 5.1.1 of this article, or Section 5.3.2 in Paper II).

Also in Figure 11, differences exist in the temperature map between the mesh-based and particle-based codes, particularly in the regions next to the galaxies and filaments. This difference manifests itself as a diverging distribution in the density-temperature PDF near $\sim 10^{-27}\,\mathrm{g\,cm^{-3}}$, $\sim 10^{5-6}$ K in Figure 6. We recall, however, from Section 5.2.1 that the observed temperature differences become irrelevant as soon as the stellar feedback is activated, and thus have little impact on the results of the final zoom-in cosmological runs (`CosmoRun`) in Section 6.

5.4. *Calibration Step Four* (`Cal-4`)*: "Favorite" Stellar Feedback Prescription By Each Code*

The objective of this last calibration step (`Cal-4`) is to get convergence on the stellar mass of the main progenitor at $z = 4$ within 0.5 dex, to the value predicted by the semi-empirical models based on the abundance matching techniques (e.g., Rodríguez-Puebla et al. 2017). The main motivation for `Cal-4` is to come up with a *realistic* simulation resembling observed galaxies, by adopting each code group's "favorite" feedback — as close to the most widely-used one for research in each code community. Each code group's cosmological simulation has been carried out with GRACKLE-v3.1.1, a common star formation prescription, and its own choice of stellar feedback and metal production (see Table 1 and Section 3.2). Each group has been asked to provide a reference with detailed information on their "favorite" feedback prescription (as in Section 3.2). Although time-consuming, at the end of `Cal-4` we establish a common ground based on which we can compare the effects of each group's "favorite" feedback on the evolution of galaxies and CGM.

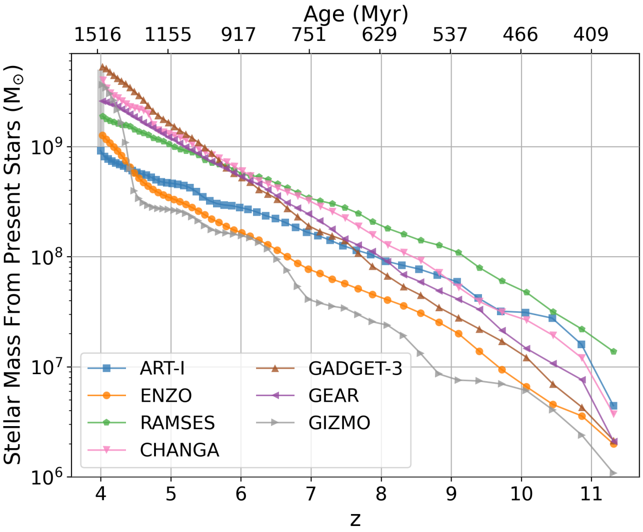


Figure 12. Stellar mass growth histories for the `Cal-4` runs inside a R_{200} sphere centered at the target progenitor. The curve is computed using the ages or creation times recorded in star particles at $z = 4$. The stellar mass range at $z = 4$ targeted in our calibration is $M_\star \sim 1-5\times10^9\,\mathrm{M_\odot}$, as motivated by semi-empirical models. What we show here is an upper limit for the total $M_\star$ formed inside R_{200}. It is in Figure 13 where we can make a fair comparison of $M_\star$ formed inside the galaxy with predictions from semi-empirical models. See Section 5.4 for more information on `Cal-4` and this figure.

5.4.1. *Calibration Target In* `Cal-4`

According to the predictions by the aforementioned semi-empirical models, the expected stellar mass inside the main galactic system of a $M_{200} = 2\times10^{11}\,\mathrm{M_\odot}$ halo at $z = 4$ is $\sim 1-1.5\times10^9\,\mathrm{M_\odot}$. Since our selected halo (see Section 2) experiences a relatively violent assembly history by $z = 4$, we have extended the target range of the stellar mass $M_\star$ to $\sim 1-5\times10^9\,\mathrm{M_\odot}$ at $z = 4$. The width of the target mass range is to allow flexibility when each code group selects its stellar feedback scheme. `Cal-4` has required the most amount of time among all calibration steps. Typically, the process was not over with a single simulation, but required several iterations carried out by each participating code group. The sim-

---

[20] Note that ENZO produces 2-3 times fewer stars than the other codes. Unlike the other codes, the only tunable parameters in ENZO's star formation module is the star formation efficiency and the density threshold, both being fixed in this work (see Section 3.1). Thus, it has been difficult to further adjust ENZO's star formation to acquire better convergence.

ulations they acquire after these iterations become the final entries in Section 6 (dubbed `CosmoRun`). In this subsection we briefly discuss only the calibration process in `Cal-4`, not the detailed analysis of each code group's final simulation entry — the latter will be discussed in full detail in Section 6.

5.4.2. *Findings From* `Cal-4`

At the end of `Cal-4`, the participating code groups have found a need to use stronger stellar feedback than they commonly used in their communities in order to achieve the target stellar mass at $z = 4$. However, none of them used unrealistic feedback parameters. In Figure 12 we show the stellar mass growth histories of final simulation entries. Each curve has been obtained using the star particles residing inside a R_{200} sphere centered on the target progenitor galaxy at $z = 4$. Therefore, Figure 12 is the stellar mass assembly history (SMAH) inside R_{200}, not the star formation history (SFH) of the main galactic system, thus it is only an upper limit for the generated stellar mass.[21] The plot demonstrates how all codes successfully converge to the agreed $M_\star$ range, although the SPH codes tend to have higher $M_\star$ at $z = 4$. Comparing Figure 12 with Figure 9, in each code we observe the expected decrease of the stellar mass growth due to the stellar feedback (notice the change in the y-axis). The shape of the SMAH differs from one code to another because of the different stellar feedback prescriptions implemented in the codes, that can affect star formation differently at a given epoch. The "timing discrepancies" among the codes in the halo assembly history could also cause differences in the SMAHs. Indeed, the exact timing of a major merger occurring at $z \sim 4$ could precipitate sizable variations in the SMAH, and the gas and stellar properties discussed in Section 6 (see Sections 6.2 and 6.3 for more discussion).[22] Lastly, readers may notice that the inter-code differences are larger at early times (e.g., the variation is ~ 1.5 dex at $z = 10$ but ~ 0.5 dex at $z = 4$). Indeed, previous research have found that different stellar feedback implementations can exacerbate the discrepancy at high redshift (e.g., Hayward & Hopkins 2017).

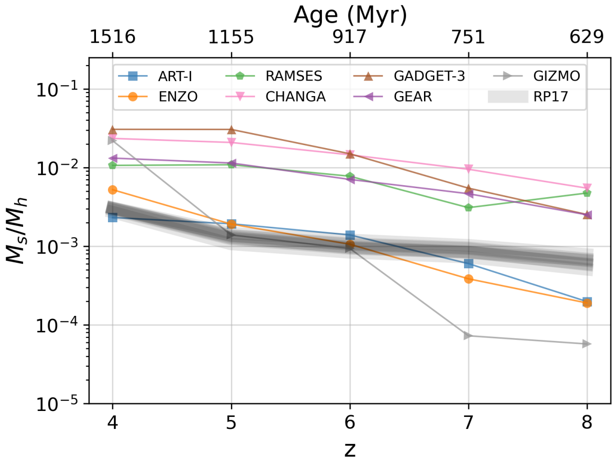


Figure 13. Evolution of the stellar-to-halo mass ratio, $M_{\star,\mathrm{gal}}/M_{200}$, from $z = 8$ to $z = 4$ in the `CosmoRun` simulations (rightmost column in Table 2). Gray shadowed regions indicate the predicted ranges of the ratio by the semi-empirical model of Rodríguez-Puebla et al. (2017), obtained using the halo mass at each redshift, in each simulation. See Section 6 for more information on `CosmoRun`, and Section 6.1 in particular on this figure.

With this final result, we conclude the entire calibration procedure. The code groups that completed the four calibration steps, `Cal-1` to `Cal-4`, have obtained the final `CosmoRun` simulations. In the next Section, we present and analyze the properties of these final simulation entries from the codes groups down to $z = 4$.

6. THE *AGORA* `CosmoRun` SIMULATIONS

In this section, we introduce the *AGORA* `CosmoRun` simulations acquired from the rigorous calibration steps in Section 5. As we present the analysis of their stellar and gas components, we focus on five redshifts, $z = 8, 7, 6, 5$ and 4.[23] The simulations have been running down to even lower redshift, and the full analysis — the CGM evolution down to e.g., $z = 2$, in particular — will be presented in the forthcoming papers from the *AGORA* Collaboration.

6.1. *Global Properties of The Target Galaxy Progenitor*

We start by analyzing the global bulk properties of the target galaxy progenitor in `CosmoRun`. In Table 2 we list the total virial mass, M_{200}, and gas and stellar masses enclosed inside a sphere whose radius is the mean R_{200} among the codes. We also include the gas masses inside the main galaxy vs. those in the CGM (i.e., $M_{\mathrm{gas,gal}}$ for $R < 0.15R_{200}$ vs. $M_{\mathrm{gas,CGM}}$ for $0.15R_{200} < R < R_{200}$), and the stellar-to-halo mass ratio, $M_{\star,\mathrm{gal}}/M_{200}$, obtained by using the star particles inside $0.15R_{200}$ (rightmost column in Table 2; see

---

[21] Unlike the SFH, the SMAH includes not only the stars formed inside the target progenitor (in-situ), but also the stars formed outside and brought in by e.g., merging satellites (ex-situ). In the SMAH, the stellar mass may decrease due to the mass loss when the galaxy interacts with its neighbors. In future studies, we plan to compare the actual SFH (rather than SMAH).

[22] The discrepancies in the exact timings of mergers and star formation events, could affect the discussion of various galactic properties in Section 6. In particular, at high z, major mergers are common and can violently disturb the gas inside the galaxy and in its CGM by generating shocks and changing the gas distribution in the density-temperature plane. These perturbative events do not occur at the exact same redshift in all codes (see Section 5.3.2 of Paper I), complicating the inter-code comparison. In the future papers, we will extensively study variations in the participating codes' merger trees.

[23] 1.09, 1.22, 1.40, 1.63, 1.96 Gyr in cosmic time, respectively.

Table 2. Global properties of the target galaxy progenitor in the *AGORA* `CosmoRun` simulation suite

Code	redshift z	$M_{200}^{(a)\dagger}$ [$10^{10}\,\mathrm{M}_\odot$]	$M_\star^{(b)}$ [$10^8\,\mathrm{M}_\odot$]	$M_{\rm gas}^{(c)}$ [$10^8\,\mathrm{M}_\odot$]	$M_{\rm gas,\,gal}^{(d)}$ [$10^8\,\mathrm{M}_\odot$]	$M_{\rm gas,\,CGM}^{(e)}$ [$10^8\,\mathrm{M}_\odot$]	$\log(M_{\star,\,\rm gal}/M_{200})^{(f)}$
ART-I	8	0.92	0.48	11.36	0.19	11.18	-3.7
	7	1.49	1.04	14.87	0.38	14.50	-3.22
	6	1.83	1.52	17.80	0.56	17.24	-2.86
	5	2.77	1.98	28.50	1.29	27.21	-2.71
	4	13.23	9.22	145.41	21.68	123.72	-2.64
ENZO	8	1.16	0.23	11.03	0.17	10.86	-3.72
	7	1.84	0.43	22.37	0.83	21.54	-3.41
	6	2.26	0.96	30.05	1.58	28.46	-2.97
	5	3.84	2.04	51.41	3.67	47.74	-2.72
	4	16.04	12.72	242.62	58.39	184.23	-2.28
RAMSES	8	1.37	1.21	17.73	2.97	14.75	-2.32
	7	1.84	1.67	19.85	1.51	18.35	-2.51
	6	2.19	2.87	26.59	5.12	21.48	-2.11
	5	3.50	5.12	36.51	10.43	26.08	-1.96
	4	14.79	18.98	139.47	44.32	95.15	-1.97
CHANGA	8	1.43	1.17	29.03	5.94	23.37	-2.26
	7	2.26	2.82	43.22	7.55	35.67	-2.02
	6	2.72	5.09	58.88	17.91	40.97	-1.84
	5	4.15	10.89	72.74	11.76	60.98	-1.68
	4	15.81	39.94	203.04	85.70	117.34	-1.63
GADGET-3	8	1.32	0.48	25.16	5.62	19.54	-2.60
	7	2.17	1.47	38.84	7.41	31.43	-2.26
	6	2.61	4.23	49.25	18.06	31.20	-1.82
	5	4.05	12.75	71.65	26.46	45.20	-1.52
	4	16.15	53.17	216.98	76.24	140.74	-1.51
GEAR	8	1.72	0.67	39.52	8.28	31.24	-2.60
	7	2.52	1.55	58.84	15.51	43.33	-2.33
	6	3.23	3.71	82.14	14.93	67.21	-2.15
	5	4.60	7.77	111.38	40.51	70.87	-1.94
	4	16.34	25.92	286.33	145.52	140.81	-1.88
GIZMO	8	1.12	0.14	10.96	0.0	10.96	-4.24
	7	1.90	0.20	24.56	1.15	23.41	-4.14
	6	2.35	0.92	33.02	0.98	32.04	-3.03
	5	3.65	1.64	41.18	1.32	39.86	-2.86
	4	15.39	36.23	165.59	41.21	124.38	-1.66

† Each column lists the following quantities at the corresponding redshift: $^{(a)}$total halo mass, $^{(b)}$stellar mass, $^{(c)}$gas mass inside the mean R_{200} among codes, where the R_{200} values found are 5.8, 7.5, 8.4, 11.4 and 25.4 proper kpc at $z = 8, 7, 6, 5$ and 4, respectively, $^{(d)}$gas mass inside the main galaxy or the ISM (which we define as regions with $R < 0.15\,R_{200}$), $^{(e)}$gas mass in the CGM (which we define as regions with $0.15\,R_{200} < R < R_{200}$), $^{(f)}$the ratio of stellar mass (in the main galaxy) to halo mass.

also Figure 13). It should be noted that we do not expect to find perfect convergence in all the properties here, but expect substantial dependence on the stellar feedback prescriptions adopted by each code group. This dependence will be especially evident in the spatial distribution of gas in and around the target halo, and also in its temperature and metallicity.

Table 2 illustrates that all the participating codes converge on the stellar and total masses within < 0.5 dex from one another. This convergence is not surprising as it is a consequence of the calibration strategy used (`Cal-4`; Section 5.4.1). The small deviations from code to code in the total mass, M_{200}, are due to the "timing discrepancies" in the halo assembly history (Section 5.4.2 and footnote 22). On the other hand, relatively larger deviations in the gas mass inside the virial radius, $M_{\rm gas}$, or the ratio of gas masses in the main galaxy vs. in the CGM (i.e., $M_{\rm gas,\,gal}$ vs. $M_{\rm gas,\,CGM}$), are a direct consequence of the different stellar feedback strategies adopted. In fact, the strength of the outflows generated by stellar feedback has a strong impact not only on the amount of gas remaining inside the virial radius, but also on how efficiently the cold inflows replenish the galaxy with fresh gas. A detailed analysis of the thermodynamics and kinematics of gas is in Sections 6.2 and 6.4.

In Figure 13 we show the stellar-to-halo mass ratios, $M_{\star,\mathrm{gal}}/M_{200}$, in the `CosmoRun`, computed at $z = 8, 7, 6, 5$ and 4 (see also the rightmost column in Table 2), compared with predictions from semi-empirical models (e.g., Rodríguez-Puebla et al. 2017). The gray shadowed regions indicate the stellar-to-halo mass ratio obtained from a semi-empirical model using M_{200} at each redshift, in each simulation. Since in `Cal-4` we calibrated each simulation's stellar feedback so that the stellar mass produced is in the range of $\sim 1-5\times10^{9}\,\mathrm{M}_{\odot}$ at $z = 4$ (see Figure 12 and Section 5.4.1), all seven lines do not deviate more than one dexfrom one to another at $z = 4$. In addition, the difference between the simulated stellar-to-halo mass ratios and the semi-empirical predictions is less than 1 dex at $z = 4$, because it is designed as such in `Cal-4`. However, the semi-empirical predictions lie below the simulated values in most codes. The mismatch is because our target halo does not have an assembly history of a prototypical halo of $10^{12}\,\mathrm{M}_{\odot}$ at $z = 0$, but that of a halo which assembled early and had a quiescent period from $z = 2$ to 0 (Section 2). This bias yields a higher-than-expected stellar mass at $z \gtrsim 4$. At higher redshift ($z \gtrsim 7$), the differences among the simulated stellar-to-halo mass ratios, and that between the simulated ratios and the semi-empirical predictions are significantly larger. They are due to the variations in the feedback prescriptions, causing changes on the amount of star-forming gas available at each redshift, hence on the star formation history.

6.2. *Gas Properties*

Because deviations in stellar feedback are better reflected in gas, gas properties in simulations can be used to compare and calibrate the stellar feedback prescriptions employed. It is not in the scope of this paper to determine which stellar feedback in which code better fits the observations. Instead, we aim to show which gas properties are more sensitive to feedback, and to provide the community with a common ground to make new comparisons. In this subsection, we present only a general analysis of the gas properties. This first analysis is currently being extended and will be presented in a future paper focused on the evolution of the CGM.

6.2.1. *Gas Density and Temperature*

The first figure of this subsection, Figure 14, displays the gas density-temperature PDF, that can be compared with Figures 3, 4 and 6 from our calibration steps `Cal-1` to `Cal-3` (see Section 5). Note that, in this plot, we only show the gas inside R_{200} (see the caption of Table 2), while Figures 3, 4 and 6 include gas out to the IGM. From Figure 14, we see that, once the stellar feedback is activated, the convergence we always get is only the shape of the $\sim 10^{4}$ K cooling curve. Notable differences between the codes in Figure 14 include: *(1)* The blue bins with small mass per bin in the mesh-based

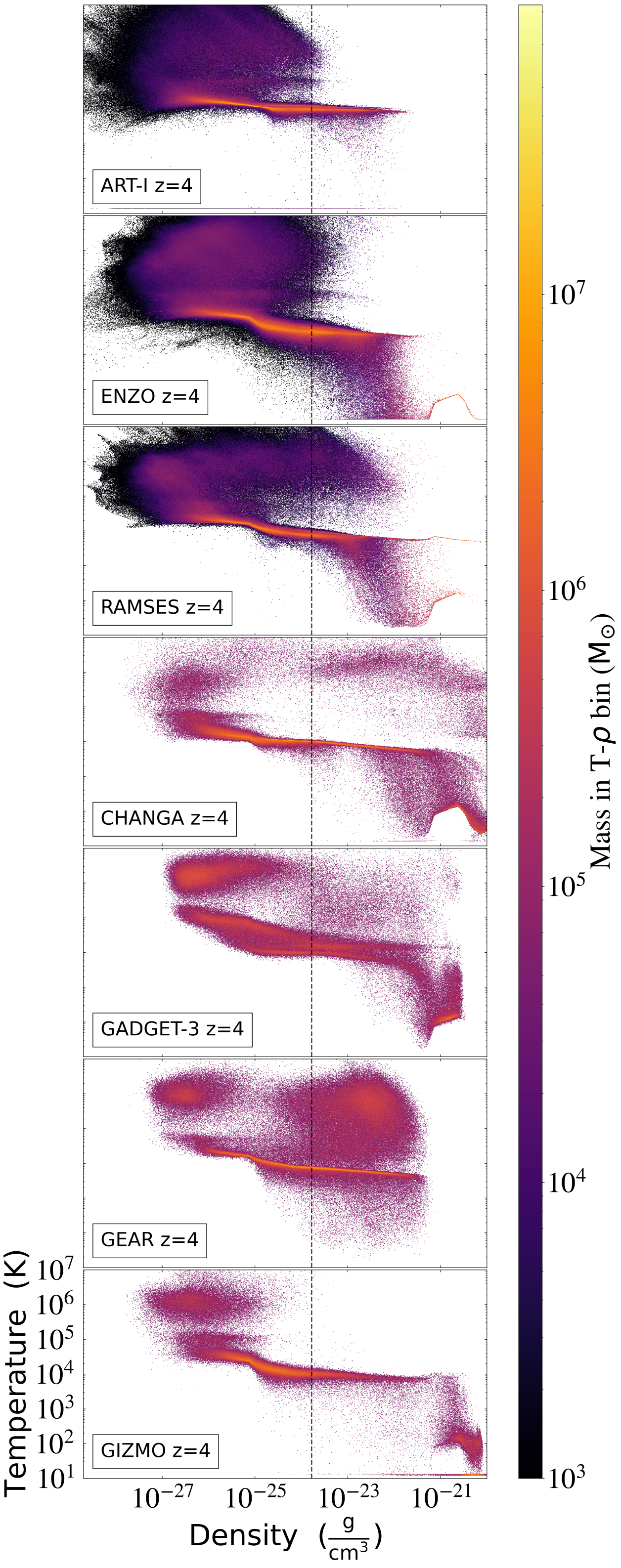


Figure 14. The $z = 4$ composite of 2-dimensional PDF of density and temperature for the gas within the mean R_{200} among the codes (~ 25.4 kpc) from the target galaxy's center in the `CosmoRun` simulations. It is similar to Figures 3, 4 and 6; but, unlike the previous figures, a sphere of R_{200} encloses the main galaxy and CGM, but not the IGM. Colors represent the total gas mass in each 2-dimensional bin. A black dashed vertical line marks the density threshold for star formation. See Section 6 for more information on `CosmoRun`, and Section 6.2 in particular on this figure.

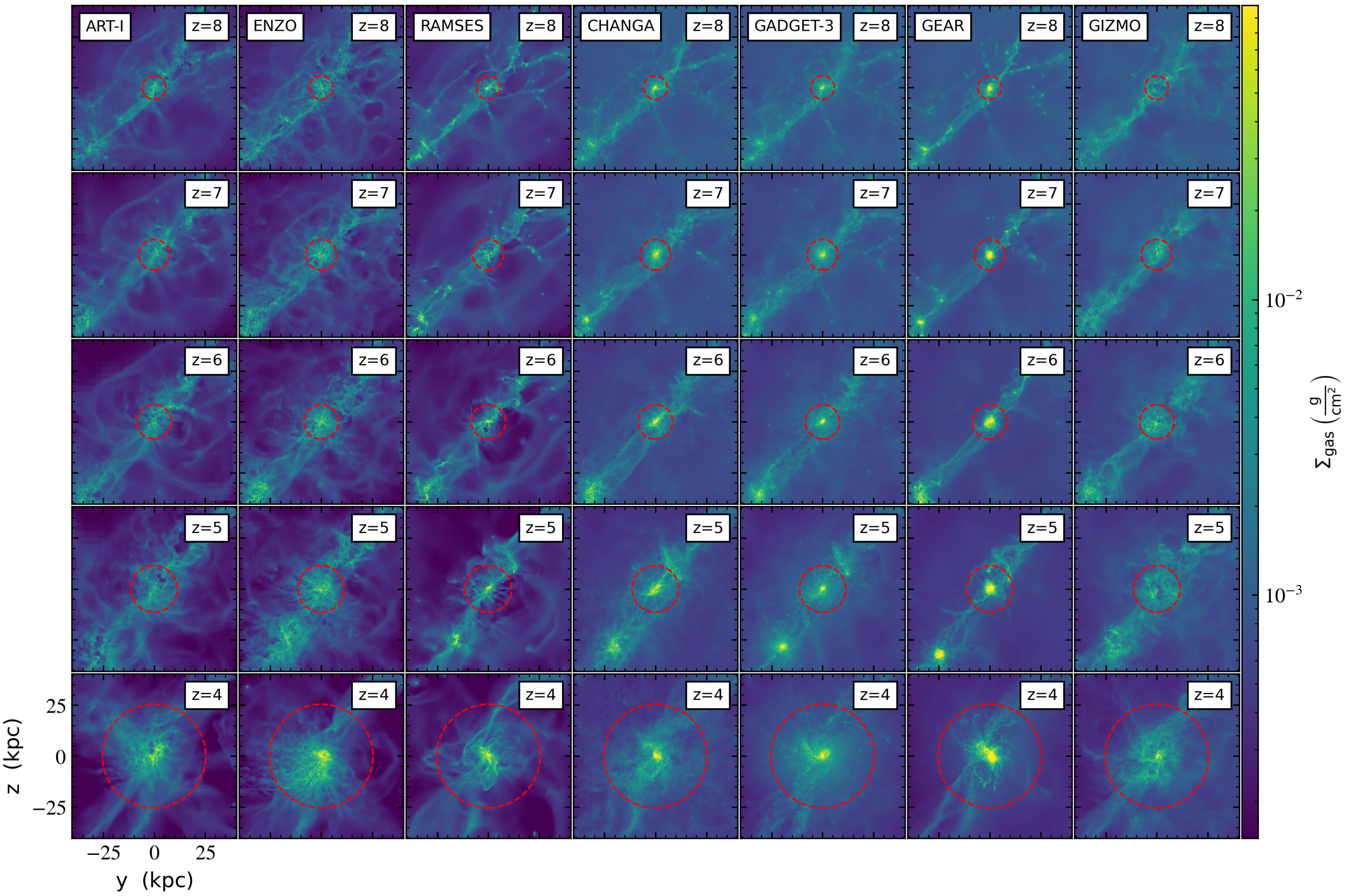


Figure 15. Gas surface densities at $z = 8$ to 4 from our final `CosmoRun` simulation suite, centered on the center of mass of stars and dark matter belonging to the target galaxy progenitor. Here and in the following figures we indicate the mean R_{200} among the codes at each redshift with a red dashed circle (5.8, 7.5, 8.4, 11.4 and 25.4 proper kpc at $z = 8$, 7, 6, 5 and 4, respectively). Units are proper kpc. The projections along the other axes are available as digital supplements to this article. See Section 6.2 for more information on `CosmoRun` and this figure.

codes reflecting very diffuse gas, that are not well represented in the particle-based codes (as discussed in Sections 5.2.1 and 5.3.1). *(2)* The total gas mass $M_{\rm gas}$ inside R_{200} changes significantly between codes due to the different stellar feedback strategies adopted (see Section 3.2) and the "timing discrepancies" (see Section 5.4.2 and footnote 22), for which a clear example appears when comparing the total $M_{\rm gas}$ (number of bins and colors) in R_{200} of e.g., ART-I and CHANGA (see also Figure 17). The exact timing of a major merger occurring at around $z \sim 4$ partly explains the discrepancy in the PDF between different codes. For example, while ART-I still undergoes the merger at $z = 4$, other codes already experienced it at slightly earlier times (see Section 6.3 and Figure 21). *(3)* In addition to driving the gas out of R_{200}, the different stellar feedback strategies may also instigate other differences in the PDF, in particular in the warm-hot gas phase ($\sim 10^{5-7}$ K) above the threshold for star formation, $n_{\rm H,\,thres} = 1\,{\rm cm}^{-3}$. Indeed, the gas in star-forming regions is sensitive to variations in the stellar feedback strategies used to release energy and momentum from newly-formed stars. Particularly, the use of a delayed cooling strategy (in RAMSES, GADGET-3 and GEAR) may result in the accumulation of warm-hot gas in a dense state, around star forming regions. The superbubble feedback scheme used in CHANGA produces a similar effect on the warm-hot dense gas. *(4)* Lastly, the cold diffuse gas near the CMB floor, visible only in ART-I, is due to the code's stochastic star formation recipe and its particular pressure floor implementation (as discussed in footnote 18 and Section 5.3.1).[24]

---

[24] As a final note to Figure 14, the gas at $\gtrsim 10^{-21}\,{\rm g\,cm}^{-3}$ is seen heated up to $\sim 10^2$ K (except in ART-I and GEAR in which such dense gas is nonexistent for the moment). This heated gas is caused by GRACKLE's redshift-dependent UVB with self-shielding (Section 3.1), and is observed even in a simple one-zone test using GRACKLE. The source of the heating is assumed to be re-emission of absorbed radiation inside the dense gas cloud. The shielded CLOUDY tables were made by integrating into the star-forming cloud for a distance set by the Jeans length at a given density and temperature (with a maximum of 0.1 kpc). Over this length, UVB radiation absorbed by

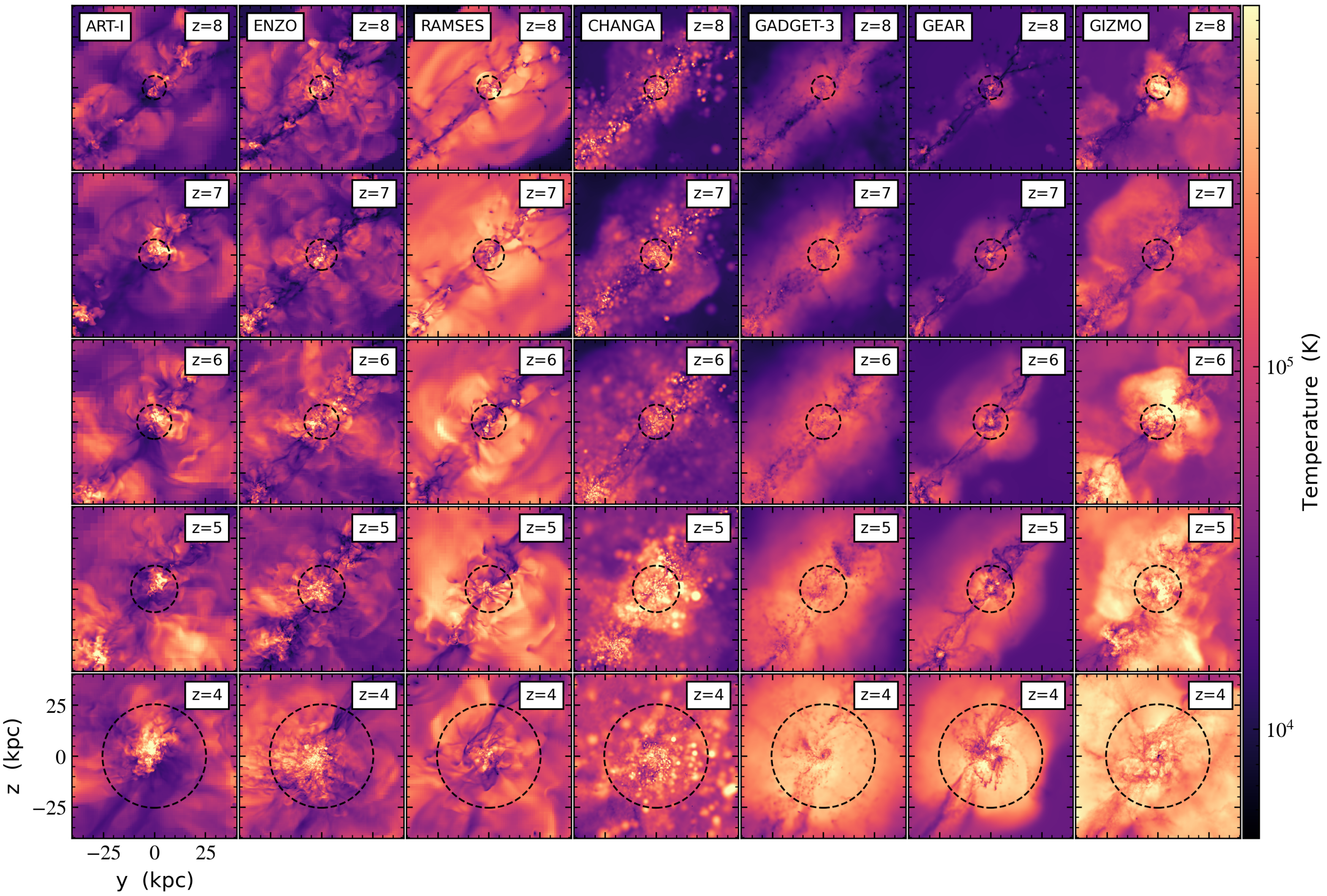


Figure 16. Similar to Figure 15, but now showing density-square-weighted projections of gas temperature in our `CosmoRun` simulation suite. Units are proper kpc. See Section 6.2 for more information on `CosmoRun` and this figure.

To better illustrate the effect of stellar feedback on the gas in the galaxy, the CGM, and the IGM, we show the evolution of the projected density and temperature in each code in Figures 15 and 16. The mean virial radius, R_{200}, at each redshift (see the Figure 15 caption) is marked with a red/black dashed circle. In these figures, we confirm the differences in the spatial distribution and thermal structure of gas, due to variations in the stellar feedback strategies, despite the fact that all the participating codes produce similar stellar mass at our target epoch, $z = 4$. Although differences in gas density and temperature may appear dramatic in Figures 14 to 16, we find a good agreement in the density distribution, especially in the nonextreme density range. This result can be observed in Figure 17, where we show the evolution of the gas density PDF of all the gas inside R_{200} from $z = 8$ to $z = 4$. We clearly see that most codes agree on the total gas mass — the area below the curve — in the intermediate density range, [$\sim 10^{-27}$, $\sim 10^{-23}$] $\mathrm{g\,cm^{-3}}$. Obviously, discrepancies in the lowest and highest density bins exist, produced by various reasons discussed in Figure 14 (note that Figure 17 shows the values of Figure 14 integrated along its y-axis).

6.2.2. *Gas Metallicity*

Metallicity is a good tracer of changes in galactic evolution. The metal content of gas inside the galaxy and its CGM, depends on how efficiently the outflows remove the metal-rich gas from the dense star-forming regions. The metal enrichment of the IGM is also dictated by the ouflows, as the IGM is the recipient of the gas pushed out of the virial radius. The exchange of metals between the CGM and IGM also determines the gas evolution in time on the density-temperature plane, as it strongly affects how quickly the gas cools and regulates the interplay between star formation and feedback. Metallicity indeed provides important information on the differences between the feedback schemes employed, and their

---

the outer layers of the cloud can be re-emitted, causing some heating on the inner layers. We caution GRACKLE users when they use the default shielded CLOUDY table provided with the package (e.g., depending on the simulation setup and resolution, one may want to disable UVB above a certain density).

ability to fit observations (Suresh et al. 2015; Kacprzak et al. 2019; Lehner et al. 2020).

Before presenting the next figures on metallicities, it is important to remind the readers that all code groups used metal yields in supernovae that are similar to the ones in the *AGORA* common physics (see Section 3.1). Using metal yields similar to the common ones allows us to conjecture that the differences observed in gas metallicity are explained mostly by the variations in stellar feedback — and/or the metal diffusion schemes — presented in Section 3.2. As a consistency check, in each `CosmoRun` simulation we have computed the ratio of the total metal mass and the total stellar mass inside the entire simulation box at $z = 4$ (i.e., "effective" metal yields in the fourth column of Table 1). Our calculation confirms that, although each code group is using its favorite metal production strategy, its "effective" yield value matches what each group assumes in the code's deposit scheme, and is in agreement within less than half a dex from other codes.

First, in Figure 18, we show the projected gas metallicity at $z = 8, 7, 6, 5$ and 4. It is important to mention that a correct interpretation of this figure requires the information on the total gas distribution (Figure 15), e.g., most metals in GEAR are in low-metallicity dense gas in the inner parts of the halo. Some codes show high metallicity around the main galaxy (e.g., RAMSES, CHANGA, and GADGET-3), while others exhibit lower values (e.g., ART-I, ENZO, and GIZMO). The former codes are the ones that tend to keep gas and metals around the star-forming regions, while the latter codes are able to push them out to the CGM, or even the IGM (see also Figure 23). The discrepancy seen here is also because the spatial distribution of metals is highly sensitive to how efficient the stellar feedback is at driving the metal-enriched outflows (see Figure 23), and to how efficient the metal diffusion is at polluting the neighboring cells/particles.

We reach a similar conclusion by analyzing the probability distribution function of metallicity and metal mass in Figures 19 and 20, respectively. Here we include only the gas inside a sphere of R_{200} from the target progenitor's center. Figure 19 shows that RAMSES, CHANGA, GADGET-3 and GIZMO exhibit large amounts of high-metallicity ($\gtrsim 1\,Z_\odot$) gas in and around the main galaxy, while ART-I, ENZO, and GEAR shows less amounts. This difference confirms that the overall gas metallicity distribution depends strongly on the efficiency of stellar feedback. Furthermore, in Figure 20, — while the global features in the PDF have been discussed in the section relevant to Figure 14 — we find variations in the total metal mass kept inside R_{200}. The stellar feedback in ART-I and ENZO rapidly push the metals out to the low-density and low-metallicity gas in the CGM and then to the IGM, leaving only a few dense star-forming regions with high metallicity. In contrast, the remaining codes keep most of the metals inside R_{200}, showing more regions with high

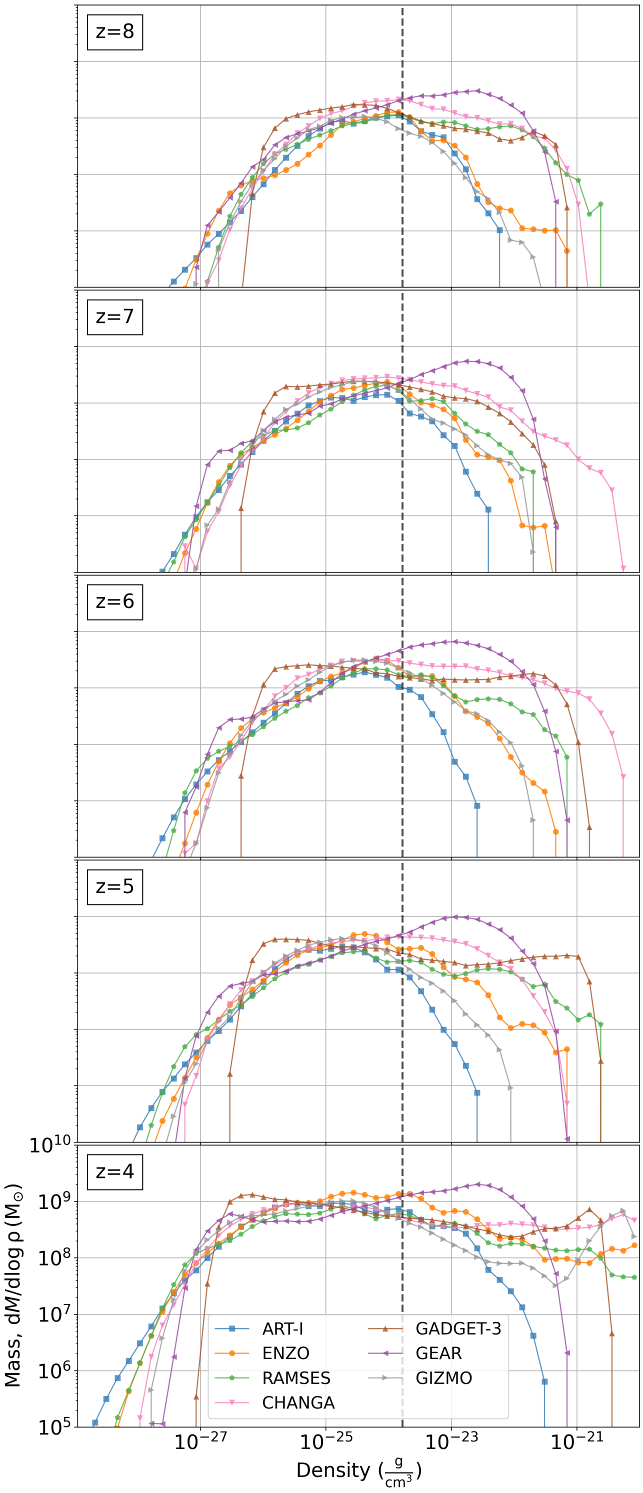


Figure 17. Distribution of gas mass as a function of gas density at $z = 8, 7, 6, 5$ and 4 from our `CosmoRun` simulation suite. Each panel is for all the gas inside the target progenitor's R_{200}. The vertical black dashed line denotes the star formation threshold, $n_{\rm H,thres} = 1\,{\rm cm}^{-3}$. See Section 6.2 for more information on `CosmoRun` and this figure.

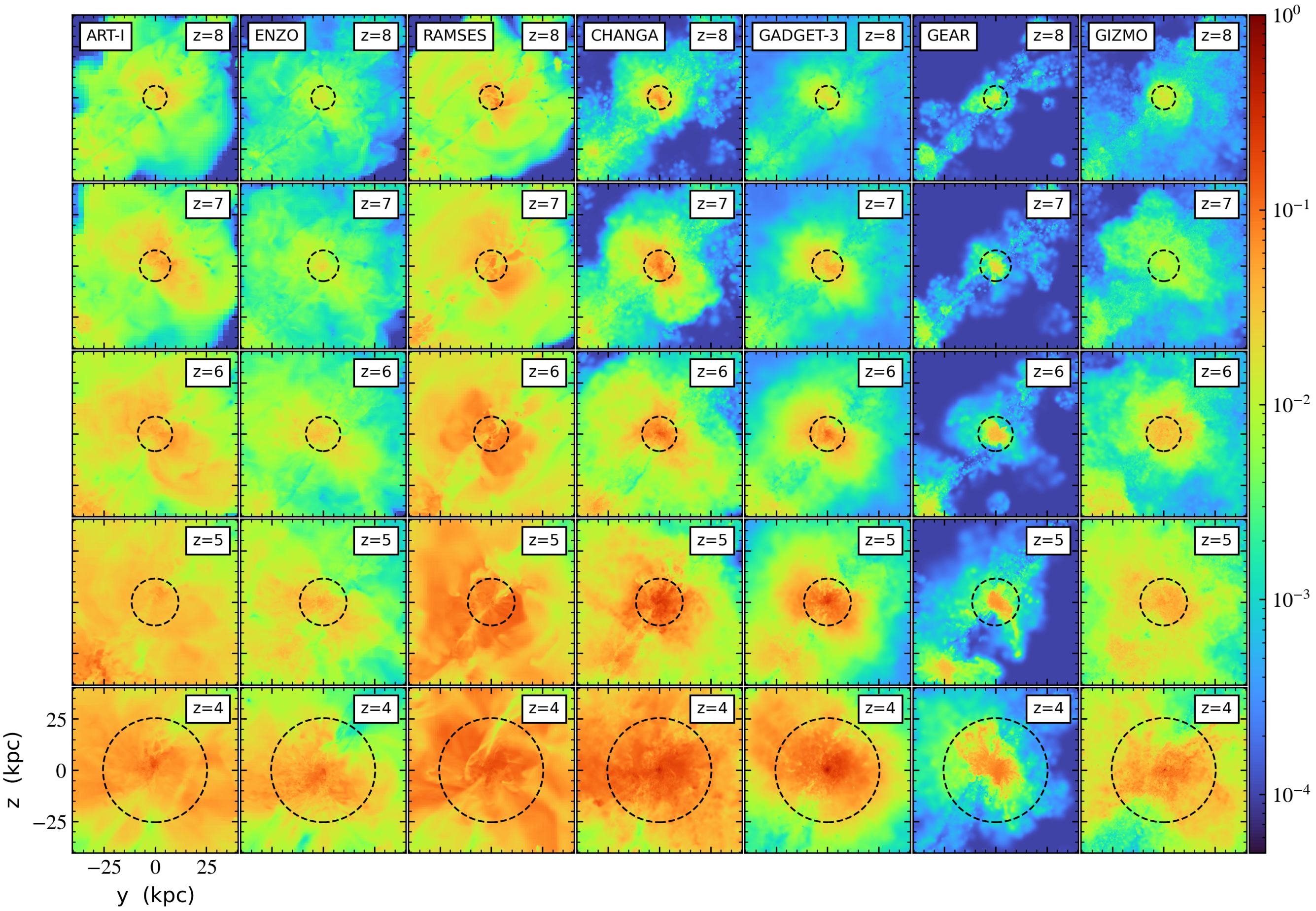


Figure 18. Similar to Figure 15 and 16, but now showing density-square-weighted projections of gas metallicity in our `CosmoRun` simulation suite. Colors represent the metallicity in units of $Z_\odot$. Units are proper kpc. See Section 6.2 for more information on `CosmoRun` and this figure.

metallicity in the gas density-temperature plane, particularly inside the regions of delayed cooling.

6.3. *Stellar Properties*

In this section, we carry out a global analysis of the stellar components in the `CosmoRun` simulations, but only focusing on their spatial distribution and metallicity. A more detailed analysis of the stellar component, including kinematics, SFHs, in-situ vs. ex-situ origin, and low-z evolution will be presented in a future paper by the *AGORA* Collaboration.

In Section 5.4.2 for `Cal-4`, we have examined the stellar mass growth histories (Figure 12). There, we detect occasional increases in stellar masses in most codes — the kinds of increases that are not contemporaneous between the codes. In fact, these are signs of the major mergers, which can be best observed in the stellar surface density maps in Figure 21. The mean virial radius, R_{200}, at each redshift (see the Figure 15 caption) is marked with a white dashed circle in each panel. In this figure, it is easier to perceive that major/minor mergers do not occur at the same time in every simulation due to the aforementioned "timing discrepancy" (see Sections 5.4.2 and 6.2). The $z = 4$ row is particularly interesting. By $z = 4$, most codes have gone through a recent major merger event, but they are at different stages of halo relaxation. This observation warns us of the need to be careful when comparing properties of galaxy-scale systems in cosmological simulations between different codes; it is indeed prudent to avoid the times when a strong perturbation is ongoing. Simulations presented here will be further analyzed in a future paper, also at lower redshifts when major mergers are rare and comparisons are more straightforward.

We conclude this subsection by investigating stellar metallicities and comparing the results with the distribution of metals in the gas component. By construction, stars form in regions where gas reaches the imposed star formation threshold, thus they inherit the properties of their progenitor gas. Among the inherited properties, metallicity is the one that should follow a similar trend between stars and the high-density gas. Additionally, in the gas metallicity PDF within R_{200} (Figures 19 and 20), we expect to find that a significant fraction of gas in the high-density, high-metallicity bins

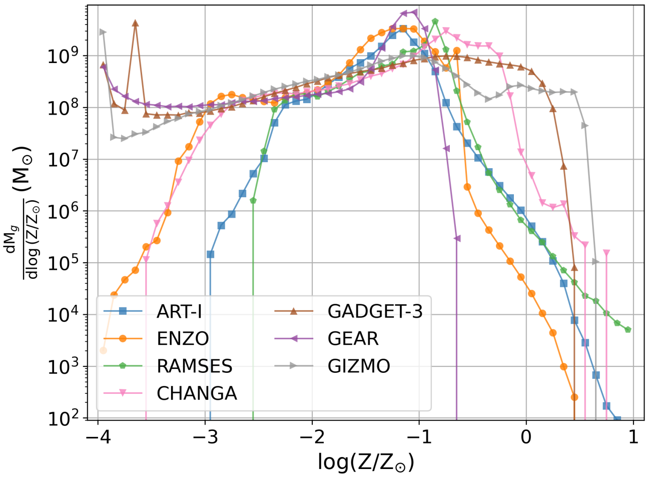


Figure 19. Distribution of gas mass as a function of gas metallicity at $z = 4$ for all the gas inside the target progenitor's R_{200} in our `CosmoRun` simulation suite. The y-axis range is kept identical as in Figure 22 for easier comparison. See Section 6.2 for more information on `CosmoRun` and this figure.

is star-forming. This argument is in agreement with what we observe in Figure 22, in which we show the stellar mass per metallicity bin. As can be also inferred from Figures 17 and 19, the stellar metallicity distribution peaks at a similar value to the gas metallicity in each code. Nevertheless, the distribution tends to be narrower in the stellar metallicities (Figure 22) than in the gas metallicities (Figure 19), as most star particles form in the densest pockets of gas. The low-metallicity stars could be either the early generation of stars formed in the gas that has not been heavily metal-enriched yet, or the later generation of stars formed in the CGM only lightly metal-enriched by galactic outflows.

6.4. *Circumgalactic Medium (CGM) Properties*

The *AGORA* collaboration plans to work on a full analysis of the CGM properties and evolution of the presented `CosmoRun` simulations, from high z's down to $z = 2$. The results of this extensive analysis will be presented in a forthcoming paper. In this section, however, we demonstrate how the multi-platform studies like *AGORA* could be useful to better understand the thermal and kinematic states of the CGM, in which disparities exist between contemporary cosmological simulations carried out with different codes, by presenting the first analysis of gas kinematics in four different temperature bins at $z = 4$. The temperature bins are defined following the observationally-motivated temperature thresholds proposed in Roca-Fàbrega et al. (2019) and in Strawn et al. (2021).

In Figure 23, we show the probability distributions of the velocity magnitude (top row) and the radial velocity (bottom row) for the gas inside a sphere of radius R_{200} from the cen-

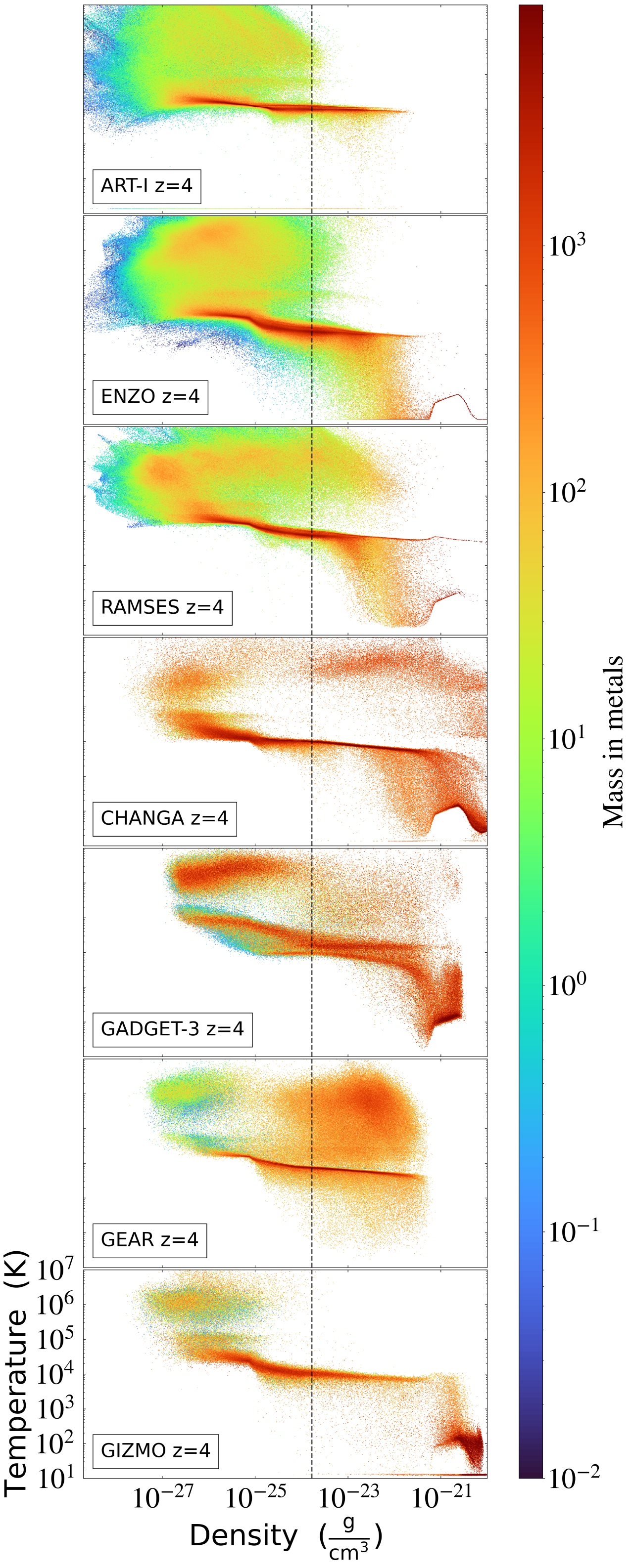


Figure 20. Similar to Figure 14, but now with colors representing the total metal mass in each 2-dimensional bin in our `CosmoRun` simulation suite. Note that the PDF is for the gas within R_{200} from the center of the target galaxy in the `CosmoRun` simulations. A sphere of radius R_{200} encloses the main galaxy and CGM, but not the IGM. See Section 6.2 for more information on `CosmoRun` and this figure.

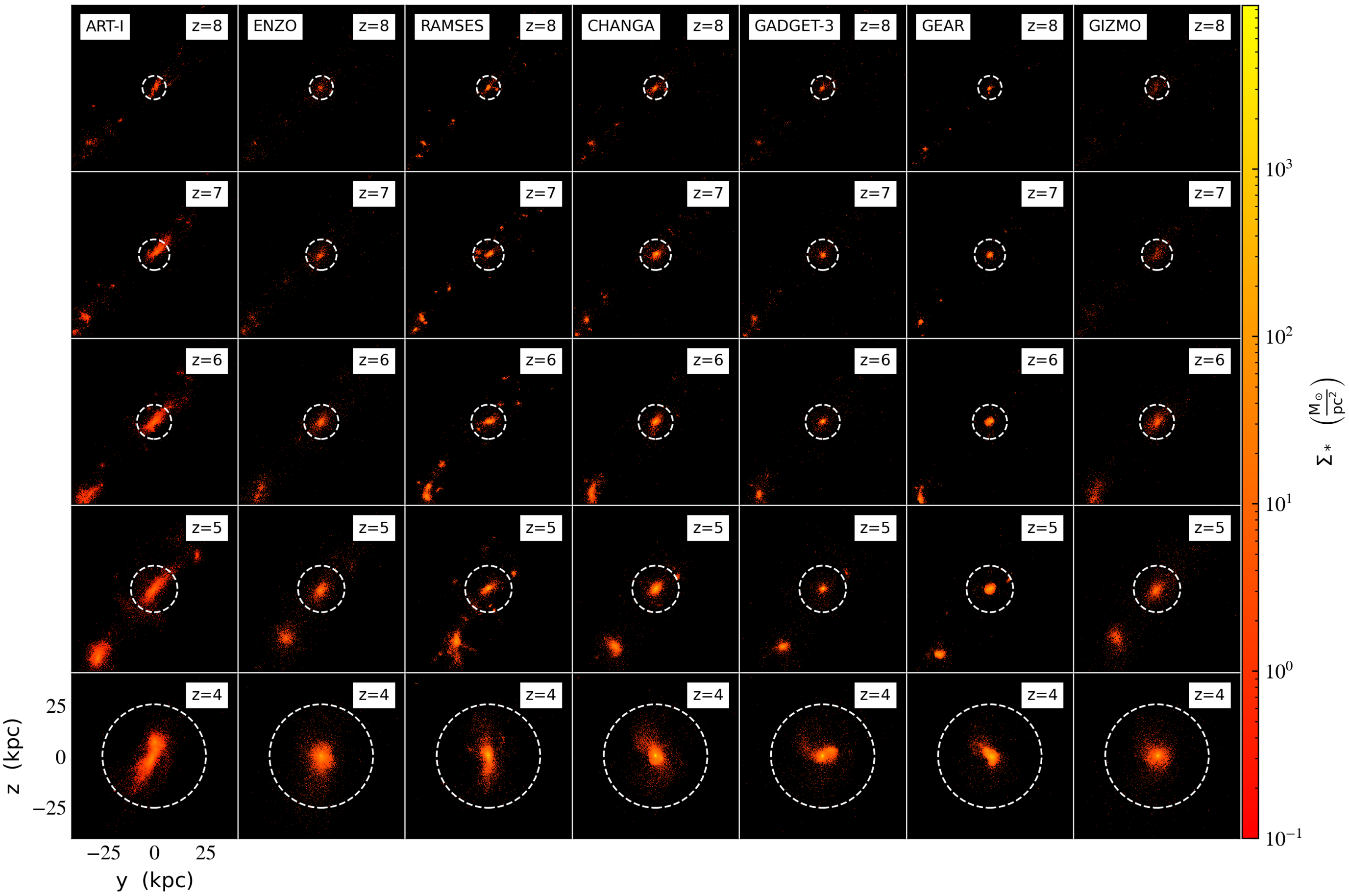


Figure 21. Similar to Figure 15, 16 and 18, but now presenting stellar surface densities from our `CosmoRun` simulation suite. Colors represent the total stellar mass in each 2-dimensional bin. Units are proper kpc. See Section 6.3 for more information on `CosmoRun` and this figure.

ter of the target progenitor galaxy. The panels are for all the gas, cold gas ($T < 10^{3.8}$ K), cool gas ($10^{3.8} < T < 10^{4.5}$ K), warm gas ($10^{4.5} < T < 10^{6.5}$ K), and hot gas ($T > 10^{6.5}$ K) from left to right. The velocity magnitude PDFs (top row) show that there is a reasonably good agreement on the kinematics of the gas. This agreement is particularly good in the cool and warm gas; in these temperature phases, the mesh-based codes and the particle-based codes agree well with each other. The convergence is not as good in the hot gas, though, where Art-I and Enzo exhibit slightly larger gas fraction with high velocity than the rest of the participating codes, due to stronger feedback-driven outflows (rightmost panel; as discussed in Section 6.2). The Ramses run presented here shows lower velocities than Art-I and Enzo in the hot gas component as expected from our analysis of metal distribution (see a full discussion in Section 6.2). Additionally, in the Changa, Gadget-3, Gear and Gizmo runs, the hot gas with the largest velocities typically belongs to regions with very low density that are not well represented by their particle-based approach. In agreement with our conclusions on the gas metallicity distribution (see Section 6.2.2), Gear generates the slowest outflows, keeping most of the metals in the dense gas around the galaxy.

In the bottom row of Figure 23, we show the distribution of gas mass in radial velocity bins. Radial velocity informs us of the presence of inflowing or outflowing gas, and the strength thereof. As discussed in the previous paragraph and in Section 6.2, the strong feedback-driven outflows in Art-I and Enzo are evident in the hot gas phase (rightmost panel; also in the warm phase for Art-I). This outflowing hot gas transports a large fraction of metals to the IGM, leaving the CGM in Art-I and Enzo with lower metallicity relative to the other codes. The Ramses, Changa, Gadget-3, Gizmo and particularly Gear runs do not show as strong outflows as in Art-I or Enzo, keeping most of the metals and gas inside the CGM (as also seen in Figures 17 and 18). The cool gas follows a smooth distribution centered at zero velocity but slightly inflowing (3rd panel from the left), with a very good agreement among all the codes.

The very preliminary analysis of the gas properties in the CGM and, in particular, of its kinematics in four different temperature bins, teaches us that the kinematics of the cold

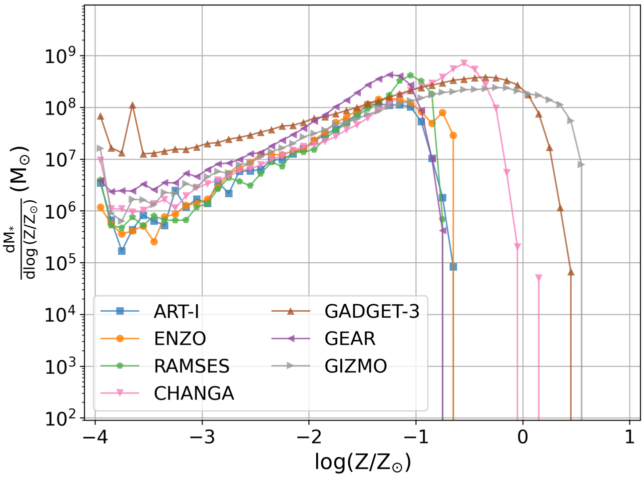


Figure 22. Distribution of stellar mass as a function of stellar metallicity at $z = 4$ for all the stars inside the target progenitor's R_{200} in our `CosmoRun` simulation suite. The y-axis range is kept identical as in Figure 19 for easier comparison. See Section 6.3 for more information on `CosmoRun` and this figure.

and hot gas is a good tracer of differences in the adopted stellar feedback prescriptions. We suggest that the research groups interested in testing their feedback models include the study of cold and hot gas kinematics in their comparisons.

7. DISCUSSION AND CONCLUSION

In this paper, we have presented a suite of seven high-resolution cosmological zoom-in simulations to $z = 4$ of a halo with a Milky Way mass at $z = 0$, obtained using seven contemporary astrophysical simulation codes — 3 AMR codes and 4 SPH codes — widely used in numerical galaxy formation. The physics prescriptions in the simulations include the common gas cooling and heating by GRACKLE-V3.1.1 that are similar to what was used in the previous *AGORA* comparisons, and the standardized *AGORA* subgrid physics such as star formation and stellar evolution (Section 3.1). However, the code groups participating in the comparison use the stellar feedback prescription that resembles the most widely used in their code community for research (Section 3.2). The simulations also account for the effects of cosmological processes such as the expansion of the Universe, and the cosmic UVB radiation emitted by massive stars and quasars.

The simulations presented here have been obtained after a careful, four-step process of calibrations (Section 5). The calibration strategy designed by the Collaboration is to reduce the number of tunable simulation parameters to be accounted for when studying the effects of stellar feedback on galaxy evolution. By completing this set of calibrations, the participating code groups establish a common ground to make a robust and unbiased comparison of different simulations focusing on stellar feedback effects on the gas and SFH of the target galaxy. The calibration procedure includes four steps. In the first step (`Cal-1`) the code groups control the effects of the different gravity and hydrodynamics solvers, and refinement strategies in radiative cooling/heating-free simulations. In the second step (`Cal-2`), we ensure that the GRACKLE cooling and UVB are correctly implemented in each code. The third step (`Cal-3`) aims for convergence in the total stellar mass produced with the common star formation prescription in stellar feedback-free simulations. Finally, in the last calibration step (`Cal-4`), we ask each code group to test a stellar feedback prescription that is as close to the most commonly used one in each code community as possible, while aiming for convergence in the stellar-to-halo mass ratio at $z = 4$ to the prediction by semi-empirical models. Designing and executing the calibration procedure has required formidable efforts by the Collaboration members to (re)run the simulations while revising, when necessary, the physical prescriptions they use for the final cosmological simulations.

After all the participating code groups successfully completed the calibration steps, we reach a suite of cosmological zoom-in simulations with very similar mass assembly histories down to $z = 4$ (`CosmoRun`; Section 6). With numerical accuracy that resolves the internal structure of a target halo ($\lesssim 100$ physical pc at $z = 4$), we find that the codes overall agree well with one another in many aspects. We argue that, if adequately tested in accordance with our proposed calibration steps and common parameters, results of a modern high-resolution cosmological zoom-in simulations are robust and their predictive power can be maximized. While this calibration does lead to substantial agreement on critical parameters, differences still remain between the codes — in the properties of the gas, stars and the CGM — due to different stellar feedback strategies adopted in each of the participating codes, as well as the diversity in implementations of the hydrodynamics. We show that the gas distribution in the density-temperature space is globally affected by differences in the stellar feedback, particularly in the coldest and hottest gas, while achieving solid convergence in the cool and warm gas. We also confirm that the spatial distribution of gas metallicity from metals released in the supernova explosion is a key parameter when testing stellar feedback prescriptions in cosmological models. This is because they play an important role in the gas cooling rates, amplifying the differences in the feedback prescriptions. A similar effect is observed when analyzing stellar metallicities. We also confirm that the expected timing discrepancies in halo mergers need to be accounted for when making code-to-code comparisons, since variations in the host's post-merger relaxation highly impacts the gas properties. The analysis presented in this paper, that includes only five redshift epochs (i.e., $z = 8, 7, 6, 5$

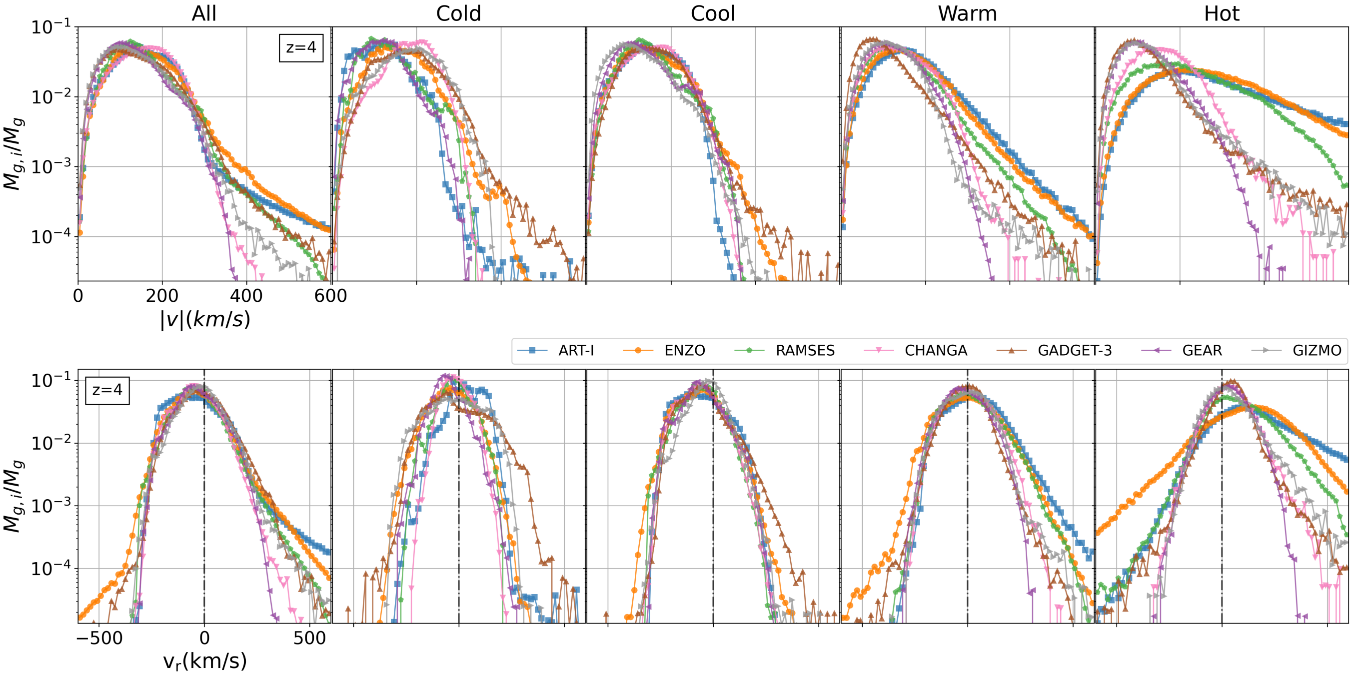


Figure 23. Distribution of gas mass as a function of velocity at $z = 4$ — velocity magnitude (*top*) and radial velocity (*bottom*) — for the gas inside the target progenitor's R_{200} in our `CosmoRun` simulation suite. The y-axis indicates the fraction of gas mass in each velocity bin with respect to the total mass in each temperature phase. The panels are for all the gas, cold gas ($T < 10^{3.8}$ K), cool gas ($10^{3.8} < T < 10^{4.5}$ K), warm gas ($10^{4.5} < T < 10^{6.5}$ K), and hot gas ($T > 10^{6.5}$ K) from left to right. See Section 6.4 for more information on `CosmoRun` and this figure.

and 4), serves as a first presentation of our suite of cosmological zoom-in simulations, and we are currently running them down to lower redshift and saving snapshots at finer timesteps.

It is important to briefly note a few points about our study presented in this work: *(1)* Our comparison in this paper across different code platforms was possible only because we have established a solid baseline through rigorous calibration steps (Section 5). The proposed calibration procedure has enabled us to trust that any differences can only be attributed to the chosen stellar feedback prescriptions and the (relatively minor) intrinsic variations of the codes' numerics. *(2)* The process of running cosmological simulations through multiple calibration steps and production stages has required Herculean endeavor by many *AGORA* members. It was also facilitated by close discussions between the code representatives, through 3 workshops and more than 30 telecons (for the `CosmoRun` simulations alone; as of May 2021), hosted by the Collaboration. This type of inter-platform collaboration is somewhat novel in the field of numerical cosmology. *(3)* Throughout this invaluable learning process, participants have used *AGORA* as a forum to talk to and learn from one another about other codes, and sometimes surprisingly, about their own. Many participants have been able to improve their codes and simulation strategies. The new versions of GRACKLE and `yt` were tested on multiple code platforms during this work, providing useful feedback to the respective developer communities.

We pride ourselves on our contribution to the galaxy formation community, by helping to maintain the reproducibility of galaxy formation simulations in general. *AGORA* helps to raise the predictive power of numerical experiments — this time, in particular, of cosmological zoom-in simulations — in building and testing the theory of structure formation in Universe, thereby benefiting researchers who rely on the robustness of simulations. Furthermore, we have demonstrated how the multi-platform approach like *AGORA* could be useful to better understand how the Universe works. For example, in *AGORA*, the thermal and kinematic states of the CGM — in which disparities exist between contemporary numerical simulations on different code platforms — can be easily investigated with multiple codes and increased fidelity, as showcased in Section 6. Indeed, *AGORA* enables a well-controlled science case in which we test various stellar feedback prescriptions and confront simulations with the ones from other codes. The novel infrastructure presented in this work will provide the *AGORA* community (or the broader simulation community) with a tool to undertake a number of new comparison projects, including the analysis of the CGM properties in simulations with different stellar feedback, the formation of clumps at high redshift, and many others. It should be noted that the code groups involved in other ongoing projects in *AGORA* or in any upcoming new projects are not limited to the seven codes that participated in this paper. Our Collaboration is open to the participation of new code groups, and we encourage interested community members to test their code's compatibility on their own, by adopting the common initial conditions, the common physics package, and the proposed calibration steps, and comparing their results with the ones from the models presented by the *AGORA* Collaboration.

We thank all of our colleagues participating in the *AGORA* Project for their collaborative spirit which has allowed the *AGORA* Collaboration to remain strong as a platform to foster and launch multiple science-oriented comparison efforts. We thank Aldo Rodríguez-Puebla for sharing results from the abundance matching semi-empirical models, and Volker Springel for providing the original versions of Gadget-3 to be used in the *AGORA* Project. We also thank the anonymous referee for his/her insightful comments and suggestions. This research used resources of the National Energy Research Scientific Computing Center, a DOE Office of Science User Facility supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231. Santi Roca-Fàbrega acknowledges support from a Spanish postdoctoral fellowship, under grant number 2017-T2/TIC-5592. His work has been supported by the Madrid Government (Comunidad de Madrid-Spain) under the Multiannual Agreement with Complutense University in the line Program to Stimulate Research for Young Doctors in the context of the V PRICIT (Regional Programme of Research and Technological Innovation). He also acknowledges financial support from the Spanish Ministry of Economy and Competitiveness (MINECO) under grant number AYA2016-75808-R, AYA2017-90589-REDT and S2018/NMT-429, and from the CAM-UCM under grant number PR65/19-22462. Ji-hoon Kim acknowledges support by Samsung Science and Technology Foundation under Project Number SSTF-BA1802-04. His work was also supported by the National Institute of Supercomputing and Network/Korea Institute of Science and Technology Information with supercomputing resources including technical support, grants KSC-2018-CRE-0052 and KSC-2019-CRE-0163. Kentaro Nagamine acknowledges the support by the MEXT/JSPS KAKENHI Grant Number JP17H01111, 19H05810 & 20H00180, as well as the travel support from the Kavli IPMU, World Premier Research Center Initiative (WPI), where part of this work was conducted. Alessandro Lupi acknowledges funding by the MIUR under the grant PRIN 2017-MB8AEZ. Daniel Ceverino is a Ramon-Cajal Researcher and is supported by the Ministerio de Ciencia, Innovación y Universidades (MICIU/FEDER) under research grant PGC2018-094975-C21. Héctor Velázquez acknowledges support from PAPIIT-UNAM under grant number IN101918 and also by the Centro Nacional de Supercomputo (CNS-IPICYT-CONACYT). Art-I simulations were performed on the Brigit/Eolo cluster at the Centro de Proceso de Datos, Universidad Complutense de Madrid, and on the AtócatlSupercomputer at the LAMOD/IAUNAM. LAMOD is a collaborative effort between the IA, ICN and IQ institutes at UNAM. Ramses simulations were performed on the Miztli supercomputer at the LANACAD, Universidad Nacional Autónoma de México, within the research project LANCAD-UNAM-DGTIC-151 and on the Laboratorio Nacional de Supercḿputo del Sureste-Conacyt. Changa simulations were performed on the Atócatl supercomputer at the Instituto de Astronomía de la UNAM, and on the Extreme Science and Engineering Discovery Environment (XSEDE) allocations TG-AST20020 and TG-MCA94P018. XSEDE is supported by the National Science Foundation (NSF) grant ACI-1053575. Gadget3-Osaka simulations and analyses were performed on the XC50 systems at the Center for Computational Astrophysics (CfCA) of the National Astronomical Observatory of Japan (NAOJ), Octopus at the Cybermedia Center, Osaka University, and Oakforest-PACS at the University of Tokyo as part of the HPCI system Research Project (hp190050, hp200041). The publicly available Enzo and `yt` codes used in this work are the products of collaborative efforts by many independent scientists from numerous institutions around the world. Their commitment to open science has helped make this work possible.

REFERENCES

Aoyama, S., Hou, K.-C., Hirashita, H., Nagamine, K., & Shimizu, I. 2018, MNRAS, 478, 4905

Aoyama, S., Hou, K.-C., Shimizu, I., et al. 2017, MNRAS, 466, 105

Brummel-Smith, C., Bryan, G., Butsky, I., et al. 2019, Journal of Open Source Software, 4, 1636. https://doi.org/10.21105/joss.01636

Bryan, G. L., Norman, M. L., O'Shea, B. W., et al. 2014, ApJS, 211, 19

Ceverino, D., Dekel, A., & Bournaud, F. 2010, MNRAS, 404, 2151

Ceverino, D., & Klypin, A. 2009, ApJ, 695, 292

Ceverino, D., Klypin, A., Klimek, E. S., et al. 2014, MNRAS, 442, 1545

Ceverino, D., Primack, J., Dekel, A., & Kassin, S. A. 2017, MNRAS, 467, 2664

Dehnen, W., & Aly, H. 2012, MNRAS, 425, 1068

Dubois, Y., Volonteri, M., Silk, J., et al. 2015, MNRAS, 452, 1502

Ferland, G. J., Porter, R. L., van Hoof, P. A. M., et al. 2013, Revista Mexicana de Astronomia y Astrofisica, 49, 137

Grisdale, K., Agertz, O., Renaud, F., et al. 2019, MNRAS, 486, 5482

Haardt, F., & Madau, P. 2012, ApJ, 746, 125

Hahn, O., & Abel, T. 2011, MNRAS, 415, 2101

Hayward, C. C., & Hopkins, P. F. 2017, MNRAS, 465, 1682

Hernquist, L., & Katz, N. 1989, ApJS, 70, 419

Hinshaw, G., Larson, D., Komatsu, E., et al. 2013, ApJS, 208, 19

Hopkins, P. F. 2013, MNRAS, 428, 2840. http://adsabs.harvard.edu/abs/2013MNRAS.428.2840H

Hopkins, P. F. 2015, MNRAS, 450, 53

Hopkins, P. F., Wetzel, A., Kereš, D., et al. 2018, MNRAS, 477, 1578

Hosono, N., Saitoh, T. R., & Makino, J. 2016, ApJS, 224, 32

Hubber, D. A., Falle, S. A. E. G., & Goodwin, S. P. 2013, MNRAS, 432, 711

Jia, J.-Y., Zhu, W.-S., Gao, L., & Feng, L.-L. 2020, Research in Astronomy and Astrophysics, 20, 095. http://dx.doi.org/10.1088/1674-4527/20/6/95

Kacprzak, G. G., Pointon, S. K., Nielsen, N. M., et al. 2019, ApJ, 886, 91

Keller, B. W., Wadsley, J., Benincasa, S. M., & Couchman, H. M. P. 2014, MNRAS, 442, 3013

Kim, J.-h., Wise, J. H., Abel, T., et al. 2019, ApJ, 887, 120

Kim, J.-h., Abel, T., Agertz, O., et al. 2014, ApJS, 210, 14

Kim, J.-h., Agertz, O., Teyssier, R., et al. 2016, ApJ, 833, 202

Kobayashi, C., Tsujimoto, T., & Nomoto, K. 2000, The Astrophysical Journal, 539, 26. http://adsabs.harvard.edu/abs/2000ApJ...539...26K

Komatsu, E., Smith, K. M., Dunkley, J., et al. 2011, ApJS, 192, 18

Kravtsov, A. V. 2003, ApJL, 590, L1

Kravtsov, A. V., Klypin, A. A., & Khokhlov, A. M. 1997, ApJS, 111, 73

Lehner, N., Berek, S. C., Howk, J. C., et al. 2020, ApJ, 900, 9

Lukić, Z., Stark, C. W., Nugent, P., et al. 2014, Monthly Notices of the Royal Astronomical Society, 446, 3697. http://dx.doi.org/10.1093/mnras/stu2377

Lupi, A., Pallottini, A., Ferrara, A., et al. 2020, MNRAS, 496, 5160

Matteucci, F., & Greggio, L. 1986, A&A, 154, 279

Morris, J. P. 1996, PASA, 13, 97

Nuñez-Castiñeyra, A., Nezri, E., Devriendt, J., & Teyssier, R. 2020, arXiv e-prints, arXiv:2004.06008

Oh, B. K., Smith, B. D., Peacock, J. A., & Khochfar, S. 2020, MNRAS, 497, 5203

Okamoto, T., Eke, V. R., Frenk, C. S., & Jenkins, A. 2005, MNRAS, 363, 1299. http://arxiv.org/abs/astro-ph/0503676

Power, C., Navarro, J. F., Jenkins, A., et al. 2003, MNRAS, 338, 14

Raiteri, C. M., Villata, M., & Navarro, J. F. 1996, A&A, 315, 105

Revaz, Y., Arnaudon, A., Nichols, M., Bonvin, V., & Jablonka, P. 2016, Astronomy and Astrophysics, 588, A21. http://adsabs.harvard.edu/abs/2016A%26A...588A..21R

Revaz, Y., & Jablonka, P. 2012, Astronomy & Astrophysics, 538, A82. http://arxiv.org/abs/1109.0989

—. 2018, Astronomy and Astrophysics, 616, A96. http://adsabs.harvard.edu/abs/2018A%26A...616A..96R

Roca-Fàbrega, S., Dekel, A., Faerman, Y., et al. 2019, MNRAS, 484, 3625

Roca-Fàbrega, S., Kim, J.-h., Primack, J. R., et al. 2020, arXiv e-prints, arXiv:2001.04354

Rodríguez-Puebla, A., Primack, J. R., Avila-Reese, V., & Faber, S. M. 2017, MNRAS, 470, 651

Rosdahl, J., Schaye, J., Dubois, Y., Kimm, T., & Teyssier, R. 2017, MNRAS, 466, 11

Saitoh, T. R. 2017, AJ, 153, 85

Saitoh, T. R., & Makino, J. 2009, ApJL, 697, L99

Saitoh, T. R., & Makino, J. 2013, The Astrophysical Journal, 768, 44. https://doi.org/10.1088%2F0004-637x%2F768%2F1%2F44

Scannapieco, C., Wadepuhl, M., Parry, O. H., et al. 2012, MNRAS, 423, 1726

Shen, S., Wadsley, J., & Stinson, G. 2010, MNRAS, 407, 1581

Shimizu, I., Todoroki, K., Yajima, H., & Nagamine, K. 2019, MNRAS, 484, 2632

Shin, E.-j., Jung, M., Kwon, G., et al. 2020, ApJ, 899, 25

Smith, B. D., Bryan, G. L., Glover, S. C. O., et al. 2017, MNRAS, 466, 2217

Springel, V. 2005, MNRAS, 364, 1105

—. 2010, MNRAS, 401, 791

Stinson, G., Seth, A., Katz, N., et al. 2006, MNRAS, 373, 1074

Strawn, C., Roca-Fàbrega, S., Mandelker, N., et al. 2021, MNRAS, 501, 4948

Suresh, J., Bird, S., Vogelsberger, M., et al. 2015, MNRAS, 448, 895

Taylor, P. A., & Miller, J. C. 2012, MNRAS, 426, 1687

Teyssier, R. 2002, A&A, 385, 337

Tornatore, L., Borgani, S., Dolag, K., & Matteucci, F. 2007, MNRAS, 382, 1050. http://arxiv.org/abs/0705.1921

Tsujimoto, T., Nomoto, K., Yoshii, Y., et al. 1995, MNRAS, 277, 945. https://academic.oup.com/mnras/article/277/3/945/1080907

Turk, M. J., Smith, B. D., Oishi, J. S., et al. 2011, ApJS, 192, 9

Wadsley, J. W., Keller, B. W., & Quinn, T. R. 2017, MNRAS, 471, 2357

Wiersma, R. P. C., Schaye, J., Theuns, T., Vecchia, C. D., & Tornatore, L. 2009, MNRAS, 399, 574. http://arxiv.org/abs/0902.1535

SWIFT Part III

4 Theory

Far he journeyed in that ship, even into the starless voids; but most often was he seen at morning or at evening, glimmering in sunrise or sunset, as he came back to Valinor from voyages beyond the confines of the world.

— J. R. R. Tolkien, The Silmarillion

My main work during my thesis was to implement the models used with `GEAR` into the code `SWIFT`. In this chapter and the two next, my implementation (called `SWIFT-GEAR`) will be discussed starting from the models to the scaling. In this chapter, I will provide the required theory for our models including the numerical discretization. I will start with a quick overview of the different physical processes that we are aiming to solve before moving to the details of each one.

`SWIFT-GEAR` is aimed at simulating dwarf galaxies at high resolution and at producing predictions that can be directly matched to observations. To produce realistic dwarf galaxies, obviously the gravity (section 4.2) needs to be taken into account. It is also desirable to include a cosmological context for the hierarchical formation of galaxies along with the impact of the Hubble flow due to the expansion of the universe that slows down the large scale gravitational collapse (section 4.1). Both cosmology and gravity are the only processes dominated by the dark component of the universe (dark matter and dark energy). While the impact of the baryonic physics is not as important as the gravity, it still plays an important role in order to produce observable quantities. The baryonic matter is found mainly in the gas and stars (Fukugita et al., 1998). Initially, only the gas is present and is well described by the hydrodynamics (section 4.3). The equations are extended by the addition of the radiative cooling that consists in the interaction between photons and the elements contained within the gas (section 4.4). Even if it is called cooling, it can also heat the gas if the radiations are strong enough and the gas cold enough. An important source of such radiations is called the UV background that is present almost everywhere in the universe and produced by star forming galaxies and AGNs. Once the gas reaches high enough densities, it starts to produce stars (section 4.5) that will explode in supernovae after a few millions years (section 4.6). During the explosions, they release a large amount of energy along with the metals processed during their evolution into the surrounding gas. The metals are then "diffusing" in the gas (section 4.7) and deeply enhance the radiative cooling.

4.1 Comoving Coordinates

As mentioned in the introduction, the evolution of the universe is well described by the standard cosmological model ΛCDM shown in Figure 1.2. Thus in `SWIFT-GEAR`, we assume a ΛCDM framework. During the first Gyrs, the universe considerably increased its size and thus the average physical density was reduced. Such reduction impacts the growth of structures. A convenient modelization of this expansion is done through the usage of comoving variables that depends on the evolution of the scale factor (Bertschinger, 1998; Peebles, 1993). In the next sections, most of the computations are made using the comoving frame and the comoving variables (r'). They are based on Schaller, 2019a and Schaller, 2019c. In order to obtain the non-cosmological equations, one simply needs to assume a static universe at redshift 0 (e.g. $a = 1$, $\dot{a} = 0$, $H = 0$). The comoving coordinates are given by $r' = r/a$, and we decided to make the entropy an invariant between the comoving and physical coordinates $A' = A$. Using the previous relations, the definition of the density and the ideal gas law (that we assume in the hydrodynamics), we obtain the following relations for the thermodynamic quantities:

$$\rho = \frac{\rho'}{a^3}, \qquad u = \frac{u'}{a^{3(\gamma-1)}}, \qquad P = \frac{P'}{a^{3\gamma}}, \qquad c_s = \frac{c_s'}{a^{3(\gamma-1)}}$$

where c_s is the speed of sound, P the pressure, u the specific internal energy, γ the adiabatic coefficient and ρ the density. Concerning the velocity, the most logical definition would be $v' = \dot{r}'$. A more practical choice used in `SWIFT` (as in `GADGET`) that leads to the simplification of equations is the following definition $v' = a^2\dot{r}'$.

In the next sections, it will be important to be careful when deriving the equations in a cosmological context. Indeed, some scale factors are present in the spatial derivatives and kernels. The time differentials need also to be applied on the scale factors. As mentioned in the introduction, the uncertainties on the Hubble constant are not large enough to motivate the usage of the "little h". As it can be source of confusion, it is worth to try to stop using it in `SWIFT` and thus, as `GEAR` is still using h, the units between `SWIFT` and `GEAR` are not necessarily the same.

4.2 Gravity

In cosmological simulations, the main driver for the formation and evolution of large scale structures is gravity. If the velocities are not large enough, gravity can quickly grow any fluctuations into massive objects. It means that an accurate treatment of gravity is necessary in order to correctly reproduce the structure of the universe. In principle, we should develop all the equations in the General Relativity (GR) framework. Some simulations are done in a GR framework using a weak field approximation, but they lack baryonic physics. They are also not efficient enough, due to the extra computations required by GR, to produce simulations comparable to the `EAGLE` or `ILLUSTRIS` ones. Thus, the Newtonian gravity is still largely used and is correct under the assumption of weak gravitational fields and non-relativistic speed (Adamek et al., 2016). While the first assumption is justified on such large scales due to the lack of resolution for the most compact objects (e.g. black holes), the second one is more complex and depends on the physics we want to solve. As some type of dark matter (e.g. hot dark matter such as standard neutrinos) have relativistic velocities, they should be impacted by GR. Fortunately, the standard model (ΛCDM) contains a cold dark matter that respects this condition. Another source of relativistic matter are the supernovae where the ejected gas can reach speed up to a few percent of the speed of light (Hillebrandt and Niemeyer, 2000). As this relativistic gas represents only a small fraction of the overall mass in the galaxy, it can therefore be safely neglected at the galaxy scale.

Recently, the first cosmological simulation including GR was published (Adamek et al., 2016) and they have shown that the main impact of GR through the frame dragging effect has an almost negligible impact (about 1%). Rigopoulos and Valkenburg, 2015 also conclude on the correctness of Newton's gravity for cosmological simulations through the computation of Newton and GR's trajectories.

Therefore, we aim at solving Newton's gravity given by the following set of equations:

$$\frac{\mathrm{d}v'}{\mathrm{d}t} = \frac{g'}{a} \tag{4.1}$$

$$\Delta'\phi' = 4\pi G\rho' \tag{4.2}$$

where ϕ is the gravitational potential, $g' = -\nabla'\phi'$ is the gravitational acceleration and G is the gravitational constant.

4.2.1 Numerical Approximations for Gravity

Now that we have the general equation, we need to discretize it for our numerical simulations. As we will see later, `SWIFT` is a particle based code and the simplest approach for gravity would be to use a simple N-body approach where we compute all the interactions between two particles. While this approach is exact, it is far too slow ($O(n^2)$) and thus a more elaborate technique is implemented based on Dehnen, 2002; Bagla and Ray, 2003 and Schaller, 2019b.

As we are working with a linear equation for gravity (Equation 4.2), we can split the potential in the sum of two contributions. The first one computes the forces locally using the Fast Multipole Method (FMM) and the second one computes the long range [1] and periodic forces using a spectral method that will both be described in details later. The spectral method consists in a grid based solver that uses a Fourier transformation and a Cloud In Cell (CIC) method to project the density into the grid. To separate the contribution from the small and large scales, we introduce a cutoff in the potentials. The potential is thus given in Fourier space by $\hat{\phi} = (1 - \hat{F}(k))\hat{\phi}(k) + \hat{F}(k)\hat{\phi}(k)$ where $\hat{\cdot}$ is the Fourier transform of the function and F is the cutoff function that drops to zero quickly above a given radius r_s (selected in order to match the resolution of the CIC). Thus, we can define

$$\hat{\phi}_s = \hat{F}\hat{\phi} \qquad \hat{\phi}_l = (1 - \hat{F})\hat{\phi} \tag{4.3}$$

as the short (computed with the FMM) and the long range (computed with the spectral method) potentials.

For the cut-off function, different choices are possible. While `GADGET` uses a function easily computed in Fourier space, `SWIFT` has a function quickly computed in real space in order to speed up the large number of evaluations done in the FMM method:

$$F(r) = \frac{1}{r}\left[2 - 2\sigma\left(\frac{2r}{r_s}\right)\right] \tag{4.4}$$

$$\hat{F}(k) = \frac{\pi}{2} k r_s \operatorname{csch}\left(\frac{\pi}{2} k r_s\right) \tag{4.5}$$

$$\sigma(x) = \frac{e^x}{e^x + 1} \tag{4.6}$$

where σ is the sigmoid function. In real space, the short range potential of a single particle is given by:

$$\phi_s(r) = -\frac{aG}{r}\left[2 - \sigma\left(\frac{2r}{r_s}\right)\right]. \tag{4.7}$$

As it can be seen, ϕ does not include the mass here. In this section, this choice has been made in order to show explicitly the mass dependencies in the equations.

Fast Multipole Method

As mentioned before, the simplest approach would be to compute the gravitational forces through the Newtonian forces between each pair of particles. To speed up this technique, the Fast Multipole Method (FMM; $\mathcal{O}(n)$) groups particles together using a multipole expansion and then interacts the particles group by group when they are sufficiently far away from each other. The

---

[1] As you will see later, this long range does not correspond to the task grav_long_range.

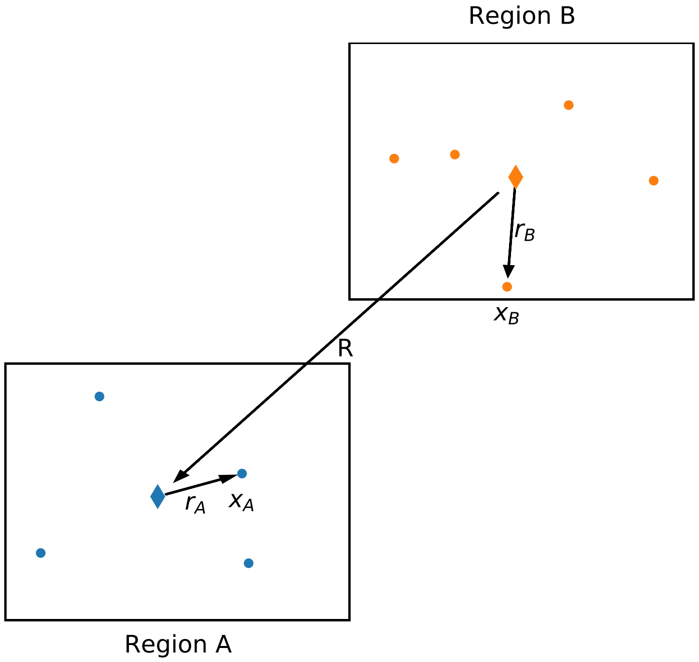


Figure 4.1 – Representation of the FMM methods interacting region A with region B. The dots represent particles and the diamond the center of mass of each region. x_A and x_B are the absolute position of the particles and r_a, r_b their relative position according to the center of mass.

main advantages of the FMM from traditional tree codes are the symmetrical operations that ensures the third law of Newton and the conservation of the momentum but also reduces the computational cost (Dehnen, 2002).

In order to derive the FMM equations, we first start by considering two particles interacting together from the absolute positions x_a and x_b and each particle belonging to a different group of particles. In Figure 4.1, the situation is represented where the particles are given by dots and the centers of mass by the diamonds. Let us start with the potential at the position x_a and from a single particle in x_b:

$$\phi_s(x_a - x_b) = \phi_s(r_a - r_b + R) \tag{4.8}$$

where r are the positions relative to the local center of mass and R the distance between the two centers of mass. The basic idea now is to Taylor expand the potential around R in power of $(r_a - r_b)$ and gives:

$$\sum_{\mathbf{k}} \frac{1}{\mathbf{k}!}(r_a - r_b)^{\mathbf{k}} \nabla^{\mathbf{k}} \phi_s(R) \tag{4.9}$$

where $\mathbf{k}$ and in the next equations $\mathbf{n}, \mathbf{m}$ are multi-indices (see Dehnen, 2014 for more details on this notation). They all are 3D vectors with the norm given by $n = |\mathbf{n}| = n_x + n_y + n_z$. We can use them as exponent of vectors $\mathbf{r^n} = r_x^{n_x} r_y^{n_y} r_z^{n_z}$. In the case of scalars, the exponent is replaced by its norm as defined previously. Finally, the factorial is defined as $\mathbf{n}! = n_x! n_y! n_z!$.

The term $(r_a - r_b)^{\mathbf{k}}$ can now be expanded and the different terms are regrouped in order to give the final form of the interaction between two particles:

$$\sum_{\mathbf{n}} \frac{1}{\mathbf{n}!} r_a^{\mathbf{n}} \sum_{\mathbf{m}} \frac{1}{\mathbf{m}!} (-r_b)^{\mathbf{m}} \nabla^{\mathbf{n}+\mathbf{m}} \phi_s(R). \tag{4.10}$$

From this equation, we can now consider the impact on a of all the particles from the region of b and reorder a bit the terms. Here I use Φ and not ϕ in order to differentiate the potential with and without the impact of the mass:

$$\Phi_s(x_a) = \sum_b G m_b \phi_s(x_a - x_b)$$

$$= G \sum_{\mathbf{n}} \frac{1}{\mathbf{n}!} r_a{}^{\mathbf{n}} \sum_{\mathbf{m}} \frac{1}{\mathbf{m}!} \sum_b m_b (-r_b)^{\mathbf{m}} \nabla^{\mathbf{n}+\mathbf{m}} \phi_s(R) \tag{4.11}$$

The three sums can be separated into three terms:

$$M_{B,\mathbf{m}}(R) = \frac{1}{\mathbf{m}!} \sum_{b \in B} m_b (-r_b)^{\mathbf{m}} \tag{4.12}$$

$$F_n(R) = G \sum_{\mathbf{m}} M_{B,\mathbf{m}}(R) D_{\mathbf{n}+\mathbf{m}}(R) \tag{4.13}$$

$$D_{\mathbf{n}+\mathbf{m}}(R) = \nabla^{\mathbf{n}+\mathbf{m}} \phi_s(R) \tag{4.14}$$

$$\Phi_s(x_a) = \sum_{\mathbf{n}} \frac{1}{\mathbf{n}!} r_a^{\mathbf{n}} F_{\mathbf{n}}(R) \tag{4.15}$$

where Equation 4.12 is the P2M term (particle to multipole), 4.13 is the M2L term (multipole to local expansion) and 4.15 is the L2P term (local expansion to particle). The first and last terms are only applied once per particles while the M2L term needs to be computed for each region A. Now the gravitational acceleration can be easily obtained by taking the gradient of the potential and limiting the sum to the desired order p (by default 4 in `SWIFT`). We have now an efficient way of computing the (non-periodic) gravitational forces at long distance.

The FMM cannot be applied in all situations as it requires a Taylor expansion and thus implies large distances between regions in comparison to their size:

$$\frac{\rho_A + \rho_B}{R} < 1 \tag{4.16}$$

where $\rho_A = \max_{a \in A}(r_a)$. Dehnen, 2014 introduced an estimation of the FMM errors that can be used as an additional criterion:

$$E_{BA,p} \frac{M_B}{R^2} < \epsilon_{\mathrm{FMM}} \min_{a \in A}(a_a) \tag{4.17}$$

where ϵ_{FMM} is a parameter provided by the user, a_a is the acceleration of the particle a, M_B the total mass within region B and $E_{BA,p}$ is given by:

$$8 \frac{\max(\rho_A, \rho_B)}{\rho_A + \rho_B} \frac{1}{M_B R^p} \sum_{n=0}^{p} \binom{p}{n} P_{B,n} \rho_A^{p-n} \tag{4.18}$$

where $P_{B,n} = \sum_{|\mathbf{m}|=n} \frac{\mathbf{m}!}{|\mathbf{m}|!} M_{B,\mathbf{m}}^2$. In this expression, we ensure that the error will not be larger than a given fraction of the minimal acceleration within the region. This creates a loop where we need the acceleration in order to define how to compute this same acceleration. In `SWIFT`, we break it by using the acceleration computed in the previous time step. For the first step, `SWIFT` simply computes a first fake step where only the criteria on the distances is used in order to get a first approximation. The main issue with this criteria is the lack of symmetry, therefore in some cases, a region A might be able to use the FMM for region B while B is not able to do it with A.

Finally, it is worth to mention that the gravitational forces are softened at short distances in order to produce collisionless simulations (as expected from galactic evolution). In `SWIFT`, the dirac function representing the position of the particles is convolved with the C2 kernel (`GEAR` uses a cubic spline). The specific potential is then:

$$\phi(r, H) = \begin{cases} \frac{1}{H} f\left(\frac{r}{H}\right) & \text{if } r < H, \\ \frac{1}{r} & \text{if } r \geq H \end{cases} \tag{4.19}$$

where $H = 3\epsilon_{\mathrm{Plummer}}$ is an input parameter and $f(u) = -3u^7 + 15u^6 - 28u^5 + 21u^4 - 7u^2 + 3$. The main advantage of this function over `GEAR`'s one is its simpler expression that can be evaluated faster while keeping the overall same shape. As this expression reduces the acceleration at short distances and the cutoff function F reduces the acceleration at large distances, the multipole acceptance criteria given in Equation 4.17 is slightly modified. As this equation is often evaluated, `SWIFT` uses a polynomial approximation ($f_{\mathrm{MAC}}(r)$) of the true gravitational forces that is always in between the Newtonian forces and the softened ones. The equation is then simply modified by replacing $1/R^2$ with this function:

$$E_{BA,p} M_B f_{\mathrm{MAC}}(R) < \epsilon_{\mathrm{FMM}} \min_{a \in A}(a_a) \tag{4.20}$$

Spectral Method

In the periodic case, the forces need to include the contribution from periodicity. As the Fourier transform assumes periodic boundaries, the periodic Poisson equation can be easily solved thanks to it. In order to derive the equations, I will only consider the case of a single particle at the origin. The case of multiple particles can be trivially derived from the equations for a single particle.

First the CIC method is applied in order to compute the density field from the particle:

$$\rho(x) = \frac{1}{\Delta x^3} m K\left(\frac{x}{\Delta x}\right) * \delta(x), \tag{4.21}$$

where $*$ is the convolution operator, K is the smoothing kernel (e.g. top hat or triangle function), $\Delta x = L/N$, L is the box size and N the number of cells.

Then the FFT is applied on the density and the Poisson equation is solved for both the full potential and the long range potential with a simple division (Equation 4.3) [2]:

$$\hat{\rho}(k) = \frac{m}{(2\pi)^{3/2}} \hat{K}\left(k\Delta x\right), \tag{4.22}$$

$$\hat{\Phi}(k) = -\frac{4\pi G}{k^2}\hat{\rho}(k), \tag{4.23}$$

$$\hat{\Phi}_l(k) = -\frac{4\pi G}{k^2}\hat{\rho}F(k). \tag{4.24}$$

where the term Δx^3 is removed due to the Fourier transform of the kernel.

Finally, the long range potential is simply computed through the inverse FFT and by applying the kernel again in order to interpolate the field:

$$\Phi_l(x) = K\left(\frac{x}{\Delta x}\right) * \Phi_l(x). \tag{4.25}$$

The acceleration can be computed with the same technique, but the kernel is replaced with its gradient.

We have now a complete expression for the potential at any point, but due to the kernel, the forces are not exact. We can see it easily by using the FFT on Φ_l and expressing all the terms in it using equations 4.22, 4.24 and 4.25:

$$\hat{\Phi}_l = -\frac{4\pi m G}{(2\pi)^{2/3} k^2} F(k) K^2\left(\frac{k}{\Delta x}\right) \tag{4.26}$$

---

[2] The Fourier transform of $\Delta f(x)$ is given by $-k^2 \hat{f}(x)$

Therefore, the potential ϕ_l needs to be divided by $\hat{K}^2$ before the FFT in order to compensate the impact of the kernel during the density computation and the future interpolation of the potential.

In `SWIFT`, the kernel selected is the triangle function. As we are interested in the acceleration, we need to be able to derive the kernel without having a 0 derivative:

$$K(x) = \prod_i^3 \max\left(1 - |x_i|, 0\right), \tag{4.27}$$

$$\hat{K}(k) = \prod_i^3 \operatorname{sinc}\left(\frac{k_i}{2\pi}\right). \tag{4.28}$$

where x_i and k_i are the coordinates of x and k.

4.3 Hydrodynamics

Cosmology and gravity together dominate the large scale structures and are physical processes dominated by the dark component of the universe (dark energy and dark matter). On the opposite side, the hydrodynamics have an important effect on smaller structure (e.g. galaxies) and act only on the gas. In the current standard model, only the baryonic matter is directly observable. Therefore, hydrodynamics are essential in order to compare the simulations with observations.

The main equations have been derived around the first half of the 19th century by Leonhard Euler (Euler equations), Claude-Louis Navier and Sir George Stokes (Navier-Stokes equations). The first set of equations (Euler) can be seen as a special case of the Navier-Stokes equations where the fluid's viscosity is negligible. As the Navier-Stokes equations derive directly from the conservation of mass, momentum and energy, the Euler equations derive from them too. In order to evaluate if the Euler equations are compatible with our simulations, we need to ensure that the viscosity is negligible. The Reynolds number $\mathcal{R}_e$, that represents the ratio between the inertial and viscous forces, provides a good indication. In the case of cosmological simulations, the gas has a low viscosity due to its low density and thus produce high Reynolds number. For example, in the intracluster and intergalactic media, the Reynolds number reaches values around 50 and the interstellar medium values around $10^5 - 10^7$ (Price, 2012a; Bauer and Springel, 2012). Therefore, it means that Euler's equations describing an adiabatic and inviscid flow are sufficient.

Two different forms of the equations exist depending if the derivatives follow the flow or not (respectively Lagrangian or Eulerian). Martel and Shapiro, 1998; Teyssier, 2015 provide a good description of the Eulerian equations. In the next chapters, we will focus on the Lagrangian equations given by:

$$\frac{\mathrm{d}\rho'}{\mathrm{d}t} = 3H\rho' - \rho'\nabla' \cdot \left(\frac{v'}{a^2} + Hr'\right) \tag{4.29}$$

$$\frac{\mathrm{d}v'}{\mathrm{d}t} = -\frac{\nabla' P'}{a^{3(\gamma-1)}\rho'}, \tag{4.30}$$

$$\frac{\mathrm{d}u'}{\mathrm{d}t} = -\frac{P'}{a^2\rho'}\nabla' \cdot v' \tag{4.31}$$

where $\frac{\mathrm{d}(\cdot)}{\mathrm{d}t} = \frac{\partial(\cdot)}{\partial t} + \mathbf{v} \cdot \nabla(\cdot)$ is the Lagrangian derivative. In this case, we are aiming to find the evolution of the variables ρ, u, v and P with only 3 equations, therefore an additional one is required to close the system of equations. This equation links the pressure to the density and internal energy, and is called the equation of state. Due to the low density of the gas, the ideal gas law is well suited to our cosmological simulations:

$$P = (\gamma - 1)\, u\rho \tag{4.32}$$

where γ is the adiabatic index.

4.3.1 Numerical Approximations for Hydrodynamics

In numerical hydrodynamics, the two opposite views based on the frame of reference also exist. In the Lagrangian form, the equations are discretized according to the mass while the Eulerian equations are discretized according to the volume. Each approach has its advantages and disadvantages when solving numerically the equations. For example, the Eulerian approach is producing an artificial diffusion when a constant velocity is added (Pontzen et al., 2021) and SPH has some trouble with shocks and discontinuities (described further in this section). If the user is interested in the topic, I recommend to look at the publications done with the codes `AREPO` (Springel, 2010), `RAMSES` (Teyssier, 2002) and `GIZMO` (Hopkins, 2015).

Let us start with a quick word on the Eulerian approach and then to the main topic: the Lagrangian approach and the Smoothed Particles Hydrodynamics (SPH). To solve the hydrodynamics equations in their Eulerian form, simulations often rely on a regular grid on which the equations are solved using conservative methods such as the Godunov method. Even if the numerical methods can be of any convergence order, when solving partial differential equations, it is desirable to keep a first order scheme in at least some parts of the simulation. Indeed, in some situations (e.g. shocks as seen in Figure 8.10 in Hesthaven, 2018), solving the equations without producing new artificial extrema (e.g. strong oscillations around shocks) cannot be done with a convergence order larger than 1 according to Godunov's theorem (Hesthaven, 2018). Flux limiters (or slope limiters) have been introduced in order to modify locally (and when required) a method with a high convergence order into lower order method and thus to remove the artificial oscillations. A second issue comes from the discretization of the equations that introduces a new partial differential that corresponds to a diffusion term (Pontzen et al., 2021). While the shocks are well captured (at first order) with the Godunov's method along with the required Riemann solver, this numerical diffusion can strongly impact the galaxies in cosmological simulations. Most of the galaxies will have a non-null velocity with respect to the simulation's volume, thus the numerical diffusion will reduce the density of the galaxies and impact the star formation history (Pontzen et al., 2021). Some hybrid methods (Springel, 2010; Hopkins, 2015) have been introduced to solve this issue (and the ones related to SPH) and rely on an unstructured mesh or a meshless approach. Unfortunately, they introduce some new difficulties due to the unstructured mesh (e.g. possible absence of neighbors).

For the SPH method (used in `SWIFT`), the equations are derived from a Lagrangian point of view. While this method is only first order accurate, due to the Godunov's theorem, it is equivalent to the Eulerian approach in the presence of shocks. Its main advantage is the conservation of mass, momentum and energy. Indeed, as the equations derive from a time independent Lagrangian, they are naturally conserved. Another feature well appreciated is that it is easily coupled to gravity. The main idea of SPH is to compute the local properties of the gas (e.g. density, velocity divergence, ...) through the smoothing over all the neighboring particles using a convolution with a kernel:

$$\frac{1}{y_i}\sum_j f_j x_j W_{ij}(h_i) \tag{4.33}$$

where f is the quantity to smooth, x is a weight and y the corresponding normalization, h_i is the smoothing length (defined later) and W the kernel. For clarity, it is worth to mention that we do include $i = j$ in the sum. Different choices for x and y exist and are called flavors. As they deeply impact the results of a SPH simulation, they will be discussed in details later. The classical choice for x is the mass and thus y represents the density.

In this section, I will start with some details on the kernel and derive the equations of motion. Then I will present and compare the different flavors of SPH and conclude with the pressure floor that is required in simulation containing both gravity and hydrodynamics. Before starting, it is worth mentioning that the following equations are based on Schaller and Borrow, 2019, Schaller, 2019c, Hopkins, 2013 and Saitoh and Makino, 2013.

Kernel

All the kernels are required to have the following properties: isotropic, positive, monotonically decreasing and twice differentiable. Usually the kernel is also required to have a small compact support in order to avoid to convolve with too many particles. Therefore, we can write the kernel in 3D with the following form.

$$W_{ij}(h) = W(r_{ij}, h) = \frac{1}{H_{\mathrm{sup}}^3} f\left(\frac{r_{ij}}{H_{\mathrm{sup}}}\right) \tag{4.34}$$

where $r_{ij} = r_i - r_j$ is the difference of position between two particles, H_{sup} is the support radius, h the smoothing length and $f(u)$ is a function respecting the previous properties, that drops to 0 for $u \geq 1$ and normalized such that

$$\int f(u)\mathrm{d}^3u = 1 \tag{4.35}$$

Here h is not necessarily equal to H_sup and it is worth to mention that `SWIFT` is using h as the smoothing length while `GADGET` is using H_sup. By defining the smoothing length as $h = 2\sigma$ and the kernel's standard deviation

$$\sigma^2 = \frac{1}{3}\int u^2 W(u, h)\mathrm{d}^3u, \tag{4.36}$$

the support radius can be computed along with $\gamma_K = H_\mathrm{sup}/h$. As it will be seen later, the smoothing lengths are usually defined from the number of neighbors. In `SWIFT`, we follow a slightly different approach by using η for the resolution (here given in 3D):

$$N_\mathrm{ngb} = \frac{4}{3}\pi\left(\frac{H_\mathrm{sup}}{h}\eta\right)^3. \tag{4.37}$$

The advantage of this definition is that for a given η, the resolution will be approximately the same in any number of dimensions while the number of neighbors will largely change (48 neighbors in 3D to 4.28 in 1D with `SWIFT`'s default parameters).

The kernel can have a strong impact on the behavior of the hydrodynamics. Thus, it cannot be chosen without an understanding of the pairing instability described by Dehnen and Aly, 2012. Depending on the kernel and the number of neighbors, particles tends to have preferential positions relative to each others and thus generate a non-homogeneous distribution that reduces the quality of the sampling (e.g. Figure 7 in Dehnen and Aly, 2012). The previous publication has shown that, in the case of large number of neighbors and kernels that are not only positive in Fourier space, this instability cannot be prevented.

In `SWIFT`, the default kernel is the cubic spline with an η equivalent to 48 neighbors in 3D. While it is not only positive in Fourier space and might suffer from pairing instability, it does not show any pairing instability with this number of neighbors. It is also more computationally efficient than the Wendland kernels required to solve the pairing instability. Finally, it is worth to mention that the Wendland kernel might solve the pairing instability but suffers from other instabilities.

Derivation of the SPH Equations of Motion

As the name indicates it, the Lagrangian hydrodynamics equations can be derived from a Lagrangian (Eckart, 1960). In the SPH approach, this Lagrangian is discretized (Hopkins, 2013; Price, 2012b; Springel and Hernquist, 2002). We start with the following one:

$$L'(q', \dot{q}') = \frac{1}{2}\sum_i m_i a^2 \dot{r}_i'^2 - \sum_i m_i \frac{u_i'}{a^{3(\gamma-1)}} \tag{4.38}$$

where m is the mass of a particle, u the internal energy, γ the adiabatic index and q are not only the position r but also the smoothing lengths h. This Lagrangian is obtained from the physical Lagrangian and by using a Gauge transformation as shown in Peebles, 1980; Schaller, 2019c. Here, as we are only focused on the hydrodynamics, we do not include the gravity term. All the following equations are based on Hopkins, 2013. Assuming isentropic process (e.g. $\mathrm{d}s = 0$), the variation of internal energy is given by:

$$\mathrm{d}u' = -\frac{P'}{m}\mathrm{d}V' \tag{4.39}$$

where m is the particle mass, P is the pressure given by the equation of state and V is an estimation of the volume.

For the smoothing length, we require h to always contain a given number of (weighted) particles through the following equation:

$$\phi'(q') = \frac{4\pi}{3}h'^3\frac{1}{\tilde{V}'} - N_{\mathrm{ngb}} = \frac{4\pi}{3}\left(\frac{h'^3}{\tilde{V}'} - \gamma_K^3\eta^3\right) = 0 \tag{4.40}$$

where $\tilde{V}$ is an estimate of the volume and not necessary the same as V and N_{ngb} is the required number of neighbors. It is worth to point out that the number of neighbors does not require to be a natural number, it can be any real numbers. It does not mean that you will effectively have N_{ngb} neighbors, but you will have a weight corresponding to this number of neighbors, due to the (future) definition of $\tilde{V}$ and most of the kernel used. For example in the case of a strong gradient in density, a particle could find 1-2 orders of magnitude more particles than N_{ngb} but will still have the correct weighted number of neighbors. The volumes are defined through the weights x and $\tilde{x}$ and the normalization $y'_i = a^{\xi-3}\sum x'_j W'_{ij}(h'_i)$ (same for $\tilde{y}$) where ξ is the scale factor dependency of x_i. Thus, the volumes can be defined as $V'_i = x'_i/y'_i$.

Now the equations of motion can be derived from the Lagrangian (Equation 4.38), via the Euler-Lagrange equations, with the additional constraint on the number of neighbors (Equation 4.40):

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L'}{\partial \dot{q}'} - \frac{\partial L'}{\partial q'} = \sum_j \lambda'_j \frac{\partial \phi'_j}{\partial q'} \tag{4.41}$$

where λ_j are the Lagrange multipliers. They can be computed thanks to the elements where q is the smoothing length and gives:

$$\lambda'_j = -\frac{3P'_j\tilde{V}'^{2}_j}{4a^{3(\gamma-1)}\pi_j h'^{3}_j}\psi' \tag{4.42}$$

$$\psi'_j = \frac{h'_j}{3\tilde{V}_j{}'}\frac{\partial V'_j}{\partial h'_j}\left(1 - \frac{h'_j}{3\tilde{V}_j{}'}\frac{\partial \tilde{V}_j{}'}{\partial h'_j}\right)^{-1} \tag{4.43}$$

And now the case where q are the positions, we get:

$$m_i\frac{\mathrm{d}v'_i}{\mathrm{d}t} = \sum \frac{P'_j}{a^{3(\gamma-1)}}\left(\nabla'_i V'_j + \psi_j \nabla'_i \tilde{V}_j{}'\right) \tag{4.44}$$

where the ∇_i corresponds to the gradient with respect to r_i. The derivatives of V (and $\tilde{V}$) can be computed from the previous definition. For the derivative in h, we recall that $W_{ij}(h_i) \propto 1/h_i^3$.:

$$\frac{\partial V'_i}{\partial h'_i} = -\frac{x'_i}{y'^2_i}\frac{\partial y'_i}{\partial h'_i}, \tag{4.45}$$

$$\frac{\partial y'_i}{\partial h'_i} = -\sum_j \frac{x'_j}{h'_i}\left[3W'_{ij}(h'_i) + \frac{|r|'_{ij}}{h'_i}\left.\frac{\partial W(z)'}{\partial z}\right|_{|r|'_{ij}/h'_i}\right], \tag{4.46}$$

$$\nabla'_i V'_j = -\frac{x'_j}{y'^2_j}\nabla'_i y'_j, \tag{4.47}$$

$$\nabla'_i y'_j = x'_i \nabla'_i W'_{ij}(h'_j) + \delta_{ij}\sum x'_k \nabla'_i W'_{ik}(h'_i). \tag{4.48}$$

The final equation of motion is given by:

$$\boxed{m_i\frac{\mathrm{d}v'_i}{\mathrm{d}t} = -\frac{1}{a^{3(\gamma-1)}}\sum_j x'_i x'_j\left[\frac{P'_i}{y'^2_i}f'_{ij}\nabla'_i W'_{ij}(h'_i) + \frac{P'_j}{y'^2_j}f'_{ji}\nabla_i W'_{ij}(h'_j)\right],} \tag{4.49}$$

$$f'_{ij} = 1 - \frac{\tilde{x}_j{}'}{x'_j}\frac{h'_i}{3\tilde{y}_i{}'}\frac{\partial y'_i}{\partial h'_i}\left[1 + \frac{h'_i}{3\tilde{y}_i{}'}\frac{\partial \tilde{y}_i{}'}{\partial h'_i}\right]^{-1} \tag{4.50}$$

where the first term in pressure is due to the special case $i = j$ and the second one is the case where $i \neq j$. The same procedure can be applied on the energy equation (Equation 4.39):

$$\frac{\mathrm{d}V'_i}{\mathrm{d}t} = -\frac{x'_i}{y'^2_i}\frac{\mathrm{d}y'_i}{\mathrm{d}t} = -\frac{x'_i}{y'^2_i}\sum x'_j \frac{\mathrm{d}W'_{ij}(h'_i)}{\mathrm{d}t} \tag{4.51}$$

Developing the total derivative gives:

$$\frac{\mathrm{d}W_{ij}(h_i)}{\mathrm{d}t} = \frac{\partial W_{ij}(h_i)}{\partial t} + \frac{\mathrm{d}(r_i - r_j)}{\mathrm{d}t}\nabla W_{ij}(h_i) + \frac{h_i}{3\tilde{V}_i}\frac{\partial W_{ij}(h_i)}{\partial h_i}\frac{\mathrm{d}\tilde{V}_i}{\mathrm{d}t} \tag{4.52}$$

The partial derivative with time is null and the last term is due to the time derivative of h and uses the constraint on the number of neighbors ϕ. Now we can work on the last part of the previous equation using exactly the same computation as before:

$$\frac{\mathrm{d}\tilde{V}_i}{\mathrm{d}t} = -\frac{\tilde{x}_i}{\tilde{y}_i{}^2}\left(v_{ij}\nabla W_{ij}(h_i) + \frac{h_i}{3\tilde{V}_i}\frac{\partial W_{ij}(h_i)}{\partial h_i}\frac{\mathrm{d}\tilde{V}_i}{\mathrm{d}t}\right). \tag{4.53}$$

With this we can finally obtain the energy equation (f_{ij} is the same as in the equation of motion):

$$\boxed{m_i\frac{\mathrm{d}u'_i}{\mathrm{d}t} = -\frac{1}{a^2}\sum_j x'_i x'_j \frac{P'_i}{y'^2_i} f_{ij} v'_{ij}\cdot\nabla' W'_{ij}(h'_i)} \tag{4.54}$$

Viscosity

As the SPH method conserves perfectly the entropy, it cannot produce entropy as expected in shocks. Consequently, the shocks are not well resolved. A solution to this problem is to add some artificial viscosity to the fluid when a shock is present. A commonly used artificial viscosity is the Monaghan's one with the Balsara switch (Balsara, 1995; Monaghan and Gingold, 1983):

$$\Pi'_{ij} = -a^{(7-3\gamma)/2}\frac{1}{4}\frac{\alpha(B'_i + B'_j)\mu'_{ij}v'_{\text{sig, ij}}}{\rho'_{ij}}, \tag{4.55}$$

$$\mu'_{ij} = \frac{1}{a}\begin{cases} \dfrac{v'_{ij}\cdot x'_{ij} + a^2Hx'^2_{ij}}{x'_{ij}} & \text{if } v'_{ij}\cdot x'_{ij} < -a^2Hx'_{ij}, \\ 0 & \text{otherwise,} \end{cases} \tag{4.56}$$

$$v'_{\text{sig, ij}} = \frac{1}{a^{3(\gamma-1)/2}}\left(c'_i + c'_j - a^{3\gamma-5}\beta\mu'_{ij}\right), \tag{4.57}$$

$$B'_i = \frac{\left|\nabla'\cdot v'_i + 3Ha^2\right|}{\left|\nabla'\cdot v'_i + 3Ha^2\right| + \left|\nabla'\wedge v'_i\right| + 10^{-4}a^{(1-3\gamma)/2}c'_i/h'_i} \tag{4.58}$$

where c is the speed of sound, α and β are two parameters, $\rho_i = \sum m_jW_{ij}(h_i)$, $\nabla\cdot v_i = \frac{1}{\rho_i}\sum m_jv_{ij}\cdot\nabla_xW_{ij}(h_i)$ and $\nabla\wedge v_i = \frac{1}{\rho_i}\sum m_jv_{ij}\wedge\nabla_xW_{ij}(h_i)$. More complex models exist where the viscosity coefficient becomes a particle based and time dependent variable (Morris and Monaghan, 1997; Rosswog et al., 2000; Cullen and Dehnen, 2010). The idea is to include the viscosity only around the shocks by increasing it in front of them and decreasing it behind. In `GEAR`, the model presented in Rosswog et al., 2000 is used. `SWIFT` propose different SPH flavors along with different viscosity model. In the future, we plan to use `SPHENIX` that includes such coefficient (Borrow et al., 2020).

This viscosity is then simply added to the equation of motion and the energy equation:

$$m_i\frac{\mathrm{d}v'_i}{\mathrm{d}t} = \ldots + \frac{1}{a^{3(\gamma-1)}}\sum_j\frac{1}{2}a^{3(\gamma-5)/2}m_j\Pi'_{ij}\left(\nabla_iW_{ij}(h_i) + \nabla_jW_{ij}(h_j)\right), \tag{4.59}$$

$$m_i\frac{\mathrm{d}u'_i}{\mathrm{d}t} = \ldots + \frac{1}{a^2}\sum_j\frac{1}{4}a^{3(\gamma-5)/2}m_j\Pi'_{ij}v'_{ij}\cdot\left(\nabla'_iW'_{ij}(h'_i) + \nabla'_jW'_{ij}(h'_j)\right). \tag{4.60}$$

The scale factor in front of the sum is extracted in order to have the same coefficient as the rest of the equation.

SPH Flavors

Until now, we did not assume any form for the weights x and $\tilde{x}$ and not all weights are equivalent. Here we will focus only on three flavors but an infinity of flavors exists (see for example Schaller and Borrow, 2019 or Hopkins, 2013).

Gadget / Density-Entropy

This flavor uses the weights $x = \tilde{x} = m$ and therefore $y = \tilde{y} = \bar{\rho}$.

$$m_i \frac{\mathrm{d}v'_i}{\mathrm{d}t} = -\frac{1}{a^{3(\gamma-1)}} \sum_j m_i m_j \left[\frac{P'_i}{\bar{\rho}'^2_i} f'_{ij} \nabla'_i W'_{ij}(h'_i) + \frac{P'_j}{\bar{\rho}'^2_j} f'_{ji} \nabla'_i W'_{ij}(h'_j) \right], \tag{4.61}$$

$$\frac{\mathrm{d}A'_i}{\mathrm{d}t} = \frac{1}{4a^2} A_{\mathrm{eos}} \left(\rho_i, \sum_j a^{3(\gamma-5)/2} m_j \Pi'_{ij} v'_{ij} \cdot \left[\nabla'_i W'_{ij}(h'_i) + \nabla'_j W'_{ij}(h'_j) \right] \right), \tag{4.62}$$

$$f_{ij} = \left[1 + \frac{h'_i}{3\bar{\rho}_i{}'} \frac{\partial \bar{\rho}_i{}'}{\partial h'_i} \right]^{-1} \tag{4.63}$$

where A_{eos} is the equation of state linking the entropy to the density and energy. As we are assuming an isoentropic process, the entropy does evolve only through the viscosity (in shocks).

Pressure-Energy

This flavor uses the weights $x = (\gamma - 1)mu$, $\tilde{x} = 1$ and therefore $y = \bar{P}$, $\tilde{y} = \bar{n}$.

$$m_i \frac{\mathrm{d}v'_i}{\mathrm{d}t} = -\frac{1}{a^{3(\gamma-1)}} \sum_j (\gamma - 1)^2 m_i m_j u'_i u'_j \left[\frac{P'_i}{\bar{P}_i{}'^2} f'_{ij} \nabla'_i W'_{ij}(h'_i) + \frac{P'_j}{\bar{P}_j{}'^2} f'_{ji} \nabla'_i W'_{ij}(h'_j) \right], \tag{4.64}$$

$$m_i \frac{\mathrm{d}u'_i}{\mathrm{d}t} = -\frac{1}{a^2} \sum_j (\gamma - 1)^2 m_i m_j u'_i u'_j \frac{P'_i}{\bar{P}_i{}'^2} f'_{ij} v'_{ij} \cdot \nabla'_i W_{ij}(h_i)' \tag{4.65}$$

$$+ \frac{1}{4} a^{3(\gamma-5)/2} m_j \Pi'_{ij} v'_{ij} \cdot \left(\nabla'_i W'_{ij}(h'_i) + \nabla'_j W'_{ij}(h'_j) \right) \tag{4.66}$$

$$f'_{ij} = 1 - \frac{h'_i}{3(\gamma - 1) m_j u'_j \bar{n}_i{}'} \frac{\partial \bar{P}_i{}'}{\partial h'_i} \left[1 + \frac{h'_i}{3\bar{n}_i{}'} \frac{\partial \bar{n}_i{}'}{\partial h'_i} \right]^{-1} \tag{4.67}$$

When simulating an ideal gas without pressure floor, the pressures are equivalent and can be simplified.

SPHENIX

`SPHENIX` is a density energy scheme (same weights than `GADGET` but the energy is evolved and not the entropy) introduced by Borrow et al., 2020. It includes some artificial diffusion in addition to the artificial viscosity in order to resolve correctly the discontinuities. This diffusion requires the computation of gradients and thus an extra loop over the neighbors that increases the computational time of the method.

Importance of different SPH schemes

The inability of `GADGET`'s SPH to correctly simulate contact discontinuities has been an important topic starting from Agertz et al., 2007, therefore a large collection of papers exists on the topic with deep comparisons (see for example Hopkins, 2013 or Borrow et al., 2020 for `SPHENIX`). Figure 4.2 shows the density of the Kelvin-Helmholtz instability for the three different schemes. The instability consists in two fluids (blue and red) moving in opposite direction with the same pressure but different densities and energies. In the initial conditions, we set a sinusoidal vertical velocity to the fluid near the discontinuity in order to seed the instability. As it can be seen in the first image, `GADGET` is unable to properly evolve the instability due an artificial surface tension arising from the discontinuity. The pressure-energy in the second image is better but still has trouble to properly mix the fluid around the discontinuities due to the lack of energy diffusion and finally `SPHENIX` in the last image correctly reproduces the Kelvin-Helmholtz instability.

The reason of the failure of `GADGET` is that the weights for the density-entropy formulation are the same on each side of the discontinuity (mass of a particle) while in pressure-energy, the weights include the internal energy that changes on each side of the discontinuity. Thus, the discontinuity in the weights impacts strongly the ability to simulate the contact discontinuity (Springel, 2010). While `SPHENIX` uses a density-energy formulation, it is still able to resolve the contact discontinuity thanks to the artificial diffusion of the energy.

Pressure Floor

When simulating a homogeneous gas with self gravity, the gas should not be able to collapse except if a perturbation is generated. As the discretization produces some perturbations, they are amplified by gravity and initialize an artificial gravitational collapse. The natural collapse is called the Jeans instability and its related mass is given by:

$$M_J = \frac{\pi^{5/2} c_s^3}{6\sqrt{G^3\rho}} \tag{4.68}$$

where c_s is the speed of sound, G the gravitational constant and ρ the density. A similar relation can be obtain for the Jeans length. In nature, a cloud will start to collapse if its size is larger than this length. In the simulations, it means that if the smoothing length is larger than the Jeans length, the gas can start to artificially collapse. To avoid this effect, we need to ensure that enough particles (N_J) are contained within the smoothing length of the particles: $N_J m_p = M_J$ where m_p is the mass of a single particle and is related to both the density and the smoothing length: $4\pi h^3/3 = m/\rho$. By combining the three previous results, a critical pressure can be derived:

$$P'_{\text{Jeans}} = \frac{a^{\gamma-4}\rho'}{\gamma}\frac{4}{\pi}Gh'^2\rho' N_{\text{Jeans}}^{2/3} \tag{4.69}$$

where γ is the adiabatic index. It is also possible to derive a critical temperature or energy from the same relation. To fix this artificial collapse, three different solutions exist: applying

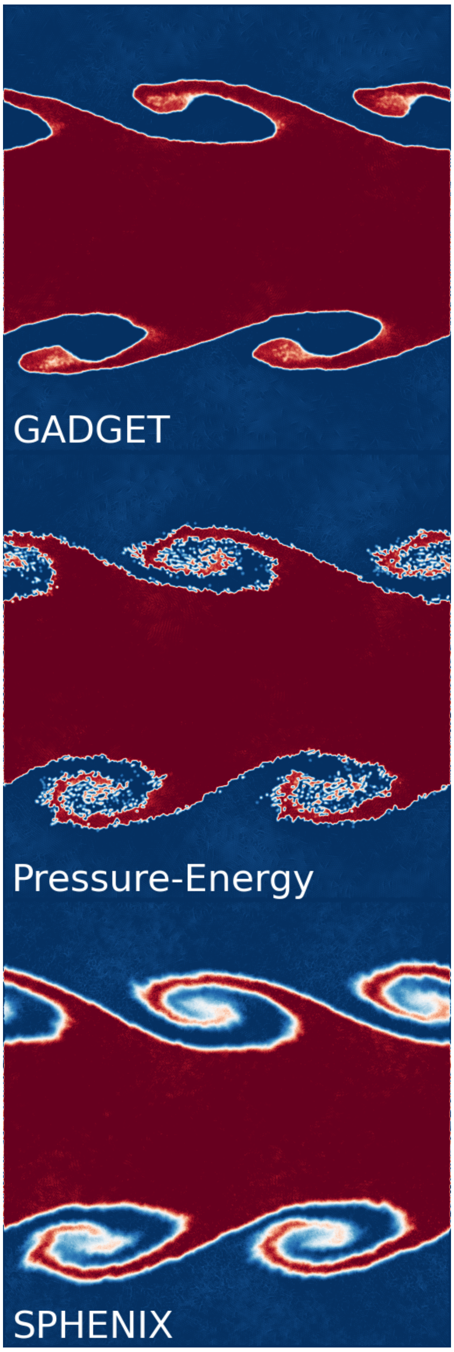


Figure 4.2 – Density projection of a 2D Kelving-Helmholtz instability with the three different flavors. `GADGET` is not able to grow correctly the instability while both the pressure-energy and `SPHENIX` can. The pressure-energy is producing a strongly non homogeneous medium due to its lack of diffusion. `SPHENIX` fixes this issue thanks to its artificial diffusion.

a temperature / energy floor (e.g. `EAGLE`; Robertson and Kravtsov, 2008), a pressure floor (e.g. `SWIFT-GEAR`; Hopkins et al., 2011) or using a multiphase gas model (e.g. `ILLUSTRIS`; Springel and Hernquist, 2003). For both the multiphase and temperature floor, the energy of the unresolved particles is increased in order to reach their own definition of the critical temperature. For the pressure floor, the situation is a bit different. We do not wish to modify the energy of the particles, therefore the pressure used in the equation of motion is increased to the critical pressure. This additional pressure/energy can be seen as the pressure produced by the unresolved and turbulent scales that avoids the artificial collapse.

The equations 4.30 and 4.31 can be easily modified in order to include the pressure floor by taking the maximum of P and P_{Jeans} as the pressure. For the SPH, the situation is a bit more complicated depending on the choice of flavors. With the pressure-energy flavors, two possible implementations exist either we consider y as the pressure and apply the pressure floor also on it (through an energy floor on x) or we can consider x/y simply as measurement of the volume and only apply the pressure floor on P. In `SWIFT`, we decided to follow the second technique as it requires less computation and thus only P is modified with the maximum and not $\bar{P}$ in equations 4.64 and 4.65.

4.4 Radiative Cooling

The continuity equation given in Equation 4.31 does not take into account non-adiabatic source of cooling/heating, thus the equation needs an additional term:

$$\frac{\mathrm{d}u}{\mathrm{d}t} = \frac{\Gamma - \Lambda}{\rho} \tag{4.70}$$

where Γ (Λ) is the heating (cooling) function due to the absorption or emission of photons. Through some chemical reactions such as the photodissociation or photorecombination, the gas can change its internal energy. Thus, the radiative cooling consists in modeling the temperature evolution of the gas through the evolution of its composition, emission and absorption of photons. In principle to compute the evolution of the composition, a chemical network needs to be solved including the external contribution from the UV background generated by the star forming galaxies and quasars in the universe (Haardt and Madau, 2012). These chemical networks quickly increase in size and complexity with each element added, thus it is currently complicated to run a cosmological simulation including more than about 10 different species without some strong simplifications such as assuming an equilibrium.

Luckily, the gas is mainly composed of hydrogen (73.9% of the Milky Way in mass (Arnett and Cameron, 1967)) and helium (24.0%), thus for a large fraction of the gas, a simple network containing the hydrogen, helium and free electrons is enough to accurately follow its evolution. Such a network already requires 6 species (each element plus its ionized states) and 11 reactions (see table 4.1).

The remaining fraction is composed of metals[3]. Even if they represent only a tiny fraction of the matter (2.1% in the Milky Way), their impact is extremely important as their electrons are less bound to the nucleus and thus will more easily interact with the incoming photons or emit photons in order to return to a less excited state.

4.4.1 UV Background and Self-Shielding

Star forming galaxies are producing a large amount of ionizing photons and especially in the UV band (Haardt and Madau, 2012). When the radiations are able to leave the galaxy, they form what we call the UV background (UVB) that is present almost everywhere in the universe. Its evolution is shown in Figure 4.3 through the photoionization rate of HI, HII and HeI as function of time. The UVB quickly raised since the creation of the first stars until reaching a peak at $z \sim 2$. This radiation considerably impacts the smallest galaxies as it heats the gas contained within them and can stop the formation of new stars. Thus for dwarf galaxies, surviving the peak plays a central role on their late evolution.

At high densities, the density of H becomes large enough to change the gas from being optically thin to optically thick and therefore the UV background is not able to penetrate the gas anymore. In this regime, the gas is said to be self shielded by H against the UV background, and it means that the gas at the center of a dense cloud will not be heated by the UV anymore. To simulate correctly this behavior, the radiation needs to be followed through the radiation transfer equations, but it requires a large amount of computational power and more code development. This is one of the next step for `SWIFT-GEAR` and should be available in the next 2 years. Until then, the gas is simply assumed to be self-shielded as soon as it reaches a high density without any condition on the geometry and thus the UV background is removed from the cooling function. In `SWIFT-GEAR`, we use $n_{\mathrm{H}} = 0.007\ \mathrm{cm}^3$ for the self shielding based on Aubert and Teyssier, 2010.

4.4.2 Models used by `GRACKLE`

In `SWIFT-GEAR`, we rely on `GRACKLE` (Smith et al., 2017) to compute the radiative cooling. The library has two main types of mode, the first one assumes ionization equilibrium and the two other ones do not make this assumption. The three chemical networks available in `GRACKLE` are given in table 4.1 and includes the recombination, photoionization, collisional photoionization, collisional dissociation and three body reaction for the hydrogen, helium and optionally deuterium. In the first mode, the cooling function becomes independent of the gas' chemical composition and therefore depends only on the density, temperature, redshift[4] and metallicity. It means that the chemical network is not solved on the fly, but simply precomputed for a set of parameters and then simply interpolated. This approach greatly speedup the computation and relies on the code `CLOUDY` (Ferland et al., 2017) to computes a network containing all the important elements

---

[3] Elements more massive than He.

[4] for the UV background

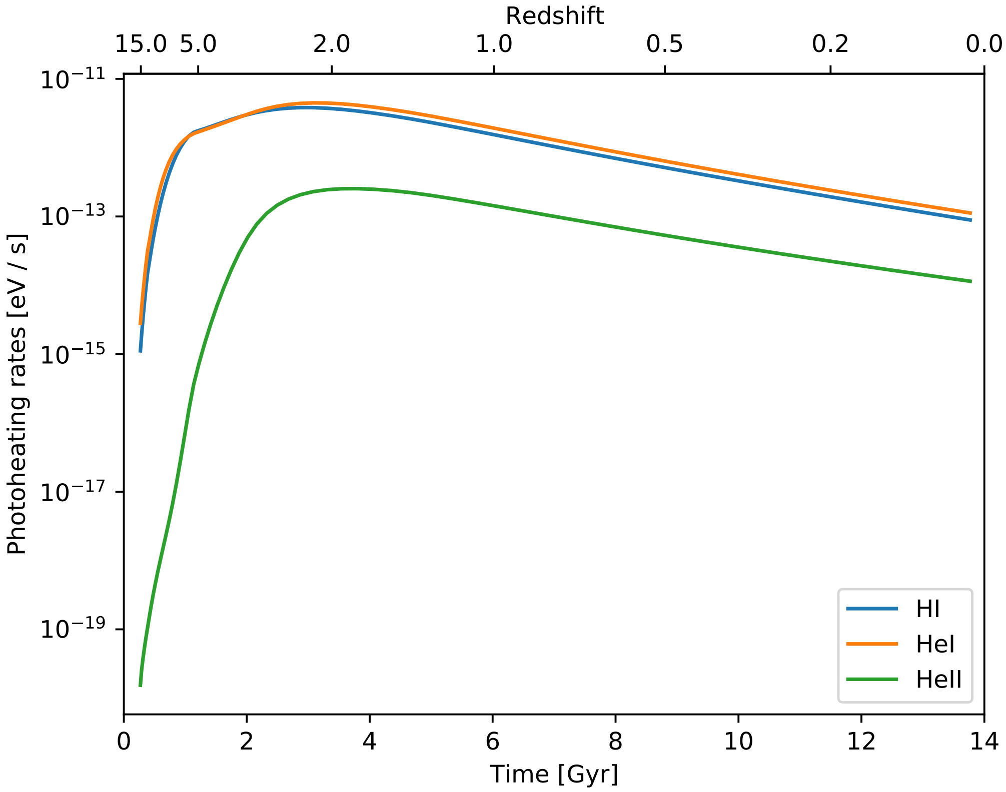


Figure 4.3 – Photoheating rate from the UVB as function of time for HI, HII and HeI. The UVB has been raising since the formation of the first stars until a peak at $z \sim 2$. Since then, the UVB has been decreasing. This model is taken from `GRACKLE` and is based on Haardt and Madau, 2012.

and writing the interpolation table[5]. Usually the cooling libraries split the cooling into two parts: the primordial cooling (H and He) and the cooling assuming that the interstellar medium is at solar metallicity. While the first part is simple enough to be computed during the simulation, the second one is far more complicated and usually requires the supposition of ionization equilibrium. Once both cooling are computed, the libraries interpolate the two cooling functions at the correct metallicity:

$$\Lambda_{\text{tot}} = \Lambda_{\text{prim}} + z\Lambda_{\odot} \tag{4.71}$$

where z is the metallicity of the ISM, and $\Lambda_{\text{tot}}, \Lambda_{\text{prim}}, \Lambda_{\odot}$ are the total cooling, the cooling due to the primordial gas and gas at solar metallicity. While this approach avoids the increased computational cost of following each metal individually, it is not fully consistent with our simulations where the ratio between the different elements is not necessarily the same as in the Sun. The second type of modes does not make the equilibrium assumption and therefore is more accurate, but due to the limited computational time, fewer elements can be simulated. Therefore only the primordial elements are simulated without the equilibrium assumption.

`GRACKLE` includes different cooling tables with different models of cooling, heating, UVB and, in some cases, a model for the self shielding following the prescription from Rahmati et al., 2013. In our simulations, we use the table called `CloudyData_UVB=HM2012.h5` and, currently, we are still using the first mode.

In Figure 4.4, the cooling rate produced by `GRACKLE` for the equilibrium mode is shown as function of temperature for different densities and metallicities. At low temperature (below about 10^4 K for the blue line), the gas is heated by the UV background and above the gas is cooled down by the different chemical reactions. Three main features can be seen in the cooling without metals (straight lines). The first (second) peak above the limit between cooling and heating is due to the ionization of the hydrogen (helium). At high temperature, the cooling rate constantly increases due to the Bremsstrahlung. Finally the impact of the metals is to globally increase the cooling. Thus, the metals decrease the equilibrium temperature between the cooling and heating. It means that two same clouds in the same environment, but different metallicities will not have the same equilibrium temperature. Thus, the one with the most metals will be able to collapse and produce stars faster.

4.5 Star Formation

The star formation is a process spanning a large range of scales. It starts with the collapse of the gas from a galactic scale (parsec) to almost a planetary scale (100 $R_{\oplus}$) and is clearly out of reach in cosmological simulations. Luckily, even if the details of the formation of a single star are difficult to reproduce and observe, its effect at the galactic scale is well understood through the Kennicutt-Schmidt law (Kennicutt, 1998). While some people are observing better relations through the observation of cold molecules (Morselli et al., 2020; Bigiel et al., 2008), the main

---

[5] The generation process is described in the appendix.

Table 4.1 – Chemical network solved by `GRACKLE` (Smith et al., 2017). Three different networks with increasing complexity exists, the equations contained in each network are indicated in the last column.

Reaction			Mode
$H + e^-$	$\longrightarrow$	$H^+ + e^- + e^-$	1,2,3
$H^+ + e^-$	$\longrightarrow$	$H + \gamma$	1,2,3
$He + e^-$	$\longrightarrow$	$He^+ + e^- + e^-$	1,2,3
$He^+ + e^-$	$\longrightarrow$	$He + \gamma$	1,2,3
$He^+ + e^-$	$\longrightarrow$	$He^{++} + e^- + e^-$	1,2,3
$He^{++} + e^-$	$\longrightarrow$	$He^+ + \gamma$	1,2,3
$H + H$	$\longrightarrow$	$H^+ + e^- + H$	1,2,3
$H + He$	$\longrightarrow$	$H^+ + e^- + He$	1,2,3
$H + \gamma$	$\longrightarrow$	$H^+ + e^-$	1,2,3
$He + \gamma$	$\longrightarrow$	$He^+ + e^-$	1,2,3
$He^+ + \gamma$	$\longrightarrow$	$He^{++} + e^-$	1,2,3
$H + e^-$	$\longrightarrow$	$H^- + \gamma$	2,3
$H^- + H$	$\longrightarrow$	$H_2 + e^-$	2,3
$H + H^+$	$\longrightarrow$	${H_2}^+ + \gamma$	2,3
${H_2}^+ + H$	$\longrightarrow$	$H_2 + H^+$	2,3
$H_2 + H^+$	$\longrightarrow$	${H_2}^+ + H$	2,3
$H_2 + e^-$	$\longrightarrow$	$H + H + e^-$	2,3
$H_2 + H$	$\longrightarrow$	$H + H + H$	2,3
$H^- + e^-$	$\longrightarrow$	$H + e^- + e^-$	2,3
$H^- + H$	$\longrightarrow$	$H + e^- + H$	2,3
$H^- + H^+$	$\longrightarrow$	$H + H$	2,3
$H^- + H^+$	$\longrightarrow$	${H_2}^+ + e^-$	2,3
${H_2}^+ + e^-$	$\longrightarrow$	$H + H$	2,3
${H_2}^+ + H^-$	$\longrightarrow$	$H_2 + H$	2,3
$H + H + H$	$\longrightarrow$	$H_2 + H$	2,3
$H + H + H_2$	$\longrightarrow$	$H_2 + H_2$	2,3
$H^- + \gamma$	$\longrightarrow$	$H + e^-$	2,3
${H_2}^+ + \gamma$	$\longrightarrow$	$H + H^+$	2,3
$H_2 + \gamma$	$\longrightarrow$	${H_2}^+ + e^-$	2,3
${H_2}^+ + \gamma$	$\longrightarrow$	$H^+ + H^+ + e^-$	2,3
$H_2 + \gamma$	$\longrightarrow$	$H + H$	2,3
$H + H + \mathrm{grain}$	$\longrightarrow$	$H_2 + \mathrm{grain}$	2,3
$H^+ + D$	$\longrightarrow$	$H + D^+$	3
$D^+ + H$	$\longrightarrow$	$D + H^+$	3
$H_2 + D^+$	$\longrightarrow$	$HD + H^+$	3
$HD + H^+$	$\longrightarrow$	$H_2 + D^+$	3
$H_2 + D$	$\longrightarrow$	$HD + H$	3
$HD + H$	$\longrightarrow$	$H_2 + D$	3
$D + H^-$	$\longrightarrow$	$HD + e^-$	3

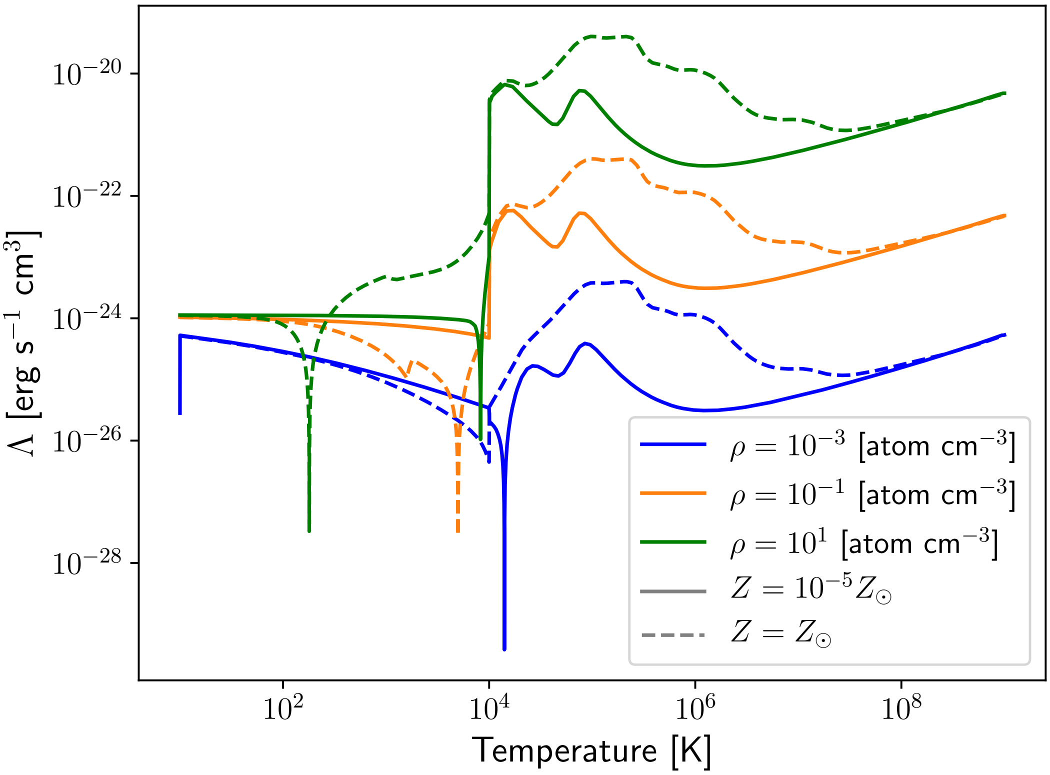


Figure 4.4 – Cooling rate as function of the temperature for different metallicities and densities. The cooling is shown in absolute value, thus the part before the large drop (at 10^4 K for the blue line) is a negative cooling (heating) and after it the gas effectively cools down. The heating is due to the presence of the UV background that heats the gas through the photoionization and photo-dissociation of the different elements. In the straight lines, the two peaks just after the switch between cooling and heating are due to the ionization of the hydrogen and helium. At high temperature, the Bremsstrahlung dominates the cooling. Finally the metals increase the cooling with respect to the primordial gas.

idea is that a high gas density will produce a high number of stars:

$$\Sigma_{\rm SFR} \propto \Sigma_{\rm gas}^n \tag{4.72}$$

where $\Sigma_{\rm SFR}$ is the surface density of star formation, $\Sigma_{\rm gas}$ is the surface density of gas and n is a fitting coefficient (usually around 1.4). This relation shows a direct link between the surface density of the gas and the surface density of star formation. As in simulations, the quantities are not projected but directly computed from 3 dimensional quantities, we can rewrite the equation in the following form:

$$\frac{\mathrm{d}\rho_\star}{\mathrm{d}t} = \frac{\epsilon_\star \rho_{\rm gas}}{t_{\rm ff}} \tag{4.73}$$

where $\rho_\star$ is the stellar density, $\epsilon_\star$ is the star formation efficiency, $\rho_{\rm gas}$ is the gas density and $t_{\rm ff}$ is the free fall time. As the free fall time is proportional to $\rho_{\rm gas}^{-1/2}$, the star formation density is given by $\frac{\mathrm{d}\rho_\star}{\mathrm{d}t} \propto \rho_{\rm gas}^{1.5}$ and provides a good 3D approximation to the observed one in 2D.

4.5.1 Discretization of the Star Formation

The aim of the star formation scheme is to reproduce the equation 4.73. This can be done through a stochastic approach that consists in randomly creating stars from gas particles (Katz et al., 1996; Springel and Hernquist, 2003). As stars are formed in cold and dense molecular clouds, a gas particle needs to respect the following criteria in order to have the possibility to form a star:

$$T < T_{\rm SF} \tag{4.74}$$

$$\rho > \frac{\pi}{4GN_{\rm Jeans}^{2/3}h^2}\gamma\frac{k_{\rm B}T}{\mu m_{\rm H}} \tag{4.75}$$

$$\nabla\cdot v < 0 \tag{4.76}$$

where $T_{\rm SF}$ is a temperature threshold. The three criteria correspond to a cold gas, with an unresolved Jeans length and currently collapsing. Once a particle has reached these criteria, a fraction of its mass can be converted into a stellar particle with the following probability:

$$p = \frac{m_g}{m_\star}\left(1 - \exp\left(-\frac{c_\star}{t_{\rm ff}}\Delta t\right)\right) \tag{4.77}$$

where m_g is the mass of the gas particle, $m_\star$ is the mass of the possible star[6], $c_\star$ the star formation efficiency, $t_{\rm ff}$ the free fall time and Δt the time step. Depending on the mass of the gas particles and the chosen mass for stars, two different modes are possible. If the mass of the gas particle is too low, we convert it directly into a stellar particle. In the opposite case, we remove a fraction of

---

[6] average mass of the gas particles in the initial conditions over the number of stars per gas particle required.

the mass from the hydro particle and create a stellar particle from it. When spawning a stellar particle from the gas particle (not simply fully transforming a gas particle into a stellar particle), we move both of them in a random direction and a random fraction of the smoothing length in order to avoid multiple particles at the same position. The displacement is done in order to respect the conservation of momentum.

4.6 Stellar Feedback

While only the gas is subject to the hydrodynamics, stars can still deeply impact the hydrodynamics through their different feedback mechanisms that change the properties of the gas (Katz et al., 1996). Three mechanisms often taken into account are the radiations which heats the gas through the absorption of photons, the stellar winds which continuously transfer mass from stars towards the gas and finally the supernovae which eject a large fraction of the star back into the surrounding gas (Hopkins et al., 2011). Among all stars, the most massive ones dominate the stellar feedback at the galactic scale. While the two first processes are acting during the whole life of a star, the supernovae are the final explosion that occurs at the end of a massive star's life. Different types of supernovae, and mainly two (Ia and II), exists depending on the mass and process generating the supernova. The first type is a supernova created by some binary systems while the second type concerns the most massive stars and is due to the collapse of their core.

All the previous processes strongly regulate the star formation as they heat the surrounding gas and prevent the formation of more stars. While the radiations and stellar winds are produced immediately after the formation of a star, the supernovae have an offset of a few Myr. Therefore, if the radiations and stellar winds do not reduce sufficiently the star formation rate, the supernovae can explode quickly the one after the other and remove most of the gas in the least massive galaxies.

Stars are responsible for the creation of the majority of metals present in our universe. All stars produces metals through their nuclear reactions related to their evolution (Pagel, 2009) and through mergers with other objects (e.g. neutron star mergers). The metals are then released mainly by supernovae, but in special cases, where the stellar envelop is well mixed with the internal layers of the star, the stellar winds can also contribute (e.g. Wolf-Raylet stars (Maeder, 2009)). Once the metals are released, the cooling of the gas becomes more efficient and can slightly decrease the regulation power of the feedback on the star formation.

Stars provide directly a large quantity of information observable. In the domain of dwarf galaxies, observers often rely on stellar properties to describe galaxies and thus stars provide excellent constraints on our simulations. For example, their velocity can be computed through the observation of their Doppler shift and, for the closest stars, their apparent displacement on the sky (Ramos et al., 2020; McConnachie and Venn, 2020). Their luminosity can be directly measured from the observations and the composition through the observation of spectral lines produced by the absorption of photons passing through the gas. Their age can be also computed, using color-magnitude diagrams, thanks to carefully designed model of stellar evolution taking into account the luminosity, temperature and metallicity. Finally, from the stellar ages, the star formation history of dwarf galaxies can be modeled.

In `SWIFT-GEAR`, we model the chemical enrichment including only the supernovae and, as mentioned before, are currently developing the radiative transfer that transports the radiations through the gas. Thus, once this work is done, we will be able to modelize the impact of the radiations produce by stars. Our model takes into account two different types of supernovae (SNIa and SNII). In the next sections, I will start by describing the initial mass function (IMF) that describes the distribution in mass of stars (section 4.6.1) and the stellar lifetime (section 4.6.2). By combining the two together, a supernovae rate can be derived for the SNII and also SNIa (sections 4.6.3 and 4.6.4). While they are both derived from the IMF and the lifetime, they are different in intensity but also in the delay between the formation of the stars and the first supernova. During their evolution, stars synthesize metals that will be released during the supernovae along with some energy. The quantity of metals released depends on the type of supernovae but also the mass of the progenitor in the case of SNII (section 4.6.5). To conclude with the stellar feedback, I will finish with the numerical details (section 4.6.6).

4.6.1 Initial Mass Function

In stellar evolution, the mass of a star is the most important parameter. It will decide for example the duration of each burning stage, which one can be achieved and most importantly the fate of the star. Therefore, a model for the distribution of stellar masses is required in order to determine the total number of supernovae. It can be obtained from observations (Kroupa, 2001). This distribution is called initial mass function (IMF). We define $\xi(m)\mathrm{d}m$, the fraction of stars found in the mass range $[m, m + \mathrm{d}m]$ [7]:

$$\xi(m) = B_i m^{-\alpha_i} \tag{4.78}$$

where

$$\begin{cases} \alpha_0 = 0.3, & 0.01 \leq m/M_\odot < 0.08, \\ \alpha_1 = 1.3, & 0.08 \leq m/M_\odot < 0.50, \\ \alpha_2 = 2.7, & 0.50 \leq m/M_\odot < 1.00, \\ \alpha_3 = 2.3, & 1.00 \leq m/M_\odot, \end{cases} \tag{4.79}$$

B_i are coefficients that ensure a continuity and the normalization of the IMF (see appendix VI for more details on the computation). One needs to be careful with the definition of the initial mass function as they can be defined either in number of stars (ξ) or in mass of stars (ϕ). In a power law relation such as the one of Kroupa, 2001, the exponent is simply increased by one in order to obtain the IMF in mass from the one in number and the normalization needs to be recomputed ($\phi \propto m^{1-\alpha_i}$). The lower mass limit ($0.01 M_\odot$) is due to the inability of low mass object to start the required nuclear reactions for the definition of a star and the upper mass limit is due to the

---

[7] In `SWIFT-GEAR`, we define the law as $\xi \propto m^{\alpha_i}$

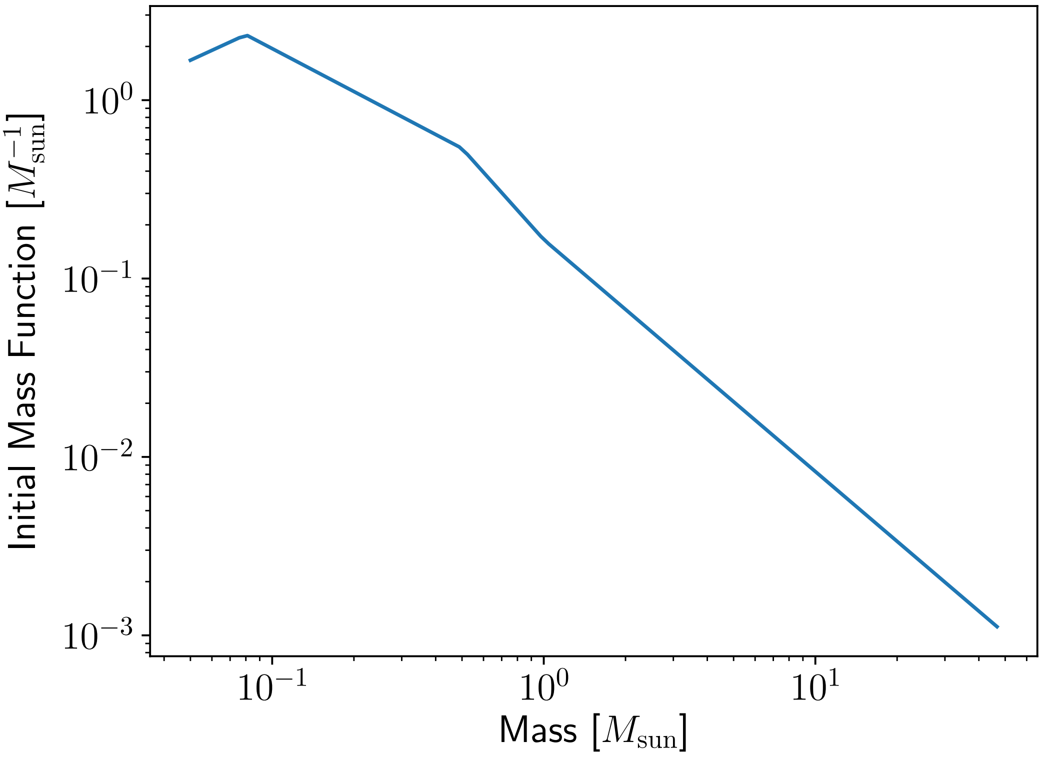


Figure 4.5 – The Kroupa Initial mass function in mass. Most of the stars have a low mass which means that due to their lifetime, they will spend almost the whole simulation on the main sequence and not produce any supernovae.

radiations that overcome the Eddington limit[8] ($50 M_\odot$ in `SWIFT-GEAR`). In Figure 4.5, the initial mass function in mass by Kroupa, 2001 is shown and, as it can be seen, most of the stars have a low mass. Low mass stars will spend almost their whole life on the so called main sequence and not produce any supernovae.

4.6.2 Lifetime

The initial mass function gives the total number of supernovae. However the information of when they are exploding is still missing. Stellar models have been developed for a long time and provide accurate predictions. Among them, the lifetime of stars is particularly important for cosmological simulations as it can be used along with the initial mass function to predict the rate of supernovae. Due to their complexity, approximate analytical models have been developed. In

---

[8] The limit is defined by the equilibrium between the gravitational and radiation forces at the surface of the star.

`SWIFT-GEAR`, we use the following approximation (Poirier, 2004):

$$\log(\tau(m)) = a(Z)\log^2(m) + b(Z)\log(m) + c(Z) \tag{4.80}$$

$$a(Z) = -\ \ 40.110Z^2 + \ \ 5.509Z + 0.7824 \tag{4.81}$$

$$b(Z) = \ \ 141.929Z^2 - 15.889Z - 3.2557 \tag{4.82}$$

$$c(Z) = -261.365Z^2 + 17.073Z + 9.8661 \tag{4.83}$$

where τ is the lifetime in yr, Z the star's metallicity and m its mass in $M_\odot$. As it can be seen in Figure 4.6, the lifetime depends mostly on the mass. The metallicity only slightly increases the lifetime of a star, thus it can delay the apparition of the first supernovae and reduces the regulation of the star formation through the stellar feedback.

4.6.3 Supernovae Type II

Supernovae of type II (SNII) are the final stage of the most massive stars. Once they start producing iron in their core, the fusion in the stars becomes endothermic and thus cannot produce enough radiation to compensate the gravity. In consequence, the star will start to collapse faster and faster until bouncing against a core dense enough to reach the neutron degeneracy and explode into a supernova.

A star will explode once reaching the end of its life, thus the SNII rate can be simply computed through a convolution between the star formation history and the initial mass function (Scalo, 1986; Tinsley, 1980; Kobayashi et al., 2000):

$$\dot{N}_{\mathrm{SNII}}(t) = \int_{M_l}^{M_u} \psi(t - \tau(m)) \frac{\phi(m)}{m} \mathrm{d}m \tag{4.84}$$

where M_l and M_u are the lower and upper mass limits to produce a SNII ($8M_\odot$ and $50M_\odot$ in `SWIFT-GEAR`), ψ is the star formation rate (SFR), τ is the lifetime function and ϕ is the initial mass function in mass. The SFR corresponds to the new mass of stars per unit of time and, in the case of a single population, is given by $m_\star\delta$ where δ is the Dirac function and $m_\star$ the total mass of stars.

4.6.4 Supernovae Type Ia

In binary systems, the Roche lobe is the area where the matter is gravitationally bound to one of the star. If one of the star expands until reaching the Roche lobe of the other star, the matter

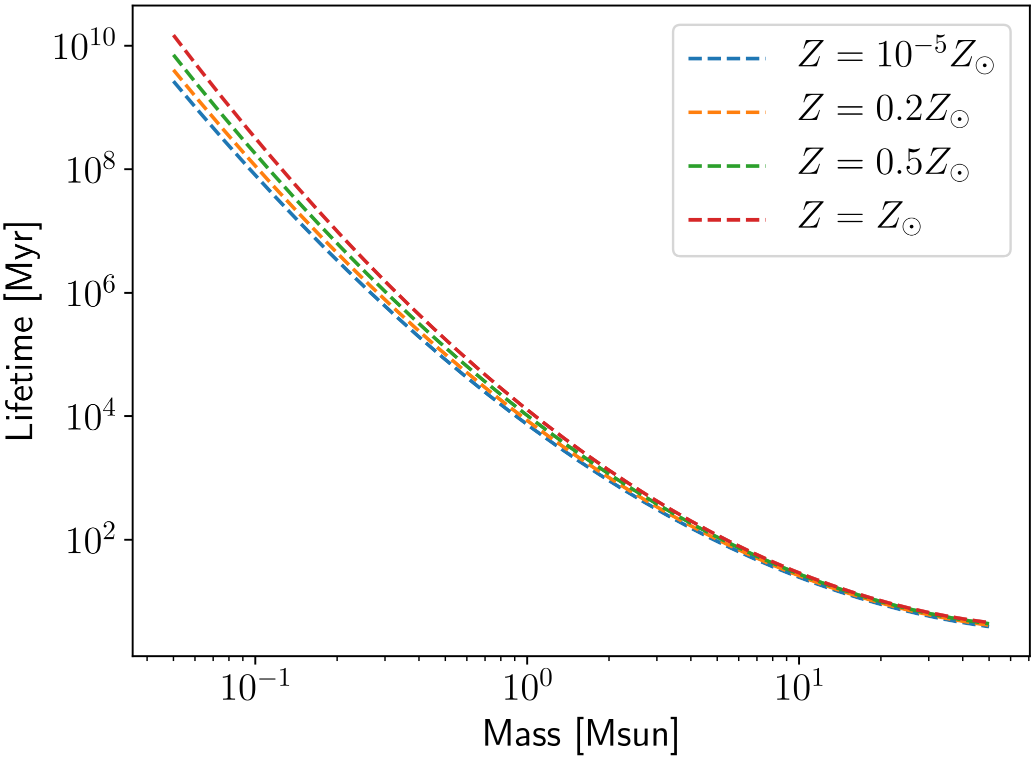


Figure 4.6 – Lifetime of the stars as a function of their mass and metallicity. The lifetime is mainly described by the mass while the metallicity is able to slightly delay the supernovae and thus the regulation of the star formation through the stellar feedback.

Table 4.2 – Parameters used for the SNIa rate.

Companion	$M_{d,l,i}$	$M_{d,u,i}$	b_i
Red Giants	0.9	1.5	0.02
Main Sequence	1.8	2.5	0.05

will be transferred from the expanding star towards the companion. In the case where one of the companion is a white dwarf, the mass can increase until reaching the Chandrasekhar limit (about $1.38 M_\odot$) where the electron degeneracy pressure is not sufficient to support the surrounding mass and thus the star explodes into a supernova of type Ia (SNIa).

The computation of the SNIa rate is done in two steps (Kobayashi et al., 2000). First the number of white dwarfs needs to be evaluated and then the fraction that will form a binary system. Knowing the mass limits of the white dwarf's progenitor, one can compute their number from the initial mass function:

$$N_{\mathrm{WD}} = \int_{M_{p,l}}^{M_{p,u}} \frac{\phi(m)}{m} \mathrm{d}m \tag{4.85}$$

where $M_{p,u}$ and $M_{p,l}$ are the upper and lower mass limits ($8 M_\odot$ and $3\mathrm{M}_\odot$ in `SWIFT-GEAR`) and ϕ the initial mass function in mass. The second step requires a model for the total number of binaries and the mass distribution of companions. As constraints on both of them are weak, the model chosen is a simple power law mass distribution $\phi_{d,i} \propto m^{-0.35}$ normalized to 1 inside the interval of a given companion and with b_i being a constant fraction of binary systems (Poirier, 2004). The rate of SNIa is finally given by:

$$\dot{N}_{\mathrm{SNIa}} = \left(\int_{M_{p,l}}^{M_{p,u}} \frac{\phi(m)}{m} \mathrm{d}m \right) \sum_i b_i \int_{M_{d,l,i}}^{M_{d,u,i}} \psi(t - \tau(m)) \frac{\phi_{d,i}(m)}{m} \mathrm{d}m \tag{4.86}$$

where the sum is over the different types of companion (red giants and main sequence stars in `SWIFT-GEAR`, see table 4.2), $M_{d,u,i}$ and $M_{d,l,i}$ are the lower and upper limits for the companion and ψ is the star formation history.

In Figure 4.7, the supernovae rate for both SNII and SNIa is shown as function of time and metallicity for a single stellar population. The two types of supernovae have very different timescales, the SNII explode in less than 100 Myr while the SNIa in more than 1 Gyr. As SNII explode quickly after their formation, they tend to explode in high density regions (e.g. molecular clouds) and disrupt them. Thus, the SNII are regulating the early star formation. In `SWIFT-GEAR`, the supernovae are supposed to release the same amount of energy into the surrounding gas, therefore SNII have a far larger impact on the galactic evolution than SNIa due to their larger number and lower lifetime. Due to the low probability of having a correct binary system for SNIa, the ratio between the number of SNII and SNIa is roughly 100. It is worth mentioning that due to the extended lifetime of metal rich stars, both supernovae rates are slightly shifted to later time with increasing metallicity.

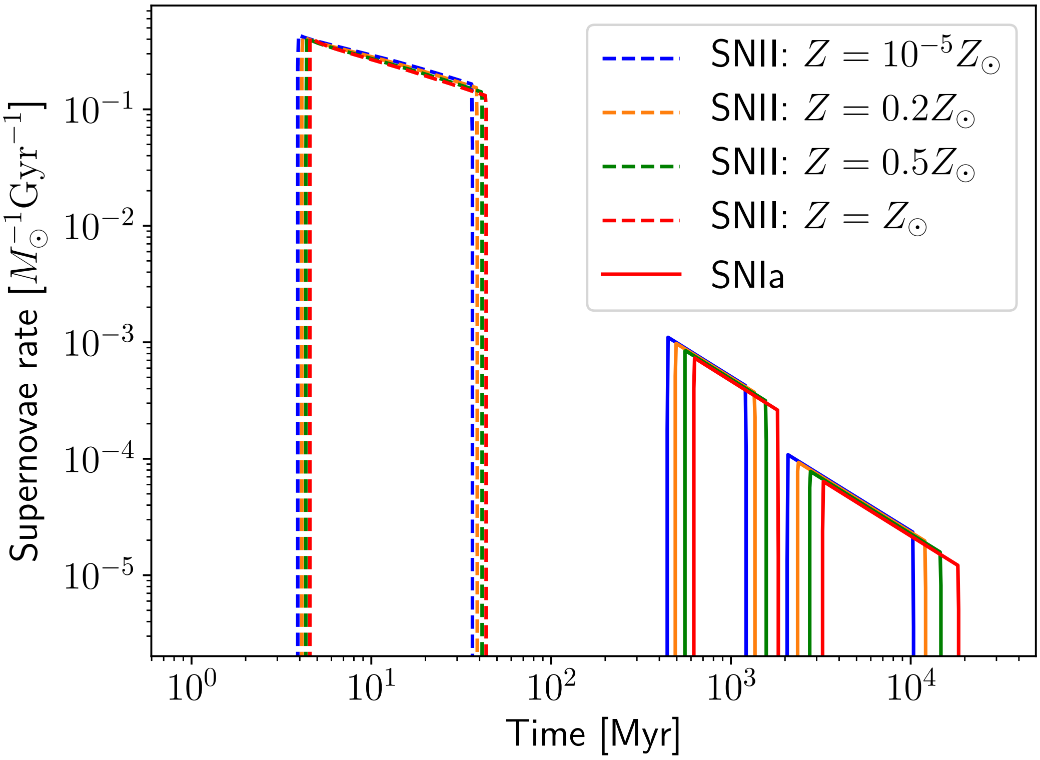


Figure 4.7 – Supernovae rate as function of time for SNII and SNIa at different metallicities. The SNII explode quickly after the formation of the stars and regulate the star formation. The SNIa explode at a late time and thus cannot prevent the formation of new stars. The metallicity slightly delays the explosion of the supernovae as it impacts the lifetime of the stars.

4.6.5 Supernovae Yields

During a supernova, a large fraction of the mass is released into the surrounding medium along with a fraction of the metals processed by the star during its life (hydrostatic fusion). Some elements are even processed during the supernova as the collapse increases both the density and temperature and allows new nuclear reactions to start (explosive fusion). The resulting elements are show in Figure 4.8 (Tsujimoto et al., 1995; Cescutti and Chiappini, 2013). While the yields of SNII are mass dependent (lines), the ones from SNIa are constant (dots). This is due to the relative importance of hydrostatic and explosive fusion. Depending on the mass of the star, the hydrostatic elements processed will change but will be almost independent from the mass for the explosive fusion (Maeder, 2009) As the SNIa are formed from low and intermediate mass stars, they do not contain a large amount of elements coming from the hydrostatic fusion and are dominated by the explosive fusion. Two elements are specially important for us: the iron and the magnesium. On one hand, the magnesium is produced during the carbon burning phase of stellar evolution reached only by massive stars. Thus they are the only one able to produce it in large quantity and release them through SNII. On the other hand, the iron is mostly coming from the explosive fusion. In this case, SNIa are strongly contributing to the iron enrichment of galaxies. Therefore, the abundances of magnesium and iron provide insights on the history of a dwarf. A high amount of iron indicates a late time as SNIa explode relatively late and many supernovae are required to get such enrichment. In the case of magnesium, a large abundance with respect to iron ($[\mathrm{Mg}/\mathrm{Fe}] = 0.5$) indicates an enrichment done only by SNII while a low one ($[\mathrm{Mg}/\mathrm{Fe}] = -1.5$) indicates a strong contribution from SNIa.

4.6.6 Discretization of the Stellar Feedback

The number of supernovae during a time step is obtained from equations 4.84 and 4.86 and is simply given by:

$$N_{\mathrm{SNIa}} = \dot{N}_{\mathrm{SNIa}}(t)\mathrm{d}t, \tag{4.87}$$

$$N_{\mathrm{SNII}} = \dot{N}_{\mathrm{SNII}}(t)\mathrm{d}t. \tag{4.88}$$

When the time step is small enough to produce less than a few supernovae, we will explode a fraction of supernovae in a continuous and thus non-physical way (e.g. 0.2 supernovae every time step and not one supernova every five steps). In order to fix this, we can remove the floating part and use it as a probability of obtaining an additional supernova. In this thesis, I will call this approach the discrete supernovae (and the other one the continuous supernovae). In Revaz et al., 2016, they are called respectively the Random IMF Sampling (RIMFS) and Continuous IMF Sampling (CIMFS).

Once the number of supernovae is computed for a given time step, we can compute the properties of the ejecta. The mass released by a supernova can be decomposed in two terms: the one

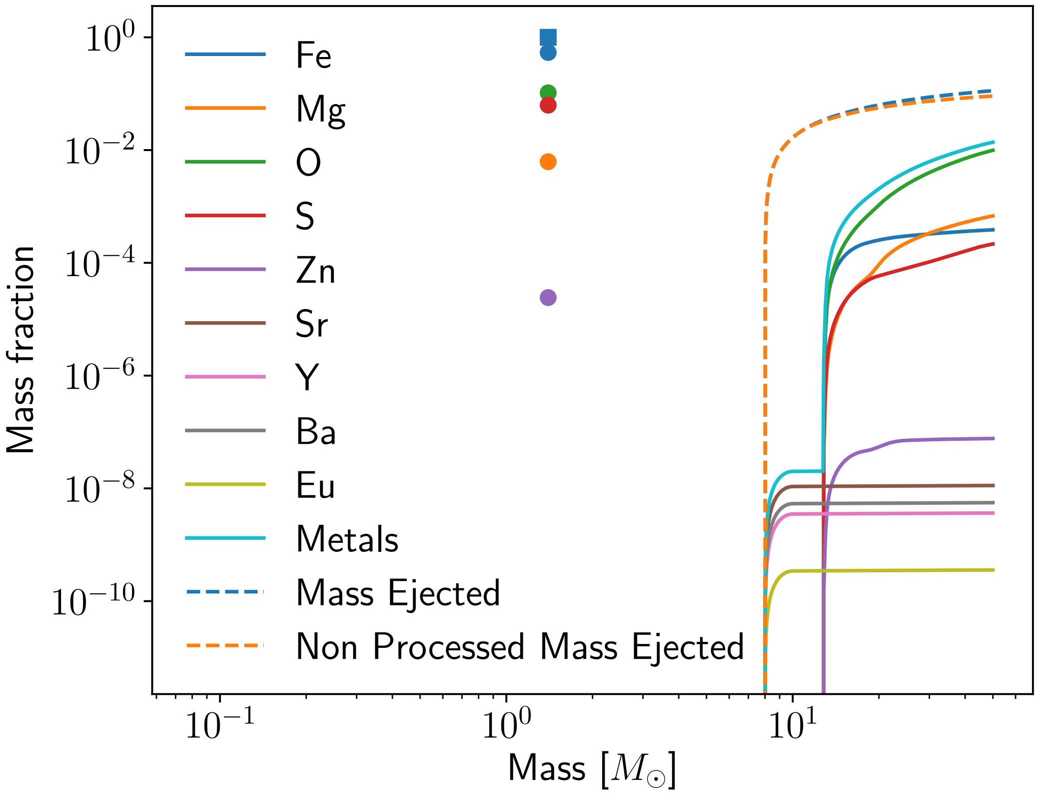


Figure 4.8 – Each line represents the mass of a given element ejected into the surrounding gas normalized by the mass of the star. "Metals" corresponds to the sum of all the elements considered as metals, "mass ejected" corresponds to the total fraction of mass ejected (with a metallicity given by the previous lines) and "non processed mass ejected" corresponds to the mass ejected that did not undergo fusion (e.g. with the initial metallicity of the star). The dots are mass fraction for SNIa and the square is for the mass fraction ejected by a SNIa.

containing metals processed by the star (through both hydrostatic and explosive fusion; in the following equations the index p) and the one that was not processed by the star (index np). Then the metals ejected are determined by two parameters: the mass of the exploding stars and the initial metallicity. For SNIa, all the stars have the Chandrasekhar mass and produce the same ejecta. It means that all the SNIa have the same properties and the metals are directly taken from the table. For the SNII, the situation is a bit more complicated, especially for the continuous sampling. Therefore let us start with the discrete supernovae.

Discrete Supernovae

The mass fractions f_i for the yields of SNII are simply taken directly from the table (shown in Figure 4.8) at a mass $M_{\mathrm{avg}} = 0.5\,(m_1 + m_2)$. The mass $m_1 = m(t)$ and $m_2 = m(t + \mathrm{d}t)$ are the mass of the stars exploding at the beginning and end of the time step. They are defined with the Equation 4.80 and as the time is a monotonically decreasing function, the maximal time and mass are inverted (e.g. $t < t + \mathrm{d}t$ but $m_1 > m_2$). Then the mass ejected (M_{ej}) and mass of metals ejected (M_Z) are given by:

$$M_{\mathrm{ej}} = N_{\mathrm{SNII}} f_{\mathrm{p}} M_{\mathrm{avg}} + N_{\mathrm{SNIa}} M_{\mathrm{p,\,SNIa}} \tag{4.89}$$

$$M_{\mathrm{Z}} = \left(f_Z + m_Z f_{\mathrm{np}}\right) M_{\mathrm{avg}} N_{\mathrm{SNII}} + M_Z N_{\mathrm{SNIa}} \tag{4.90}$$

where $M_{\mathrm{p,\,SNIa}}$ is the mass ejected per SNIa, m_z is the mass fraction of metal at the birth, M_Z is the mass of metals ejected per SNIa and f_Z represents the mass fraction of metals ejected by SNII.

Continuous Supernovae

In the case of continuous supernovae, the yields for SNII need to be averaged over the initial mass function:

$$F_i(x) = \int_{m_{\mathrm{min}}}^{x} \phi(m) f_i \mathrm{d}m, \tag{4.91}$$

where F_i is the mass ejected per mass of the single stellar population, m_{min} is the minimal mass for a SNII and ϕ is the initial mass function. Then the mass ejected is given by:

$$\Delta F_i = F_i(m_2) - F_i(m_1) \tag{4.92}$$

$$M_{\mathrm{ej}} = \Delta F_{\mathrm{p}} M_{\mathrm{init}} + N_{\mathrm{SNIa}} M_{\mathrm{p,\,SNIa}} \tag{4.93}$$

$$M_{\mathrm{Z}} = \left(\Delta F_Z + m_Z \Delta F_{\mathrm{np}}\right) M_{\mathrm{init}} + M_Z N_{\mathrm{SNIa}} \tag{4.94}$$

where M_{init} is the initial mass of the stellar population,

Energy and Metals Injection

The energy from supernovae is directly injected into the surrounding particles with a weight corresponding to the kernel as described in Stinson et al., 2006. The same is done with the total mass ejected and the mass of each metal ejected. In order to conserve the mass and the momentum, the mass of both the gas particles and stellar particles, and the velocity of the gas particles are updated:

$$w_{ij} = \frac{x_j W_{ij}(h_i)}{y_{\text{gas},i}}, \tag{4.95}$$

$$y_{\text{gas},i} = \sum_{j=\text{gas}} x_j W_{ij}(h_i), \tag{4.96}$$

$$\Delta m_j = \sum_{i=\text{stars}} M_{\text{ej},i} w_{ij}, \tag{4.97}$$

$$\Delta m_{\text{Z},j} = \sum_{i=\text{stars}} M_{\text{Z},i} w_{ij}, \tag{4.98}$$

$$\Delta u_j = \sum_{i=\text{stars}} \frac{E_{\text{SN},i}}{m_j + M_{\text{ej},i} w_{ij}} w_{ij}, \tag{4.99}$$

$$\Delta p_j = \sum_{i=\text{stars}} M_{\text{ej},i} w_{ij} \left(v_i - v_j \right). \tag{4.100}$$

where w_{ij} is the weight, $y_{\text{gas},i}$ is the SPH normalization, Δm_j, $\Delta m_{\text{Z},j}$, Δu_j and Δp_j are the mass, mass of metals, internal energy and momentum received by gas particles from stars. While we could use different flavors for x and y depending on the SPH model, in both `GEAR` and `SWIFT`, it has been fixed to a density based scheme (e.g. x is the mass and y the density). During this computation, we also ensure that particles touched by a supernova will be active during the next step[9] and set their viscosity parameter to the maximum in order to anticipate the shock due to the supernovae. Before the drift in the next step, we update all the variables. The internal energy is updated in the following way:

$$u = \frac{u m_j}{m_j + \Delta m_j} + \Delta u_j, \tag{4.101}$$

in order to take into account the impact of the additional mass to the specific internal energy. For the two masses and the momentum, a simple sum is performed. Finally, we also include a time step limiter. As its name indicates it, it limits the time step of the particles in order to keep a low difference in time step between two neighbors as presented in Durier and Dalla Vecchia, 2012. This is applied to all the particles and not only the ones touched by a supernova.

Cooling Catastrophe

As mentioned in the introduction, a gas particle heated by a supernova ends up in a regime that cools down extremely rapidly (see Figure 4.4). It can remove all the energy inserted by

---

[9] This is called synchronization in `SWIFT`.

a supernova in a few steps. Thus, the radiative cooling strongly reduces the stellar feedback efficiency. This phenomenon is called the cooling catastrophe and can be fixed by applying a delay between a supernova and the next computation of the radiative cooling. In `SWIFT-GEAR`, we use a delay of 5 Myr. The radiative pressure produced by young stars is believed to reduce this problem along with non-thermal effects such as magnetic field, turbulence and cosmic rays (Teyssier et al., 2013; Smith et al., 2018).

4.7 Smoothed Metallicity

As the metals deeply impact the cooling function, each gas particle contains the mass fraction of the different metals. This mass fraction increases each time a supernova explodes close to a gas particle and thus produces strong scatters between the particles polluted by supernovae and the one that did not. Therefore, in order to follow the idea of SPH, the metallicities are smoothed over the kernel radius for all the particles. At our current resolution (about 1000 $M_\odot$), it reproduces correctly the scatter in the abundances of stars. The details of the method are given in Wiersma et al., 2009b; Okamoto et al., 2005; Tornatore et al., 2007 and can be summarized by the following equation:

$$\bar{f}_{Z,i} = \frac{1}{y_i} \sum_j x_j f_{Z,j} W_{ij}(h_i) \tag{4.102}$$

where $f_{Z,i}$ ($\bar{f}_{Z,j}$) is the (smoothed) mass fraction of metal. As before, even if we could adapt the weights with the SPH, we are always using a density based scheme in this equation.

It is worth to mention that this smoothing is not equivalent to solving a diffusion equation even if it is able to produce comparable solution in our simulations. The metals are never exchanged between particles, only smoothed over the smoothing length. As the characteristic length for the smoothing depends on the resolution, this method will smooth the metals on distances far too large for the diffusion in low resolution simulations and far too small in high resolution. The smoothed metallicity is therefore only a way to stay within the concept of the smoothed particle hydrodynamics and will need an additional diffusion mechanism.

4.8 Time Integration

When simulating a gravitational problem over a long time, it has been shown that symplectic methods are better than traditional methods (Euler, Runge-Kutta, ...). This is due to the fact that Hamiltonian systems are not structurally stable against non-Hamiltonian perturbations and the symplectic methods are designed to produce a Hamiltonian perturbation when discretizing the equations (Hairer et al., 2010; Springel, 2005). When a single time step is used with such symplectic methods, the energy oscillates around the initial energy and the amplitude of the oscillation depends on the size of the time step (see for example Fig. 2.3 in Hairer et al., 2010).

Thus, it conserves the energy in average. Unfortunately, this behavior is not guaranteed with an adaptive time step as the perturbation becomes time dependent. Furthermore, the individual time steps for the particles destroy the symplectic nature of the Hamiltonian. As we are simulating collisionless systems, the individual time steps do not really matter. The motions are dominated by the collective potential, and thus we recover more or less a Hamiltonian system (Springel, 2005). For the error due to the adaptive time step, no good solution currently exists (see Fig. 6 in Springel, 2005 for an illustration of the error). It means that, as with other methods, we will not be able to properly conserve the energy, but globally, the conservation of the energy will be better than with traditional methods. Therefore, we use the symplectic method called the kick-drift-kick leapfrog integrator that consists in two half kick[10] and a drift[11] (kick1: $t \rightarrow t + \frac{1}{2}\mathrm{dt}$, drift: $t \rightarrow t + \mathrm{dt}$, kick2: $t + \frac{1}{2}\mathrm{dt} \rightarrow t + \mathrm{dt}$).

As `SWIFT` performs cosmological simulations, the concept of time is not as trivial as usually and requires the definition of four different time operators that are used within the drift and kick:

$$x\,(t + \Delta t) = x(t) + v(t)\mathrm{dt}_{\mathrm{drift}}, \tag{4.103}$$

$$u\,(t + \Delta t) = u(t) + \frac{\mathrm{d}u(t)}{\mathrm{d}t}\mathrm{dt}_{\mathrm{therm}}, \tag{4.104}$$

$$v\,(t + \Delta t) = v(t) + a_{\mathrm{hydro}}(t)\mathrm{dt}_{\mathrm{kick,\ hydro}} \tag{4.105}$$

$$+ a_{\mathrm{grav}}(t)\mathrm{dt}_{\mathrm{grav,\ hydro}} \tag{4.106}$$

where x, v and a are the position, velocity and acceleration, and dt are the operators. The acceleration is split into a term due to the hydrodynamics and one for the gravity due to the separation of the computation within `SWIFT`[12]. Usually, the operators are simply the difference in time between two steps: $\mathrm{dt} = \int_t^{t+\Delta t} \mathrm{d}t$. In `SWIFT`, they include also the scale factor dependency

---

[10] update of the velocities

[11] update of the positions

[12] In fact, even the gravity is split in two as the spectral method is computed on larger time steps than the FMM.

of each equation in order to increase the accuracy of the numerical integration:

$$\mathrm{dt}_{\text{drift}} = \int_t^{t+\Delta t} \frac{1}{a^2} \mathrm{d}t, \tag{4.107}$$

$$\mathrm{dt}_{\text{therm}} = \int_t^{t+\Delta t} \frac{1}{a^2} \mathrm{d}t, \tag{4.108}$$

$$\mathrm{dt}_{\text{kick, hydro}} = \int_t^{t+\Delta t} \frac{1}{a^{3(\gamma-1)}} \mathrm{d}t, \tag{4.109}$$

$$\mathrm{dt}_{\text{kick, grav}} = \int_t^{t+\Delta t} \frac{1}{a} \mathrm{d}t. \tag{4.110}$$

$$\tag{4.111}$$

These integrals can be computed numerically with a high resolution allowing a more accurate time integration of all the equations.

5 Implementation

Deeds will not be less valiant because they are unpraised.
— J. R. R. Tolkien, The Return of the King

Now that the models have been presented, we can focus on how they are implemented within the code `SWIFT` along with the architecture of the code. `SWIFT` is a project that started around 2012 by M. Schaller and P. Gonnet with a strong focus on modularity and algorithmic improvements. For a physicist, the modularity of a code is an important point as it means that someone can improve the physics without any knowledge of the underlying structure of the code. We were able to achieve this thanks to a very diverse group developing `SWIFT`. As each subgroup focuses on different physics (2 cosmological, a planetary and some engineering models), it forced us to develop a common High Performance Computing (HPC) architecture with a strong separation of the physics.

The main issue with our old code `GEAR`, based on `GADGET`, comes from the large idle time of the CPUs. This is due to the bad repartition of the work between different MPI ranks. To reduce this problem, `SWIFT` is using pthreads to take advantage from the shared memory on a single node of a computational server. Most of the work within `SWIFT` is split into small tasks that are shared between pthreads in order to help each others without the need to wait. As large simulations require large amount of memory and more cores than a single computational node can provide, at some point, the simulations will be run on machines that do not share memory anymore. In such cases, `SWIFT` is using MPI to send messages between the different nodes. The repartition is based on the amount of work of each task and thus is able to accurately predict the amount of work of each node. `SWIFT` is now in a state where new models can be easily implemented and has shown to be largely faster than `GADGET` (Borrow et al., 2018).

In this chapter, I will start by providing the general ideas of `SWIFT` in more details than the previous paragraph along with a few definitions. Then, as the main improvement of `SWIFT` is its task based system, a complete description of the tasks is provided. First, I will start with a general picture of the task system and some functions that are necessary and outside the task system. Then, I will describe, in the order of the dependencies, the tasks for the drift operators, gravity, hydrodynamics, cooling, kick operator, star formation, stellar feedback and finish quickly with the time step limiter. Here, I consider only `SWIFT-GEAR` with the pressure-energy flavor as it would take far too many pages to describe all the different physical modules implemented in `SWIFT`, and it would not represent my own work. I will conclude this chapter with some information on the snapshots provided with `SWIFT` and two small new models for `SWIFT-GEAR` that I have implemented.

To clearly separate my own work with the one of the collaboration, I will start by quickly mentioning my contributions related to this chapter (from the largest to the smallest):

- The tasks related to the stellar feedback,
- All the subgrid models presented in the previous chapter,
- The creation of the modules for the pressure floor and the chemistry,
- Designing and implementing the graphs for the task dependencies that were totally absent from `SWIFT` when I joined.
- Some small improvements to make `SWIFT` more user friendly. Two examples are the configuration option "–with-subgrid=GEAR" that allows to easily run a simulation with a given model without knowing all the options to use, and the writing of two YAML files before the first step. One file represents all the parameters used (including the default ones that were not written in the configuration file) and the second one all the parameters that were defined but not used.

Finally, the SPH, gravity and cosmology, provided in this thesis, were the work of J. Borrow and M. Schaller.

5.1 General Description of SWIFT

A key concept behind the code efficiency is the task based approach that enables a very low idle time for the CPUs. Let us start by explaining how it works with the simple example of task dependency graph given in Figure 5.1 and then I will explain in details how it is done within `SWIFT`. This example describes the first steps of a day in a nice task based and theoretical world. The graph is composed of ellipses and diamonds that represent all the different tasks.

Initially, all the tasks are stored within a scheduler that enqueues the tasks when they are ready to be used. To figure out which tasks should be inside the queue, the scheduler looks at the arrows that indicate the dependencies between tasks and picks only the tasks without any unsolved incoming dependencies. Here it means that initially the queue contains both “Receiving Comments” and “Waking up”. Both tasks can be done in parallel without any issue, so let us focus on the first one. This one represents a MPI communication (represented by a diamond shape) and the main work is done when the scheduler enqueues the task by creating the asynchronous communication. Once the communication is received, this task is flagged as done and removed from the queue.

Now, the only remaining task within the queue is to wake up. Once done, this task unlocks and enqueues both cooking tasks. As they both represent the action of cooking, they are grouped together within a cluster (rectangle) and cannot be done at the same time. Indeed, in this theoretical world, a stove can be used only for one ingredient at the time and thus a conflict exists between the two tasks (not indicated in this type of graph). After this hard work, the cook is allowed to spend a bit of time daydreaming where he does nothing directly useful for the breakfast. Such tasks are called implicit and are shown in gray.

This world puts a lot of pressure on the cooks as they always need to send a picture of their meal to someone else and to wait on an answer before finally being able to eat it. Again sending the picture is immediately done asynchronously when the task is enqueued and marked as done only once the communication is over. Now the task receiving comments can finally be marked as done and unlocks the possibility to eat the eggs and bacon. Through this example, I have introduced the different type of tasks: implicit, communication and the others that I will call computational tasks, the dependencies with the arrows and the conflicts that are not shown.

Let us go back to `SWIFT` now and give a more detailed picture of the architecture. Each task represents a fraction of the work required in a single step. They are defined by a volume, the particles (approximately) contained within it and a subset of the equations solved (e.g. computation of the gas density, smoothing lengths, stellar feedback, star formation, ...). As the equations have an order, some dependencies exists between the tasks. For example, the gas density needs to be computed before the hydrodynamics forces and thus the corresponding tasks needs to be done in the same order. It is worth to mention that `SWIFT` considers only the **active** tasks for the dependencies. So in this previous example, if the density is inactive, the forces would not inherit the dependencies of the density and, thus, would have no dependencies.

Once all the tasks required for a simulation are defined, they are sorted within the scheduler and added into a queue if all their dependencies are solved. Every time a task is completed, the tasks depending on it are added to the queue if they have no other missing dependencies. Then, the pthreads ask the scheduler to provide a new task from the queue. As all the pthreads share the same queue, they help each other to finish all the tasks at the same time. With large quantity of tasks and totally independent ones (e.g. without conflicts and without dependencies), it ensures that the pthreads are never idle during a step except for the last few tasks. As we will see in the next chapter, this is not necessary the case in reality.

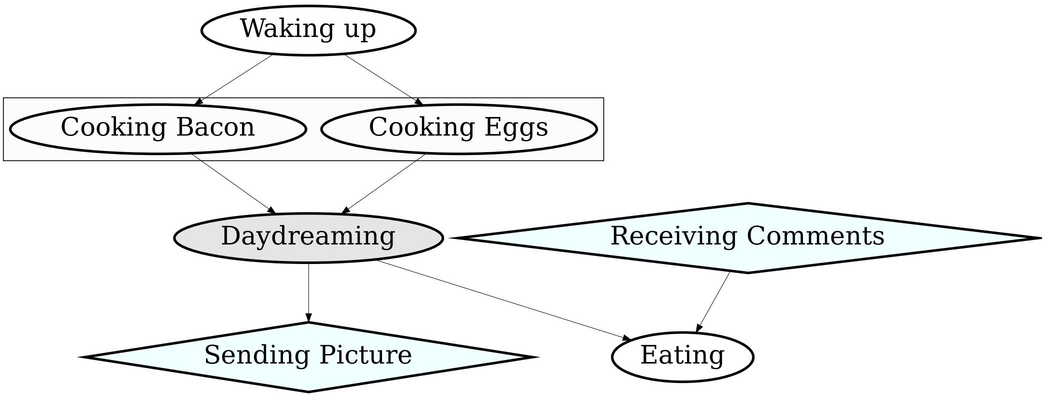


Figure 5.1 – Example of task dependencies for a wake up routine. The first step is to wake up, of course, and, then, to cook the breakfast. As it is impossible to start cooking without waking up first, a dependency is indicated with the arrow. Cooking the bacon and the eggs are two different tasks, but still represent the same action and are thus grouped within a cluster (rectangle). Both require a stove and, in this theoretical world, they cannot be done simultaneously. This is called a conflict between the two tasks (not indicated in the dependency graph) and prevents race conditions. Once the meal prepared, in this model, some time is dedicated to daydreaming and thus nothing directly productive (called an implicit task and represented in grey). Here, it is impossible to eat something without sending a picture of the food first and receiving comments about it (communication tasks are shaped in diamond). Once the comments are received, the breakfast can be finally eaten and the task system stopped.

To define the volume of each task, the total volume of the simulation is divided in top level cells according to a regular mesh and then all the top level cells are divided in octree[1] until each leafs contains a maximal number of particles (by default 400). Different types of tasks exist according to the volume they act on and can be split into two categories: self and pair tasks. The self tasks act on a single cell while the pairs interact two cells together. All are initially created on the top level cells and then through a recursion pushed towards the leafs until some criteria are meet. Among them, the most important criterion is based on the smoothing lengths of the stars and gas. A task cannot act on a cell smaller than the maximal smoothing length of its particles in order to ensure that all the neighbors of the particles are within the closest cells. `SWIFT` also uses implicit tasks that are not doing any computation, but greatly simplify the dependencies. In some cases, two tasks sharing the same volume can be running at the same time and create a race condition where two pthreads try to modify the same variable at the same time. To solve this issue `SWIFT` contains the principle of conflicts between tasks where only a single task can act on a given volume at a time. When the scheduler obtain a second task acting on a volume currently in use, it simply skip the task for now and try the next one in the queue.

For the distributed memory parallelization (done using MPI in `SWIFT`), each processing unit is called a rank. `SWIFT` distributes the work by assigning each top level cell to a single rank. As the memory is not shared anymore, each rank will need some information from the computations done by the other ranks during steps. In order to optimize the code, the communications should always be hidden by some computations and kept as low as possible. Thanks to its task based approach, `SWIFT` defines asynchronous communication tasks that will be automatically hidden, and can accurately measure the computational weight of each task contained within a top level cell (including both communication and physical tasks). The measurement can then be used in the graph partitioning library `METIS` [2] in order to equalize the work between ranks.

5.1.1 Particle Types

7 different types of particles are available in order to simulate the different type of physics and each have their associated tasks: gas (in the code it corresponds to the structure `part`), gravity (`gpart`), gravity background (`gpart`), sinks (`sink`; discussed in chapter 8), stars (`spart`), black holes (`bpart`; not discussed in this thesis) and neutrinos (`gpart`; not discussed in this thesis).

- The `parts` are subject to the hydrodynamics, radiative cooling, star formation, stellar feedback and time step limiter.
- The `sparts` are subject only to the stellar feedback and are created from the star formation tasks.
- The `gparts` are subject only to the gravity tasks and are the only ones to be used within such tasks.

---

[1] An octree is a way to recursively split the cells. In 3D, the cells are split in 8 cubes.

[2] http://glaros.dtc.umn.edu/gkhome/metis/metis/overview

Table 5.1 – List of the main variables contained within hydrodynamics particles (`parts`).

Variable Name	Description
x	Position
v	Drifted velocity (for inactive particles)
v_{full}	Velocity at the last active step
a_{hydro}	Acceleration due to hydrodynamics
a_{grav}	Acceleration due to gravity
m	Mass
u	Internal (specific) energy
h	Smoothing length
ρ	Density
$\bar{P}$	Weighted pressure (y in the general description of SPH)
n_i	Weighted number of neighbors ($\tilde{y}$ in the general description of SPH)
Z_{frac}	Non-smoothed metallicity
$\bar{Z}_{\text{frac}}$	Smoothed metallicity
c_s	Speed of sound
B	Balsara switch (equation 4.58)
Δp	Momentum received by supernovae
Δm	Mass received by supernovae
Δu	Internal (specific) energy received by supernovae
v_{sig}	Maximal signal velocity (speed of sound modified by the viscosity)
dt	Time step

As only the `gparts` are used in the gravity tasks, for each non `gpart`, we have an equivalent `gpart` that allows us to compute independently the gravity from the rest of the physics (meaning without task conflicts). For example, in the case of `parts`, the `part` is linked to the corresponding `gpart` with a pointer and in the other direction by the ID of the `gpart` (negative value of the index within the global array). Every time we need to drift all the particles, we synchronize the position of the two particles and in every kick we synchronize the velocities. In the case of `gparts` and in simulations using the zoom technique (see section 9.2 for more details), we use two different types of particles that share the same structure. The low resolution particles that corresponds to boundary conditions are flagged as being `swift_type_dark_matter_background` while the high resolution particles are flagged as being `swift_type_dark_matter`. This distinction allows to quickly select the particles belonging to the area of interest. In Tables 5.1 - 5.3, the variables used in this chapter and related to the different particles are given.

5.1.2 Task System Within `SWIFT`

As `SWIFT` is a task based code, it is important to understand how the tasks depend on each others. The best way to do it is through the so-called task dependency graph produced by the code at

Table 5.2 – List of the main variables contained within star particles (`sparts`).

Variable Name	Description
x	Position
v	Velocity
m	Mass
m_{init}	Initial mass of the stars (before any supernovae)
h	Smoothing length
n_i	Weighted number of neighbors ($\tilde{y}$ in the general description of SPH)
ρ_{gas}	Surrounding gas density
Z_{frac}	Metallicity of the star at birth
M_{ej}	Mass ejected by the particle during the current step
$M_{\text{Z},ej}$	Mass of metals ejected by the particle during the current step
E_{ej}	Energy ejected by the particle during the current step
dt	Time step

Table 5.3 – List of the main variables contained within gravity particles (`gparts`).

Variable Name	Description
x	Position
v	Velocity
a_{spec}	Acceleration due to the spectral method
a_{grav}	Acceleration due to: pair interactions, FMM and spectral (initially only two firsts)
m	Mass
type	Type of particles (e.g. `part`, `spart` or true `gpart`)
dt	Time step

each start[3]. In the next page, the full dependency graph is shown for reference. As it is pretty large and not necessarily very readable, some subgraphs are produced in Figures 5.3, 5.4, 5.5, 5.6. The first subgraph focuses on the overview of the task dependencies and the following ones focus each on a different physics (in the order: gravity, hydro and stars).

It is worth to mention that, as in the example, we use different shapes for the communication (diamond) and computational (ellipses) tasks and in the following text, I will not discuss the communication tasks as they are not interesting in terms of physics. In the graphs, the colors correspond to the type of physics (blue for hydro, red for gravity, yellow for stars and black for the common tasks) and the tasks in grey are implicit tasks. The tasks inside a rectangle are a cluster of tasks computing the same physics but with different optimizations. `SWIFT` will start with the tasks at the top of the graph (all the drifts, some communication tasks and `init_grav`) and will move towards the bottom. As the dependencies are resolved at the cell level, `SWIFT` does not need to compute all the drift tasks before moving to the first sort task.

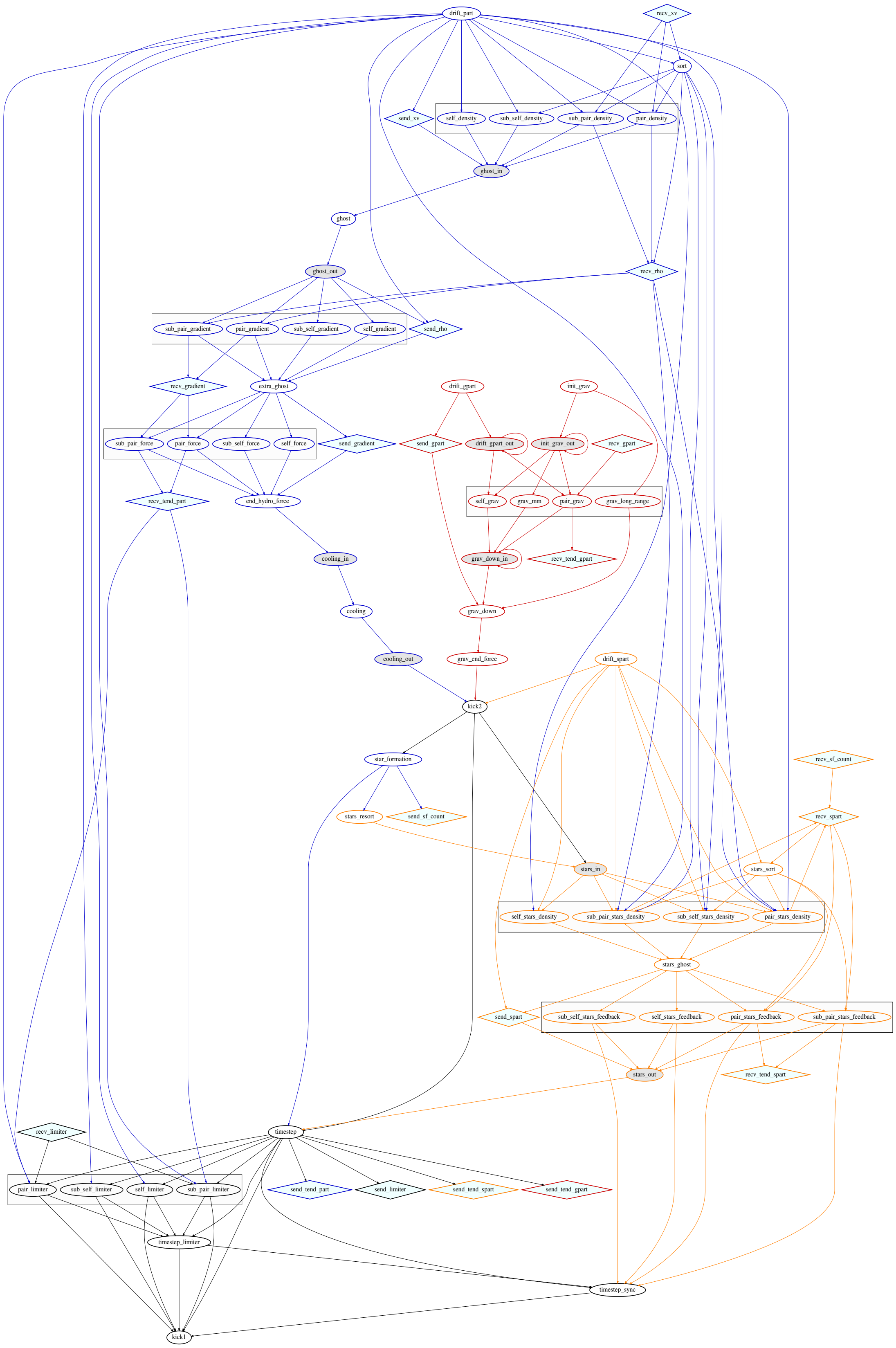
Let us take a look at Figure 5.2 (or 5.3 for less details). Three of the first tasks (in the sense that they do not depend on anything else) are the drifts (`drift_gpart`, `drift_part` and `drift_spart`). Once the `parts` and `gparts` have been drifted, we can proceed with the hydrodynamics (all the tasks in blue except from the cooling and the star formation) and gravity forces (all the tasks in red) that will be used in the two kick tasks (`kick2` and `kick1`) along with the computation of the time step (`timestep`). For the gas, we follow the hydrodynamics by the cooling (`cooling`). On the gravity side, only the gravitational forces are required. As the equation of motion for stars is only described by gravity, we only need to synchronize them with the `gparts` in the kicks. The next step is to kick the particles by half a time step, and then we can start with the star formation (`star_formation`) and by the stellar feedback due to the supernovae (all the tasks in orange). Now that all the physics has been computed, we can compute the time step of all the active particles for the next step (`timestep`) and ensure that all the gas particles have comparable time steps than their closest neighbors (all the tasks called `limiter`). Due to the feedback, we also need to activate in the next step the particles that received some feedback from the stars (`timestep_sync`) before moving to the first kick.

Initially, the order of the tasks in `SWIFT-GEAR` was different from the `EAGLE`'s one, but through different comparisons (not shown in this document), I have shown that the order has little

---

[3] `SWIFT` generates the file `dependency_graph.csv` at each start, and it can be transformed into a figure with the script `tools/plot_task_dependencies.py`

Figure 5.2 – Complete task dependencies for a simulation using the `SWIFT-GEAR` model. Each color represents a different kind of physics. The tasks in blue are related to the gas, in orange to stars and in red to gravity. The different shapes of task represent the types: ellipses for the computational tasks, diamond for the communications and with a grey background for the implicit ones. In Figures 5.3, 5.4, 5.5 and 5.6, the same graph is presented in a simplified way in order to focus on the different parts.


drift_part
recv_xv
sort
send_xv
self_density
sub_self_density
sub_pair_density
pair_density
ghost_in
ghost
ghost_out
recv_rho
sub_pair_gradient
pair_gradient
sub_self_gradient
self_gradient
send_rho
recv_gradient
extra_ghost
drift_gpart
init_grav
sub_pair_force
pair_force
sub_self_force
self_force
send_gradient
send_gpart
drift_gpart_out
init_grav_out
recv_gpart
recv_tend_part
end_hydro_force
self_grav
grav_mm
pair_grav
grav_long_range
cooling_in
grav_down_in
recv_tend_gpart
cooling
grav_down
cooling_out
grav_end_force
drift_spart
kick2
star_formation
recv_sf_count
stars_resort
send_sf_count
recv_spart
stars_in
stars_sort
self_stars_density
sub_pair_stars_density
sub_self_stars_density
pair_stars_density
stars_ghost
send_spart
sub_self_stars_feedback
self_stars_feedback
pair_stars_feedback
sub_pair_stars_feedback
stars_out
recv_tend_spart
recv_limiter
timestep
pair_limiter
sub_self_limiter
self_limiter
sub_pair_limiter
send_tend_part
send_limiter
send_tend_spart
send_tend_gpart
timestep_limiter
timestep_sync
kick1

importance on the properties of the galaxies. Therefore the order presented in the previous paragraph is the same than in `EAGLE`. Anyway, we can justify the order picked through the separation of the equations. The first group are evolved with the leapfrog scheme (gravity, hydrodynamics and cooling) and should be done in between the drift and the second kick. As the cooling and the hydrodynamics influence themselves through the density and internal energy of the gas, they need to be done the one after the other. The second group modify the properties of the gas instantaneously and cannot be properly integrated within the leapfrog scheme (star formation and stellar feedback) and should be kept in between the time steps (e.g. between the two kicks). As stars need to be born before producing any feedback, the star formation task is done before the stellar feedback.

5.1.3 Common Operations Performed by `SWIFT`

While most of the work should be spent within the task system, some important operations are required in between two steps. They can be split in two groups. One deals with the tasks and the second one with the octree on which the tasks are built. The tasks are initially created in `engine_maketasks` as described in the introduction of this chapter. In theory, they should be recreated at every steps, but it would require too much computations. A solution is to reuse the tasks and to tag them as being not done. Two different functions exist depending on if the tasks where just created or already reused (`engine_marktasks` and `engine_unskip`). Finally, once all the tasks are available and correctly tagged, they are sorted according to the dependencies (`scheduler_ranktasks`). In order to run the tasks, a queue is created and always contains all the tasks that have already solved their dependencies. Within the queue, the tasks are sorted according to some weights. They are defined by the type of tasks and the number of particles to use. In an effort to push forward the tasks that have many dependencies, their weight is added to the tasks that unlock them. At the end of a task, the scheduler adds into the queue (and keeps it sorted) all the tasks that have their dependencies fixed by the task.

For the second group, the most important function is the rebuild (`space_rebuild`). This function is triggered every time the tree is considered as not being good enough. To estimate the quality of the tree, we use two different criteria. In `SWIFT`, cells contain particles that could have left the cell's boundaries and thus both criteria try to estimate how far away they can leave. The first one is based on the tasks related to baryons. As we assume that all the neighbors are within the closest cells, we need to ensure that the particles have not moved too much with respect to the maximal size of the smoothing lengths and the cell's size. The second criterion is for gravity and, in this case, we do not have a strict criterion telling us when the tree does not behave correctly. We aim at using as many multipoles as possible and to do so, they should not be too extended. Therefore our criterion tries to figure out when it is most costly to continue with a direct double sum for the worst cells than rebuilding the full space. `SWIFT` uses the same criterion than `GADGET` based on the number of particle updates since the last rebuild. Once this number is larger than a given fraction of the particles, a rebuild is done. It is worth to mention that here the particles updated can consists in the same particles being updated multiple times. While this criterion seems to be a bit artificial, the rebuilds in cosmological simulations are mostly triggered by the baryons and not gravity.

The second operation is the repartition (`engine_repartition`) that exchanges top level cells between MPI ranks in order to equalize the amount of work between the ranks. Different strategies exist for this function within `SWIFT`. By default, every time a rank is spending more time on its tasks than the others (5 % imbalance), a repartition is done. In this case, `SWIFT` uses the amount of time spent in the tasks to estimate how much each cell costs (both in term of computation and communication) and relies on the library `METIS` to find the best distribution possible. When there is a lot of fluctuations in the time spent for the tasks (depends on the tasks used, the MPI implementation, the usage of the cluster, ...), this criteria can be triggered at almost every step. In such rare cases, it is possible to switch to a simpler criteria that consists in equalizing the amount of memory. While this is far more stable and strongly reduces the amount of repartition, it does not guarantee that the ranks will have an equivalent amount of computations. For example, in zoom simulations, a rank could have all the background particles with a long timescale while another one only the high resolution area with a small timescale.

The last operation worth mentioning consists in drifting all the particles (`engine_drift_all`). In `GADGET`, all the particles are drifted without consideration on if they will be used or not during this time step (which helps to get a good scaling of the code). As it can take a large amount of time, `SWIFT` uses the cells in order to know if the particles are active or if they will interact with an active particle. If it is not the case, the particle will completely skip the current step and do a larger drift later. In some cases, all the particles are required to be at the same time and thus `SWIFT` drifts them all to the current time. This is always done before a rebuild, a repartition, a computation of gravity with the spectral method, or dumping a snapshot or statistics. In the same way, when writing a snapshot, the velocities written are kicked (but not the velocity in the particles).

Before starting with the description of the tasks, I would like to mention that I am only providing an overview and that tasks can change with time. Therefore, I am not mentioning some details and some others might be wrong depending on `SWIFT`'s version. Finally, it is worth to mention that, in `SWIFT`, we assume that all the units are in internal units except if stated otherwise.

5.2 Drift

As we wish to compute the forces with the correct positions (meaning the correct time), the drift is applied to all the particles that will be needed in the current step. As gravity interacts all the particles together, it would mean that we need to drift all the particles like in `GADGET`. A more elegant solution consists in reusing the multipoles computed in the previous steps and simply drift their center of mass. This multipole drift is performed within the gravity tasks whenever an interaction with an undrifted multipole is required. As the velocities are required for the hydrodynamics, the velocities are "drifted" for the forces and a second variable records the

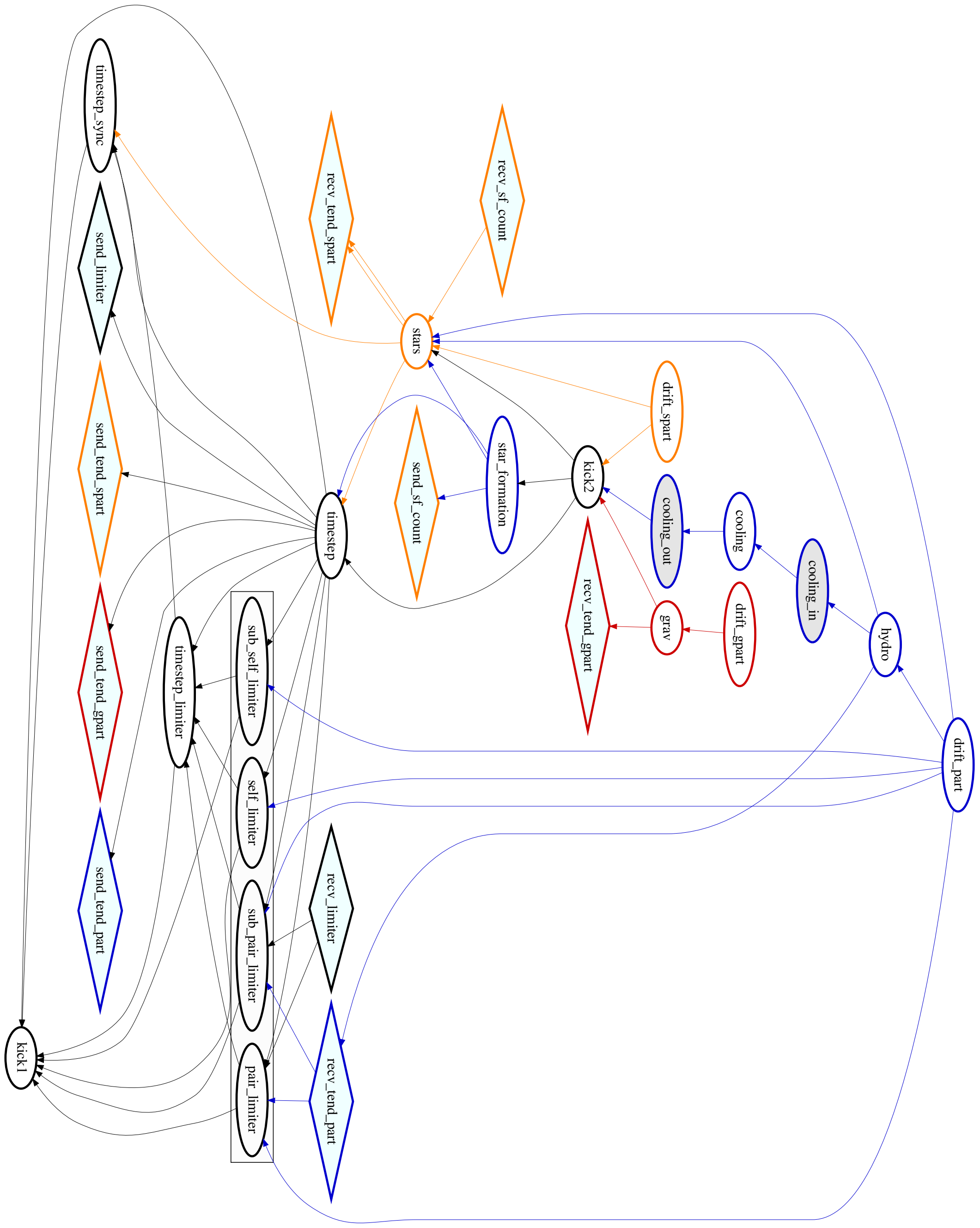


Figure 5.3 – Overview of the task dependencies. `SWIFT` will start from the tasks at the top and finish at the bottom. The tasks related to the hydrodynamics, gravity and stars are reduced into a single box in order to make the graph more readable. The details are provided in Figures 5.4, 5.5 and 5.6.

velocity at the end of the previous active step for the real evolution of the particle. The stellar feedback only acts on gas particles and thus they are the only particles to have some operations related to the feedback here.

The drift of the gas particles is shown as an algorithm in Algorithm 5.1 where all the variables are presented within the tables 5.1-5.3. The time step operators are described in the section 4.8. It is worth mentioning that the internal energy here is not allowed to go below the minimal internal energy.

For gravity and stellar particles, the equations are a lot simpler and shown in Algorithm 5.2. As mentioned before, the `parts` and their corresponding `gparts` are evolved independently and all the `parts` need their own drift and all the `gparts` (including the `gparts` corresponding to a `part`) also need their own drift (same for the `sparts`).

Algorithm 5.1 drift_part

```
    Feedback update
    Update the mass, energy, velocity and metallicity of the gas due to the supernovae that
    exploded at the previous step (see section 4.6.6).

    Drift
    $x \leftarrow x + v_{\text{full}} \mathrm{dt}_{\text{drift}}$
    $v \leftarrow v + a_{\text{hydro}} \mathrm{dt}_{\text{kick, hydro}} + a_{\text{grav}} \mathrm{dt}_{\text{kick, grav}}$
    $u \leftarrow u + \frac{\mathrm{d}u}{\mathrm{d}t} \mathrm{dt}_{\text{therm}}$
    $h \leftarrow h \exp\left(\frac{\mathrm{d}h}{\mathrm{d}t} \frac{\mathrm{dt}_{\text{drift}}}{h}\right)$
    $\rho \leftarrow \rho \exp\left(-3 \frac{\mathrm{d}h}{\mathrm{d}t} \frac{\mathrm{dt}_{\text{drift}}}{h}\right)$
    $\bar{P} \leftarrow \bar{P} \exp\left(-3 \frac{\mathrm{d}h}{\mathrm{d}t} \frac{\mathrm{dt}_{\text{drift}}}{h}\right)$
    Update derived variables.
```

Algorithm 5.2 drift_gpart, drift_spart

```
    $x \leftarrow x + v_{\text{full}} \mathrm{dt}_{\text{drift}}$
```

5.3 Gravity

The full task dependencies for gravity is given in Figure 5.4. First the particles are drifted and the multipoles are set to 0 in order to recompute them (`init_grav`). This is followed by the gravity cluster (`self_grav` and `pair_grav` for interacting particles directly between them or using a multipole, `grav_mm` for multipole-multipole interactions and `grav_long_range` for multipole-multipole interactions between top level cells) that computes the pairwise interactions or the multipoles. Then the multipoles are applied to the particles inside (`grav_down`) and, finally, the full acceleration is computed (`grav_end_force`). It is worth to mention explicitly that `grav_long_range` does not compute the contribution from the spectral method but only the long range FMM.

The particle to multipole (P2M) computation is done only in the rebuild (therefore not at all steps):

$$M_{\mathbf{m}}(z_B) \leftarrow \frac{1}{\mathbf{m}} \sum_b m_b(-r_b)^{\mathbf{m}}. \tag{5.1}$$

and, then, the multipoles are drifted to the required time in order to decrease the amount of computation required each step:

$$z_A \leftarrow z_A + v_{\text{mean}}\, \text{dt}_{\text{drift}} \tag{5.2}$$

where z_A is the center of mass and v_{mean} is the average (mass weighted) velocity of the particles inside a cell.

With a periodic mesh, the long range forces can be computed through a spectral method as shown in section 4.2. While `GADGET` is assuming a constant force per particle due to the spectral method between two computations, `SWIFT` assumes a constant potential field and then at each step computes the force at the current position of the particle.

For the tasks, we start first with the initialization of the gravity shown in Algorithm 5.3 that consists in setting the multipoles that will be recomputed to 0.

Algorithm 5.3 init_grav

```
    $F_{\mathbf{m}}(z_A) \leftarrow 0$
```

$F_{\mathbf{m}}$ corresponds to the M2L term defined in equation 4.13.

Then the gravitational forces at low and medium distances are computed either through the FFM method or a direction computation. It is done through 4 different tasks: `self/pair_grav`, `grav_mm`, `grav_long_range`. The self and pair tasks can use either the Fast Multipole Method (FMM) or the direct computation, `grav_mm` and `grav_long_range` are only using the FMM. While `grav_mm` can interact any kind of cells, the long range one will only work at the top level. The computations done by all these tasks is shown in Algorithm 5.4 where r_{ij} is the distance between the particles.

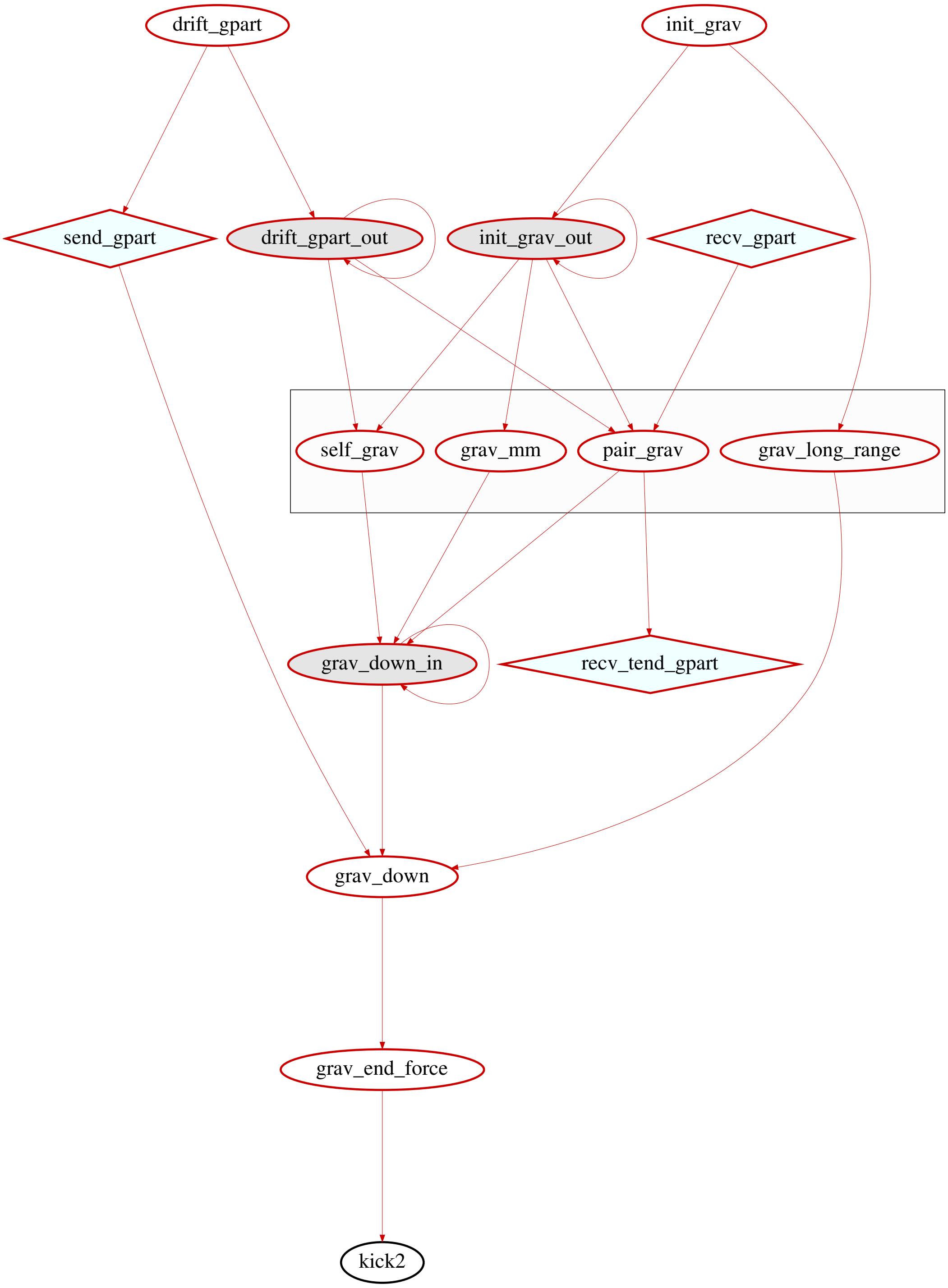


Figure 5.4 – The task dependency graph for gravity is shown here. The overview of the global dependencies is shown in Figure 5.3. `SWIFT` starts with the task at the top of the graph and finishes at the bottom. The tasks in gray are implicit tasks that are only present to simplify the dependencies.

Algorithm 5.4 FMM: grav_long_range, grav_mm, self_grav, pair_grav

```
Depends on: drift_gpart, init_grav
  if can do Fast Multipole Method then
    Compute the multipole to local expansion
    $F_{\mathbf{n}}(z_A) \leftarrow F_{\mathbf{n}}(z_A) + \sum_{\mathbf{m}}^{p-|\mathbf{n}|} M_{\mathbf{m}}(z_B) D_{\mathbf{n}+\mathbf{m}}(R)$    ▷ Equation 4.13
  else
    Compute the direct pair interaction
    $a_{\mathrm{grav},i} \leftarrow a_{\mathrm{grav},i} - \sum_j \frac{m_j}{r_{ij}^3} r_{ij}$
```

where p is the maximal multipole order taken into account.

The contribution from the multipoles is then added to the acceleration in `grav_down` shown in Algorithm 5.5. Finally, the missing coefficient is added in `grav_end_force` shown in Algorithm 5.6.

Algorithm 5.5 grav_down

```
Depends on: FMM
  Apply the local expansion to the particles.
  $a_{\mathrm{grav},i} \leftarrow a_{\mathrm{grav},i} + \sum_{\mathbf{n}}^{p} \frac{1}{\mathbf{n}!} r^{\mathbf{n}} F_{\mathbf{n}}(z_A)$    ▷ Equation 4.15
```

Algorithm 5.6 grav_end_force

```
Depends on: grav_down
  $a_{\mathrm{grav},i} \leftarrow G a_{\mathrm{grav},i}$
```

5.4 Hydrodynamics

The full task dependencies for the hydrodynamics are shown in Figure 5.5. For all the hydrodynamics methods, at least 2 loops over the neighbors are required. The first loop (density) computes the averaged quantities such as the density and the second one computes the forces. In some cases, the gradients are also computed and require an extra loop (e.g. energy diffusion or hybrid methods such as `SHADOWFAX` and the meshless methods). The gradient tasks are present only if the hydrodynamics scheme requires it.

The first computation after the drift is the density loop (`self_density`, `pair_density`, `sub_self_density`, `sub_pair_density`) shown in Algorithm 5.7 where we compute the

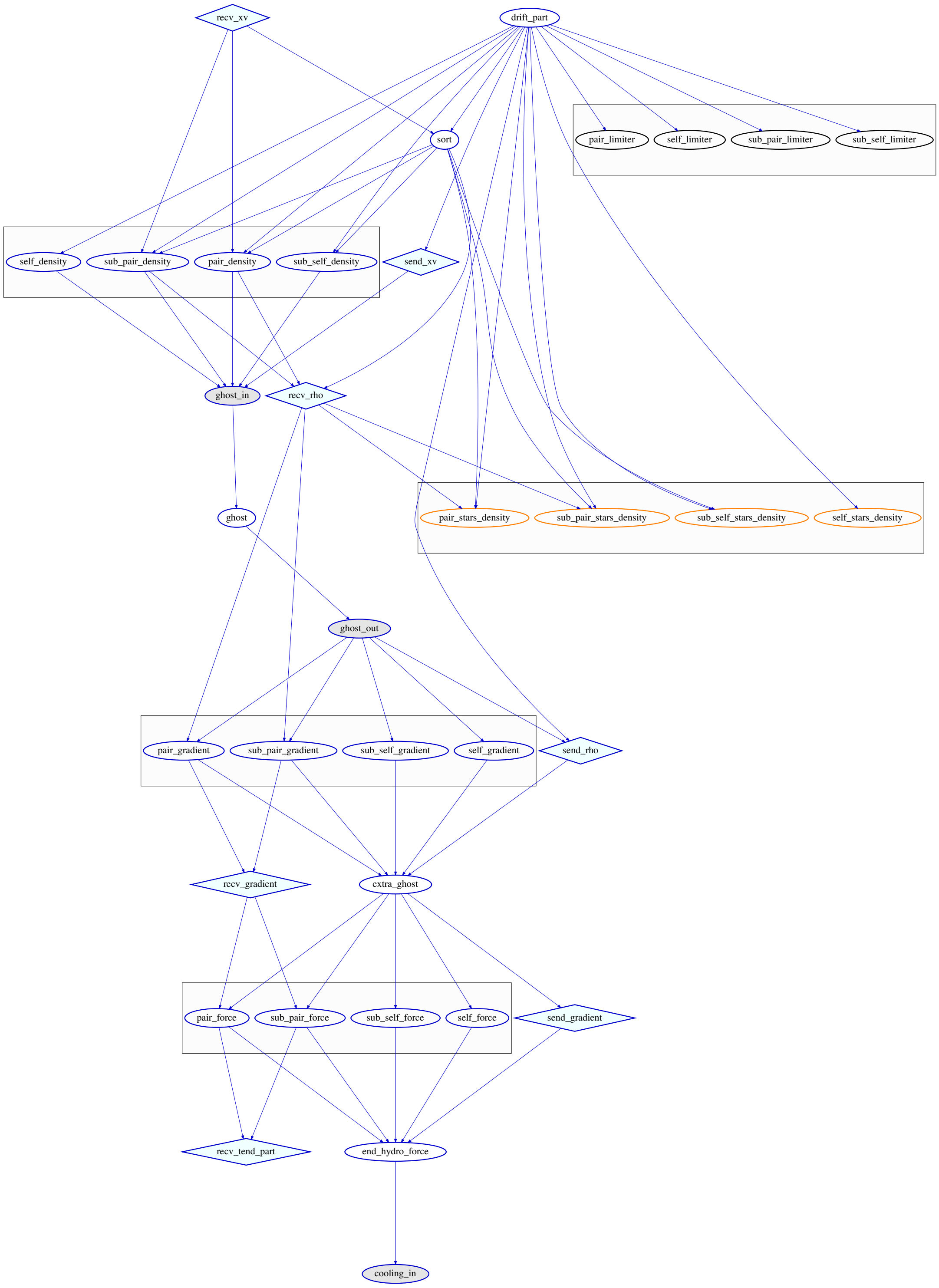


Figure 5.5 – The task dependencies for the hydrodynamics are shown here. The overview of the global dependencies is shown in Figure 5.3. `SWIFT` will start with the task at the top of the graph and finish at the bottom. It is worth mentioning that in the case of the `GADGET`'s SPH, the tasks for the gradient are not created (along with the extra ghost).

density, weighted pressure, weighted counts of particles, divergence and rotational of the velocity and finally the weighted metallicity for the smoothed metallicity.

Algorithm 5.7 Hydro density: self_density, sub_pair_density, pair_density, sub_self_density

```
Depends on: drift_part
    Hydro
    $\rho_i \leftarrow \sum_j m_j W_{ij}(h_i)$
    $\bar{P}_i \leftarrow \sum_j m_j u_j W_{ij}(h_i)$
    $\frac{\mathrm{d}\bar{P}_i}{\mathrm{d}h} \leftarrow -\sum_j m_j u_j \left(3W_{ij}(h_i) + \frac{r_{ij}}{h_i}\frac{\mathrm{d}W_{ij}(h_i)}{\mathrm{d}x}\right)$
    $n_i \leftarrow \sum_j W_{ij}(h_i)$
    $\frac{\mathrm{d}n_i}{\mathrm{d}h} \leftarrow -\sum_j 3W_{ij}(h_i) + \frac{r_{ij}}{h_i}\frac{\mathrm{d}W_{ij}(h_i)}{\mathrm{d}x}$
    $\nabla \cdot v_i \leftarrow -\sum_j m_j \frac{\mathrm{d}W_{ij}(h_i)}{\mathrm{d}x}\frac{v_{ij} \cdot r_{ij}}{r_{ij}}$
    $\nabla \wedge v_i \leftarrow \sum_j m_j \frac{\mathrm{d}W_{ij}(h_i)}{\mathrm{d}x}\frac{v_{ij} \wedge r_{ij}}{r_{ij}}$

    Chemistry
    $\bar{Z}_{\mathrm{frac,i,k}} \leftarrow \sum_j m_j Z_{\mathrm{frac,j,k}} W_{ij}(h_i)$        ▷ Equation 4.102
```

In `SWIFT` the kernels are not normalized with the smoothing lengths, this normalization is done in the ghost task to reduce the computational cost. In this document I suppose that the kernels contain the smoothing lengths in order to simplify the description of the tasks.

The ghost task is shown in Algorithm 5.8 where P_{Jeans} is the Jeans pressure and a the scale factor. The aim of this task is to solve the equation giving the number of neighbors (equation 4.40) with the Newton-Raphson scheme [4]. This operation provides the value of the smoothing length by successive evaluations of the density loop (manually recomputed within this task without any rescheduling of the density tasks). It also finishes the density loop and initializes the force loop.

---

[4] This scheme is similar to the Newton method that finds the solution to an equation through an iterative process.

Algorithm 5.8 ghost

```
Depends on: Hydro density
    while $h_i$ has not converged do Hydro density
    Definitions
    $\lfloor P_i \rfloor = \max(P_i, P_{\text{Jeans, i}})$

    End Density
    $\rho_i \leftarrow \rho_i + m_i W(0, h_i)$
    $\bar{P}_i \leftarrow (\gamma - 1)\left(\bar{P}_i + m_i u_i W(0, h_i)\right)$
    $\frac{\mathrm{d}\bar{P}_i}{\mathrm{d}h} \leftarrow (\gamma - 1)\left(\frac{\mathrm{d}\bar{P}_i}{\mathrm{d}h} - 3 m_i u_i W(0, h_i)\right)$
    $n_i \leftarrow n_i + W(0, h_i)$
    $\frac{\mathrm{d}n_i}{\mathrm{d}h} \leftarrow \frac{\mathrm{d}n_i}{\mathrm{d}h} - 3W(0, h_i)$
    $\nabla \cdot v_i \leftarrow \frac{1}{a^2 \rho_i} \nabla \cdot v_i + 3H$
    $\nabla \wedge v_i \leftarrow \frac{1}{a^2 \rho_i} \nabla \wedge v_i$

    Chemistry
    $\bar{Z}_{\text{frac,i,k}} \leftarrow \frac{1}{\rho_i}\left(\bar{Z}_{\text{frac,i,k}} + m_i Z_{\text{frac, i}} W(0, h_i)\right)$

    Prepare force
    $\nabla h_i \leftarrow \frac{\mathrm{d}\bar{P}_i}{\mathrm{d}h} \frac{h_i}{3 n_i (\gamma - 1)} \left(1 + \frac{h_i}{3 n_i} \frac{\mathrm{d}n_i}{\mathrm{d}h}\right)^{-1}$
    $c_{s,i} \leftarrow \sqrt{\gamma \frac{\lfloor \bar{P}_i \rfloor}{\rho_i}}$
    $B_i \leftarrow \alpha |\nabla \cdot v_i| \left(|\nabla \cdot v_i| + \nabla \wedge v_i + 0.0001 \frac{c_{s,i} a^{(1-3\gamma)/2}}{h_i}\right)^{-1}$
```

Once the ghost task is computed and if there is no need for a gradient task, the acceleration and evolution of the energy of each particle can be computed in the force tasks shown in Algorithm 5.9 where H is the Hubble constant, β a parameter for the viscosity.

Algorithm 5.9 Hydro force: pair_force, sub_pair_force, sub_self_force, self_force

```
Depends on: ghost (or extra_ghost with the gradient)
    Definitions
    $f_{ij} = 1 - \left(\frac{\nabla \cdot h_i}{m_j u_j}\right)$
    $a_{\text{SPH, ij}} = \sum_j u_j u_i (\gamma - 1)^2 \frac{1}{r_{ij}} \left( f_{ij} \frac{\lfloor \bar{P}_i \rfloor}{\bar{P}_i^2} \frac{\mathrm{d}W_{ij}(h_i)}{\mathrm{d}x} + f_{ji} \frac{\lfloor \bar{P}_j \rfloor}{\bar{P}_j^2} \frac{\mathrm{d}W_{ji}(h_j)}{\mathrm{d}x} \right)$    ▷ Equation 4.49
    $\mu_{ij} = a^{\frac{3\gamma-5}{2}} \frac{\min\left(v_{ij} \cdot x_{ij} + a^2 H r_{ij}^2, 0\right)}{r_{ij}}$
    $v_{\text{sig}} = c_i + c_j - \beta \mu_{ij}$
    $\nu_{ij} = -\frac{1}{4} \frac{v_{\text{sig}}}{r_{ij}} \mu_{ij} (B_i + B_j)$
    $a_{\text{visc}} = \frac{1}{2} \frac{\nu_{ij}}{r_{ij}} \left( \frac{\mathrm{d}W_{ij}(h_i)}{\mathrm{d}x} + \frac{\mathrm{d}W_{ij}(h_j)}{\mathrm{d}x} \right)$    ▷ Equation 4.59
    Acceleration
    $a_{\text{hydro, i}} \leftarrow - \sum_j m_j \left( a_{\text{SPH, ij}} + a_{\text{visc}} \right) r_{ij}$

    Energy equation
    $\left.\frac{\mathrm{d}u}{\mathrm{d}t}\right|_{\text{SPH}} = (\gamma - 1)^2 u_j u_i f_{ij} \frac{\lfloor \bar{P}_i \rfloor}{\bar{P}_i^2} \frac{\mathrm{d}W_{ij}(h_i)}{\mathrm{d}x} v_{ij} \cdot \frac{r_{ij}}{r_{ij}}$    ▷ Equation 4.54
    $\left.\frac{\mathrm{d}u}{\mathrm{d}t}\right|_{\text{visc}} = \frac{1}{2} a_{\text{visc}} \left( v_{ij} \cdot r_{ij} + a^2 H r_{ij}^2 \right)$
    $\frac{\mathrm{d}u_i}{\mathrm{d}t} \leftarrow \sum_j m_j \left( \left.\frac{\mathrm{d}u}{\mathrm{d}t}\right|_{\text{SPH}} + \left.\frac{\mathrm{d}u}{\mathrm{d}t}\right|_{\text{visc}} \right)$    ▷ Equation 4.60

    Smoothing length evolution
    $\frac{\mathrm{d}h_i}{\mathrm{d}t} \leftarrow - \sum_j m_j \frac{v_{ij} \cdot r_{ij}}{\rho_j} \frac{\mathrm{d}W_{ij}(h_i)}{\mathrm{d}x}$

    Signal velocity
    $v_{\text{sig, i}} \leftarrow \max(v_{\text{sig, i}}, v_{\text{sig}})$
```

Finally, the missing factors are added to the required quantities within the task `end_hydro_force` given in Algorithm 5.10.

Algorithm 5.10 end_hydro_force

```
Depends on: Hydro force
    $\frac{\mathrm{d}h_i}{\mathrm{d}t} \leftarrow \frac{1}{3} \frac{\mathrm{d}h_i}{\mathrm{d}t}$
```

5.5 Cooling

The cooling is computed with the library `GRACKLE`. It provides a few different functions, and we are mainly using `local_solve_chemistry` that computes the evolution of the chemical network and the particle energy (assuming constant density and metallicity during a given time step) within the cooling task (`cooling`). This task depends indirectly on `end_hydro_force` and is shown in Algorithm 5.11 where u_{min} is the minimal energy allowed (usually 10 K or below) and H is the Heaviside step function[5]. As the variable $\frac{\mathrm{d}u}{\mathrm{d}t}$ already contains the hydro term in `SWIFT`, it needs to be carefully treated, and we need to ensure that the energy does not become smaller than the minimal energy.

Algorithm 5.11 cooling

Depends on: end_hydro_force

```
Radiative cooling
$u_{\mathrm{ad,\,i}} = \max\left(u_{\mathrm{min}}, u_i + \mathrm{d}t_{\mathrm{therm,\,i}}\frac{\mathrm{d}u_i}{\mathrm{d}t}\right)$
if $t - t_{\mathrm{SN,\,i}} < t_{\mathrm{ad}}$ then
    $\left.\frac{\mathrm{d}u_i}{\mathrm{d}t}\right|_{\mathrm{cooling}} = 0$    ▷ No radiative cooling after a supernova
else
    $\left.\frac{\mathrm{d}u_i}{\mathrm{d}t}\right|_{\mathrm{cooling}} = \Gamma_{\mathrm{UVB}} H(\rho_i - \rho_{\mathrm{shield}}) - \Lambda(u_{\mathrm{ad,\,i}})$    ▷ Computed by GRACKLE

Update $\left.\frac{\mathrm{d}u_i}{\mathrm{d}t}\right|_{\mathrm{cooling}}$ if the energy goes below the minimal energy after the cooling.
$\frac{\mathrm{d}u_i}{\mathrm{d}t} \leftarrow \frac{\mathrm{d}u_i}{\mathrm{d}t} + \left.\frac{\mathrm{d}u_i}{\mathrm{d}t}\right|_{\mathrm{cooling}}$
```

5.6 Kicks

In the velocity leapfrog scheme, there is two kicks that are doing the same computation but with variables synchronized at different time. The first one (`kick1`) is moving from t to $t + \frac{1}{2}\Delta t$ and the second one (`kick2`) from $t + \frac{1}{2}\Delta$ to $t + \Delta t$. A step is therefore composed in the order of a kick, a drift and a second kick. To simplify the individual time step scheme and the synchronization between particles, the first kick is computed at the end of the task system and thus the time step finishes in the middle of the task system. It means that we have `kick2` before `kick1` within the tasks but not according to the time evolution. In `GADGET-2`, both kicks were applied at the same time along with the computation of the time step to reduce the computation time. In

---

[5] This is the only place where H corresponds to this function. All other occurrences are the Hubble constant except if stated otherwise.

`SWIFT` and more recent versions of `GADGET`, they have been separated in two in order to insert some operations such as the time step limiter in between. It is worth to mention that the only point in time when both the positions and velocities are synchronized is in between the two kicks. The kicks for gas particles are shown in Algorithm 5.12 and the other types of particles in Algorithm 5.13. As for the drift, the kicks are applied separately to the baryonic particles and their `gparts`.

Algorithm 5.12 kick2 / kick1 (`parts`)

Depends on: For kick2: grav_end_force, cooling; For kick1: timestep, timestep_limiter, timestep_sync

```
$v_{\text{full, i}} \leftarrow v_{\text{full, i}} + a_{\text{hydro, i}} \text{dt}_{\text{kick, hydro, i}} + a_{\text{grav, i}} \text{dt}_{\text{kick, grav, i}}$
$u_{\text{full, i}} \leftarrow u_{\text{full, i}} + \frac{\text{d}u_i}{\text{d}t} \text{dt}_{\text{therm, i}}$
Ensure that the energy does not go below the minimal energy.
```

Algorithm 5.13 kick2 / kick1 (`gparts`and `sparts`)

Depends on: For kick2: grav_end_force, cooling; For kick1: timestep, timestep_limiter, timestep_sync

```
$v_{\text{full, i}} \leftarrow v_{\text{full, i}} + a_{\text{grav, i}} \text{dt}_{\text{kick, grav, i}}$
```

5.7 Star Formation

In the star formation task, some gas particles are converted into stellar particles according to equation 4.73 and presented in Algorithm 5.14 where t_{ff} is the free fall time, m_g the mass of the gas particle, T_{max} the maximal temperature allowed, ρ_{Jeans} the Jeans density (Equation 4.74) and $c_\star$ is the star formation efficiency. The mass of a star ($m_\star$) is given by the average mass of the gas particle in the initial conditions divided by the number of stars spawn from a single gas particle. For the last star spawned from a gas particle, we fully convert the gas particle into a star particle and adapt accordingly $m_\star$ in the equation. This avoids gas particles with extremely low masses and does conserve the mass. Therefore $m_\star$ is not the exact mass of all the particles but only an approximation.

Algorithm 5.14 star_formation

Depends on: kick2

```
if $(\nabla \cdot v_i > 0) \vee (T_i > T_{\max}) \vee (\rho_i < \rho_{Jeans})$ then return
$\frac{1}{t_{\mathrm{ff,\,i}}} = \sqrt{\frac{32}{3\pi} G \rho_i}$
$p_i = \frac{m_{g,i}}{m_\star} \left(1 - \exp\left(-c_\star \frac{\mathrm{d}t_i}{t_{\mathrm{ff,\,i}}}\right)\right)$        ▷ Equation 4.77
if random(0, 1) $< p_i$ then
    Create a new spart and, if required, flag the part as being removed.
    Copy the properties from the part to the new spart.
    If a new spart is spawn (e.g. the part is not fully transformed into a spart), the two particles are randomly moved inside a fraction of the kernel.
    Save the birth properties (e.g. mass, time, smoothed metallicity, ...).
```

5.8 Star Formation and Stellar Feedback

The task dependencies for the stellar feedback are shown in Figure 5.6. Two loops over the neighbors are required, the first one computes the smoothing length of each `spart` in order to have a given weighted number of gas neighbors as done for the gas (`self_stars_density`, `pair_stars_density`, `sub_self_stars_density` and `sub_pair_stars_density`). The second one applies the stellar feedback to the surrounding gas particles (`self_stars_feedback`, `pair_stars_feedback`, `sub_self_stars_feedback` and `sub_pair_stars_feedback`). During the feedback loop, we simply compute the total energy and mass to inject. They are then injected during the drift (see Algorithm 5.1 that corresponds to Equation 4.101 and the text around). Injecting during the feedback loop, as in `GEAR`, would generate a situation where the order of the interactions would change the properties of the gas particle as the energy update depends on the mass.

The stellar evolution is computed one step in advance and allows to skip the `sparts` that will not produce any feedback during the two loops. This amelioration provides a large speedup as stars do not produce any feedback on most of the time steps (see Figure 4.7).

The first interesting task is the density task for stars which consists in the same computations as the hydrodynamics for the smoothing length (except that we do not add the self contribution) and is described in Algorithm 5.15. In the ghost, again we redo the density loop until reaching a correct number of weighted neighbors.

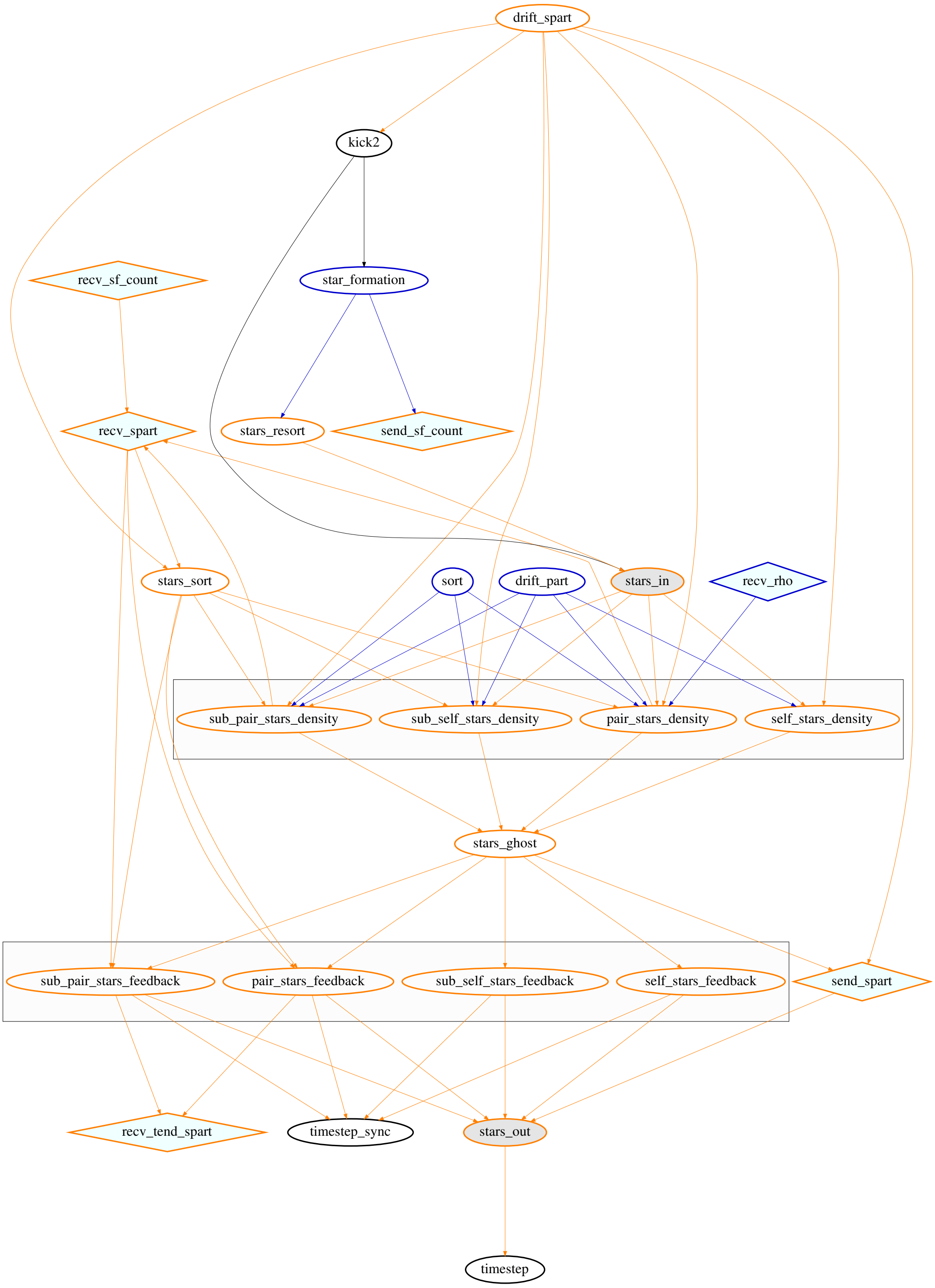


Figure 5.6 – The task dependency for stars is shown here. The global dependencies are shown in Figure 5.3. `SWIFT` will start with the task at the top of the graph and finish at the bottom.

Algorithm 5.15 Stars density: sub_pair_stars_density, sub_self_stars_density, pair_stars_density, self_stars_density

Depends on: drift_parts, sort, drift_sparts, stars_sort, stars_resort

```
if will not produce any feedback then return
$n_i \leftarrow \sum_{j=\text{gas}} W_{ij}(h_i)$
$\frac{\mathrm{d}n_i}{\mathrm{d}t} \leftarrow -\sum_{j=\text{gas}} 3W_{ij}(h_i) + \frac{r_{ij}}{h_i}\frac{\mathrm{d}W_{ij}(h_i)}{\mathrm{d}x}$
$\rho_{\text{gas, i}} \leftarrow \sum_{j=\text{gas}} m_j W_{ij}(h_i)$
```

Thanks to the computation of the stellar evolution during the last step, the energy and mass ejected by the different stars can be summed in the surrounding particles during the feedback loop. The loop is shown in Algorithm 5.16 where only the gas particles are updated.

Algorithm 5.16 Star feedback: sub_pair_stars_feedback, pair_stars_feedback, sub_self_stars_feedback, self_stars_feedback

Depends on: Star ghost

```
if will not produce any feedback then return
$w_{ij} = \frac{m_j W_{ij}(h_i)}{\rho_{\text{gas},i}}$
$\Delta m_j \leftarrow \sum_{i=\text{stars}} M_{\text{ej},i} w_{ij}$
$\Delta u_j \leftarrow \sum_{i=\text{stars}} \frac{E_{\text{SN},i}}{M_{\text{ej},i} w_{ij}} \frac{m_j W_{ij}(h_i)}{\rho_{\text{gas},i}}$
$\Delta p_j \leftarrow \sum_{i=\text{stars}} M_{\text{ej},i} w_{ij} \left(v_i - v_j\right)$
Impose the maximal viscosity to the particle in order to capture the forthcoming shock.
Synchronize the particle in order to make it active at the next step.
```

5.9 Time Step

`SWIFT` uses adaptive and individual time steps and thus includes a task dedicated to their computation. A common criterion for all the particles is based on the accelerations. In the case of hydrodynamics, a Courant-Friedrichs-Lewy (CFL) condition is also required, and we ensure that the smoothing lengths are not changing too much between two steps for stability. Therefore the time step is computed after the stellar feedback as some models within `SWIFT` directly modify the internal energy of the gas and thus the CFL condition. The time step computation is shown in Algorithm 5.17 for the `parts` and 5.18 for both the `sparts` and `gparts` where η is an accuracy parameter, a is the scale factor, γ is the adiabatic index, ϵ is the comoving softening length, C is the maximum change allowed and $\alpha_{\text{CFL}} < 1$ the CFL parameter.

Algorithm 5.17 timestep (`part`)

Depends on: Star feedback, star formation, kick2

```
    Gravity
    $\mathrm{dt}_{\text{grav, i}} = \sqrt{2\frac{a\eta\epsilon_i}{\left(\frac{a_{\text{hydro, i}}}{a^{3\gamma-2}} + \frac{a_{\text{grav, i}}}{a^2}\right)}}$
    Hydro
    $\mathrm{dt}_{\text{CFL, i}} = 2\gamma_K\alpha_{\text{CFL}}\frac{ah_i}{a^{3(1-\gamma)/2}v_{\text{sig, i}}}$
    $\mathrm{dt}_{\text{h, i}} = C\left|\frac{h_i}{\frac{\mathrm{d}h_i}{\mathrm{d}t}}\right|$
    $\mathrm{dt}_i \leftarrow \min(\mathrm{dt}_{\text{grav, i}}, \mathrm{dt}_{\text{CFL, i}}, \mathrm{dt}_{\text{h, i}})$
```

Algorithm 5.18 timestep (`gpart` and `spart`)

Depends on: Star feedback, star formation, kick2

```
    $\mathrm{dt} \leftarrow \sqrt{2\frac{a^3\eta\epsilon}{a_{\text{grav}}}}$
    Stellar evolution (only for sparts)
    Compute the mass of a star that explodes at the beginning and the one at end of the next time step (equation 4.80).
    Compute the supernovae rates (equations 4.84 and 4.86) with the integral boundaries given by the mass previously computed.
    if using discrete yields then
        $N_{\text{SNII}}, N_{\text{SNIa}} \leftarrow \lfloor N_{\text{SNII}} \rfloor, \lfloor N_{\text{SNIa}} \rfloor$
        Add randomly a supernova of each type with a probability given by the remainder.
        Compute the properties of the ejecta based on the average of the two mass previously computed (Equation 4.89).
    else
        Compute the integral of the mass ejected and the metallicity between the two previous masses (Equation 4.92).
```

Field	`GEAR`	`SWIFT`
Coordinates	$\frac{hx}{a}$	$\frac{x}{a}$
SmoothingLengths	$\frac{\gamma_K hH}{a}$	$\frac{H}{a}$
Masses	mh	m
Velocities	$v\sqrt{a}$	v
InternalEnergies	u	$u \,/\, a^{3(\gamma-1)}u$
Densities	$\frac{a^3\rho}{h^2}$	$a^3\rho$

Table 5.4 – Examples of objects stored in the snapshots for GEAR and SWIFT. The names correspond to SWIFT's snapshots. The smoothing length is indicated by H in order to avoid confusion with the "little h" from the Hubble constant. They also differ in their definition due to γ_K (defined in section 4.3.1). Thus, H for both codes corresponds to `SWIFT`'s definition. For `SWIFT`, the units depends on the SPH, therefore the one reported here are for `GADGET`'s SPH. To be safe, I recommend using the information contained in the attributes of each field. For the internal energy, the inputs are not in the same coordinates (physical) than the output (comoving).

5.10 Time Step Limiter and Synchronization

As mentioned in the theory, in order to improve the modelization of supernovae, the neighboring particles need to have similar time steps (`timestep_limiter`, `self_limiter`, `pair_limiter`, `sub_self_limiter`, `sub_pair_limiter`) and to be activated as soon as they receive some feedback (`timestep_sync`). As it is very technical, not interesting in term of physics and not my own work, I will not go through the implementation details.

5.11 Snapshot

Now that all the tasks have been described, it is time to move to a quick description of the snapshots written by `SWIFT`. They are saved with the library `HDF5`. It is worth mentioning that they are different from `GADGET`'s ones and table 5.4 shows the differences (the name of the dataset can also change). The main difference is the absence of the "little h" in the quantities that avoids people forgetting them in the analysis. Each dataset produced with `SWIFT` has a few attributes that provides the units of the arrays (including scale factor and "little h") along with a description of the data.

5.12 New physics in Development

I have also implemented some new physics that we are planning to have in the future, but this is not a work large enough to be presented in details within this thesis. Furthermore, only a basic implementation exists, therefore I will only quickly mention them. The two new models are the metal diffusion and the first stars. Both models have been developed in order to solve an issue in the abundances of the stars.

5.12.1 Metal Diffusion

As we are trying to go to higher resolution, the smoothed metallicity will become insufficient to reproduce the low scatter in the abundances of stars due to its scale dependency. Indeed, the metals are smoothed over the smoothing lengths that correspond to the resolution. The solution to this problem is to add some diffusion of the metals in the gas due to the unresolved scales of the hydrodynamics. A few papers exist on this topic and I have decided to follow Shen et al., 2010. The implementation is still lacking a time step condition, an analysis of the impact of the diffusion coefficient and some tests in a cosmological context.

5.12.2 First Stars

Due to the low cooling rate of the primordial gas, some people are considering the metal free and first stars to be a lot more massive than typical stars seen today (Haemmerlé et al., 2020; Bromm et al., 2002). It would mean that these stars would have a totally different evolution and production of metals than other stars and could explain why our faint dwarf galaxies have a lower average metallicity than seen in observations (Figure 6 in Revaz and Jablonka, 2018). Therefore, in `SWIFT-GEAR`, we simply read two different tables and select them depending on the metallicity of the stellar particle.

6 Verification and Scaling

> ‘Precious, precious, precious!’ Gollum cried. ‘My Precious! O my Precious!’ And with that, even as his eyes were lifted up to gloat on his prize, he stepped too far, toppled, wavered for a moment on the brink, and then with a shriek he fell. Out of the depths came his last wail precious, and he was gone.
>
> — J. R. R. Tolkien, The Return of the King

`SWIFT-GEAR` was designed to replace `GEAR` with better performances. Before doing so, we need to ensure that we obtain comparable results and also that `SWIFT` is indeed faster. In this chapter, I will first take a look at the conservation laws. This will be done with a simple example ran with different parameters and will show the impact of some assumptions. Next, the simulations produced by `SWIFT` are compared with Revaz and Jablonka, 2018 within a cosmological context. Finally, a discussion about the efficiency is given with a description of the tools available in `SWIFT` to optimize the code.

6.1 Verification of the Conservation Laws

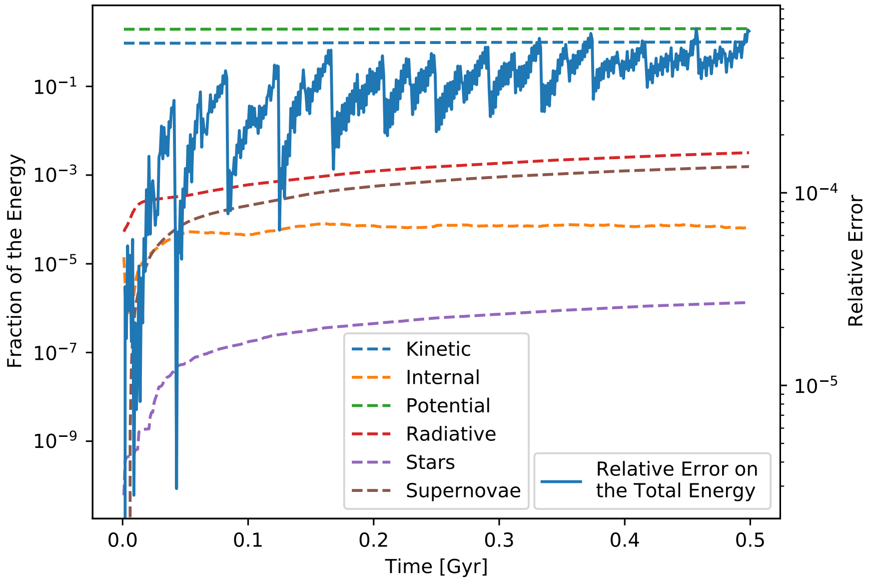
As usual in numerical simulations, it is important to ensure that we are conserving the energy, momentum, angular momentum and mass. While the last one is trivially done thanks to the Lagrangian approach and as we do not modify the particles’ masses except from the star formation and the stellar feedback, the 3 others require a more careful implementation and will be analyzed here.

Unfortunately, the integration scheme velocity Verlet, also known as leap-frog, is in theory conservative only with a fixed time step (see for example Springel, 2005). Without the individual and adaptive time steps, it would be difficult to run a small cosmological simulations even at low resolution and thus `SWIFT` cannot guarantee the conservation. Anyway, to verify the quality of the conservation, the `AGORA` isolated disk (Kim et al., 2016) was run for 0.5 Gyr (one dynamical time) with the physics presented in the previous chapter. In the first graph of Figure 6.1, the different forms of energies are shown with dashed lines along with the conservation of the total energy represented by the solid line. The different forms of energies tracked are the classical kinetic, internal and potential energies. It also includes the energy radiated by the radiative cooling (lost to the simulation but accounted for), the internal energy of gas particles transformed into stars and the energy released by supernovae. A last form of energy injection is through the minimal temperature of 10 K that ensures the stability of the simulation. This mechanism is not tracked but an upper bound will be provided in the next paragraph. In this example, the energy is conserved with a relative error of less than 10^{-3}.

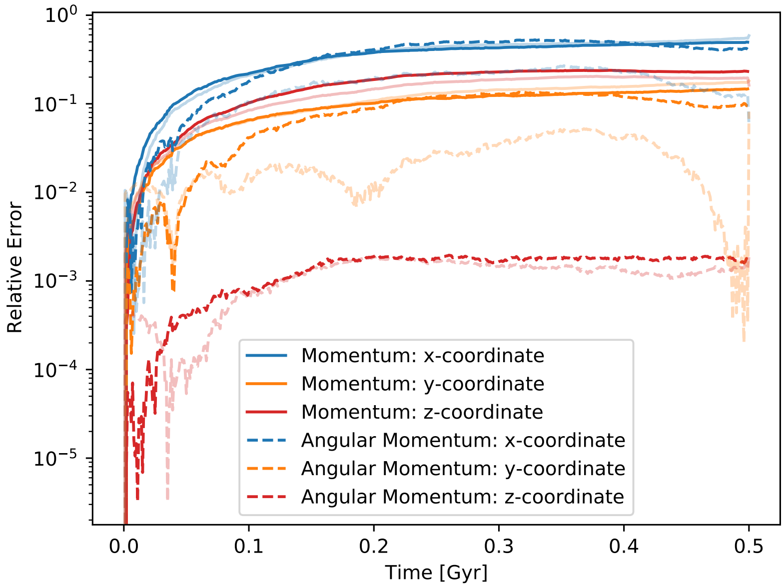
The second graph shows the conservation of both the momentum and the angular momentum. A dark matter only version of this simulation, with a single time step shared by all particles, produces the same level of conservation (light lines) and proves that gravity is mainly responsible for the low conservation shown in the graph. The most probable explanation is the non symmetric multipole acceptation criterion within the FMM.

In both graphs, the errors strongly increase at the beginning. As the initial conditions are computed assuming an hydrostatic equilibrium, they do not take into account the perturbation of the subgrid models. Thus, the system quickly changes in order to go back to a dynamical equilibrium. As one could expect from a numerical simulation, all the quantities are sufficiently conserved, even if the error of the momentum is relatively large.

Let us now focus on the energy conservation in order to show that the individual time steps are responsible for a large fraction of the error. In Figure 6.2, the energy conservation is shown for 4 different simulations. The blue solid line corresponds to the simulation presented just before and the red solid line corresponds to the same simulation without any subgrid model (e.g. no pressure floor, no radiative cooling, no stars and no chemistry; only hydrodynamics and gravity). The dashed lines correspond to the two previous simulations but with a single time step for all the particles. The time step is chosen such that it corresponds to the lowest time step found in the previous simulations. While the two first simulations both include the minimal energy, only the first one triggers it as the radiative cooling enables such low temperatures. Therefore the difference in between them represents an upper bound to the lack of conservation from it. This comparison does not allow to differentiate between the effect of the minimal energy and the subgrid models, but it shows that their contribution to the error is small. The two simulations with a single time steps show a far better conservation (one order of magnitude), therefore our error budget is clearly dominated by the necessary approximation of the individual time steps.



(a) Energy conservation



(b) Momentum conservation

Figure 6.1 – Conservation of both the energy and momentum for the `AGORA` disk including the full physics. In the first graph, the absolute value of the energies are indicated with dashed lines and the relative error for the total energy is indicated with the solid line. “Radiative” corresponds to the energy lost due to the radiative cooling, “Stars” corresponds to the internal energy contained within the gas particles transformed into stars and “Supernovae” corresponds to the energy released by the stars. In the momentum conservation graph, a dark matter only simulation is shown in light.

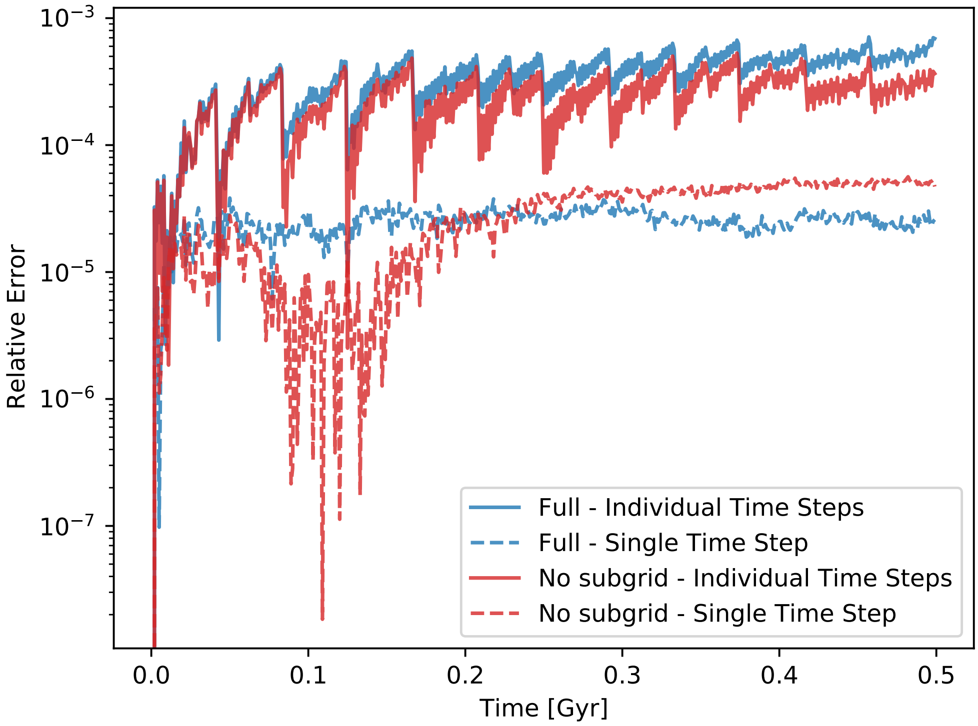


Figure 6.2 – Comparison of the energy conservation for 4 different simulations. Two simulations are run with the subgrid models (in blue) and two without (only hydrodynamics and gravity; in red). To show the impact of the individual time steps, each set of simulations are run once with them (solid lines) and once with a single time step equivalent to the lowest one found in the previous simulations (dashed lines).

6.2 Verification of the Implementation

The physics implemented was verified by comparing the results from `SWIFT-GEAR` and observations of the Milky Way and Andromeda's satellites. The verification is based on Revaz and Jablonka, 2018. The simulations were done in a cosmological context and using the zoom technique within a box of 5 Mpc. They started from redshift 70 with an initial gas temperature of 80K. The resolution is 1500 $\mathrm{M}_\odot$ for the stars, 6100 $\mathrm{M}_\odot$ for the gas and $33'400$ $\mathrm{M}_\odot$ for the dark matter. For the following figures, 22 dwarf galaxies were simulated with the same parameters than in Revaz and Jablonka, 2018. They were selected from a dark matter only simulation in order to have virial masses in between $6 \cdot 10^8$ $\mathrm{M}_\odot$ and $1.5 \cdot 10^{10}$ $\mathrm{M}_\odot$. The most computational heavy were not run until z=0 (red dots in the next figures) as it would have require far too much computational resources for the purpose of this comparison, but as it will be seen, they are already sufficient to verify the implementation. Only the 4 most massive galaxies are at redshift above 4.3. Thus the galaxies that have their evolution significantly influenced by the UVB are at a redshift below the moment where the star formation starts to decrease according to Figure 10 in Revaz and Jablonka, 2018.

In this section, I will start by showing the scaling relations from the simulations previously mentioned. While theses relations are a good indication of the quality of the models, they are not sufficient. Indeed, due to the slight changes between `SWIFT-GEAR` and `GEAR` (e.g. `SPHENIX`, random numbers generation, gravity, ...), a new calibration is required as it will be shown in Figure 6.8. Thus the end of this section focuses on two individual galaxies run until $z = 0$. They were picked in order to represent the faintest and the largest galaxies in the previous set of simulations. The largest galaxy provides the results of one of the 4 massive galaxies at $z = 0$ (after recalibration) that were stopped above $z = 4.3$ for the scaling relation (without recalibration).

6.2.1 Scaling Relations

In Figure 6.3, the velocity dispersion of stars is compared to the V-band luminosity of the dwarf galaxies (derived from the population synthesis model of Vazdekis et al., 1996). The velocity dispersion is computed along 100 different line of sights (LoS) in a radius of 1 kpc using only the component along the LoS. The value and its uncertainty are computed from the mean and the standard deviation of all the LoS. Only a single simulation is producing a luminosity below 10^4 $\mathrm{L}_\odot$. While it lies on the scaling relation, the simulation is not well resolved and should not be fully taken into account. Our galaxies are able to reproduce the observed relation between the velocity dispersion and the luminosity of dwarfs over 3 order of magnitude and could even go as far as 4 order of magnitude if we include the less luminous galaxy. While we lack of luminous galaxies at z=0, the ones at higher redshift are already on the scaling relation and will continue to increase their luminosity as they are still producing a large amount of stars. This increase of luminosity should allow to reach the 4 order of magnitude mentioned in Figure 5 of Revaz and Jablonka, 2018.

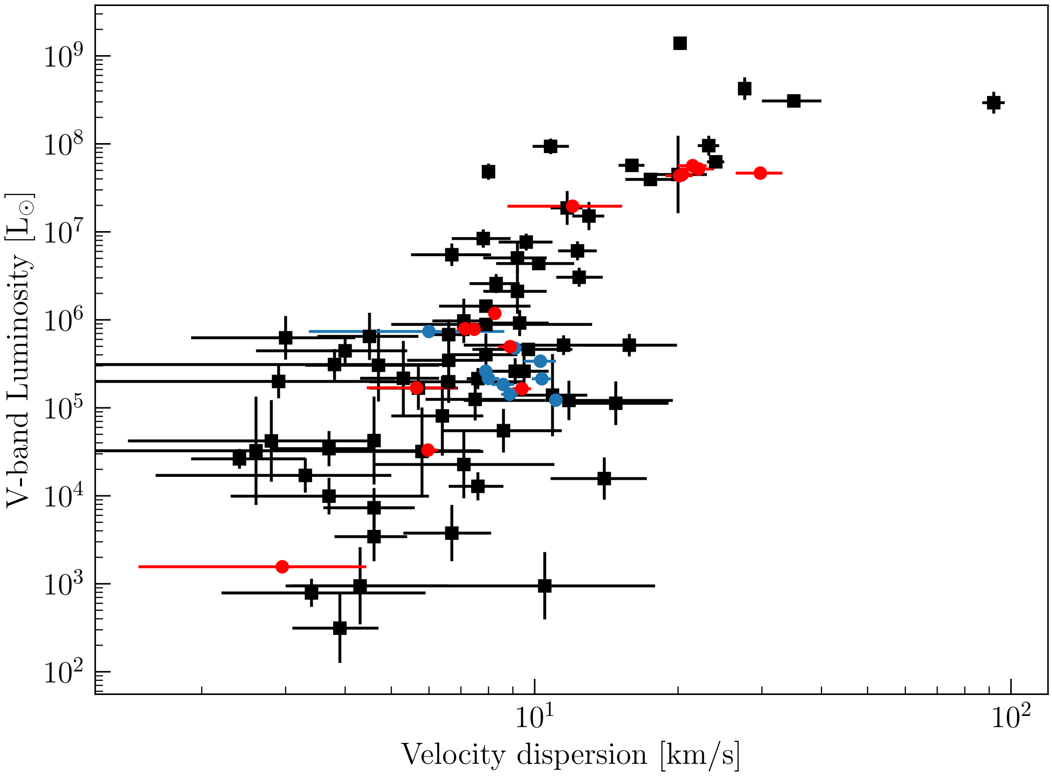


Figure 6.3 – V-band luminosity of dwarf galaxies as function of the stellar velocity dispersion. Observations of galaxies around the Milky Way and Andromeda are shown in black (McConnachie, 2012) and the simulations from `SWIFT-GEAR` in blue ($z = 0$) and red ($z \neq 0$). This Figure corresponds to Figure 5 in Revaz and Jablonka, 2018.

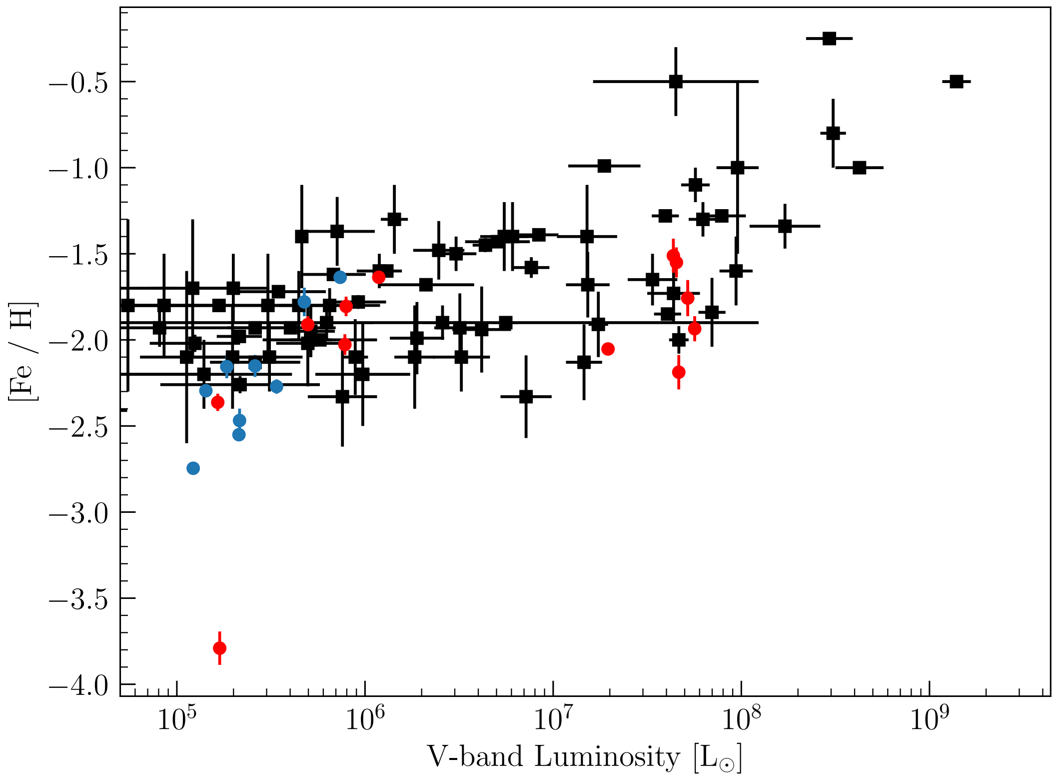


Figure 6.4 – The median [Fe/H] of stars within dwarf galaxies as function of the V-band luminosity. Observations of galaxies around the Milky Way and Andromeda are shown in black (McConnachie, 2012) and the simulations in blue ($z = 0$) and red ($z \neq 0$). This Figure corresponds to Figure 6 in Revaz and Jablonka, 2018.

In Figure 6.4, the median [Fe/H] value of stars are compared with the luminosity of dwarf galaxies. At low luminosity, our dwarfs are not producing enough iron and therefore are not on the observed scaling relation. This can be seen in other groups (e.g. Wheeler et al., 2019; Grand et al., 2021) and could be due to the missing physics of the first stars that are dominant in such low mass galaxies. At high luminosity, our galaxies are at the bottom of the scaling relation, and it is not seen in Figure 6 of Revaz and Jablonka, 2018 due to the non-zero redshift of the simulations presented here. As mentioned before, the galaxies are still producing stars and will continue to increase the median [Fe/H]. Thus, the red dots represent lower limits in term of metallicities.

Figure 6.5 displays the relation between the remaining gas of dwarf galaxies and the luminosity. The mass of gas is taken inside a radius of r_{200} and below a temperature of 10^4 K. For the observations, HI is the main component of the gas in such systems and can be used for reference (Cormier et al., 2014). The scaling relation corresponds well to Figure 8 in Revaz and Jablonka, 2018.

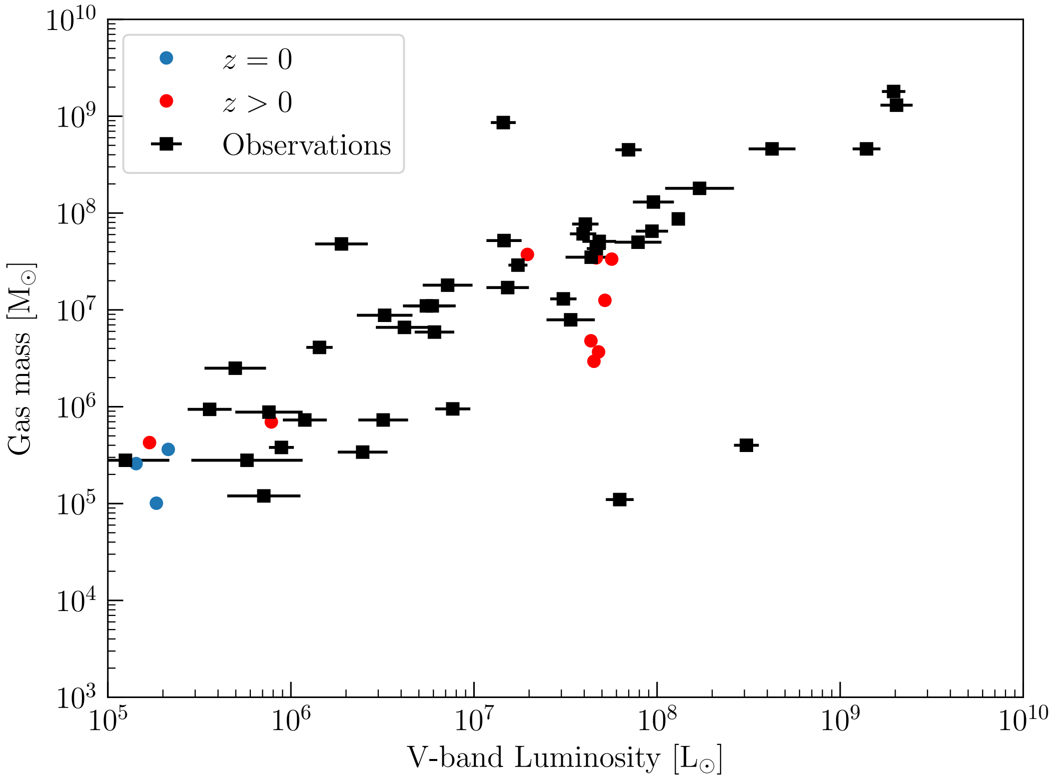


Figure 6.5 – The remaining gas within r_{200} in the galaxies as function of the V-band luminosity. The masses of neutral hydrogen (HI) of dwarf galaxies around the Milky Way and Andromeda are shown in black for comparison (McConnachie, 2012) and the simulations in blue ($z = 0$) and red ($z \neq 0$). Only the gas with a temperature below 10^4 K is shown for the simulations. This Figure corresponds to Figure 8 in Revaz and Jablonka, 2018.

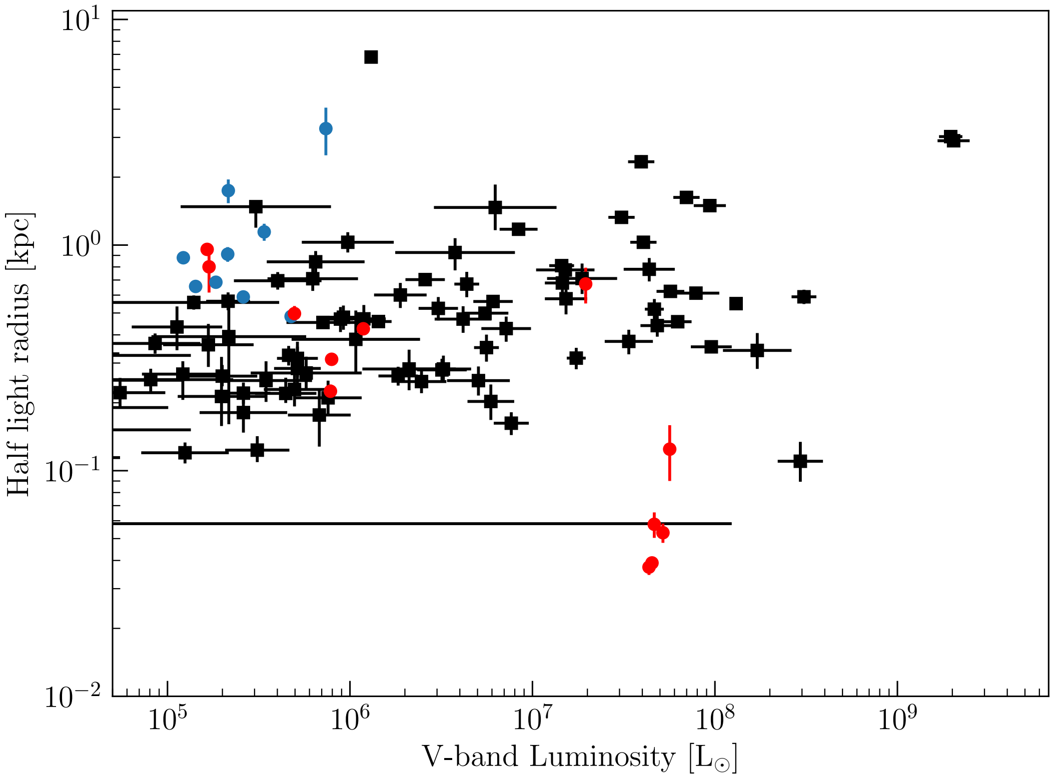


Figure 6.6 – Half-light radius as function of the V-band luminosity. Observations of galaxies around the Milky Way and Andromeda are shown in black (McConnachie, 2012) and the simulations in blue ($z = 0$) and red ($z \neq 0$). As reported in Revaz and Jablonka, 2018, the stars are dynamically heated due to the difference of masses between stars and dark matter particles (ratio between masses of ≈ 22). This Figure corresponds to the top of Figure 7 in Revaz and Jablonka, 2018.

In Figure 6.6, the luminosity is compared with the half light radius. As mentioned in Figure 7 of Revaz and Jablonka, 2018, the large difference of mass between stars and dark matter particles produce a mass segregation and thus the half light radius is artificially increasing with time after the end of the star formation and explains the larger size of our galaxies at z=0. As the half light radius increases with time due to the creation of new stars, the most luminous galaxies which did not reach z=0 are smaller than expected from the observations.

6.2.2 Individual Galaxies

Before moving to a picture of individual galaxies, let us see the star formation history (SFH) of most of the galaxies simulated in order to show the necessity to recalibrate the simulations. In Figure 6.7 and 6.8, a comparison between the SFH of `GEAR` and `SWIFT` is shown. As it can be

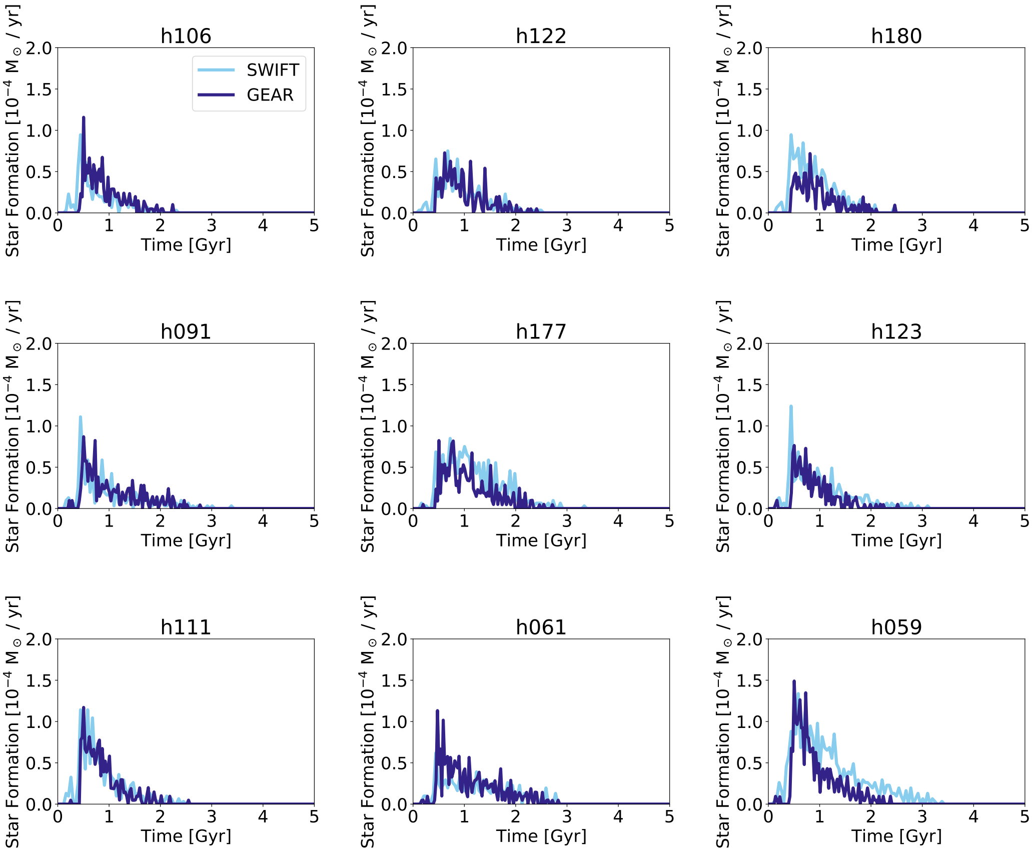


Figure 6.7 – Comparison of the star formation histories between `SWIFT` in light blue and `GEAR` in dark blue. The galaxies are ordered approximately from the lowest to highest star formation rates. All the galaxies, except from the 4 most massives, are plotted with the same axis and can be directly compared together. The comparison is continued in Figure 6.8. While most of the galaxies show approximately the same star formation history between the two codes, a few massive ones show large differences. It shows the need to slightly re-calibrate the code.

seen, the history for the smallest galaxies is more or less matched. The differences, in both low mass and high mass dwarf galaxies, can be explained by the chaotic behavior of galaxies and the slight changes in the models (e.g. `SPHENIX`). As low mass galaxies are mostly quenched by the UV background and already have a low star formation rate, increasing the stellar feedback has little effect on their star formation rate. On the other hands, the more massive galaxies, with their high star formation rate, are deeply impacted by it. In order to have the same star formation rate between `GEAR` and `SWIFT-GEAR`, the simulations require a recalibration. It was done through an increase of the feedback efficiency from 10% to 25%. Only two simulations where rerun with this increased feedback and were selected in order to represent the smallest and largest galaxies within the previous sample.

With Figure 6.9, we can move from a global point of view to a single galaxy. This figure shows the properties of a single low mass galaxy (h177) in blue compared to Sextans in red. As it

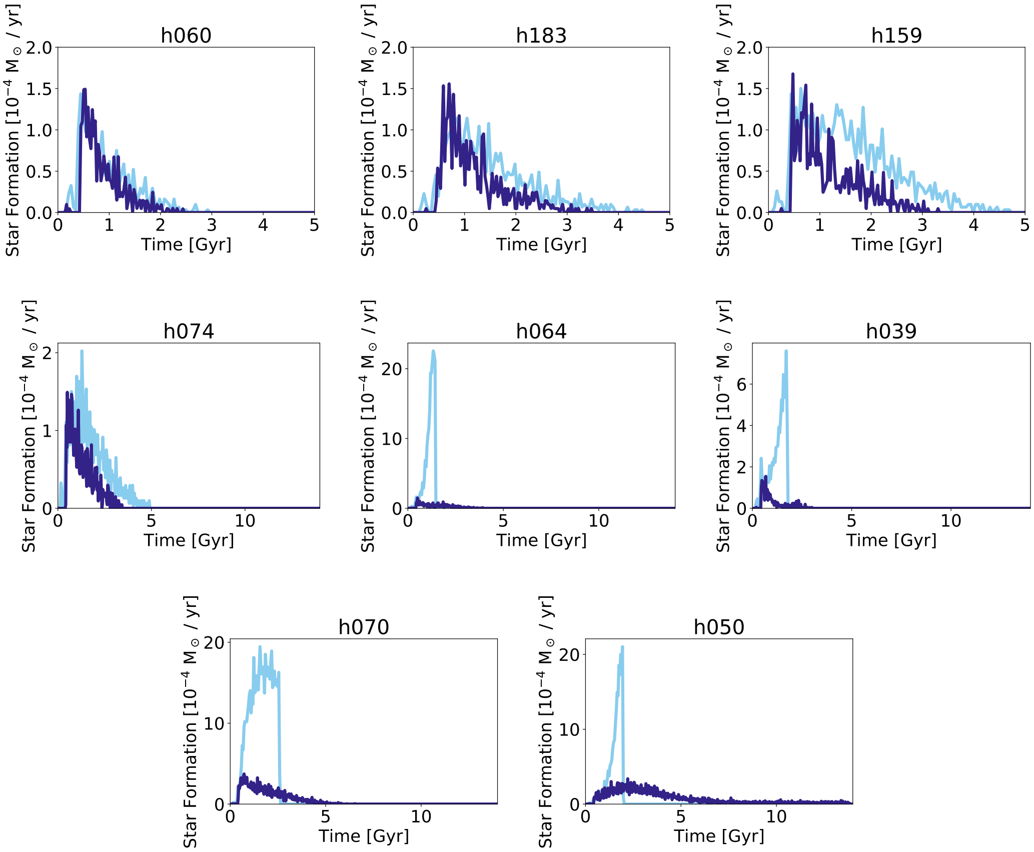


Figure 6.8 – End of Figure 6.7. `GEAR` is given in dark blue and `SWIFT` in light blue.

can be seen, the simulation correctly reproduces the observations of metallicities and velocity dispersion in stars. It also correctly reproduces the results from `GEAR` (see Figure 10 in Revaz and Jablonka, 2018). In the upper left corner, the star formation history of the simulated dwarf is shown. This galaxy produces stars until the UV background starts (around 0.5-1 Gyr) and then the external heating of the gas completely stops the formation of new stars. In the upper right corner, the distribution of stars with respect to their [Fe/H] is shown. In the lower left corner, the velocity dispersion of the stars along the line of sight is shown. In order to compute the uncertainties, 1000 lines of sight have been computed from the simulation. In black the circular velocity is shown and is computed from the distribution of mass in the simulation. In the lower right corner, the abundances of the stars are shown for [Mg/Fe] vs [Fe/H]. As explained in the introduction, the first abundance shows the impact of supernovae II and Ia and the second one indicates the time as galaxies are continuously producing more metals. The type II supernovae produces a roughly constant [Mg/Fe] ≈ 0.4 while the type Ia supernovae have [Mg/Fe] ≈ -0.6. At early times only the supernovae of type II explode and thus produce a plateau around 0.4 until [Fe/H]≈ -2.5. Then the supernovae of type Ia start to explode and slowly decrease the [Mg/Fe] of the next generation of stars. `SWIFT-GEAR` is predicting a slightly larger amount of magnesium that could indicates a star formation efficiency that is too strong in comparison to observations.

In Figure 6.10, a different simulation is shown in the same way and shows the properties of one of the galaxies with, initially, the most extreme difference observed between `SWIFT-GEAR` and `GEAR` (h070). As one can see, thanks to the calibration, the simulation behaves in a similar way than in `GEAR` (see Figure 10 in Revaz and Jablonka, 2018) and the observations are well matched. The star formation rate is slightly below the one in Revaz and Jablonka, 2018, but it does stop around the same time (5 Gyr). The iron distribution peaks a bit higher than the observations of Sculptor but the overall distribution is well reproduced. In the case of the velocity dispersion, the central velocity is slightly above the observations, as in `GEAR`, and can be explained by the difference of mass mentioned before. Finally the abundances show a knee at the same position than observations and, as in Revaz and Jablonka, 2018, the slope is slightly lower.

From the two previous figures, `SWIFT-GEAR` is clearly able to reproduce the observations and the results from `GEAR`. Before fully moving to `SWIFT-GEAR` for our simulations, we still need to ensure that the code is indeed faster. Thus in the next section, we will look at the scaling of the two codes.

6.3 Optimization of the Code for Zoom Simulations

As `SWIFT` was initially designed by people working on the `EAGLE` simulations, the code is well optimized for large volume and low resolution simulations (Schaller et al., 2016). Unfortunately, it does not mean that the code will perform well on zoom simulations at high resolution. Indeed, `SWIFT` required a few optimizations specific to such simulations in order to be faster than `GEAR`. The largest issue is how to properly distribute the work when most of the volume will contain boundaries particles that will require almost no work. In this section, we first explore the time

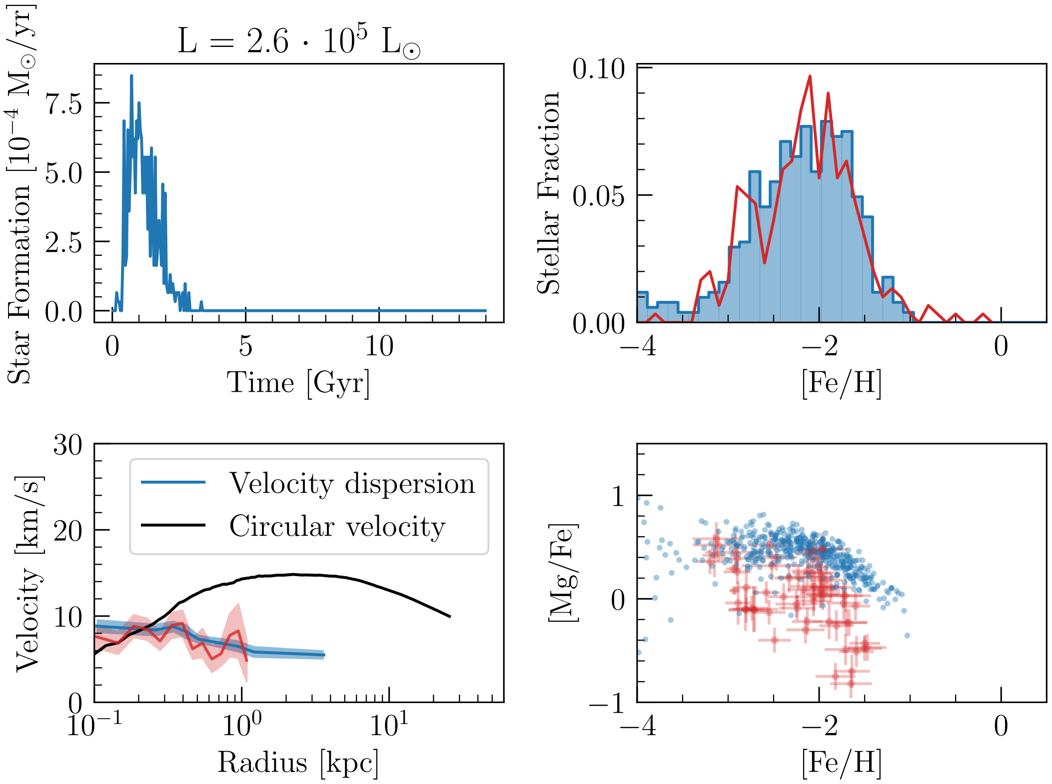


Figure 6.9 – Comparison of the properties of Sextans (red) with one of our simulation (h177 in blue). The V-band luminosity of the simulated galaxy is given at the top of the figure. The upper left panel shows the star formation rate as function of time. The upper right panel shows the distribution of [Fe/H] among stars. The lower left panel shows the velocity dispersion of stars (in blue and red) and circular velocity (in black) as function of the radius. The lower right panel shows the metallicity of stars through [Mg/Fe] vs [Fe/H].

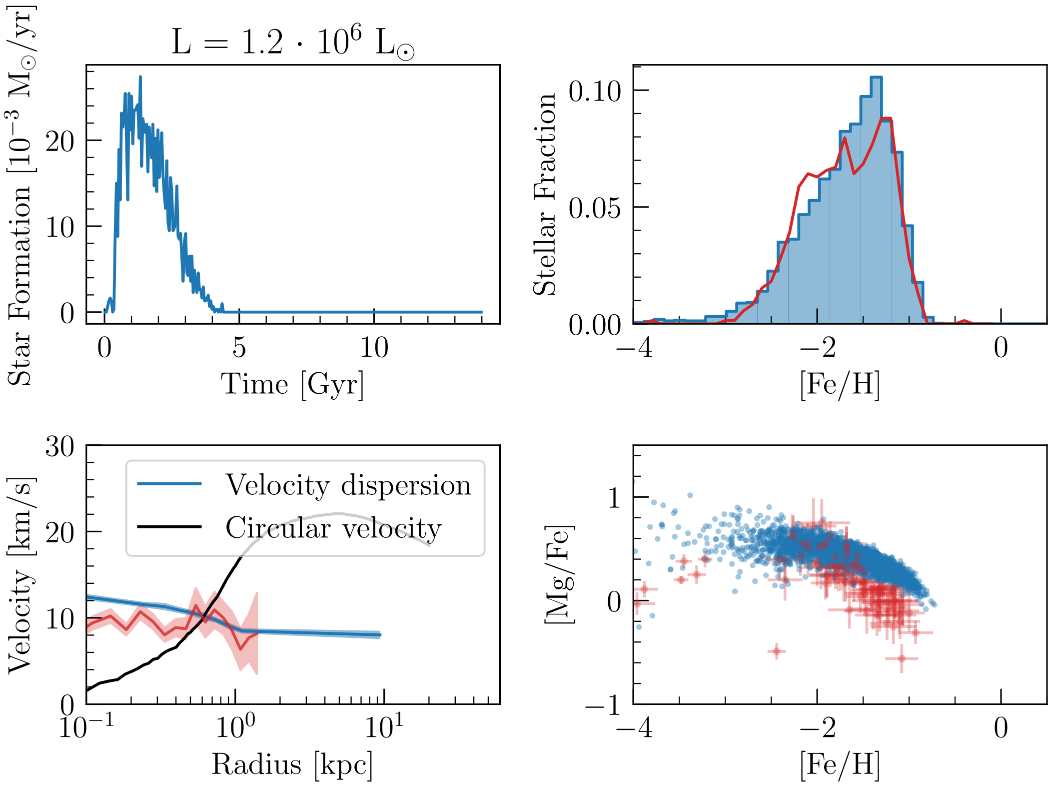


Figure 6.10 – Comparison of the properties of Sculptor (red) with one of our simulation (h070 in blue) as done in Revaz and Jablonka, 2018. The V-band luminosity of the simulated galaxy is given at the top of the figure. The upper left panel shows the star formation rate as function of time. The upper right panel shows the distribution of [Fe/H] among stars. The lower left panel shows the velocity dispersion of stars (in blue and red) and circular velocity (in black) as function of the radius. The lower right panel shows the metallicity of stars through [Mg/Fe] vs [Fe/H].

taken in each step. We will then discuss the scaling. Before starting, It is worth to mention that `SWIFT` was designed to run very large simulations and thus will not be good for simulations with a low number of particles such as in the simulations presented before. Therefore, this section is concluded with the `AGORA` cosmological zoom simulations that contains far more particles.

6.3.1 Optimization of a Single Step

Before analyzing the graphs in Figure 6.11, let us start with an explanation of their content. On each sub-figure, the work done by the tasks is represented for a single step of h050 at late time ($z \approx 1.6$). On each line, the timeline of a single CPU core is shown and each color represents a type of task. The important message obtained from this type of graph is given by the white spaces that represent dead time where the threads are not able to find any task. If the code was perfectly parallel, the tasks should form a single block without any white space in between as in the first part of the last graph. In all cases, the white spaces are generated by a lack of available tasks. It is usually due to too many dependencies or conflicts to solve that allows less tasks to be run than threads, but also happens at the end of the steps where almost all the tasks except a few have been ran. When running over multiple MPI ranks, the communication tasks are an important source of missing dependencies especially on small steps (meaning low number of active particles). Finally, the two vertical dashed lines represent the start and end of a step. While a step always finishes with a task, the start consists in a few functions that are not done within the task system (such as the activation of the tasks).

Let us now move to the analysis. The first graph of Figure 6.11 represents a simulation without any optimizations specific to zoom simulations. Even if the step contains a small fraction of particles, the required time is well above 3000 ms. This is due to some tasks that are not able to run deep enough in the octree and thus cannot be run on more than 1 thread at the time due to the conflicts between the tasks (e.g. tasks acting on the same particles). Another issue present in this simulation, but not seen for this step is the large amount of time used before the task based computation. This is due to the activation of the tasks (about 400ms on some steps).

In order to improve the scaling, some tasks have been pushed deeper in the tree. To do so, two important changes were made and both consists in adapting the criterion for splitting the tasks. `SWIFT` was designed to ensure that all the neighbors are contained within the closest neighboring cells. Thus, a task cannot be split if the largest smoothing length (SL) is comparable to the cell size. The first optimization was to use only the SL of the active particles. As the time step is proportional to it, it means that the most active particles have also the smallest SLs and thus, most of the time, the tasks will be run deeper than the largest SL would allow. The second optimization was to partially split the tasks. It means that the particles with large SLs are run with large cells and particles with small SLs are run with a task deeper in the tree [1]. This reduces the impact of the particles in a low resolution environment (e.g. at the outskirt of the galaxies). In such cases, the distribution of particles is not homogeneous and can create situations where a single particle

---

[1] Both optimizations are currently not implemented within the master.

contains a complete galaxy inside its smoothing length.

For the activation of the tasks, I have implemented a more clever walk of the tree. When activating the tasks, `SWIFT` goes into all the cells and activate the tasks if the particles are active. While this was already implemented in parallel and based on the top level cells, in zoom simulations, a single thread was doing all the work as only a low number of top level cells contain a deep hierarchy. To speedup this function, I have implemented a flag system to ensure that each cell is visited only once. This resulted in a speedup of roughly 10x for this single function. In Figure 6.11 (b) and (c), the same graph has been produced but with a new version of the code and on far larger steps. Two different steps have been selected.

To show the impact of the optimizations, Figure 6.11 (b) shows an average number of active particles for h050 and Figure 6.11 (c) a step where all the particles are active. As it can be seen, the largest steps with the optimizations are faster than the smallest steps without them. While on the largest steps, the dead time is relatively small (white space in the graph), on the smallest and medium steps, a large improvement could still be done.

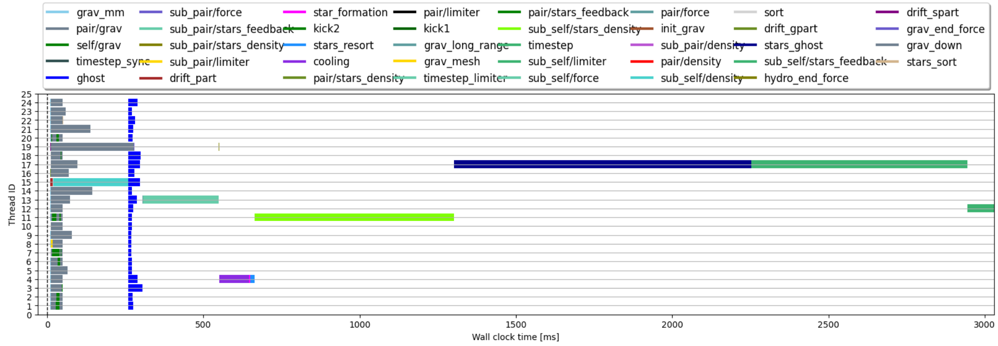
6.3.2 Strong Scaling

In this section, I will start with an example of strong scaling at low redshift. The choice was made to have the most accurate strong scaling of the codes individually and not necessarily have a fair comparison. Therefore any comparison should be done with care. In the second part, I will show the strong scaling at high redshift. In this case, it is possible to make a fair comparison between the code.

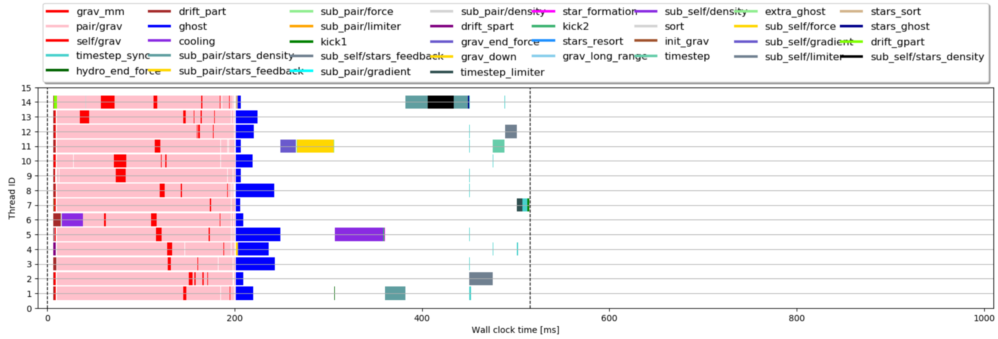
In Figure 6.12, the scaling of `GEAR` is compared with the one of `SWIFT` for h050 at $z \approx 1.6$ (without recalibration). The top left graph represents the speedup obtained by increasing the number of threads, the top right represents the parallel efficiency of the code (speedup divided by the number of threads) and the last graph the time to solution. All the graphs are done as function of the number of threads. The parallel efficiency is the best choice to look at for the scaling of a code while the last one for comparing the overall speed. While both codes could restart from the same IC, it was not done as it does not represent an accurate scaling of a cosmological simulation due to the perturbation of the restarting procedure (both in terms of physics and numerical computations such as when the octree is rebuilt). Both codes were run until the required redshift and restarted using their own files (the restart files for `SWIFT` and a snapshot for `GEAR`[2]). Each simulation was run for 1000 steps and only the time spent during the time steps is used (e.g. not including the restarting mechanism but including the octree rebuild, communications, tasks activation, ...). While the figure accurately represents the individual scaling of each code at low redshift, it does not mean that the two codes are simulating exactly the same object due to the lack of recalibration. In this case, `GEAR` only simulates $\approx 7'000$ stellar particles and `SWIFT` $\approx 25'000$. As the stellar particles impact more the scaling than the other types of particle and the SPH

---

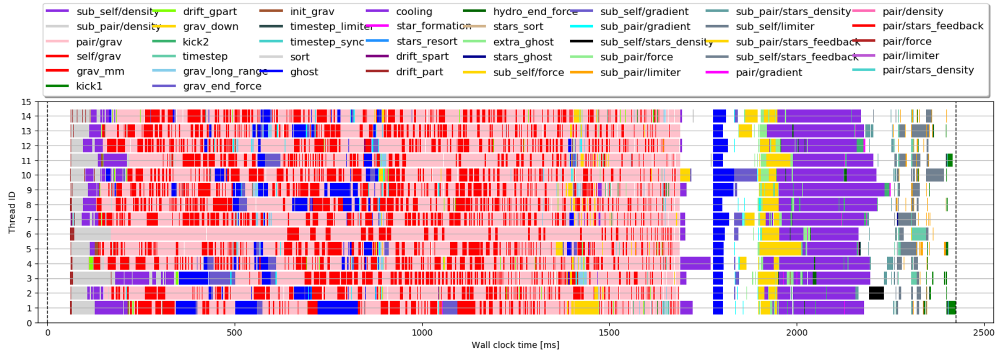
[2] `GEAR` is unable to restart with a different number of threads from the restart files.



(a) Small fraction of particles updated (before any optimizations)



(b) Moderate fraction of particles updated (after optimizations)



(c) Large fraction of particles updated (after optimizations)

Figure 6.11 – Work done by the tasks during a single time step for the dwarf galaxies h050 at $z \approx 1.6$. While the first graph shows the situation before any optimizations specific to our simulations, the two others show the situation with some optimizations done (see the text for details on the optimizations). The graph (b) represents a step with an average number of active particles while the graph (c) represents a step with all the particles active. On each graph, each line represents a different thread and each color represents a different task (not necessarily the same colors between the graphs). The vertical dashed lines represents the beginning and end of the step. The white space between the dashed line and the first tasks corresponds to the work done before any task based computation (e.g. activation of the tasks, drift of the multipoles, task scheduling, ...).

flavor in `GEAR` is simpler, the speedup offered by `SWIFT` in comparison to `GEAR` is certainly underestimated in this graph. In addition, `SWIFT` is using `SPHENIX` that requires three different loops over the neighbors for the hydrodynamics while `GEAR` requires only two. Anyway, it seems that `SWIFT` is roughly 2 times more efficient than `GEAR` on this example but as it can be seen for the parallel efficiency, it scales better than `GEAR` with the number of threads. This example is relatively small (about 1 million particles) and thus `SWIFT` does not have enough tasks to properly reduce the dead time (67 % on 28 threads). Therefore, we can expect a better scaling on larger simulations.

In Figure 6.13, the scaling of the two codes is shown again but in a situation more favorable to `SWIFT` (e.g. larger number of particles) and at high redshift. The simulation used is the one presented in chapter 3. It consists in a zoom simulation of a Milky Way like galaxy with a gas resolution of $5.65 \cdot 10^4$ $M_\odot$ ($2.8 \cdot 10^5$ $M_\odot$ for the dark matter). The simulation contains initially about 15 millions gas particles and 19 millions dark matter particles. As the situation is more homogeneous, only 100 steps were done as they already represent a good estimation of the scaling. `GEAR` was not run with less than 4 cores as it requires to restart the code. Indeed, the simulation needs more time than the wall clock limit on the EPFL's clusters (3 days) and could bias the results. In order to properly compute the speedup and the parallel efficiency for `GEAR`, a perfect scaling was assumed from the simulation with 4 threads to get the time for a single thread. In this case, we can clearly see the advantage of `SWIFT`. Even if the scaling of `GEAR` is slightly better, `SWIFT` is 7.65 times faster than `GEAR`. The good scaling of `GEAR` is mostly due to the simplicity of many unnecessary drifts. Indeed, `GEAR` is drifting all the particles at every time step while `SWIFT` is only drifting the required particles.

To conclude this chapter, as expected, `SWIFT` has been shown to reproduce the observations and the behavior of `GEAR`. It is also largely faster than `GEAR` even if the optimizations related to zoom simulations are only at their beginning and many more should arrive in the future.

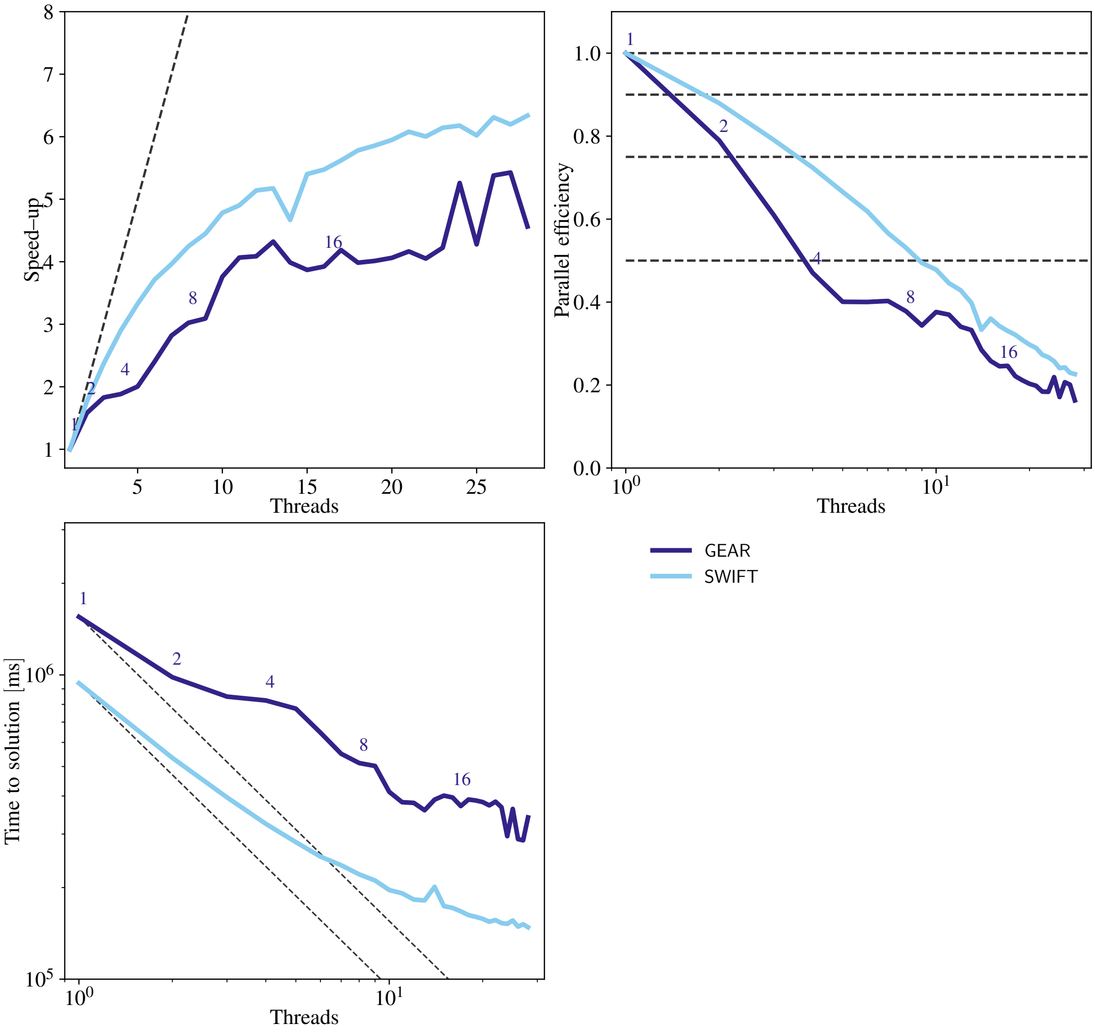


Figure 6.12 – Scaling of the codes on h050 at $z \approx 1.6$ and for 1000 steps. Each code restarted from its own output and thus does not contain the same number of stellar particles. `GEAR` contains approximately 7’000 stellar particles while `SWIFT` contains about 25’000. On the upper left, the speedup (total time for 1 thread / total time) is shown as function of the number of threads. On the upper right, the parallel efficiency (speedup / number of threads) is shown as function of the number of threads. On the lower left, the total time as function of the number of threads is shown.

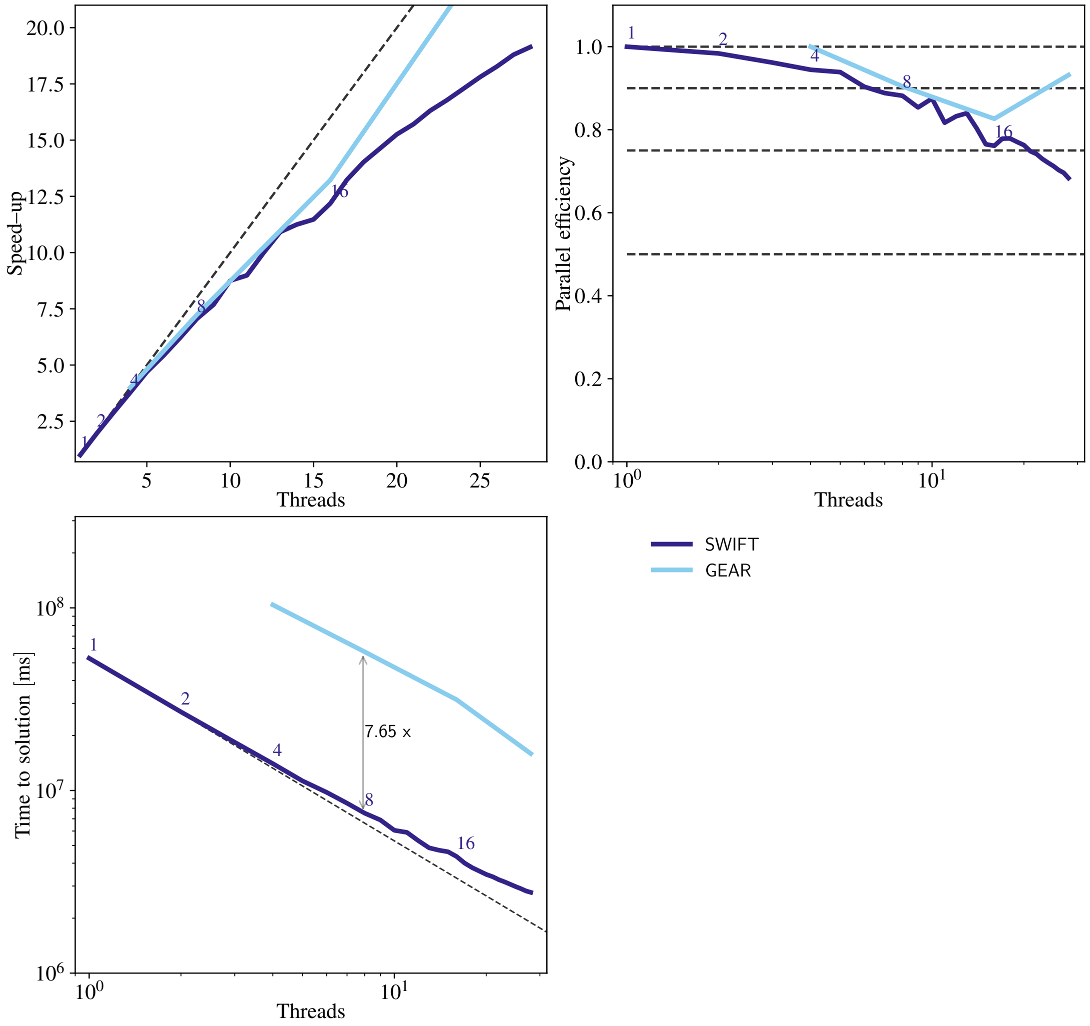


Figure 6.13 – Scaling of the codes on the `AGORA` cosmological simulation at high redshift and for 100 steps. Each code started from a common initial condition and thus simulated exactly the same part of the universe. `SWIFT` uses a more complex hydrodynamics scheme (`SPHENIX`) than `GEAR`. Thus, it requires an additional loop over the neighbors here. On the upper left, the speedup (total time for 1 thread / total time) is shown as function of the number of threads. On the upper right, the parallel efficiency (speedup / number of threads) is shown as function of the number of threads. On the lower left, the total time as function of the number of threads is shown. As `GEAR` was not run with less than 4 threads, a perfect scaling is assumed from 4 threads in order to get the total time for 1 thread. In this case, `SWIFT` is 7.65 times faster than `GEAR`.

Next Generation of Simulations Part IV

7 Continuous Simulation Data Stream (CSDS)

> It's the job that's never started as takes longest to finish.
> — J. R. R. Tolkien, The Fellowship of the Ring

While most of the work on the simulation outputs has been made towards faster writing (Xiao et al., 2012; Ross et al., 2008; Ma et al., 2006; Mitra et al., 2005), and abstraction level (Godoy et al., 2020; Lüttgau et al., 2018; Zheng et al., 2013; Abbasi et al., 2009), little work has been done on how and what to write into the storage. An interesting example of such work are the output for light cones images [1] (Evrard et al., 2002). Cosmological simulations are especially well designed to rethink the traditional approach of storing the complete state of the simulation at regular intervals.

As the ratio of timescales between the lowest and highest density regions can easily reach 1000 (Springel, 2005), using a single time step for all the particles would be largely inefficient. Indeed, the code would spend most of its time evolving particles with large time scales at an extremely large and unwanted time resolution. While recent codes are all using a multi-time step approach and resolve each particle at the same relative accuracy in time, this approach was never used, to my knowledge, in the design of the output files (traditionally snapshots). Therefore, a large fraction of the particles stored in the snapshots will be over resolved in time while the particles in the highest density (and most interesting) regions will be under resolved.

With the Continuous Simulation Data Stream (CSDS), we tried to reduce the amount of disk space required for the output by using the information about the different timescales. It means that for the same amount of storage space, we can achieve a larger time resolution. Such resolution

---

[1] Light cone images are trying to reproduce the behavior of the observations by stacking together the simulation at different epoch. Thus, looking far away is equivalent to looking in the past.

will also help, for example, to analyze the impact of fast events on the interstellar medium such as supernovae. Our system also enable us to define the output directly from parameters linked to the time resolution and not from an arbitrary number of snapshots. Finally, this approach could generate exact light cones at the end of the simulation without any need of specifying an origin during the simulation as in the approach of Evrard et al., 2002.

The idea behind the CSDS is to move from an output system where 1 file corresponds to the state of the system at a given time towards an output system composed of a single file which describes the whole evolution of the simulation (logfile). The file is composed of small chunks of data called records that can contain either a time or a single particle. At the beginning (end) of a simulation, we start (finish) with a record for each particle present in the simulation. For each step, we start by writing a time record followed by a record for each active particle meeting our writing criteria. In order to identify the records and quickly follow the evolution of a particle, each record's header contains the localization of the previous record along with the type of data contained (e.g. time record vs particle record, and also the fields contained in the particle). We complement the logfile with a set of index files that allows a quick access to the simulation at any time. Those files are a kind of snapshots where we store only the information about the localization of the particles and their IDs. As they do not contain any physical information, they can be written at a far lower rate than snapshots and do not need as many fields. In case of simulations that create or remove some particles (such as done for star formation, AGN interactions or simply removal of particles leaving the simulation volume), we also include in the index files the information of when they have been removed or created. For distributed memory parallelization, we simply consider each rank as being an independent simulation and the particles leaving a rank are considered as removed from the simulation (and created on the other rank). As not all the particles are written at the same time, an interpolation is performed between two matching records to reconstruct the state of the simulation at any time.

While, as expected, this technique has been shown to have a slower reading speed than the snapshots, the fact that we can generate snapshots from the CSDS and can largely increase our time resolution at the same disk space cost is largely compensating. In the next pages, our paper (in preparation) on the CSDS is shown and presents in details the techniques and efficiency of the CSDS along with some use cases.

Astronomy & Astrophysics manuscript no. main
August 31, 2021

Continuous Simulation Data Stream (CSDS): A dynamical, timescale-dependent, output scheme for simulations

Loic Hausammann,[1] Pedro Gonnet[2] and Matthieu Schaller[3,4]

[1] Laboratoire d'Astrophysique, Ecole Polytechnique Fédérale de Lausanne (EPFL), 1290 Sauvergny, Switzerland
e-mail: `loic.hausammann@epfl.ch`
[2] Google Switzerland, 8002 Zürich, Switzerland
[3] Lorentz Institute for Theoretical Physics, Leiden University, PO Box 9506, NL-2300 RA Leiden, The Netherlands
[4] Leiden Observatory, Leiden University, PO Box 9513, NL-2300 RA Leiden, The Netherlands

Received XXX; accepted XXX

ABSTRACT

Context. Exa-scale simulations will arrive soon but almost no new design for the output has been proposed in recent years. In simulations using the individual time step approach, the traditional snapshots are over resolving particles/cells with large time steps and are under resolving the particles/cells with low time steps. Therefore, they are unable to follow fast events and use efficiently the storage space. The Continuous Simulation Data Stream (CSDS) was designed to decrease this space while providing an accurate state of the simulation at any time. It takes advantage of the individual time step to ensure the same relative accuracy for all the particles. The outputs consist in a single file representing the full evolution of the simulation. Within this file, the particles are written independently and at their own time scales. Through the interpolation of the particles, the state of the simulation can be recovered. In this paper, we show that the CSDS can reduce the storage space by one order of magnitude for the same accuracy than snapshots with an acceptable reading speed.
Aims.
Methods.
Results.

Key words. methods: numerical – software: simulations – simulations: I/O

1. Introduction

With the arrival of the exascale era, scientists across all domains have been focused on improving the performances of their software. They have been mostly looking at the scaling of their own physical computation (examples in astrophysics include Schaller et al. (2016); Jetley et al. (2008); Potter et al. (2017); Adams et al. (2009); Müller et al. (2019)) with little attention given to the way of writing the outputs. Indeed, in recent years, the HDF5 format has become a de-facto standard and scientists rely often solely on the improvements within the library itself for their own code. This approach is generally succesful with simulations writing data in parallel from 1000s of inter-connected nodes. This format is particularly well adapted for snapshots that consist in writing the state of the simulation at a few discrete times (Nelson et al. 2015; Norman et al. 2007). Anyway, there is still some studies or software that have been focused on improving the performances of the I/O (Xiao et al. 2012; Ross et al. 2008; Ma et al. 2006; Mitra et al. 2005) or to increase the level of abstraction (Godoy et al. 2020; Lüttgau et al. 2018; Zheng et al. 2013; Abbasi et al. 2009), but little work has been done on rethinking the way we store the simulations.

An interesting case where such rethinking was done is for the light cone images. Such images consist in reproducing the behavior of observations where looking further away corresponds to looking back in time. As only the particles contained within the light cone surface are interesting, techniques have been developed to increase the performance of the simulation (Garaldi et al. 2020) and to write the outputs in a more efficient way (Evrard et al. 2002). This last technique consists in writing a kind of snapshot where the particles are written only when they cross the light cone surface. While this approach needed to rethink the outputs, it is not a general solution for astrophysics.

In cosmological simulations (but also in other astrophysical simulations), the gravity is producing large difference of time scales through the simulated volume. As it would require too much computational time to use a single time step, a multi and individual time step approach was designed where each particle or cell is evolved at its own time scale. This approach ensures that all the particles/cells are evolved at the same relative accuracy (Aarseth 1963; Springel 2005). In Figure 1, we show the distribution of time steps within a cosmological simulation run with the EAGLE model (Schaye et al. 2015). Only an almost negligible fraction of the particles actually needs a small time step and the analysis often focus on it. Even if this simulation is relatively small (13 millions of particles), already a ratio of 1000 is seen between the lowest and largest time steps. For the outputs, it means that writing all the particles/cells together provides a far too high accuracy for particles/cells within voids and far too low for particles/cells within galaxies. A consequence is that the snapshots are unable to follow accurately fast events such as supernovae and render their comprehension more complicated. Therefore the snapshots are not efficient in term of storage space and accuracy for such simulations.

In SWIFT (Schaller et al. 2018), we are developing a new approach called the Continuous Simulation Data Stream (CSDS)

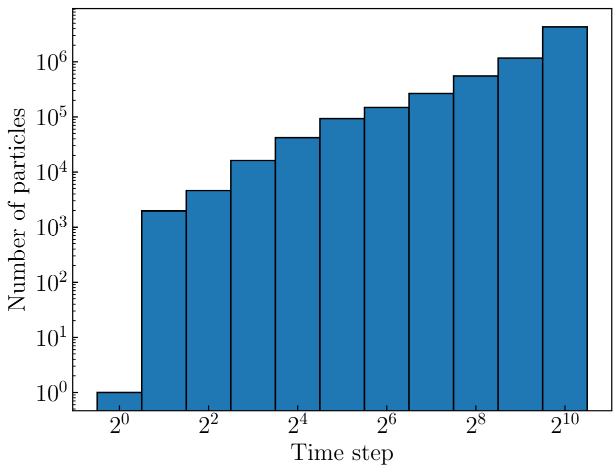


Fig. 1: Distribution of time steps (normalized by the smallest time step) for a cosmological simulation run with the EAGLE model within a periodic box of 12 Mpc at $z = 3$. On this simulation, the ratio between the largest and smallest time steps is already above 1000 and only a small fraction of the particles really need a high time resolution.

that try to fix this issue and will allow us to write more information for a lower memory cost. The idea is to move from one snapshot corresponding to the complete state of the simulation at a given time towards an approach where a single file describes the whole simulation across time. In this file, the particles are written independently at their own timescale in the form of records. A snapshot can then be reconstructed at any time with an interpolation between records belonging to the same particle. While this approach is slower than the snapshots, it is particularly well designed to study the evolution of a system.

It is worth to mention that everything within this paper is open source and available within the git repository of SWIFT[1] and its own repository for the reader [2]. The code presented here corresponds to the version 1.5 [3]. From now on, we will only discuss about particle based codes but this approach could also be applied to Adaptive Mesh Refinements (AMR).

The paper is organized as follows. In section 2, we describe in details the theory behind the CSDS. In section 3, the implementation within SWIFT is provided followed by section 4 where the reading strategy is explained. Our results are separated in two sections (5 and 6). The first one presents the efficiency of the CSDS and the second one shows some examples of applications.

2. Description of the Continuous Simulation Data Stream

The Continuous Simulation Data Stream (CSDS) is a new output system that aims at replacing, at least partially, the traditional snapshot system. Its main advantage resides in the individual writing of the particles. It allows to capture them according to their own time scale and not at a time resolution far too low or too high for most of the particles. Thus, it perfectly adapts to simulations using individual time steps and can achieve extremely high time resolution while keeping the output size reasonable.

Before describing the CSDS, let us state a few important points:

- While the snapshots represent the state of a simulation at a given time, the CSDS contains a single file (logfile) that represents the whole evolution of the simulation,
- It is a particle based output system that takes into account the time scale of each particle, thus the particles are never written all together and there is no need to synchronize the particles before writing them (e.g. drifting the inactive particles),
- As the particles are not synchronized, an interpolation is necessary to reconstruct the simulation.

The different objects that we will use also need a quick definition (see Figure 2 for their relations):

- The *logfile* contains the full evolution of the particles,
- The *records* are the basic elements that compose the logfile and describe a single particle or a single time step.
- The *index files* are used to speedup the reading and contain the last offset of the particles in the logfile along with the history of the particles created or removed particles[4],
- The *metadata file* contains information about the simulation (such as the compilation options, the parameters, ...),

Let us start with the logfile that contains the information about the entire simulation. Except from the header, the data are stored in the form of records that contain either the current time or a given particle at a given time. When the simulation starts, a header is written in the logfile followed by a time record for the initial time and a particle record for each particle. At each step, the CSDS writes a time record and a particle record for the required particles. It means that the time of the particle records can be obtained through the position of the time records and the CSDS does not require the time information to be stored inside the particle records. Finally, at the end of the simulation, all the particles are generated and a time record concludes the logfile.

As the logfile can reach large sizes, some index files are written and contain the information about the position of the last record of each particle. They also include the particles that have been created or removed from the simulation (e.g. MPI exchanges, star formation and black hole interactions for cosmological simulations). Theses files are only used to speedup the reading and generated after the simulation.

At the beginning of the simulation, a YAML file is written and contains metadata about the simulation (e.g. all the parameters, configuration options, ...). This file will not be discussed as it is code dependent and its design does not impact the CSDS.

2.1. Description of the records

In Figure 2, a record is shown at the bottom. It consists in a header and the data.

In the header, we store the *mask* and the *offset*. The mask is simply a bitmask that describes the fields contained in the data (e.g. coordinates, velocities, internal energies, ...). This means that based only on the bitmask, a record can be identified as being the current time or a particle, but it also means that we can

---

[1] `www.swiftsim.com`

[2] `https://gitlab.cosma.dur.ac.uk/lhausammann/csds-reader`

[3] This version can be obtained from commit d078e2dd in SWIFT and c073baf0 in the reader.

[4] They can be seen as snapshots containing only the ids and the offset of the particles. As they do not contain any physical information, they can be written at much lower frequency than snapshots.

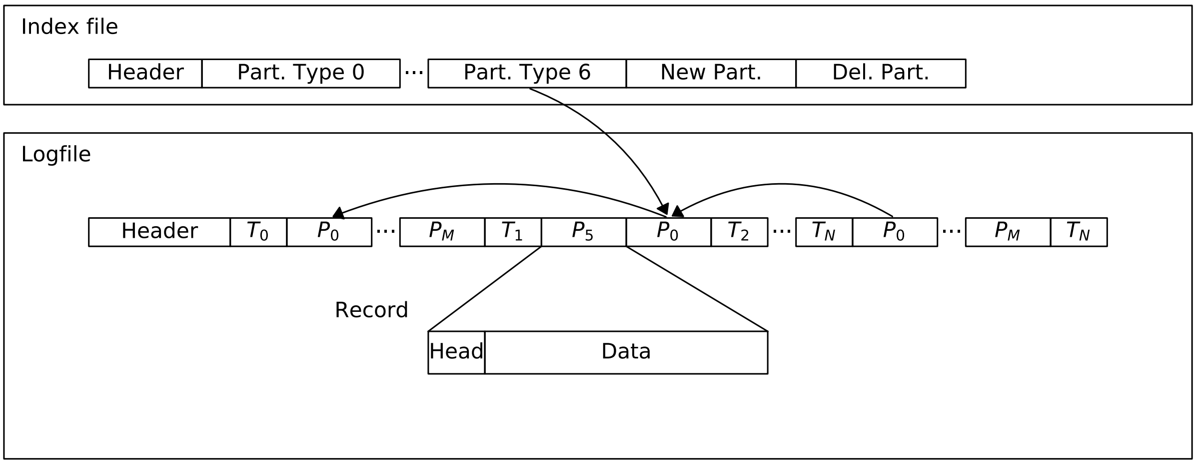


Fig. 2: Representation of the different files. The logfile contains the output of the simulations and the index files can be used to speed up the reading of the logfile. The logfile starts with a header containing information about the format (e.g version numbers and the different masks, see Section 2.2). Then the file will only contain records such as the one in the zoom. They can either contain a time or a particle and are always composed of a header and the data. In the header, a mask is stored in order to describe the data part and also an offset to the previous or next corresponding record (see Section 2.1). At the beginning of the simulation, we first write down the time (T_0) and the initial conditions (P_0 to P_M). Until the end of the simulation, the CSDS writes down the time step at the beginning of each step and only the required active particles (e.g. P_0 and P_5, see Section 2.4 for more information about the writing criterion). Finally, the code writes the particles at the final time and ends up with a second writing of the time step as a sentinel indicating the end of file. The index files are written once the simulation is done in order to speedup the reading. They start with a header (e.g. simulation time, number of particles, ...) and then the particles sorted by type. For each particle, the position of the last record (offset) and their ID is written. At the end, the file contains the information about the particles created and removed since the last index file. They are still written in the form an offset and an ID.

have different writing frequencies for different fields. The offset is the distance in the file to the previous record of the particle (or time step). It means that the evolution of a particle can be quickly followed thanks to this offset. As usually people are interested in moving forward in time, the offset of the records are reversed during the first read of a logfile (initially they point to the previous record and are reversed to point to the next one) [5].

In the data, we simply copy the fields marked with the mask into the file. As the different fields are written the one after the other without any information in between, the order when writing and reading a record must be respected. This order is indicated in the logfile header.

2.2. Description of the Logfile

In Figure 2, the structure of the logfile is shown. The logfile starts with a header that contains the version, the direction of the offset (in order to know if they need to be reversed), the size of the strings, the number of different masks, all the available masks (name and data size), and finally the masks for each particle types (in the writing order). It could be extended with more data without risk as we also store the position of the first record.

When the simulation starts, the initial time is written followed by a record for each particle presents in the initial conditions. As the CSDS requires some interpolations, this step is required in order to avoid any extrapolation. Then at each step, the CSDS writes a time record followed by the required particle records (see section 2.4 for a discussion about the writing criterion). On the steps with a low number of active particles, it can go as low as 0 particle record and on some of steps with the largest number of active particles, it can be all of them. At the end of the simulation, we write the final time followed by all the particles in order to avoid any extrapolation and conclude with a copy of the last time record as a sentinel marking the end of the file.

As most of the computational servers have time limits for the jobs, a strategy to restart a simulation is required. The usual approach is to simply dump the memory and reload it later. To restart the CSDS, it is enough to dump the position of the last record in the file. When restarting, the file only need to be reopen at this position and everything afterwards can be safely discarded.

2.3. Description of the index file

The index files are totally optional as they contain information that exists in the logfile, but in a different form that allows an efficient usage of the logfile. For example, if the simulation at the final time is requested, without the index files, it would be required to read the whole logfile in order to find all the existing particles (assuming the creation of new particles or simply some exchanges over MPI) and then update them until reaching the

---

[5] This operation is one of the slowest. Thus we might drop it in the future and encourage the users to move backward in time.

final time. To solve this problem, the index files contain enough information to reconstruct quickly the state of a simulation at a given time using the logfile.

The index files are generated when reading the logfile and are set at regular interval in time. It is worth to mention that a low number of index files requires a larger amount of random access memory (RAM) than a large number of index files. Indeed, the history of all the particles created or removed in between two index files is kept in memory.

An index file starts with the current time step and number of particle and then an array of ID and offset to the last record is written for each particle (sorted by particle's type). Next, we write down the history of the created and suppressed particles since the last index file. The history consists first in the number of new particles along with an array of ID and offset (sorted by type) and then the same for the suppressed particles.

2.4. Particle writing criterion

The CSDS writes down particles at different times and, therefore, uses interpolations to reconstruct a snapshot at any time. It means that the criterion for writing a particle should depends on the required quality for the reconstruction.

A possibility would be to keep in memory some information about the particle since the last record and use this information to evaluate the quality of the future interpolation. For example, if one wishes to do a linear interpolation of a field f, the error ($\propto \mathrm{d}t^2 \max |f''(t)|$) could be evaluated with the time difference since last record (dt) and the second derivative of f. This method should work well with the CSDS as it can be applied independently to each field and then write only a subset of the fields in each record. It would therefore reduce the size of the logfile without losing accuracy. Unfortunately, it requires several new fields for each field written in the logfile and therefore increases considerably the RAM memory required for the simulation.

A simpler solution is to use a criterion based on the number of steps since the last record (every N active steps). As the time step is computed from the time scale in order to correctly integrate the particle's properties, the quality of the future interpolation is directly linked to it.

2.5. Programming techniques

As the CSDS relies heavily on the access to the logfile (both for the reading and writing), it requires a careful design of the I/O and good strategies to deal with shared/distributed memory parallelization. Firstly the I/O needs to ensure a low number of access to the data storage device in order to reduce the impact of the storage latency. It can be done through the memory mapping of the file. Secondly the CSDS needs to be protected against race conditions due to parallel processing. The solution differs depending on the type of memory (shared or distributed) and are therefore explained separately.

2.5.1. Memory-mapped file

The accesses to the logfile can dramatically decrease the performances of the CSDS due to the storage latency, therefore direct access through the functions `write` and `read` is strongly discouraged as they will perform an operation on the file storage at each call.

In order to avoid this problem, operating systems (OS) provide a function that memory-map a file and perform lazy operations on the file (e.g. `mmap` for the POSIX standard). It means that the OS will load new pages [6] when requested. If the memory access are predictable, the OS can even load them in advance. The pages will be kept in the random access memory until the OS estimates that they will not be accessed anymore. When unloading a page, the OS simply copies it to the file storage.

It is also worth pointing the fact that the file is very easily manipulated through the use of a pointer. This pointer behaves exactly as if the whole file was loaded in memory and the OS manages it for the user.

2.5.2. Concurrency

In shared memory parallelization, two threads can try to access and update the same bits at the same time which results in a race condition. To avoid this problem, atomic operations have been developed in most programming languages in order to ensure that a single thread access a variable at a given time. Unfortunately, the atomic operators slow down the execution of the code and therefore should be used as little as possible.

For the CSDS, when writing a record to the logfile, a race condition can arise as two threads may decide to write at the same place and erase the work of the other. To avoid this problem, the memory-mapped file does not need to be protected, but only the pointer to the next free bytes. When a thread needs to write a record, it simply computes the record's size, increments the pointer by the size with an atomic operator and then can freely write on the assigned memory. This can be done for multiple particles at a time to decrease the amount of atomic operations.

2.5.3. MPI

Unfortunately, the atomics work only in shared memory, thus in distributed memory parallelization, a different technique is required.

Two different possibilities are often used when dealing with files. Either each rank owns a single file or we can predict the required amount of memory used by each rank and can then safely write synchronously in different part of the file. Even if the memory can be easily predicted by checking all the active particles before the beginning of a time step, we decided to use a single logfile per rank for the CSDS as it avoids additional computations. For the index files, we follow the same approach and write a set of files for each rank. Another advantage is that with MPI, the simulation is usually split in subvolume and each volume belongs to a single rank. Thus it avoids reading the entire logfiles when looking at only a small volume of the simulation.

As the different ranks are exchanging particles, the CSDS needs to follow the particles when they are leaving / entering a new rank. In the logfile, a particle record is written when the particle leaves (enters) the rank and contains a flag giving the ID of the other rank. In the index file, the particle is simply considered as being created or deleted and thus is written in the history.

3. Implementation in SWIFT

SWIFT (Schaller et al. (2016)) is an open source code designed for cosmological simulations but also used in planetary science (Kegerreis et al. (2019)) and engineering (Chalk et al (in prep.)). We tried to implement the CSDS as independent as possible from the rest of SWIFT and recommend interested readers to

---

[6] A page corresponds to a part of a file.

copy/adapt the writer from SWIFT into their own code. While the writer may be a bit code dependent due to the differences in physics and implementation, the reader should be quite universal.

3.1. SWIFT

SWIFT uses a task based approach (Gonnet et al. (2016)) that makes it an excellent candidate for the CSDS as it is able to deal with the CSDS output while doing some other computations. Thus it reduces the stress inflicted to the storage device due to the lower number of threads accessing the files at the same time and should provide a higher efficiency than if all the threads were writing at the same time.

SWIFT is designed as an HPC code that interacts the different types of particles[7] together with a minimal knowledge of the underlying interaction models. It means that the code can easily switch between different type of simulations (e.g. cosmology, planets or engineering). During the development of the CSDS, we also tried to follow this modular approach and included different writing strategy for the different particles and physics modules.

3.2. Implementation of the CSDS

In the first sections, an overview of the CSDS was given. As it does not include some technical details, we will cover the most important of them in this section. Let us start with our management of the file size and then we will discuss the mask assignment.

Between every steps, we ensure that the file is large enough to write at least all the particles once with all their fields (even if it does not happen) and, if it is not the case, the size of the logfile is increased by this amount times a factor in order to avoid increasing it at every step. This operation can cost a lot for large file sizes, therefore the user should try to estimate accurately the size and over-allocate a bit.

Due to our modular approach, the masks need to be assigned in a dynamic way as each module defines its own field to write and, depending on the configuration options, the number of fields can change. The mask are written in 2 bytes of the records' header and the offsets use an additional 6 bytes. It means that we can only have 16 masks including the ones for the time and the special flag. Therefore we cannot log too many fields individually and have to group them into a single mask [8]. In an effort to reduce the number of masks needed, we check if two types of particles define the same mask and assign the same value to both of them if it is the case. It is worth to mention that, in order to simplify the reader, we force the mask of the special flag to be 1 (see next section) and for the time 2.

3.3. Special cases

During a simulation, the particles can go through different processes such as type transformation, suppression or creation. For example in cosmological simulations, the star formation is usually done through the transformation of a gas particle into a star particle or could be done, in a less conventional way, by directly creating a new star; the black holes destroy some particles through black hole mergers or gas absorption. For engineering usages, aerodynamics studies use wind tunnel simulations where some particles are created on one side and removed on the opposite side in order to simulate the wind. Finally, in the case of non periodic boundaries condition (e.g. planet simulations), the particles can leave the box and therefore are removed from the simulation.

The three different cases (creation, suppression and transformation) can be dealt with an additional mask for the particles and an additional writing of the particles before and after each event. In the data, we store a single int that includes both a flag for the type of event (creation, suppression, transformation) and the new type of the particle. For the index files, we also keep in memory the suppressed and created particle events (offset in the logfile and id of the particle) and write them in the next index file.

3.4. MPI strategy

In SWIFT, the volume is split in smaller volume that are distributed to the different ranks according to some evaluation of the work required for each subvolume. Through a simulation, the volumes tend to stay attributed to the same rank in order to avoid too much communications, therefore a particle leaving (entering) a given volume can be considered as being removed from (created in) a given rank without having a large impact on the logfile size. The only difference with the suppression and deletion presented in the previous section is that we use a different flag and store the id of the other rank. We also ensure that the offsets are correctly written between the two files.

4. Reading strategies

The reader is implemented in C++ along with a python wrapper and a documentation is provided in the repository. As for the writer, the core implementation is fully independent of the physics. Only the reading and interpolation functions need to be updated when adding new fields.

Two important possibilities offered by the reader is to follow a particle (or a set of particles) through time in an accurate and efficient way and to generate snapshots from the CSDS. While the generation of snapshots is not the best usage of the CSDS, it can greatly improve the compatibility of the CSDS with existing codes and provides a speedup when the user needs to do a deep analyze of a single specific time (e.g. z=0).

In order to speedup the reading process, the reader does not read directly the logfile (except when reversing the offset of the records during the first reading), it always starts by finding the particles in the index files and then read the logfile starting from the last offset of the particles.

When reading a subset of the particles, it is important to have an efficient way of finding the offset of the particles. It can be done by sorting the index files according to the IDs of the particles and then using a bissection search. As the initial order of the particles does not represent anything meaningful, the index files can be saved once sorted.

In the case of simulations done with multiple MPI ranks, the reader will read independently each logfile and the corresponding index files. It means that the user needs to ensure that all the logfiles are read and, then, concatenates the output together if needed.

---

[7] In cosmological simulations, we use up to 7 different types of particles representing the gas, dark matter, stars, black holes (2 types), sink particles and neutrinos.

[8] Depending on the future usage of the CSDS, we might need to change the masks from a field point of view to a particle type point of view (e.g. one mask per particle type and not per field).

4.1. Implementation

In this section, the different steps done by the reader are summarized in order to give an overview of the method. Not all the details can be discussed, but the most important ones are present.

The reader starts by reading the header of the logfile and checks if the offsets are already in the correct direction. If it is not yet the case, the operation is done immediately. This is done by starting with the first record in the logfile. If a previous record exists, its offset is modified in order to point to the current record. Once the record is done, the reader moves to the end of the current record and starts over with the next record. This requires a single linear reading of the file with random accesses to some previous part of the file. It is worth to mention that if this process is interrupted, the file will become corrupted and it will be complicated to restore it.

Once the offsets reversed, the reader reads all the time records and populate a structure with them in order to have a quick access to the time of the particle records. As this operation can be long depending on the file size, the array is saved at the end of the first index file and can be restored in future readings. The reader finalizes its initialization by reading all the index files header and storing their time.

When requesting some data, the reader starts by setting the current time. This operation is done by selecting the two correct index files: the first one gives the information about the last known state of the simulation and the second one provides the information about the particles created and removed. Then the number of available particles is computed from the index files for the allocation of the output arrays. The particles are read the one after the other assuming that they are still present at the required time. If it is not the case, the code simply skips the current particle and goes to the next one. As this approach could raise some issues if the index files or the logfile are corrupted, we do a sanity check at the end in order to verify that we have the correct number of particles and crash if it is not the case. It would be possible to predict which particles are still present at a given time thanks to the index files, but this would require too much work for the particles that are still present. As we do not expect a lot of particles to be removed from the simulation, we decided to use the first approach rather than the prediction.

The particles are read field by field in order to be more flexible when each field is written at a different frequency. As the file is memory mapped, we always read the same file pages and they should stay within the RAM memory during the whole reading of a single particle. Thus even if we read more often the file and do a bit more operations, the overall efficiency of the code stays the same.

Once all the previous operations done, the output arrays are given back to the user. As we are reading the particles in the same order than the index files, the output arrays are sorted in the same way (e.g. by particle type and then ids).

5. Results

While the snapshots and the CSDS are writing the same information, the format is far too different (single time vs whole simulation) to have a fair and complete comparison. To make it even worse, they will not behave in the same way depending on the type of simulations (e.g. only hydrodynamics vs with gravity) due to the hierarchy of time steps. The gravity produces strong differences in timescales and thus will improve the CSDS performances when compared to purely hydrodynamics simulations that can be easily described by a single time step. As the CSDS was developed specifically for simulations including gravity, the best approach to have a rough idea of the efficiency is to look at such simulations. In this section, we will first illustrate the accuracy of the Hermite interpolation with the orbit of a planet. Then, we will move to the CSDS scaling both in term of reading time and memory with the Millenium simulation (Baugh et al. 2019). Finally, we will compare the CSDS and the snapshots in term of output size and accuracy for an isolated disk galaxy.

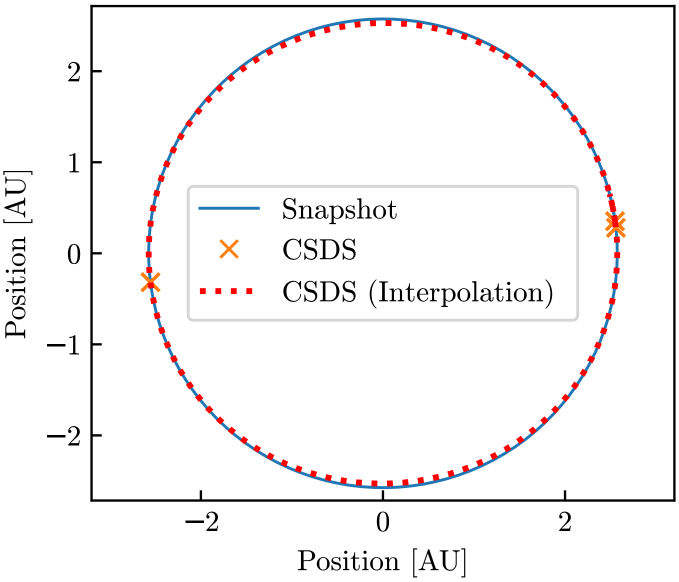


Fig. 3: Orbit of a planet around a star reconstructed with the CSDS (red line and orange crosses) and the snapshots (blue line). The orange crosses correspond to the place where the CSDS writes a record. The quintic Hermite interpolation (using the velocity and acceleration) allows to accurately reconstruct the orbit with a small number of points.

Let us start with the quintic Hermite interpolation (see Appendix A). Due to its high convergence order, it does not require many points to accurately represent the orbit of a particle. In order to show it with the CSDS, Figure 3 shows a simulation of a planet in orbit around a star. The interpolation is done with the help of the velocity and acceleration to constrain the two first derivatives in the interpolation. While the CSDS is set at low time resolution, the snapshots were written at high resolution for the comparison. During a single orbit, only three records (in orange) were written. As expected, the positions of the orange crosses perfectly match the positions of the snapshots (in blue). The interpolation given in red slightly differs with the snapshots far away from the records, but provides overall a correct interpolation. For cosmological simulations, the situation is more complicated than in this example, but we can already see here that a low number of points are sufficient to properly reconstruct the orbits.

Let us now focus on the performances of the CSDS. Figure 4 shows the scaling in term of output size and reading speed as function of the time resolution in the CSDS (Δn) and the number of particles for the Millenium simulation (dark matter only simulation within a box of 800 Mpc). The number of particles within the first graph is 384^3 and the CSDS resolution in the second graph is $\Delta n = 100$ steps. As the reading speed is dependent on the index files, two different lines are shown. The best case scenario (in red) corresponds to the case where the requested time matches an index file ($z \approx 0.12$). The worst case scenario (in orange) corresponds to the case where the requested time is just below an index file ($z \approx 0$). It means that the index file used is the same as in the best case scenario, but the particles need to be updated over 1.6 Gyr. As one could expect, the output size and reading speed depend linearly on the number of particles. The output size also roughly scales linearly as function of the time resolution. Let us now focus on the reading time as function of Δn. The best case scenario consists in reading an index file and then directly reading a value from the logfile. It means that the time resolution will have no impact on it and thus that the reading time is independent from Δn. The worst case scenario is more complex as it depends on when the index files are written and the hierarchy of time steps. In cosmological simulations, most of the particles will have large time steps. Therefore, the average number of “jumps” required to update the particles will be relatively low (between 1.0 and 7.2 for our different time resolutions). Therefore in our case, the worst case scenario is also almost independent on the time resolution.

The reading time is larger than for the snapshots (13.4s for the simulation with 768^3 particles). While this can be an issue, it is worth to explicitly mention that the main advantages of the CSDS are the accurate reconstruction at any time and the possibility to evolve the particles quickly once obtained (see the difference between the best and worst cases). Anyway, the CSDS can still write snapshots when good performances are required. The project is also still at its early steps and will certainly benefit from optimizations in the next years.

Now we will focus on the isolated disk galaxy in order to study the accuracy and output size. This simulation contains 360’000 dark matter particles within an Hernquist potential [9]. The dark matter particles resolution is 10^6 $M_\odot$. Using a snapshot as reference, Figure 5 shows the accuracy of the CSDS as function of Δn and the distance to the galaxy center. The accuracy is measured by the relative error on the radius. As one can expect, the CSDS converges towards the correct solution when increasing the output frequency (meaning decreasing Δn). Writing every 10 steps is sufficient to produce radii with an accuracy of 0.1%.

Using the same simulation, Figure 6 compares the accuracy and output size of both the snapshots and the CSDS. The relation for the snapshots is obtain through an interpolation of the reference snapshot from two snapshots written at an equal distance in time from the reference. Thus, the accuracy of the snapshot represents the worst case scenario as the interpolation is done at equal distance from the two snapshots. With the CSDS, there is no global worst case scenario, only individual ones that are represented with the upper limit of the distribution. Here, both the snapshots and the CSDS uses a cubic Hermite interpolation based on the positions and velocities. The CSDS decreases the output size by a factor of 3 for a given accuracy or two orders of magnitude in accuracy for a given output size when compared to the snapshots.

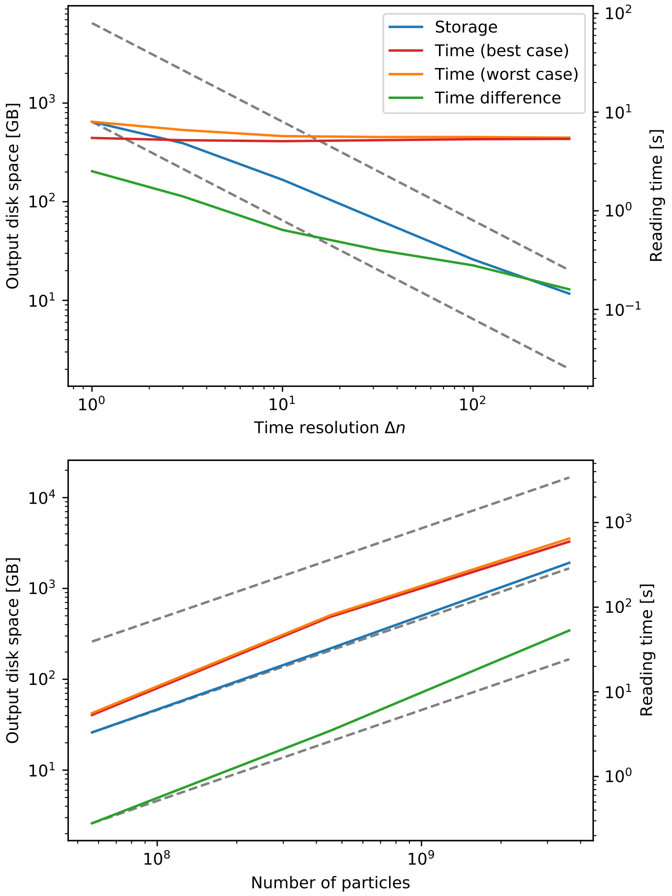


Fig. 4: Scaling of the CSDS both in term of output size and reading time. For the first graph, the number of particles is 384^3 and, for the second graph, $\Delta n = 100$. The first graph shows them as function of the time resolution of the output (number of steps between records Δn). The second graph shows them as function of the simulation size. The dashed black line corresponds to a linear scaling with a factor of 10 between them. The best case scenario is a requested time that corresponds to an index file. The worst case scenario is a requested time that is just before an index file. In green, the difference between the best case and the worst case is shown.

6. Examples of Application

The CSDS allows a higher time resolution than the snapshots and is therefore particularly useful for the production of movies as the user is not limited by the available snapshots. Thus the movie’s speed can be easily adapted in a specific part of a movie. Three movies have been produced for this paper and are hosted on YouTube (described in detail on YouTube): a cosmological simulation[10] (`https://www.youtube.com/watch?v=OKKsk0TigNo`), the chemical evolution of the same simulation (`https://www.youtube.com/watch?v=5AqmAUGndps`), and a planetary impact (`https://www.youtube.com/watch?v=6aBry0CUgVw`). While the first and last movies show the high time resolution of the output system for a cosmological simulation and a planetary impact, the second one shows that it can be used for some analysis.

A problem in the comparison of large scale simulations with observations comes from general relativity. As in observations, the further the observed object is, the older we see it, we need to produce the same behavior in our simulations. This is typically done with the method called light cone image (Evrard et al.

---

[9] The initial conditions and parameters are fully provided within SWIFT’s repository and corresponds to IsolatedGalaxy_dmparticles.

[10] The original movie without YouTube’s compression is available in the movie’s description.

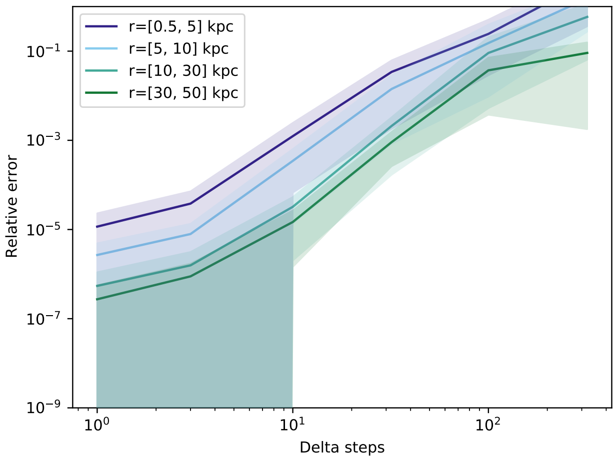


Fig. 5: Relative error of the radial positions of the particles as function of the number of steps between two writings for an isolated disk galaxy ran only with gravity. The interpolation is done through the quintic Hermite interpolation. The means of the distributions are shown with the solid lines and their 16/84 percentiles are indicated with the areas. As expected, the CSDS converges towards the solution provided by a snapshot.

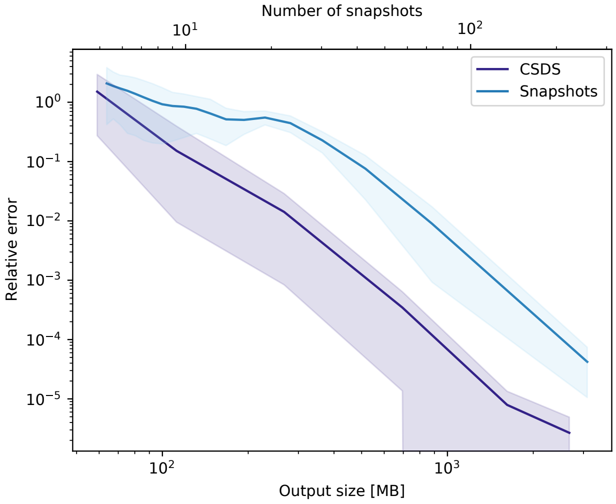


Fig. 6: Relative error of the radial position of the particles as function of the output size for both the snapshots and the CSDS. Both relations are computed from the same simulation that consists in an isolated disk galaxy simulated only with gravity. Only particles with $r \in [5, 10]$ kpc are taken into account in this figure. The average error is shown as the straight lines and the distribution is shown with the areas representing the 16/84 percentiles. For both the CSDS and the snapshots, a cubic interpolation is done.

2002; Garaldi et al. 2020) and the main idea behind it is to extract slices of volume in each snapshot and to stack them together before projecting along time. Due to the snapshots, this can be only done approximately or with an extremely large number of snapshots. In Evrard et al. (2002), they solved this issue by implementing the light cone computation directly into the simulation code and redesigned it in order to reduce the resolution in the area outside the light cone. Thus a single light cone is selected at the beginning of the simulation and cannot be changed. With the CSDS, it is possible to generate them after the end of the simulation and for as many light cones as wanted thanks to the high time resolution in the CSDS along with the interpolation. For example in Figure 7, a light cone image is produced but with the axis switched (we project both the time and position along the x axis) in order to show the time evolution.

In Figure 8, the phase diagram is shown for the cosmological simulation used for the movie. It consists in an histogram of the density and internal energy of the gas at redshift 6 and the time evolution of a reference gas particle between $z \in [23, 4]$ (yellow to purple). The gas inside the galaxies is in a dense phase (above 10^{-2} atom / cm^3) and can be split into the cold (below 10^{10} erg / g) and hot phase that has been recently touched by a supernovae. Inside a galaxy, the behavior of the gas is extremely chaotic due to the explosion of supernovae. In our simulations, supernovae are modeled by directly injecting some energy into the surrounding gas particles and provoke a quick vertical displacement in the phase diagram. While the timescale of the impact of a single supernovae is typically less than a few Myr, the cosmological simulations are usually done over 14 Gyr. The two different timescales make it very hard to accurately follow the impact of supernovae with the traditional snapshots. The CSDS is perfectly able to resolve both timescales at the same time and can help to enhance our understanding of supernovae in cosmological simulations.

7. Conclusions

We presented a new output technique that can follow the evolution of all the particles (or cells) and write them at their own timescale in order to reduce the output size and increase the accuracy at any time. This approach is particularly well fitted for simulations that includes gravity as it produces large differences in time scales. On the example presented in Figure 6, we increased the accuracy of the output by two order of magnitudes for a given storage space (or a factor of 3 in storage space for a given accuracy). This technique largely improves the traditional snapshots and will allow to improve the analysis of fast events (e.g. supernovae) within the simulations.

In the future, the technique could be expanded through the inclusion of spatial information within the index files that allows to quickly select a subvolume. It could be also possible to adapt the writing criterion according to some physical properties (e.g. extremely high resolution within a single galaxy). A last interesting improvement in term of physics could be to increase the interpolation order by the usage of more than 2 records. Lastly, some optimizations are already planned and will certainly improve the reading speed of the CSDS.

Acknowledgements. We gratefully acknowledge the help of Bert Vandenbroucke and Alexei Borissov to test the CSDS, as well as Jacob Kegerreis for his help running the planetary simulation test-case.
We also thank Alyson Brooks for reviewing this publication.
MS is supported by the Netherlands Organisation for Scientific Research (NWO) through VENI grant 639.041.749.
This work was supported by the Swiss Federal Institute of Technology in Lausanne (EPFL) through the use of the facilities of its Scientific IT and Application Support Center (SCITAS) and the University of Geneva through the usage of Yggdrasil.
The research in this paper made use of the SWIFT open-source simulation code (`http://www.swiftsim.com`, Schaller et al. (2018)) version 0.9.0.
This work used the DiRAC@Durham facility managed by the Institute

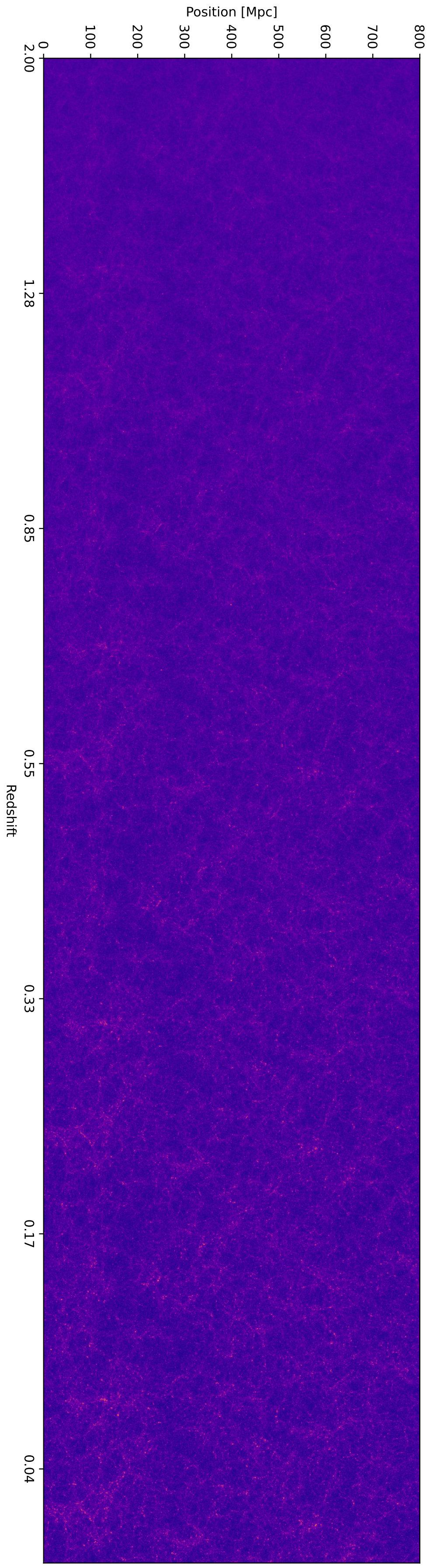


Fig. 7: Light cone image with the time axis along x. The image consists in a single volume where the time of each particle is selected in order to emit lights such as it reaches the right side at z=0. With the snapshots, it is required to read the particles by slices and thus produce discontinuities in the image. Thanks to the CSDS, this is no longer the case as we can find with high accuracy the time when a particle is crossing the light cone.

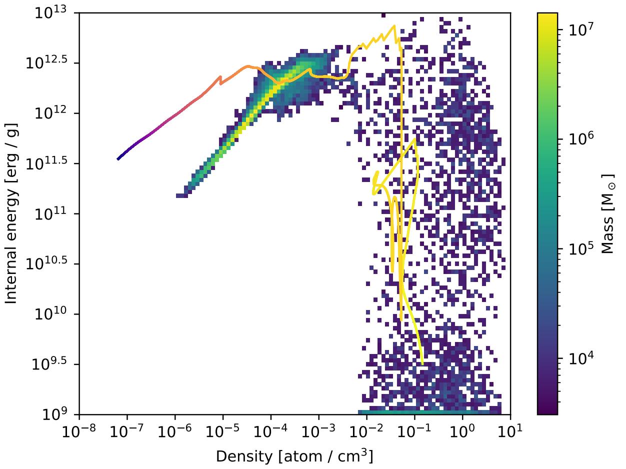


Fig. 8: Phase diagram of the density and internal energy in physical units for a simulation of the dwarf galaxy 159 with GEAR's model (Revaz & Jablonka 2018) in SWIFT at redshift 6 for the background. The line shows the evolution of a particles between redshift 23 (yellow) to 4 (purple). Thanks to the CSDS, we can accurately follow the evolution of the particle in the chaotic and dense medium and clearly see when a supernovae directly touch the particle (straight vertical line). Due to the evolution between redshift 6 and 4, the final position of the particle followed does not match the rest of the low density particles.

for Computational Cosmology on behalf of the STFC DiRAC HPC Facility (www.dirac.ac.uk). The equipment was funded by BEIS capital funding via STFC capital grants ST/K00042X/1, ST/P002293/1, ST/R002371/1 and ST/S002502/1, Durham University and STFC operations grant ST/R000832/1. DiRAC is part of the National e-Infrastructure.
We are grateful to the Numpy (Oliphant 2015), Matplotlib (Caswell et al. 2018) SciPy (Jones et al. 2001) and IPython (Perez & Granger 2007) teams for providing the scientific community with essential python tools.

References

Aarseth, S. J. 1963, Monthly Notices of the Royal Astronomical Society, 126, 223
Abbasi, H., Lofstead, J., Zheng, F., et al. 2009, in 2009 IEEE International Conference on Cluster Computing and Workshops, 1–10, iSSN: 2168-9253
Adams, M. F., Ku, S.-H., Worley, P., et al. 2009, Journal of Physics: Conference Series, 180, 012036
Baugh, C. M., Gonzalez-Perez, V., Lagos, C. d. P., et al. 2019, Monthly Notices of the Royal Astronomical Society, 483, 4922
Bertschinger, E. 1998, Annual Review of Astronomy and Astrophysics, 36, 599
Caswell, T. A., Droettboom, M., Hunter, J., et al. 2018, matplotlib/matplotlib v3.0.2
Evrard, A. E., MacFarland, T. J., Couchman, H. M. P., et al. 2002, The Astrophysical Journal, 573, 7
Garaldi, E., Nori, M., & Baldi, M. 2020, Monthly Notices of the Royal Astronomical Society, 499, 2685
Godoy, W. F., Podhorszki, N., Wang, R., et al. 2020, SoftwareX, 12, 100561

Gonnet, P., Chalk, A. B. G., & Schaller, M. 2016, arXiv:1601.05384 [cs], arXiv: 1601.05384

Jetley, P., Gioachin, F., Mendes, C., Kale, L. V., & Quinn, T. 2008 (IEEE Computer Society), 1–12

Jones, E., Oliphant, T., Peterson, P., & others. 2001, SciPy: Open source scientific tools for Python

Kegerreis, J. A., Eke, V. R., Gonnet, P., et al. 2019, Monthly Notices of the Royal Astronomical Society, 487, 5029

Lind, M. 2020, PeerJ Computer Science, 6, e304, publisher: PeerJ Inc.

Lüttgau, J., Snyder, S., Carns, P., et al. 2018, in 2018 IEEE/ACM 3rd International Workshop on Parallel Data Storage Data Intensive Scalable Computing Systems (PDSW-DISCS), 64–75

Ma, X., Lee, J., & Winslett, M. 2006, IEEE Transactions on Parallel and Distributed Systems, 17, 193, conference Name: IEEE Transactions on Parallel and Distributed Systems

Mitra, S., Sinha, R. R., Winslett, M., & Jiao, X. 2005, in 2005 IEEE International Conference on Cluster Computing, 1–10, iSSN: 2168-9253

Müller, A., Deconinck, W., Kühnlein, C., et al. 2019, Geoscientific Model Development, 12, 4425

Nelson, D., Pillepich, A., Genel, S., et al. 2015, Astronomy and Computing, 13, 12

Norman, M. L., Bryan, G. L., Harkness, R., et al. 2007, arXiv:0705.1556 [astro-ph], arXiv: 0705.1556

Oliphant, T. E. 2015, Guide to NumPy, 2nd edn. (USA: CreateSpace Independent Publishing Platform)

Peebles, P. J. E. 1993, Principles of Physical Cosmology by P.J.E. Peebles. Princeton University Press, 1993. ISBN: 978-0-691-01933-8

Perez, F. & Granger, B. E. 2007, Computing in Science Engineering, 9, 21

Potter, D., Stadel, J., & Teyssier, R. 2017, Computational Astrophysics and Cosmology, 4, 2

Revaz, Y. & Jablonka, P. 2018, arXiv:1801.06222 [astro-ph], arXiv: 1801.06222

Ross, R. B., Peterka, T., Shen, H.-W., et al. 2008, Journal of Physics: Conference Series, 125, 012099

Schaller, M., Gonnet, P., Chalk, A. B. G., & Draper, P. W. 2016, arXiv:1606.02738 [astro-ph], 1, arXiv: 1606.02738

Schaller, M., Gonnet, P., Draper, P. W., et al. 2018, Astrophysics Source Code Library, ascl:1805.020

Schaye, J., Crain, R. A., Bower, R. G., et al. 2015, Monthly Notices of the Royal Astronomical Society, 446, 521

Springel, V. 2005, Monthly Notices of the Royal Astronomical Society, 364, 1105

Xiao, Q., Shang, P., & Wang, J. 2012, in 2012 IEEE Seventh International Conference on Networking, Architecture, and Storage, 199–206

Zheng, F., Zou, H., Eisenhauer, G., et al. 2013, in 2013 IEEE 27th International Symposium on Parallel and Distributed Processing, 320–331, iSSN: 1530-2075

Appendix A: Interpolation with Comoving Coordinates

The Hermite interpolation between time t_0 and t_1 at t is given by:

$$p_n(u) = \sum_{i=0}^{(n-1)/2} p_0^{(i)} H_i^n(u) + p_1^{(i)} H_{n-i}^n(u) \tag{A.1}$$

where n is the degree of the interpolation, $p_0^{(i)}$ ($p_1^{(i)}$) is the i-th derivative at t_0 (t_1) and u is the normalized time variable ($(t - t_0)/(t_1 - t_0)$). The polynomials H_i^n are given by the line of the following matrix where the first column corresponds to the constant term (Lind 2020):

$$H^3 = \begin{pmatrix} 1 & 0 & -3 & 2 \\ 0 & 1 & -2 & 1 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 3 & -2 \end{pmatrix}. \tag{A.2}$$

For example $H_0^3(u)$ is given by $1 - 3u^2 + 2u^3$. For the quintic interpolation, the matrix is given by:

$$H^5 = \begin{pmatrix} 1 & 0 & 0 & -10 & 15 & -6 \\ 0 & 1 & 0 & -6 & 8 & -3 \\ 0 & 0 & \frac{1}{2} & -\frac{3}{2} & \frac{3}{2} & -\frac{1}{2} \\ 0 & 0 & 0 & \frac{1}{2} & -1 & \frac{1}{2} \\ 0 & 0 & 0 & -4 & 7 & -3 \\ 0 & 0 & 0 & 10 & -15 & 6 \end{pmatrix} \tag{A.3}$$

In cosmological simulation, comoving coordinates are required to take into account the expansion of the universe (for more information on comoving coordinates see for example Bertschinger (1998); Peebles (1993)). In SWIFT, they are defined as $x = ax'$, $a^2\dot{x}' = v'$ and $a\dot{v}' = g'$[11] where x', v' and g' are the comoving coordinates, velocities and accelerations, and a is the scaling factor that depends on time[12] . To use the Hermite interpolation previously described, u is replaced by $(a - a_0)/(a_1 - a_0)$ and the terms $p_0^{(i)}$ and $p_1^{(i)}$ need to include some cosmological factors. Those last terms are obtained from the derivation of the comoving coordinate with respect to the scale factor. For the positions, the velocities $p_0^{(1)}$ and $p_1^{(1)}$ need to be modified with the following expression evaluated at either a_0 or a_1:

$$\frac{\mathrm{d}x'}{\mathrm{d}a} = \frac{\mathrm{d}x'}{\mathrm{d}t}\frac{\mathrm{d}t}{\mathrm{d}a} = \frac{v'}{a^2\dot{a}}. \tag{A.4}$$

The derivative of a with respect to time is given by:

$$\dot{a} = aH_0\sqrt{(\Omega_c + \Omega_b)a^{-3} + \Omega_{\mathrm{rad}}a^{-4} + \Omega_\Lambda} \tag{A.5}$$

where Ω are the cosmological parameters and H_0 the Hubble constant. This equation assumes a flat Λ-CDM universe. For the accelerations within the position interpolation ($p_0^{(2)}$ and $p_1^{(2)}$), an additional derivative is required:

$$\frac{\mathrm{d}^2x'}{\mathrm{d}a^2} = \frac{1}{a^4\dot{a}^2}\left(ag' - 2a\dot{a}v' - \frac{a^2\ddot{a}}{\dot{a}}v'\right). \tag{A.6}$$

In the case of the interpolation of velocities, the same computation is required for the accelerations ($p_0^{(1)}$ and $p_1^{(1)}$) and gives:

$$\frac{\mathrm{d}v'}{\mathrm{d}a} = \frac{g'}{\dot{a}a} \tag{A.7}$$

Appendix B: Example of python scripts

The following code shows a quick example of the CSDS's interface. The full interface is described in the documentation provided within the reader's repository.

---

[11] The last one holds only for gravity as hydrodynamics have a different relation.

[12] Size of the universe normalized to today's value.

Listing 1: Example of the CSDS's interface.

```
import numpy as np
import libcsds as csds

# Basename of the logfile
BASENAME = "index_0000"
# The ids of the particles that we wish to follow
PARTICLE_IDS = np.array([10, 412, 213], dtype=int)
# The simulation time
TIME = 0.5


# Open the CSDS
with csds.Reader(BASENAME, verbose=0) as reader:
    # Get the time limits within the logfile
    t_min, t_max = reader.get_time_limits()

    # Ensure that the time is correct
    if TIME < t_min or TIME > t_max:
        raise Exception("The requested time is unavailable.")

    # Ensure that the fields are available for the particle type
    fields = reader.get_list_fields(part_type=csds.gas)

    if "Coordinates" not in fields or "Entropies" not in fields:
        raise Exception("Field not found in the logfile")

    # Read a subset of particles (coordinates and entropies are
    # 2 numpy arrays)
    coordinates, entropies = reader.get_data(
        fields=["Coordinates", "Entropies"], time=TIME,
        filter_by_ids=PARTICLE_IDS)

    # Read all the gas particles
    all_fields = reader.get_data(
        fields=fields, time=TIME, part_type=csds.gas)
```

8 Implementation of Sink Particles for the Star Formation

> “Go back?" he thought. "No good at all! Go sideways? Impossible! Go forward? Only thing to do! On we go!" So up he got, and trotted along with his little sword held in front of him and one hand feeling the wall, and his heart all of a patter and a pitter.”
>
> — J. R. R. Tolkien, The Hobbit

In this chapter, I am presenting my work on sink particles within `SWIFT`. Due to the structure of `SWIFT`, implementing this new set of tasks require a large amount of work, but then is almost automatically optimized to reduce the dead time. Therefore, I have prioritized to implement and test them over doing the detailed physics as this part could be easily done by someone without any experience with `SWIFT`. It means that the physical models provided later are not yet implemented but are the building blocks for any future advancement in the topic. I will introduce this chapter with a quick summary of the way sink particles will be used along with a description of why they are needed. Then, I will provide a general description of the sink particles as we will use them. Next, I will describe the different processes in more details. I will conclude with a description of the required changes to the star particles.

The sink particles will be created from gas particles in a similar way than stars in our previous model. Their initial mass is simply given by the mass of the gas particle. During their evolution, sink particles can merge together if they are close together in comparison to the resolution. They will also accrete the surrounding gas in order to grow their mass until reaching a target value corresponding to the mass of a star to produce. Every time this target is reached, a star is released, the mass of the sink particle is adapted and a new target value is set according to the initial mass function.

The star formation (SF) consists in collapsing a cloud of gas (e.g. molecular cloud) until the first nuclear reactions start within the newly formed stars, heating it from the inside and preventing further collapse. As mentioned in section 4.5, the resolution needed for SF is still out of reach for galaxy cosmological simulations and will still be for a long time. Even if it is a small scale process, the SF still has a tremendous impact on larger scales through the different form of feedback from the stars and is modelled by subgrid models. The model usually consists in the transformation of a gas particle into a stellar particle of the same mass that represents a single population of stars (see section 4.5 for more details; Katz, 1992; Katz et al., 1996).

The motivation of developing a new model can be summarized by the aim of obtaining accurate abundances within the stars. While the description of supernovae with mass dependent yields within `GEAR` has been shown to correctly reproduce the observed abundances (Revaz and Jablonka, 2018), the model is too rough for ultra-faint dwarf galaxies. Indeed, in such small galaxies, supernovae need to be fully resolved individually and not as being part of a common stellar population (with individual supernovae) as done previously. This issue can be split in three different but related ones:

1. Our resolution is too high for the stellar population hypothesis used within the star formation and stellar feedback (at least $1000 M_\odot$ according to Revaz et al., 2016 with the initial mass function (IMF) presented in section 4.6.1). This hypothesis consists in representing a population of stars within a single stellar particle.
2. As the mass of the gas particles depends on the resolution, they do not necessarily match the mass of individual stars and the approach for SF presented in Katz, 1992 cannot be used anymore in high resolution simulations.
3. The sampling of the IMF without any bias is crucial for the chemical evolution and for comparison with observations (described further in section 8.1.5). Often this sampling is made with a fixed amount of mass (e.g. the mass contained within the stellar particle) and do not necessary respect it when exploding supernovae. With low fixed masses, the IMF sampling will strongly fluctuate for high mass stars and largely miss the targeted mass. Such fluctuations will strongly bias the yields provided by the population of stars. For example, using the sampling called optimal discrete IMF sampling (OIMFS) in Revaz et al., 2016, an implicit upper limit on the mass of the most massive stars is present, or with the random discrete IMF sampling (RIMFS) a noise far larger than expected is seen.

Currently, some simulations are still being run with models comparable to Katz, 1992 at high resolution (e.g. $30 M_\odot$ in Wheeler et al., 2019, $425 M_\odot$ in Applebaum et al., 2020, $18.25 M_\odot$ in Hirai et al., 2021 or $63 M_\odot$ in Jeon et al., 2021) with more or less considerations on theses issues. While in Wheeler et al., 2019, they are not mentioning theses issues, continue to form stars the usual way and must clearly explode more stars than possible from the particles' masses, the three lasts are trying to fix them. Only Applebaum et al., 2020 try to ensure that the mass of the supernovae exploded corresponds to the stellar particle mass. Their method has an upper limit of $50 M_\odot$ (half the maximal mass of the IMF) on the total mass error given by the worst case scenario

(12% of the particle's mass at the resolution previously given). As they have a resolution close to the limit derived in Revaz et al., 2016 and ensure to have a relatively low error on the overall mass of supernovae, they should not suffer a strong bias. In Hirai et al., 2021 and Jeon et al., 2021, they group gas particles together in order to form stars more massive than their resolution without discussing the crucial issue of the IMF sampling. The gas particles are transformed into stars and their mass is assign from the IMF (Pop II stars, in Jeon et al., 2021, represent a stellar population with a mass of $500M_{\odot}$). If the mass of the gas particle is not large enough, the particle directly accretes gas from the surrounding particles in order to match the targeted mass.

As Jeon et al., 2021; Hirai et al., 2021 have done it, the most obvious solution to the first issue is to implement an individual description of the stars instead of a stellar population. This results in a simplification of the code for stellar feedback as some integrals are not needed anymore. It also moves the IMF sampling from the feedback to the star formation, thus ensuring that the amount of supernovae correspond to the total mass of the stars. Finding a solution to the second issue while guaranteeing a correct IMF sampling is the most complicated part of the new star formation. A possibility could be to simply use part of the particle's mass to generate an individual star (similarly to our method for spawning multiple stellar particles from a single particle), but this will quickly break for simulations with higher mass resolution than the most massive stars. With the Pop III stars, this assumption will even break sooner as they can reach extremely high masses depending on the models (more than $100M_{\odot}$; Haemmerlé et al., 2020). Therefore, it is better to have a method that already works with the assumption that the resolution is better than the largest possible mass of a star. Any model solving the three different issues needs to be able to recover the main observational constraint that is the Kennicutt-Schmidt law. As presented in 4.5, this law links the local star formation rate to the local density as given in equation 4.72.

A first approach studied by our group was to use a Friend of Friend (FoF) algorithm in order to identify the star forming clouds and then convert the full cloud at once into individual stars with a total mass corresponding to the cloud's mass. This approach ensure to have enough mass to form any kind of stars. Unfortunately, this static approach produces strong perturbations to the galaxy as a large volume of gas is removed instantaneously. It tends also to form many stars in a single time step, thus producing large correlated feedback events. Such events produce large unnatural outflows that tends to destroy any future star formation in large dwarf galaxies. As the time step (~ 0.1 Myr) within low resolution simulations ($1000M_{\odot}$) is already comparable to the formation of stars (e.g. 0.01 to 10^3 Myr to reach the zero age main sequence (Maeder, 2009)), a more dynamical approach should be preferred.

To fix this issue, I proposed to move towards a dynamical approach by creating sink particles. Sink particles were first introduced in the 90s in order to reduce the impact of the densest clouds on the computation time (Bate et al., 1995). In recent years, they have been used mostly in the domain of molecular clouds simulations for both AMR and SPH (Bleuler and Teyssier, 2014; Price et al., 2018) but they are slowly making their way back into cosmological simulations for the star formation such as presented in this thesis (e.g. Grudić et al., 2021). The sink particles initially represented unresolved areas contained within the simulations and can still be used in this way for the star formation as it requires a subgrid model. On the physical side, they can be seen as molecular clouds forming stars due to their own gravitational collapse. The star formation starts the collapse at large scale (e.g. molecular clouds with a maximum size of $\sim$ 200 pc) and until reaching a radius comparable to the sun ($\sim 10^{-8}$ pc). In current high resolution simulations of dwarf galaxies, the resolution limit is found slightly below the size of molecular clouds. As more than one resolution element represent a single molecular cloud, the sink particles start with the collapse of a single particle. Then they accrete the surrounding gas in order to reach the required size. When enough gas has collapsed, sink particles can spawn stars in a similar way than if we were able to resolve the cloud. This formation is modeled through a single sampling of the IMF at a time (without any assumption on the total mass) that sets the mass target for the next star and then spawn it as soon as enough mass has been accreted. As the mass selection is done without any assumption on the total mass produced, this should not have any bias as the ones presented in Revaz et al., 2016. The sink particles can also growth in mass through sink-sink mergers that should rarely occur. To come back to the three issues mentioned earlier, the sink particles can easily spawn individual stars, they are able to group gas particles together through the accretion in order to form stars and finally the sampling cannot have any direct bias as in the traditional approach.

8.1 Description of our Future Model

Let us start with a quick summary of the processes that describe our sink particles. Initially, in our cosmological simulations, the baryonic matter is only composed of gas particles and no sink particle exists yet. Around $z \sim 20$, the first stars need to be born (Frebel and Bromm, 2012) and thus their progenitors: the sink particles. Similarly to our previous star formation technique, some gas particles are directly transformed into sink particles that will evolve through sink particles mergers and accretion of gas particles. Once the sink particle is massive enough, it spawns a single star particle at a time that will evolve alone and might end its life as a supernova.

During the formation of the sink particles, we transform a gas particle into a sink and thus are removing some gas particles and introducing a perturbation to the local hydrodynamic equilibrium. Hummel et al., 2016 propose to include the sink particles in the computation of the hydrodynamics by giving a single density and temperature to all the sink particles. While it helps to regularize the flow around the sink particles, it can be problematic in the case of complex structures (Price et al., 2018). Therefore, in our implementation, we decided to not include any impact on the hydrodynamics except from the removal of gas particles through the sink formation and accretion.

As mentioned earlier, the sink particles represent unresolved areas and thus have an associated size r_{acc}. This size can be defined through three different ways. The first one is to simply follow the smoothing length approach and ensure that a given number of gas particle neighbors are within r_{acc}. As the sink particles accrete gas particles, it would mean that the smoothing lengths could increase simply because the sinks are removing particles. The last two possibilities are similar and consists in either using the smoothing length of the progenitor or using a global value shared by all sink particles (and could possibly evolve with time). The issue with the first one is that a sink particle formed within a low resolution area will not be able to decrease its size even if the resolution increases. For the second one, it means that a sink particle within a low resolution environment would almost never accrete a gas particle. To figure out the best approach, it will be necessary to do some tests with a model implemented. For now, I have implemented the last solution. Each sink particle has its own variable for r_{acc} even if they all share the same value. It means that the second method can be easily implemented. Moving towards the smoothing length definition will require more work, but should not be too complicated as it would consist in simply copying the ghost task from the stars.

As mentioned earlier, the implementation in `SWIFT` requires a set of new tasks for each of the previous processes. They are shown in green in the task dependency graph presented in the next page. In the next sections, I will describe in details the expected computations of each task and discuss them. As for the other types of particle, sink particles have a drift task presented in Algorithm 8.1. As for stars, their motion is fully described by `gparts` and simply needs to be synchronized. All the other tasks related to the sink particles represent a block of tasks that is almost independent of the rest of the tasks.

Algorithm 8.1 drift_sink

Depends on:

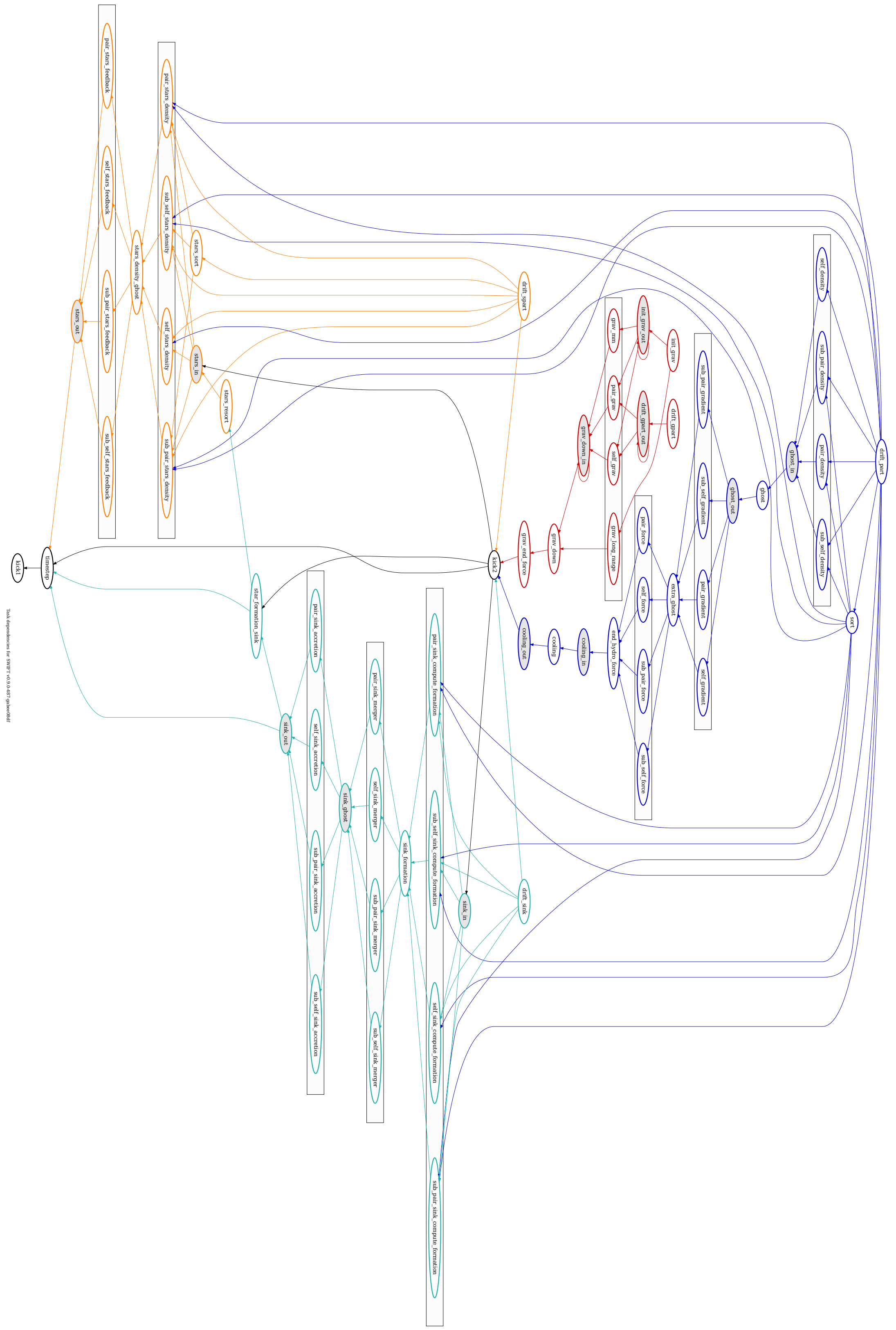
```
    $x \leftarrow x + v_{\text{full}} \text{dt}_{\text{drift}}$
    $v \leftarrow v + a_{\text{grav}} \text{dt}_{\text{kick, grav}}$
```

8.1.1 Sink Formation

As sink particles represent unresolved areas of radius r_{acc}, they should never overlap each other. Therefore, when forming a sink particle, we need to ensure that there are no sink particles around the sink forming gas particle and that we cannot form two sink particles at the same time close to each other. The second condition can be achieved by converting only particles that have the minimal gravitational potential among all their neighbors. As it is done for the formation of star particles, we can also ensure that the gas is collapsing, is above a critical density and below a critical temperature. Thus, the criteria for sink formation are given by:

1. No other sink particle within r_{acc}.

Figure 8.1 – Complete task dependencies for a simulation using the `SWIFT-GEAR` model with sink particles. The tasks corresponding to the sink particles are given in green. Except from the drift, the tasks can be considered as a single block dealing with the sink particles.


Task dependencies for SWIFT v0.9.0-487-gabee08df
kick1
timestep
kick2
pair_stars_feedback
self_stars_feedback
sub_pair_stars_feedback
sub_self_stars_feedback
stars_out
stars_density_ghost
pair_stars_density
sub_self_stars_density
self_stars_density
sub_pair_stars_density
stars_sort
stars_in
stars_resort
drift_spart
star_formation_sink
sink_out
pair_sink_accretion
self_sink_accretion
sub_pair_sink_accretion
sub_self_sink_accretion
sink_ghost
pair_sink_merger
self_sink_merger
sub_pair_sink_merger
sub_self_sink_merger
sink_formation
pair_sink_compute_formation
sub_self_sink_compute_formation
self_sink_compute_formation
sub_pair_sink_compute_formation
sink_in
drift_sink
grav_end_force
grav_down
grav_down_in
grav_mm
pair_grav
self_grav
grav_long_range
init_grav_out
drift_gpart_out
init_grav
drift_gpart
cooling_out
cooling
cooling_in
end_hydro_force
pair_force
self_force
sub_pair_force
sub_self_force
extra_ghost
sub_pair_gradient
sub_self_gradient
pair_gradient
self_gradient
ghost_out
ghost
ghost_in
self_density
sub_pair_density
pair_density
sub_self_density
sort
drift_part

2. Lowest gravitational potential ($\phi_{\text{grav,i}}$) among the neighbors,
3. Collapsing gas: $\nabla \cdot v < 0$,
4. Dense gas: $\rho > \rho_{\text{crit}}$.
5. Low temperature: $T < T_{\text{crit}}$

Here the critical density can be derived from the Jeans criterion for star formation in the previous chapters and thus can be dependent on the local resolution. Except from the first criterion, they can all be computed inside the hydrodynamics equations and thus do not need a new type of task. As mentioned earlier, only the tasks are implemented, the models are only provided to give a justification for the tasks.

In Price et al., 2018 and Hubber et al., 2013, they propose additional criterion based on the virial equation, the gravitational energy and the time step of the neighbors:

1. The neighbors are all active.
2. $\sum_{\text{ngb}} |e_{\text{thermal}}| < 0.5 \sum_{\text{ngb}} |e_{\text{grav}}|$
3. $\sum_{\text{ngb}} |e_{\text{thermal}}| + |e_{\text{rot}}| < \sum_{\text{ngb}} |e_{\text{grav}}|$
4. $\sum_{\text{ngb}} e_{\text{tot}} < 0$
5. Hill sphere criterion (reduces the risk of forming a sink particle in an area that should be sheared apart by an external gravitational field and is defined in Eq. 5 in Hubber et al., 2013).

While they are reducing the risk of producing unwanted sink particles as they ensure that the gas is really collapsing and nothing will prevent it, they are clearly linked to the previous equations (velocity divergence, density and minimum of the gravitational potential). For our first implementation, we will restrict ourselves to the first list as it is already partially taking into account the second list and it facilitates our implementation.

Finally, it is worth to mention Bleuler and Teyssier, 2014 where they use a clump finder in order to identify the star formation areas and then compute the global properties in order to decide if a sink particle should be formed inside the clump. This approach ensures that the number of sink particles is reasonable and could be interesting to study in a later implementation.

Once the formation criterion passed, the gas particle is directly transformed into a sink particle. In this case, we are not trying to follow a physical equation but only to remove the unresolved areas from the simulation and thus do not need to follow the stochastic approach done for the stars in the previous chapter. It means that the only way to regulate the formation of sink particles for a given resolution is through the feedback produced by the stars, the sink radius, and the critical density.

In Algorithm 8.2, 8.3 and 8.4, the tasks related to the sink formation are described. As the velocity dispersion and density are already computed in the hydro density, they are not presented again here.

Algorithm 8.2 in Hydro density

```
Depends on: drift_part
    can_form_i ← $\phi_{\text{grav,i}} < \phi_{\text{grav,j}}$
```

Algorithm 8.3 sink_compute_formation

```
Depends on: kick2
    can_form_i ← can_form_i ∧ not included within $r_{\text{acc}}$ of a sink.
```

Algorithm 8.4 sink_formation

```
Depends on: sink_compute_formation
    if $(\rho > \rho_{\text{crit}}) \wedge (\nabla \cdot v < 0) \wedge$ can_form then
        Convert the gas particle into a sink particle.
```

8.1.2 Sink Mergers

Even if we always create sink particles in the correct areas, gravity will produce mergers between structures (e.g. clouds, halos, galaxies, ...) and sink particles will overlap each other. Without mergers, such sink particles will compete in order to accrete gas particles and will strongly reduce their individual growing rates while representing the same unresolved area. Globally, this reduced rate will not have any impact as the same amount of mass will be transformed into stars, but the overall star formation rate will become far more noisy as the individual accretion will not be as smooth. Unfortunately, sink particles are representing extremely dense environment that cannot be resolved and therefore there is no way to know exactly if they are colliding at their scale. While Hubber et al., 2013; Price et al., 2018 are not doing any mergers, some other groups are using different merging strategies. The simplest approach is to merge any sink particles that are within the radius of each other such as done in Gong and Ostriker, 2013. More complex approaches exist such as using a clump finder or a FoF in order to identify the sink particles that belongs to the same cloud (Bleuler and Teyssier, 2014; Krumholz et al., 2004) or merging only particles that are collapsing against each others (Federrath et al., 2010). Some groups are also restricting the mergers to young sink particles (Bleuler and Teyssier, 2014; Krumholz et al., 2012).

If the sink formation is correctly done, the frequency of sink merger should stay low and therefore the details of the merging technique should not deeply impact the physics (Grudić et al., 2021). Thus, we will implement the simple and efficient approach of merging particles whenever they overlap each other such as presented in Algorithm 8.5.

Algorithm 8.5 sink_merger

```
Depends on: sink_formation
  if $r_{i,j} < \max(r_{\mathrm{acc},i}, r_{\mathrm{acc},j})$ then
    Merge the sink particles together
```

8.1.3 Sink Accretion

The accretion of gas into sink particles can be done in two different ways. The first one is specially designed for AMR codes and absorbs part of the mass contained in a cell / particle (Hubber et al., 2013). The second one is more designed for SPH and simply absorb a full particle whenever the accretion criteria is met (Price et al., 2018). While both approaches work and have been used in SPH (e.g. Hubber et al., 2013), the partial mass absorption can lead to large difference of masses between particles and produce unexpected behavior in the hydrodynamics. Therefore, the second approach should always be preferred in SPH.

As done in Price et al., 2018, we will have two different cases for the accretion. Below a given fraction f_{acc} (default to 0.8) of the accretion radius r_{acc}, a gas particle is directly absorbed and, above it, it depends if the gas particle is infalling onto the sink particle. In order to check this, we can simply verify the orbital parameters and if the particle is gravitationally bound to the sink particle:

1. The angular momentum around the sink particle is lower than the one of a Keplerian orbit at r_{acc}: $|\mathbf{r}_{ij} \wedge v_{ij}|^2 < r_{\mathrm{acc}}^2 \sqrt{GM_{\mathrm{sink}}/r_{ij}^3}$,

2. The particle is gravitationally bound to the sink particle without graviational softening ($e = 0.5v_{ij}^2 - GM_{\mathrm{sink}}/r_{ij} < 0$),

3. e is smaller than with any other sink particle.

In Algorithm 8.6, the corresponding task is shown.

Algorithm 8.6 sink_accretion

Depends on: sink_merger

```
accrete ← $\left(|\mathbf{r}_{ij} \wedge v_{ij}|^2 < r_\mathrm{acc}^2 \sqrt{G M_\mathrm{sink}/r_{ij}^3}\right) \wedge (e < 0) \wedge \left(e_{ij} == \min_\mathrm{k=sink}(e_{ik})\right)$
accrete ← $\left(r_{ij} < f_\mathrm{acc} r_\mathrm{acc}\right) \vee$ accrete
if accrete then
    Accrete the hydro particle $i$ into the sink particle $j$ and ensure the conservation of both mass (including metals) and momentum.
```

8.1.4 Sampling of the Initial Mass Function

Once the sink particle has accreted enough mass, it can spawn a star particle and eject it. The main issue in this process is to correctly sample the initial mass function (IMF). In Wall et al., 2019, they present a method using a Poisson sampling. It consists in splitting the IMF in mass bins and assigning a number of stars from a Poisson distribution based on the IMF. When a sink particle is created, they sample the IMF within each bin for a total mass of 10^4 $\mathrm{M}_\odot$. During the evolution of their sink particles, every time a star has been generated, they draw the next one from the list previously created . While they show that they are able to reproduce correctly the IMF thanks to their method, this approach can be simplified by drawing the new mass (m_SS) directly from the IMF without any Poisson distribution and binning. This is done by solving numerically the following equation:

$$\int_{m_l}^{m_\mathrm{SS}} \xi \mathrm{d}m = p \tag{8.1}$$

where m_l is the lower mass of the IMF given by ξ and p is a random value. Once the mass is selected, the sink particle will accrete some gas until reaching the previous mass and then releases a star particle.

In Figure 8.2, the sampling of the IMF is compared for the method presented here and the ones in Wall et al., 2019 and Applebaum et al., 2020. The theoretical IMF is shown in dashed line below the two first models. To produce this image, a total mass of 10^8 $\mathrm{M}_\odot$ is considered, and the IMF presented in section 4.6.1 is restricted within $[1, 50]$ $\mathrm{M}_\odot$ for simplicity (and normalized on this interval). In the case of Applebaum et al., 2020, I consider stellar particles with a mass of 50 $\mathrm{M}_\odot$ which is far lower than in their simulations. This value is clearly below the limit (as was pointed out by the authors) at which this model should be run and show a strong bias towards producing less massive stars. The bias can also be seen in the number of stars produced with 33 millions for this method and 35 millions for the others. While both the sink particles and Wall et al., 2019 do not show any bias due to their sampling of the IMF before accreting any mass, the approach presented in this thesis should be preferred due to its simplicity and lower memory requirement.

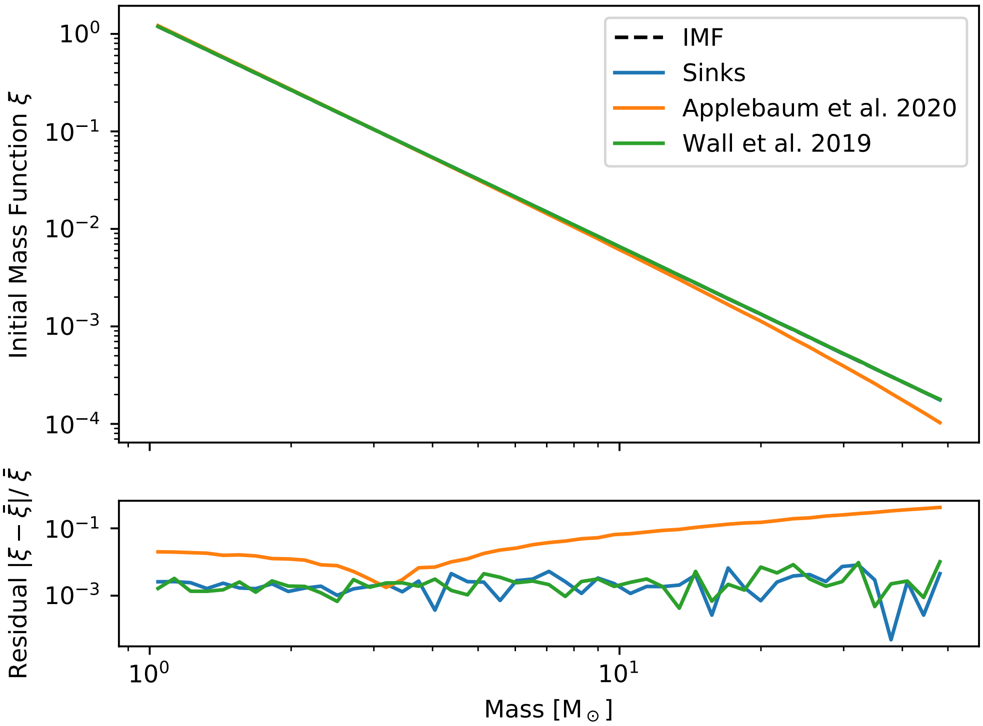


Figure 8.2 – Comparison of the IMF sampling between the sink particles presented here and in Wall et al., 2019, and the stellar particles presented in Applebaum et al., 2020. The theoretical model for the IMF is hidden behind the two first models. While the two first methods properly sample the IMF, the last method shows clearly a bias at this resolution (50 $M_\odot$). The sink particles presented here require far less memory than the one in Wall et al., 2019 without showing any bias.

8.1.5 Stellar Feedback

Once the star is formed, the feedback produced is the same for supernovae of type II except that we do not need to integrate over the IMF to know the number of supernovae and the yields. For supernovae of type Ia (SNIa), the situation is a bit more complicated as they represent the explosion of a white dwarf in a binary system. Such binary systems could in theory be fully resolved by the simulation, but they require non-softened gravitational interactions. In `SWIFT`, we consider only collisionless particles with softened interactions to avoid the low time steps resulting from the collisions. This is justified by the fact that galaxies are dominated by the global potential and not pair interactions (Dehnen, 2001). Therefore, if someone wish to resolve binary systems, the gravity within `SWIFT` should be adapted. The best solution found, that does not require to resolve the binary system, is to update slightly the process of the mass assignment. When we draw a mass that will produce a white dwarf, we randomly decide if the star will end its life in a binary system and produce a SNIa:

$$P_{\text{comp}} = \sum_i b_i \int_{M_{d,l,i}}^{M_{d,u,i}} \frac{\phi_{d,i}(m)}{m} \mathrm{d}m \tag{8.2}$$

where b_i are the normalization of the probability, $\phi_{d,i}$ is the probability to form a binary system and $M_{d,l,i}$, $M_{d,u,i}$ are the limits of the companions. This equation is obtained from Equation 4.86 and the values of the parameters are given there. In such cases, we randomly draw the mass of the companion from the IMF and the probability of binary systems. This mass is then added to the mass target of the sink particle in order to form both stars at the same time and as a single star particle. Enforcing to draw an additional star within a mass range can be dangerous for the IMF sampling and will need special care in order to do it properly. A first possibility would be to modify the IMF within the mass range of the companions in order to take into account this extra chance of being drawn. The second one would be to simply count how many companions are drawn (locally or globally). When a sink particle draw a mass corresponding to a companion, it redraws a mass if the count is not zero and decrements it in order to compensate the extra star in SNIa.

8.1.6 Hybrid Approach for the Stellar Particles

If we take a look at the initial mass function in Figure 4.5, most of the stars will have a low mass and will not produce any supernovae. Therefore, our simulations at medium resolution will spawn many stars that will not have a strong impact on the physics. Therefore, we can still use a stellar population (SP) approach for the low mass stars (with a total mass of m_{SP}) and use a single star (SS) approach for the most massive ones as presented in Applebaum et al., 2020. The masses of the SP particles and the limit between the two approaches (m_t) are free parameters. They should be chosen such that the IMF is well sampled for the stars ending their life in supernovae. A safe choice would be to set the limit to the lowest mass of a star exploding in supernovae as no sampling would be required anymore. If the SPH resolution is compatible with this, a natural choice would be to match the limit with the resolution. Another possible choice would be to follow Applebaum et al., 2020 where they assign the limit in between the lowest mass of a SNII and the largest mass of SNIa. This approach greatly simplifies the treatment of SNIa as they could still be done with the stellar population hypothesis.

With the multiple types of stellar particle, the sampling of the IMF during their formation becomes a bit more complex and requires two steps. The first one consists in randomly deciding if we are forming a SP or SS particle. This is done from the fraction of stars represented by the SP particles and normalized with the mass of the particle:

$$P_{\mathrm{SP}} = \frac{1}{m_{\mathrm{SP}}} \frac{\int_{m_l}^{m_t} m\xi \mathrm{d}m}{\int_{m_l}^{m_u} \xi \mathrm{d}m} \tag{8.3}$$

where ξ is the initial mass function by number defined in Eq. 4.78 and m_l, m_u are the lower and upper mass limits of the IMF. Then, if an SP particle is selected, the mass target is set to m_{SP}. Otherwise, the mass is randomly picked from the initial mass function restricted on the SS particle interval.

In Algorithm 8.7, the steps for the formation of stars from sink particles is shown where m_{sink} is the mass of the sink and m is the mass target of the next star. For the stellar feedback, it is equivalent to the discrete feedback provided in chapters 4 and 5 but with the mass given by the SS particle.

Algorithm 8.7 sink_formation_sink

Depends on: sink_accretion

```
while $m_{\mathrm{sink}} > m$ do
    Creates a star of mass $m$
    $m_{\mathrm{sink}} \leftarrow m_{\mathrm{sink}} - m$
    $m \leftarrow$ mass_target()                ▷ Given in Algorithm 8.8
```

Algorithm 8.8 Function mass_target()
The treatment of the bias introduced by SNIa is not shown here. See the text for two different methods

```
if random(0, 1) < $P_{SP}$ then                                          ▹ Equation 8.3
    return $m_{SP}$
Draw a mass $m$ above the limit between SP and SS particles according to the IMF.
if $m$ corresponds to a white dwarf and random(0, 1) < $P_{comp}$ then   ▹ See section 4.6.4
    Draw the mass of the companion ($m_{comp}$)
    $m \leftarrow m + m_{comp}$
return $m$
```

9 Constrained Initial Conditions

> Not all those who wander are lost.
> — J. R. R. Tolkien, The Fellowship of the Ring

Cosmological simulations have been a tremendous help to understand dwarf galaxies. In such simulations, the initial conditions (ICs or random realization) are particularly important as gravity quickly growths any initial perturbations. In the classical approach, they are constrained by the power spectrum at high redshift without any additional restriction on the morphology (Hahn and Abel, 2011). It means that obtaining a universe at z=0 comparable to the Local Universe is unlikely and often relies on a low number of parameters for the comparison (e.g. two galaxies with masses comparable to the Milky Way and Andromeda along with a comparable distance). Recently, Libeskind et al., 2020 have shown cosmological simulations where the initial conditions have been additionally constrained with observations of galaxies at $z \sim 0$. While on small and strongly non linear scales, the constrained ICs only impacts the average density field (or velocity field) and not the fluctuations, the large and linear scales are strongly constrained and reproduce the environment of the Milky Way. Such reproduction increases the quality of comparisons between simulations and the Milky Way's satellites. In the years to come, constrained initial conditions will certainly become a standard in the modeling of the Local Group.

As part of my thesis, I have developed the first, at my knowledge, open source code based on GPUs that generates constrained initial conditions [1]. The number of constraints is the main driver of the time spent generating initial conditions. Thus, my code was developed with the aim to constrain initial conditions from more than 12'000 constraints in a reasonable amount of time.

---

[1] https://gitlab.com/loikki/initialconditions

Let us start with the overview of the constrained realizations. They were first introduced in Hoffman and Ribak, 1991; Zaroubi et al., 1999 and have been further improved by Lavaux, 2016; Graziani et al., 2019. In the last publication previously mentioned, they presented a probabilistic and theoretical model of the observations depending on the constrained realization at z=0 that we aim to obtain. The observations in this model consist in both the distance modulus and redshift of galaxies. This redshift is the one observed and thus includes both the cosmological and Doppler redshifts. The model can be "inverted" to obtain a probability on our ICs according to the observations thanks to Bayes theorem. Then, using the Gibbs sampling, the model can be sampled and provides a set of constrained realizations. For each sample, the method described in Zaroubi et al., 1999 is used and requires the generation of a random realization. While the central part of the method is the same, the probabilistic model from Graziani et al., 2019 provides a self-consistent and more accurate model of the uncertainties. It is worth to mention that not all the samples are equivalent as they will have different probabilities to generate the observations, but will always be consistent with the power spectrum. As shown in Graziani et al., 2019, the average of all the samples from this Bayesian method accurately reproduces the Local Universe. This average changes the power spectrum, thus the constrained initial conditions cannot use it and are based on the best sampling according to the model mentioned previously.

As the initial conditions are generated at $z = 0$ due to the observations, they need to be rewound at higher redshift. Different techniques exist, but they are all based on the Zel'dovich approximation. The main differences are the operations before/after the Zel'dovich approximation and which objects are rewound. Currently, the most accurate method seems to be the one described in Sorce et al., 2014.

This chapter is based on Graziani et al., 2019, but it would not have been possible to fully understand the technique without the excellent explanations written in Doumler, 2012. First, I will spend a bit of time on random realizations and zoom simulations (section 9.1 and section 9.2) as the random realizations are required within the method. Next, in section 9.3, I will present the method behind all the techniques for constrained realizations (Hoffman and Ribak, 1991; Zaroubi et al., 1999). In section 9.4, I will finally present the method described in (Graziani, 2018). To conclude this chapter, I will present in section 9.5 the algorithm that rewinds the constrained realization from z=0 to high redshift. In Figure 9.8, a flowchart giving all the steps is provided. Finally, even if I will only mention the Local Group in this chapter, this method is general and can be applied to any location in the universe as long as enough observations are available.

9.1 Initial Conditions from the Power Spectrum

Different codes (Bertschinger, 2001; Prunet et al., 2008; Stadel et al., 2009; Pen, 1997; Sirko, 2005) have been developed in order to generate initial conditions from the power spectrum (also called random realization in this chapter). I will only focus on `MUSIC`, but the other codes should be comparable. In this section, I will only give a quick summary of the method and all the equations can be found in Hahn and Abel, 2011.

To compute a random realization, we first need to define the power spectrum by $P(k) = \langle\tilde{\delta}(\mathbf{k})\tilde{\delta}^*(\mathbf{k})\rangle$, where $\tilde{\cdot}$ is the Fourier transform, * is the conjugate transpose, k the wave number, $\delta(x) = \frac{\rho(x)}{\bar{\rho}} - 1$ the overdensity field, ρ ($\bar{\rho}$) the (average) density and $\langle\cdot\rangle$ is the ensemble average [2]. Let us use the follow ansatz:

$$\tilde{\delta}(k) = \sqrt{P(k)}\,\mu \tag{9.1}$$

where μ is a random complex variable and inject the value into the previous definition, we obtain that the random variable should respect: $\langle\mu\mu^*\rangle = 1$. A standard approach is to split μ into a random uniform phase and a Gaussian with a standard deviation of 1 and a mean of 0 (Hahn and Abel, 2011; Springel et al., 2005). Therefore, the overdensity field can be easily computed in Fourier space through a random variable and a known power spectrum. If a regular mesh is used for the overdensity, the Fast Fourier Transform can be used to efficiently sample the field in real space.

For AMR codes, it would be enough to stop here for the overdensity field as each cell can have a different mass and thus the overdensity can be directly used to set the masses. For SPH codes, the overdensity field is given by the particles' positions. It means that we need a way to figure out where the particles are supposed to be. The usual approach is to compute the overdensity field on a regular mesh, transform the cells into particles and move them according to the displacement field ψ: $\mathbf{x} = \mathbf{q} + \psi(\mathbf{q})$ where $\mathbf{x}$ is the final position and $\mathbf{q}$ the position on the grid. To first order, the displacement field can be easily computed from the gravitational potential ϕ and thus the overdensity (Zel'dovich, 1970):

$$\psi(\mathbf{q}, t) = -\frac{2}{3H_0^2 a^2 D_+(t)}\nabla_{\mathbf{q}}\phi(\mathbf{q}, t) \tag{9.2}$$

$$\Delta_{\mathbf{q}}\phi(\mathbf{q}, t) = \frac{3}{2}H_0^2 a^2 \delta(\mathbf{q}, t) \tag{9.3}$$

where H_0 is the Hubble constant, a the scale factor and D_+ is the growth factor of linear density perturbations given by (Eisenstein and Hu, 1998):

$$D_+(z) = (1+z)^{-1}\frac{5\Omega(z)}{2}\left(\Omega(z)^{4/7} - \Omega_\Lambda(z) + \left[1 + \frac{\Omega(z)}{2}\right]\left[1 + \frac{\Omega_\Lambda(z)}{70}\right]\right)^{-1}. \tag{9.4}$$

The parameters Ω and Ω_Λ are the density parameters as seen by an observer at redshift z (see appendix A in Eisenstein and Hu, 1998 for more details).

The initial conditions are not fully described with the overdensity field, they also require the initial velocity field that has a large impact on the structure growth. This velocity is directly obtained by a derivation of the previous equation with respect to time: $\mathbf{v} = \frac{\mathrm{d}}{\mathrm{d}t}\psi(\mathbf{q})$. While this approach is only first order accurate, it can be extended into higher order as is done by `MUSIC` (Hahn et al.,

---

[2] `CAMB` and `MUSIC` are not using exactly the same definition for the power spectrum due to a factor of 8π.

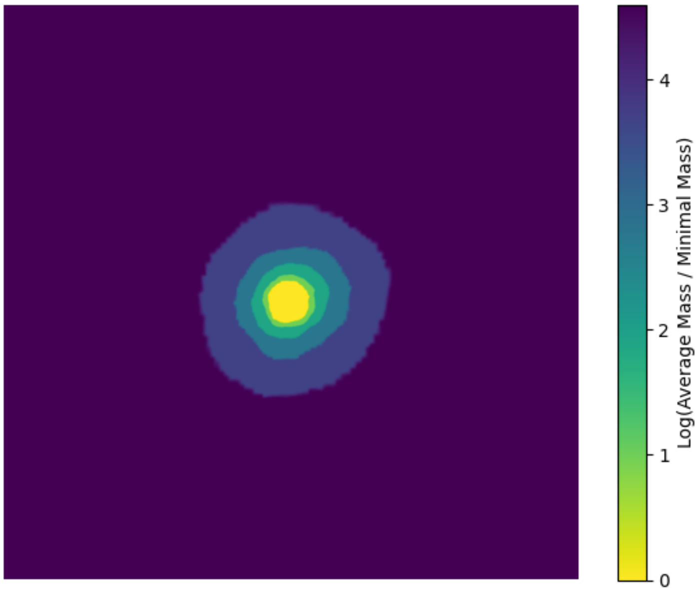
2021; Hahn and Abel, 2011).

9.2 Zoom Simulations

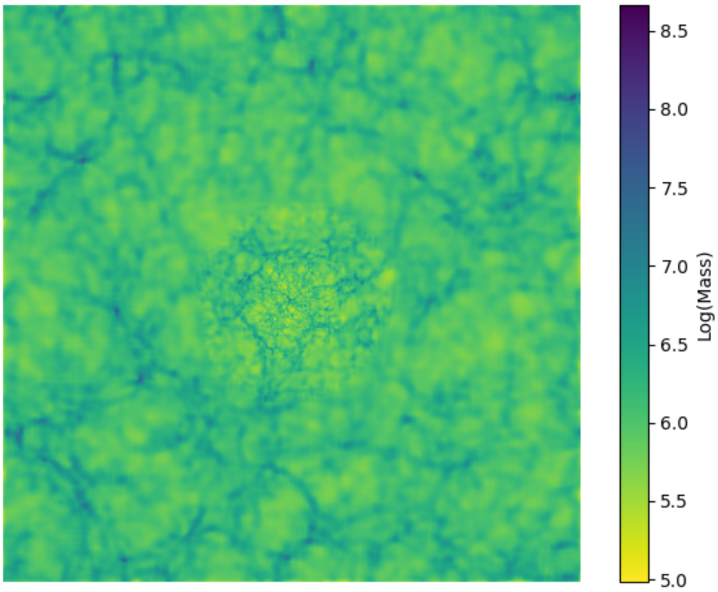
For dwarf galaxies, zoom simulations are often used in order to reach high resolution and keep an affordable computation time. They consist in a small volume at high resolution and the surrounding volume simulated at low resolution. The initial conditions can be generated with the code `MUSIC` (among other codes). Both AMR and SPH codes can use this technique, but, by its Lagrangian nature, SPH handles zoom simulations far more easily. The initial conditions consist of a large scale box (a few Mpc to Gpc) at very low resolution (typically $10^6 - 10^8 M_\odot$ per particle for simulations of dwarf galaxies) and an area of interest at high resolution (typically less than $10^4 M_\odot$ for simulations of dwarf galaxies). In order to link the two areas, the resolution in between is monotonically increasing from the low resolution to the high resolution by factor of 8 in 3D. `MUSIC` does it by having a constant number of particles on each intermediate level along each direction. In order to ensure the conservation of both the mass and the Fourier modes between two different levels, the properties of the fine levels need to be corrected by the larger scale properties of the field (Hahn and Abel, 2011).

While the high resolution particles represent the actual initial conditions that we want to evolve, the lower resolution particles can be considered as boundary conditions and represent the evolution of the surrounding large scale structures. Therefore, it is important to define an area of interest large enough in order to avoid the presence of low resolution particles close to the studied galaxy. This high resolution area is usually defined by running a low resolution DMO simulation until redshift zero. From the final snapshot, a galaxy is picked and, using the IDs of the selected particles, the corresponding area in the ICs is computed. From the positions, the high resolution area is defined along with all the lower levels. Thus, most of the ICs codes are designed such that if they use the same random seed and parameters, they can always reproduce the same ICs. Some codes, such as `MUSIC`, can even read random numbers from a file provided by the user (which will play an important role in the generation of the constrained ICs later).

Figure 9.1 shows the different resolutions for a snapshot at z=6.6 from the final model done in the `AGORA` project along a slice representing the total mass. The first image represents a slice of the full volume and is weighted in order to represent the average mass of the particles normalized by the smallest mass. At the center of the picture, the high resolution area is shown in yellow and, in the border, in purple the lowest resolution area. The second image represents the mass in each pixel and the large scale structures can be seen in both the low and high resolution areas. With `MUSIC`, the high resolution volume should always be located at the center as `MUSIC` is shifting it at this location. The ICs were generated in order to span 6 different levels (with factor of 8 in mass between each of them).



(a) Average mass of the particles



(b) Total mass

Figure 9.1 – Example of zoom simulation. Both images are slices at the center of the simulation volume. The first one represents the average mass of the particles normalized by the minimal mass while the second one the total mass per pixel. The high resolution area is situated at the center of the image and the low resolution on the borders. As it can be seen in the second image, the cosmological structures are simulated within the whole volume, but only the center is done at high resolution. The simulation shown is the simulation from `AGORA` in chapter 3 at z=6.6.

9.3 Wiener Filter and Constrained Realization

In this section, I will describe the technique used to constrain the ICs. This method was derived by Hoffman and Ribak, 1991; Zaroubi et al., 1995. However Doumler, 2012; Zaroubi et al., 1999 provide a deeper description. As before, I will only provide a summary of the equations.

With the unconstrained initial conditions, only a power spectrum is required and thus the ICs are lacking information about the local environment of the Milky Way. The Wiener Filter and Constrained Realization technique (WF/CR) improve the initial conditions by taking into account the information of observed local galaxies (both in position $\mathbf{r}$ and radial velocity $\mathbf{v} \cdot \hat{\mathbf{r}}$). The idea is to use a random realization and modify it with the large scale velocity field given by the observations. Let us start by defining the peculiar radial velocity of a galaxy:

$$v_r(\mathbf{r_i}) = \mathbf{v}(\mathbf{r_i}) \cdot \hat{\mathbf{r}}_\mathbf{i} + \epsilon_i \tag{9.5}$$

where i is the index of the observed galaxy and ϵ are the uncertainties on the radial velocity (not necessary a Gaussian distribution) and are assumed to be independent of each others.

9.3.1 Wiener Filter

The first step consists in computing the large scale velocity field with the Wiener filter. This filter was designed in signal processing to minimize the variance of the signal. In our case, we can use it in order to remove the measurement errors and recover the true linear velocities (in the sense of the linear instability theory). It can reconstruct the 3D velocity field at any position $\mathbf{r}$ from a discrete set of observations:

$$\mathcal{WF}\{v_r(\mathbf{r})\} = \sum_{ij}^{N} \langle \mathbf{v}(\mathbf{r}) v_r(\mathbf{r_i}) \rangle \, \langle v_r(\mathbf{r_i}) v_r(\mathbf{r_j}) \rangle^{-1} v_r(\mathbf{r_j}) \tag{9.6}$$

where N is the number of observations, the first term is the cross-correlation matrix (3xN matrix), the second is the auto-correlation matrix (NxN) and finally the third is a vector of size N. The notation $\langle\rangle$ denotes correlation matrices and the power $^{-1}$ is a full inversion of matrix and not simply an inversion of the coefficient. Thus, the Wiener mean field is known as long as the correlation matrices are known. As it will be seen later, the two matrices can be derived directly from the power spectrum. The same filter can be applied to the overdensity field δ and simply requires a change in the first correlation function thanks to the assumption of linear instability theory:

$$\nabla \cdot \mathbf{v} = -H_0 f(\Omega_m, z) \delta \tag{9.7}$$

where f is the linear growth rate and Ω_m is the matter density parameter. This filter is only based on the two-point correlation function and thus is often discarded in signal processing as it will neglect any higher statistical moments. In the case of the overdensity, it is in fact an advantage as, for Gaussian random fields, all the statistical properties of the field are contained within this correlation (Zaroubi et al., 1999).

9.3.2 Constrained Realization

As the Wiener filter does not conserve the power spectrum, the mean field cannot be directly used as the constrained realization. Therefore, a random realization (RR) is produced by `MUSIC` in order to complete the missing power spectrum of the mean field. The idea is to set our location at the center of the volume and then compute the radial velocities for the RR at the same positions as the observations. Then the Wiener mean field can be removed from the RR and the one from the observations is added:

$$\mathbf{v}^{\mathrm{CR}}(\mathbf{r}) = \mathbf{v}^{\mathrm{RR}}(\mathbf{r}) - \mathcal{WF}\{v_r^{\mathrm{RR}}(\mathbf{r})\} + \mathcal{WF}\{v_r(\mathbf{r})\}. \tag{9.8}$$

As the residual from the random realization $\mathbf{v}^{\mathrm{RR}}(\mathbf{r}) - \mathcal{WF}\{v_r^{\mathrm{RR}}(\mathbf{r})\}$ does not depend on the constraints $v_r^{\mathrm{RR}}(\mathbf{r})$, but only on the power spectrum, it is a valid realization of the missing power spectrum in the Wiener mean field of the observations $\mathbf{v}(\mathbf{r}) - \mathcal{WF}\{v_r(\mathbf{r})\}$ (see section 3.3.1 in Doumler, 2012 for details). The constraints from the random realization are taken at the same position than the observations. Thus, the correlation matrices are exactly the same and the previous equation can be rewritten as:

$$\mathbf{v}^{\mathrm{CR}}(\mathbf{r}) = \mathbf{v}^{\mathrm{RR}}(\mathbf{r}) + \sum_{ij}^{N} \langle \mathbf{v}(\mathbf{r}) v_r(\mathbf{r_i}) \rangle \, \langle v_r(\mathbf{r_i}) v_r(\mathbf{r_j}) \rangle^{-1} \left(v_r(\mathbf{r_j}) - v_r^{\mathrm{RR}}(\mathbf{r_j}) \right). \tag{9.9}$$

The same can be done for the overdensity field and it gives:

$$\delta^{\mathrm{CR}}(\mathbf{r}) = \delta^{\mathrm{RR}}(\mathbf{r}) + \sum_{ij}^{N} \langle \delta(\mathbf{r}) v_r(\mathbf{r_i}) \rangle \, \langle v_r(\mathbf{r_i}) v_r(\mathbf{r_j}) \rangle^{-1} \left(v_r(\mathbf{r_j}) - v_r^{\mathrm{RR}}(\mathbf{r_j}) \right). \tag{9.10}$$

As one can expect, in the absence of any constraints, the constrained realization is simply given by the random realization.

The two previous equations completely define the WF/CR and require the computation of correlation matrices that can be derived from the power spectrum. Let us first simplify the correlation matrices (see Gorski, 1988 for the details):

$$\langle v_r(\mathbf{r_i}) v_r(\mathbf{r_j}) \rangle = \hat{\mathbf{r}}_\mathbf{i} \cdot \langle \mathbf{v}(\mathbf{r_i}) \mathbf{v}(\mathbf{r_j}) \rangle \cdot \hat{\mathbf{r}}_\mathbf{j} + \epsilon_i^2 \delta_{ij} \tag{9.11}$$

$$\langle \mathbf{v}(\mathbf{r_i}) \mathbf{v}(\mathbf{r_j}) \rangle_{\mu\nu} = H(z)^2 f(\Omega, z)^2 \left[\psi_T(r_{ij}) \delta_{\mu\nu} + \left(\psi_T(r_{ij}) - \psi_R(r_{ij}) \right) \hat{\mathbf{r}}_\mu \cdot \hat{\mathbf{r}}_\nu \right] \tag{9.12}$$

$$\langle \delta(\mathbf{r_i}) v_r(\mathbf{r_j}) \rangle = -H(z) \, f(\Omega, z) \, \zeta(r_{ij}) \tag{9.13}$$

where δ_{ij} is the Dirac function, μ and ν indicates the dimension index, r_{ij} is the distance between the two galaxies, $\hat{\mathbf{r}}$ is the unit vector representing this distance, $f(\Omega, z)$ is the growth rate that depends on the cosmological density parameters (Ω) and redshift (z). A simple approximation of the growth rate is $\Omega_m(z)^{0.6}$ where Ω_m is the matter density parameter. The functions ψ_T, ψ_R and ζ are the velocity-velocity correlations (transversal and radial) and the velocity-density correlation. In theory, one correlation function should be required per dimension. As the velocities and densities at such scales are evolving only due to gravity and gravity is a radial force, only two components are necessary. For the density-velocity correlation, this is reduced to a single component as the velocity is related to the densities through the divergence (Equation 9.7). The correlation functions can be written in the following way (Gorski, 1988):

$$\psi_T(x) = \frac{1}{2\pi^2}\int_0^\infty \left[j_0(kx) - \frac{2j_i(kx)}{kx} \right] P(k)\mathrm{d}k \tag{9.14}$$

$$\psi_R(x) = \frac{1}{2\pi^2}\int_0^\infty \frac{j_1(kx)}{kx} P(k)\mathrm{d}k \tag{9.15}$$

$$\zeta(x) = \frac{1}{2\pi^2}\int_0^\infty k j_1(kx) P(k)\mathrm{d}k \tag{9.16}$$

$$\sigma^2 = \frac{1}{6\pi^2}\int_0^\infty P(k) \tag{9.17}$$

where j are the spherical Bessel function, P the power spectrum and σ the variance of the linear velocity field [3] (about 300 km/s). In order to remove the impact of the smallest (and strongly non-linear) scales, it is possible to include a Gaussian kernel to the integrals with a size given by the resolution (see for example Doumler, 2012).

In Figure 9.2, the different correlation functions are shown. The typical correlation length is around 150 Mpc (which corresponds, as it could be expected, to the baryon acoustic oscillations). It means that two points separated by more than this distance can be considered as more or less independent.

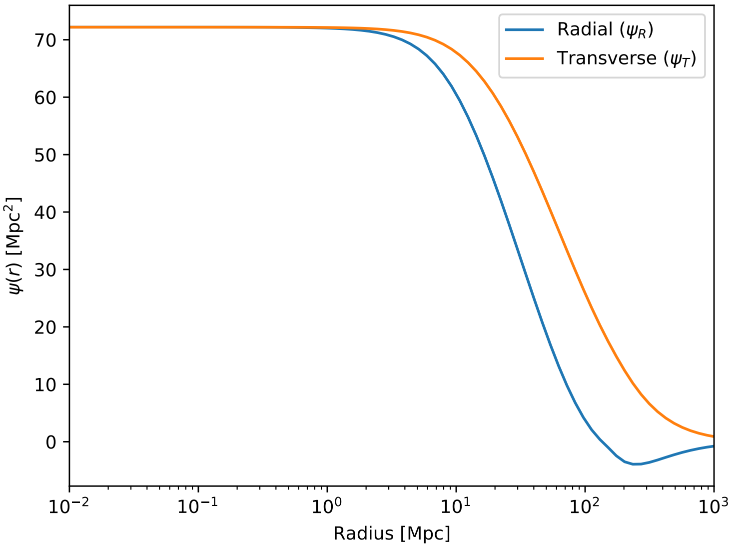
9.3.3 Inversion of the Correlation Matrix

As the correlation matrix $\langle v_r(\mathbf{r_i}) v_r(\mathbf{r_j})\rangle$ is already large due to the number of observations ($\sim$ $12'000$ in the galaxy catalog presented later) and will still increase in the future, an efficient inversion is required during the computation of the constrained realization. Unfortunately, currently the fastest inversion possible is in $O(n^{2.373})$ where n is the size of the matrix. This is done from an optimized version of the Coppersmith-Winograd algorithm, but due to the large constant computational time of such algorithms, they are rarely used. Here I am using simply a LU decomposition approach[4] with a complexity of $O(n^3)$ and parallelized on GPU.

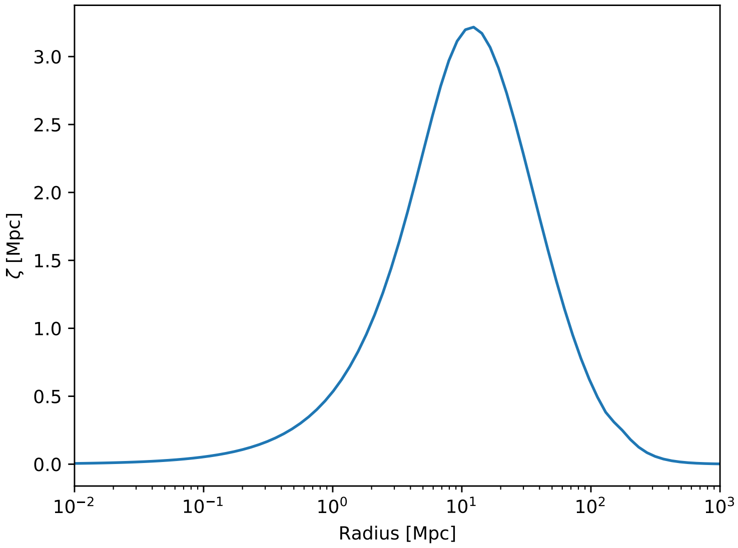
---

[3] a small error was present in the formula of Graziani, 2018

[4] First, the matrix is decomposed into a lower and an upper triangular matrix and then we can easily solve the system $LUX = \mathbb{1}$.



(a) Two points velocity-velocity correlation function (ψ_T and ψ_R)



(b) Two points cross correlation function between the density and velocity (ζ).

Figure 9.2 – Two points correlation functions for the density - velocity and velocity - velocity as function of the distance at redshift 0. The different functions are given in equations 9.14-9.16. The baryon acoustic oscillations (BAO) can be seen at a distance of 100-150 Mpc in the radial component of the velocity-velocity correlation producing an anti-correlation before dropping to 0. All the functions fall to 0 around 150 Mpc. Thus the constraints acts mostly on scales smaller than this.

9.3.4 Example

In Figure 9.3, the radial velocities extracted from a large set of constrained realizations (CR) with the same constraints are shown in gray. In this example, only a single arbitrary constraint (in red) is used in order to show how a constraint acts on the final realization. The uncertainties are assumed to be null on this constraint. The green area represents the standard deviation of all the CRs and the green line the mean value. As expected, where the realization is free from any constraint, the scatter is close to linear variance ($\sigma \sim 300$ km/s), while close to the red constraint it is given by the uncertainties. The constraint's position is chosen to match with the grid used for the overdensity in order to show the diminution of the standard deviation. In case of low resolution meshes (cells size comparable to the correlation length), a constraint between two cells would have almost no visible impact on the standard deviation, but would still impact the mean value as it can be seen at large distances in this figure (around ±200 Mpc). Thus, the WF/CR correctly enforces the initial conditions to match the observations at their location and does not bias them far away from the constraints.

9.4 Bayesian Approach to the Constrained Initial Conditions

In this section, I will present the Bayesian approach to the constrained initial conditions. It consists in finding a probabilistic model for the linear velocity field given the observations. This model can then be sampled and the field with the highest probability gives our solution. This approach has the advantage to provide more freedom on the model of the uncertainties (e.g. not necessarily gaussian) and, hence, a more accurate solution. Here, I will start by describing the observations used to constrain the initial conditions. Then, a probabilistic model for the observations assuming a known velocity field is presented. This model is then inverted thanks to Bayes theorem and gives the probability of a velocity field given the observations. Once the last model obtained, I will present the associated sampling procedure based on the Gibbs sampling. This work is based on Graziani et al., 2019 and Graziani, 2018 and all the equations are described in the previous publication and thesis.

9.4.1 Observations

To constrain our initial conditions, we first need a catalog of observations. Currently, the most appropriate choice is CosmicFlows (CF; Tully et al., 2016) as it is the largest catalog containing both the velocities and positions of galaxies in the local Universe and was specially designed to compute the local cosmography. The third data release, CF-3, consists in two different catalogs. The first contains the raw data, while the second contains galaxies grouped together when they belong to the same group or cluster. For the purpose of the constrained realization, it is important to consider only the grouped catalog as it removes part of the non-linear component from the observations and is a better representation of the velocities at large scales.

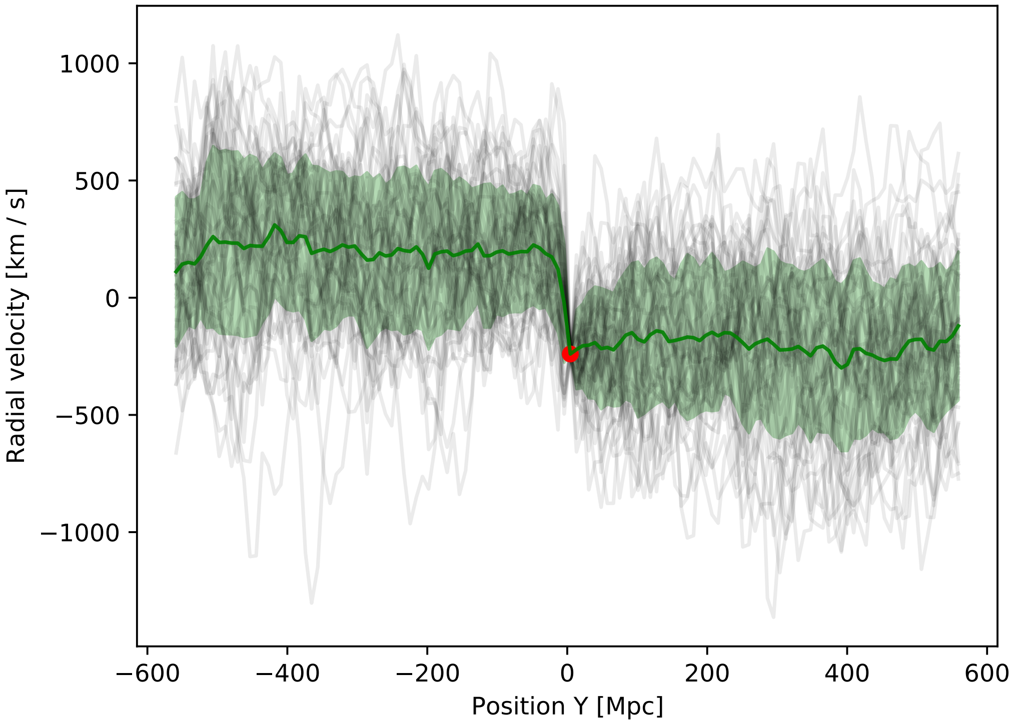


Figure 9.3 – Radial velocity along a line of constant X and Z coordinates for different constrained realization in grey and with a single constraint in red. The mean value is indicated with the green line and the standard deviation with the green area. All the realizations are strongly impacted by the constraint and have the same radial velocity close to it. Further away, the constraint has almost no impact and thus in theses regimes the constrained realizations behave in a similar way than random realizations would have.

As the CF catalogs are a combination of different catalogs, the data are not homogeneous. In CF-3, the distance measurement were done mainly with 2 different techniques: the Fundamental Plane (FP) relation and the Tully-Fisher relation (TF). While only the 6dF catalog is using the FP relation, the TF relation is applied on a few different datasets and mainly from Spitzer and Arecibo. As the observational bias are not the same between the different datasets, they are split in 5 categories: 6dF, Spitzer, Arecibo, other TF measurement and other type of measurements.

In Figure 9.4, the redshift and position distributions of the galaxies (or group) contained in CF-3 are presented. As it can be seen, the catalog contains galaxies up to redshift 0.1 and distances of 400 Mpc. The highest velocities are $\sim$ 4000 km / s. Only the uncertainties on the distance modulus are given and the maximal one is 0.54. It corresponds to a maximal relative error of 0.02 on the distance modulus. For the radial velocities, they are measured through spectroscopy and typical errors are of the order of 50 km / s. In Figure 9.5, the localization of the observations in the sky are shown in equatorial coordinates. As one can expect, observations are not performed within the Milky Way (white band). CF-3 is composed mostly of 6dF (in blue), thus most of the observations are made in the south hemisphere.

9.4.2 Description of the Model

Let us now focus on the core of our model that consists in the probabilities used for the Gibbs sampling. The first step is to summarize what we are looking for and what we know. From the observations, we know:

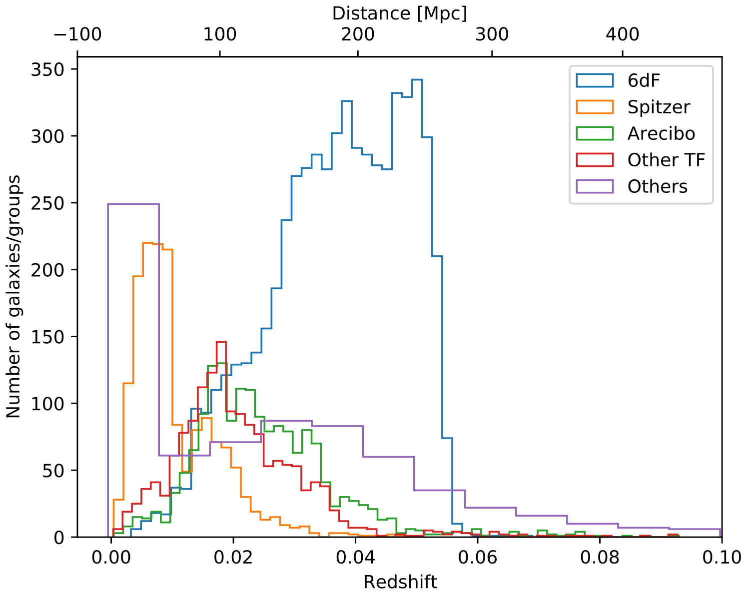
- the distance modulus (μ),
- the observed redshift [5] (z),
- the equatorial coordinates (ϕ, θ),
- the related uncertainties (σ_μ, σ_{cz} where c is the speed of light).

We would like to find the large scale overdensity field (δ) generated by galaxies to constrain the environment of the Milky Way. As given in equation 9.7, this field can be directly obtained from the linear velocity field thanks to the linear instability theory and thus both fields contain the same information. As the observations directly measure velocities, it is generally more easy to deal with the corresponding velocity field and compute the overdensity field from it. The linear velocities are discretized on a regular mesh with N_{grid} elements in each direction. Our model contains the following unknown variables:

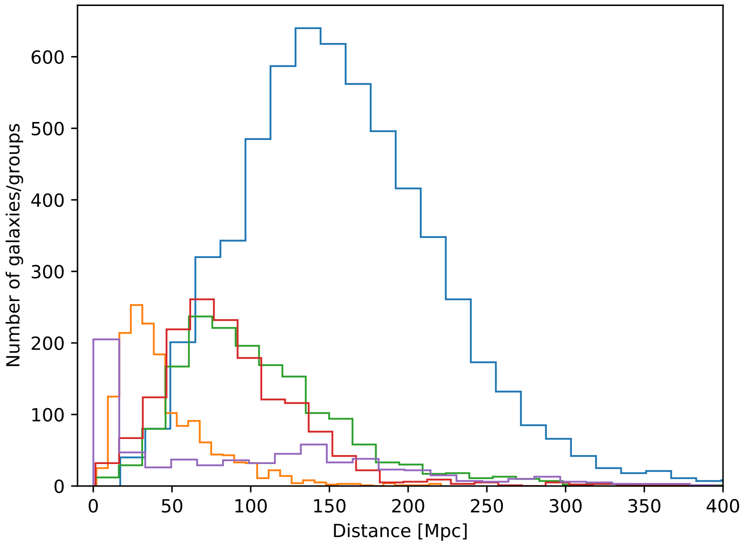
- The linear velocity field (v_L),

---

[5] It includes both the effect of the Hubble expansion and peculiar velocity, and corresponds to the shift observed in the spectrum.



(a) Redshift distribution



(b) Distance distribution

Figure 9.4 – Redshift and distance distributions of the galaxies (or group) from CF-3. The catalog reaches distance of 400 Mpc and redshift of 0.1. The highest redshift corresponds to velocities of $\sim$ 4000 km / s. In order to provide some insights on the redshift, the first graph provides a second x-axis where the distances are shown. They are computed with the assumption that the redshifts are purely due to the Hubble expansion.

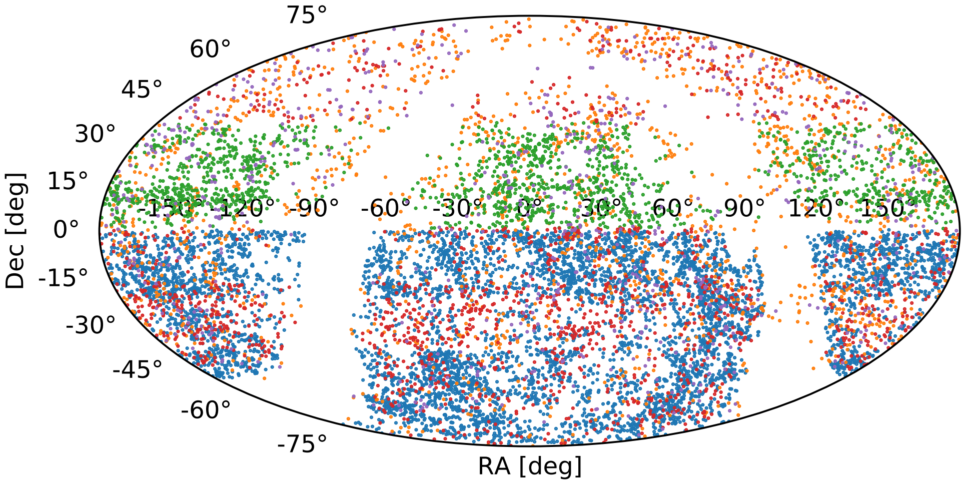


Figure 9.5 – Localization in the sky of the observations in the equatorial system of coordinates. The white band is due to the Milky Way. The colors corresponds to Figure 9.4.

Known Variable	Size	Description
N_{obs}	1	Number of galaxies/clusters observed
N_{grid}	1	Number of elements in the mesh for the linear velocity along 1 dimension
μ	N_{obs}	Distance modulus of the galaxies
z	N_{obs}	Observed redshift of the galaxies
σ_μ	N_{obs}	Error on μ
$\sigma_{cz} = c\sigma_z$	1	Error on cz (50 km / s)
ϕ	N_{obs}	Supergalactic longitude
θ	N_{obs}	Supergalactic latitude

Unknown Variable	Size	Description
h_{eff}	1	Effective Hubble constant
σ_{NL}	1	Non-linear velocity dispersion
$\mathbf{v}_L$	N_{grid} x N_{grid} x 3	Linear velocity field
d_{obs}	N_{obs}	Luminosity distance of the observed galaxies
p	4 x 3	Parameters defining the distance priors

Table 9.1 – Variables used for the Gibbs sampling (Graziani et al., 2019). The upper part of the table contains the known variables, the lower part contains the unknown variables that the Gibbs sampling will try to estimate.

- a global effective Hubble constant (h_{eff}),
- a global non-linear perturbation to the velocity field (σ_{NL}),
- a distance for each observed galaxy (d_{obs}),
- a model for the Malmquist bias (parameters given by p).

The effective Hubble constant is introduced in order to take into account the possible existence of large inflow/outflow into the Local Group and possible unmodeled observational bias. If the observations are fully consistent with the Hubble constant, this parameter should be 1 (as it is the case in Graziani et al., 2019). The Malmquist bias is described later. Table 9.1 provides a summary of all the variables.

Before moving to the probabilities, it is important to detail some important quantities. The distances of galaxies can be computed from the relation between the distance modulus and the distance:

$$\mu = 5 \log_{10} \frac{h_{\mathrm{eff}} d_{\mathrm{obs}}}{10\mathrm{pc}}. \tag{9.18}$$

Here we also include a small correction with h_{eff} in order to reduce the impact of potential external flows. While the impact is exactly the same as the Hubble parameter, it should not be taken as the true value of the Hubble parameter as it is impacted by selection functions, flows at larger scales than the observed region, and non-linear effects.

The peculiar radial velocities can be computed through the formula for the redshift and depends on both the distance modulus and the redshift:

$$1 + z = (1 + \bar{z})\left(1 + \frac{v_r}{c}\right) \tag{9.19}$$

where $\bar{z}$ is the cosmological redshift computed from the luminosity distance

$$d_{\mathrm{obs}} = c\frac{1+\bar{z}}{H_0}\int_0^{\bar{z}} \left(\Omega_\Lambda + \Omega_m(1+z)^3\right)^{-1/2} \mathrm{d}z \tag{9.20}$$

computed in Equation 9.18 where we assume the cosmological parameters for radiation Ω_r and the curvature Ω_k to be null as we are concerned only about the local Universe. As this equation is monotonically increasing, it can be easily inverted numerically to obtain the cosmological redshift from the distance. The observed velocities are then the sum of the linear field ($\mathbf{v}_L$ shared between all the galaxies) and the non-linear part $\mathbf{v}_{\mathrm{NL}}$ that is independent of the other galaxies and assumes a Gaussian distribution with a standard deviation of σ_{NL}. While galaxies tend to have correlated non-linear velocities, this last assumption is sufficient for our purpose as we are mostly concerned about the linear field.

Now that all the required variables are defined, we can go back to our model. We wish to obtain the most probable value of $\mathbf{v}_L$ according to our model of:

$$P\left(d_{\mathrm{obs}}, h_{\mathrm{eff}}, \sigma_{\mathrm{NL}}, \mathbf{v}_L \mid \{\mu, z\}\right) \tag{9.21}$$

along with the distances, effective Hubble constant and the non-linear velocity dispersion. To simplify the description of this model, we will use Bayes theorem and obtain:

$$P\left(d_{\mathrm{obs}}, h_{\mathrm{eff}}, \sigma_{\mathrm{NL}}, \mathbf{v}_L \mid \{\mu, z\}\right) \propto \mathcal{L}\; P(d_{\mathrm{obs}})P(h_{\mathrm{eff}})P(\sigma_{\mathrm{NL}})P(\mathbf{v}_L) \tag{9.22}$$

where $\mathcal{L} = P\left(\{\mu, z\} \mid d_{\mathrm{obs}}, h_{\mathrm{eff}}, \sigma_{\mathrm{NL}}, \mathbf{v}_L\right)$ is the likelihood that represents the probability of the observations according to the model and $P(\cdot)$ are the priors that represent our prior knowledge of the parameters' value.

Likelihood

The first step in any MCMC method is to define the likelihood. This probability function defines the probability of observing our galaxies (μ and z) given our model:

$$\begin{aligned} \mathcal{L} &= P\left(\{\mu, z\} \mid d_{\mathrm{obs}}, h_{\mathrm{eff}}, \sigma_{\mathrm{NL}}, \mathbf{v}_L\right) && (9.23) \\ &= P\left(\mu \mid d_{\mathrm{obs}}, h_{\mathrm{eff}}\right) P\left(z \mid \sigma_{\mathrm{NL}}, d_{\mathrm{obs}}, \mathbf{v}_L\right). && (9.24) \end{aligned}$$

Here the probability function is split in two where the first part describes the positions and the second part the velocities. The telescopes are directly measuring the distance modulus and therefore the associated error can be modeled as a Gaussian around the measured value:

$$P\,(\mu \mid d_{\mathrm{obs}}, h_{\mathrm{eff}}) = \mathcal{N}_{\mu}\left(5\log_{10}\frac{h_{\mathrm{eff}}d_{\mathrm{obs}}}{10\mathrm{pc}}\ ,\ \sigma_{\mu}^2\right) \tag{9.25}$$

where $\mathcal{N}_x(y, \sigma^2) \propto \exp\left(-\frac{(x-y)^2}{2\sigma^2}\right)$ is a Gaussian. It is worth to mention that the error is thus not symmetrical in luminosity distance. For the velocity, we also assume a Gaussian around the radial component of the linear velocity field:

$$P\,(z \mid \sigma_{\mathrm{NL}}, d_{\mathrm{obs}}, \mathbf{v}_L) = \mathcal{N}_{v_r}\left(\mathbf{v}_L(\mathbf{r})\cdot\hat{\mathbf{r}}\ ,\ \sigma_{cz}^2(1+\bar{z})^{-2} + \sigma_{\mathrm{NL}}^2\right) \tag{9.26}$$

where $\mathbf{r}$ is the position of the galaxies (depending on Supergalactic longitude and latitude ϕ, θ and d_{obs}).

Priors

Now let us focus on the priors. For both the non-linear velocity dispersion and the effective Hubble constant, the model does not make any initial assumption and thus uses a uniform prior. The effective Hubble constant is limited between 0.5 and 1.5. It should be 1 if the assumed Hubble constant is compatible with the observations and nothing else is perturbing the data (e.g. no external flow, non-linear effect, ...). As the effective Hubble constant impacts the position of the observations and thus if they are within the simulated volume, the interval cannot be too large otherwise a large fraction of the observations will be discarded in some steps and could lead to a convergence towards discarding almost all the constraints. The variance of the linear velocity field is around (300 km/s)^2 and this value can be used as a first estimate of the non-linear counterpart. The interval for the velocity dispersion has been chosen to be between 50 and 2000 km/s.

For the linear velocities, they are directly related to the overdensity field with equation 9.7. In the linear regime, the primordial fluctuations are well approximated by a Gaussian, thus a Gaussian prior is well suited for the overdensity field but also for the velocities. As seen before, the typical correlation length of the velocity field is given by the correlation matrix Ψ that can be used as the standard deviation:

$$P(\mathbf{v}_L) = \prod_{ij} |2\pi\Psi|^{-1/2} \exp\left(-v_{L,\alpha}(\mathbf{r_i})\cdot\Psi_{\alpha,\beta}^{-1}\cdot v_{L,\beta}(\mathbf{r_j})\right) \tag{9.27}$$

where $\Psi_{\alpha,\beta} = \psi_T(r_{ij})\delta_{\alpha,\beta} + \left(\psi_T(r_{ij}) - \psi_R(r_{ij})\right)$ and the product is done over all the pair of points used for the discretization (Graziani et al., 2019).

The last prior needed is for the distances of the galaxies and should take into account the Malmquist bias. In order to explain the bias, let us suppose that we have a galaxy at the true distance r and observed distance modulus μ and let us look at the probability function:

$P(r|\mu) = \dfrac{P(\mu|r)P(r)}{P(\mu)}$. As we can see, if we wish to properly sample r, we need to carefully describe $P(r)$ and $P(\mu)$. If they are omitted, the so called Malmquist bias will impact the reconstruction. Three different sources exist that can be grouped into two linear sources (selection and volume effects) and a non-linear source (density effect). The two first sources impact the probability function even in the case of a uniform distribution of the galaxies. The volume effect impacts $P(r)$ due to the increase of volume and thus number of galaxies with the distance in spherical coordinates. The selection effect impacts $P(\mu)$ due to the galaxies at large distances that are not bright enough to be observed by the telescopes. Finally, the density effect is due to the presence of dense structure that makes the distribution of galaxies $P(r)$ non-uniform. Thus to come back to the prior, it should be close to 0 at a radius of 0 and infinity.

Following the approach of Graziani et al., 2019, 3 different models are used depending on the origin of the data. For the observations coming from 6dF a piecewise normal distribution is used:

$$P(d) = \frac{1}{\sqrt{2\pi}(p_2 + p_3)} \begin{cases} \exp\left(-\dfrac{1}{2}\dfrac{(d-p_1)^2}{p_2^2}\right) & \text{if } d \le p_1 \\ \exp\left(-\dfrac{1}{2}\dfrac{(d-p_1)^2}{p_3^2}\right) & \text{if } d > p_1 \end{cases} \tag{9.28}$$

where p_1, p_2 and p_3 are parameters that are fitted to the data (and independent of the other dataset). For the observations using the Tully-Fisher relation a power law is used:

$$P(d) = \frac{1}{N} d^{p_1} \exp\left(-\left(\frac{d}{p_2}\right)^{p_3}\right) \tag{9.29}$$

where N is a normalization coefficient computed numerically and depending on p_i. This prior is used for three different datasets, but each one uses its own fitted value of p_i. For the last dataset, as it regroups data coming from different telescopes and techniques, a uniform prior is used.

9.4.3 Gibbs Sampling

Let us start with a quick explanation of the Gibbs sampling. It is a Markov Chain Monte Carlo (MCMC) method. The Markov chain part describes a sequence of events that are dependent only on the previous one and the Monte Carlo describes a random process. It means that the Gibbs sampling will sample the probability distribution through a sequence of random evaluations depending only on the state of the previous step. It is done through the separation of a multivariable distribution in single variable distributions. The single variable distributions are sampled the one after the other until enough points are obtained. Let us assume that we wish to sample $p(x, y)$. The Gibbs sampling consists, for each step, to sample $p(x|y)$ followed by $p(y|x)$. The number of steps cannot be known a priori and requires an analysis of the correlation between the sampled points. In Graziani et al., 2019, they use about 20 times the correlation length for a total of 1400 steps. This sampling is really efficient for complex multivariable distributions with weakly correlated variables. In the case of strongly correlated variables, it is possible to improve the sampling by collapsing the probability function over the correlated variables (such as done for σ_{NL} and h_{eff} in the following section).

Gibbs Sampling of the Model

Using the previous priors and the likelihood, we can now compute the conditional probability for h_{eff}, σ_{NL}, $\mathbf{v}_L$ and d_{obs} that are used within the Gibbs sampling. For all of them, the computation is based on Equation 9.24. The probabilities are simply given by this equation where a single variable is sampled and all the others are fixed. Thus, one step of the Gibbs sampling consists then in:

- Sampling $P(h_{\mathrm{eff}} \mid \sigma_{\mathrm{NL}}, \mathbf{v}_L, d_{\mathrm{obs}})$, see below for the complete expression
- Sampling $P(\sigma_{\mathrm{NL}} \mid h_{\mathrm{eff}}, \mathbf{v}_L, d_{\mathrm{obs}})$
- Sampling $P(\mathbf{v}_L \mid h_{\mathrm{eff}}, \sigma_{\mathrm{NL}}, d_{\mathrm{obs}})$. This is equivalent to drawing a constrained realization with $\epsilon^2 = (1+\bar{z})^{-2} c^2 \sigma_z^2 + \sigma_{\mathrm{NL}}^2$ (Graziani et al., 2019). This sampling calls the code `MUSIC` in order to generate a random realization[6].
- Sampling $P(d_{\mathrm{obs}} \mid h_{\mathrm{eff}}, \sigma_{\mathrm{NL}}, \mathbf{v}_L)$
- Fitting the distance priors

The different probabilities presented before are simply given by 9.22. In the case of $P(h_{\mathrm{eff}} \mid \sigma_{\mathrm{NL}}, \mathbf{v}_L, d_{\mathrm{obs}})$, it means that $P(d_{\mathrm{obs}})$, $P(\sigma_{\mathrm{NL}})$ and $P(\mathbf{v}_L)$ are all constant (and can be discarded as we can easily normalize the distribution during the sampling). The same can be done for $P(h_{\mathrm{eff}})$ as this prior is uniform. Therefore $P(h_{\mathrm{eff}} \mid \sigma_{\mathrm{NL}}, \mathbf{v}_L, d_{\mathrm{obs}})$ is given by the likelihood. All the probabilities given previously for a step of the Gibbs sampling are given here:

$$P(h_{\mathrm{eff}} \mid \sigma_{\mathrm{NL}}, \mathbf{v}_L, d_{\mathrm{obs}}) \propto \mathcal{N}_\mu\left(5\log_{10}\frac{h_{\mathrm{eff}} d_{\mathrm{obs}}}{10\mathrm{pc}}\ ,\ \sigma_\mu^2\right) \mathcal{N}_{v_r}\left(\mathbf{v}_L(\mathbf{r})\cdot\hat{\mathbf{r}}\ ,\ \sigma_{cz}^2(1+\bar{z})^{-2} + \sigma_{\mathrm{NL}}^2\right), \tag{9.30}$$

$$P(\sigma_{\mathrm{NL}} \mid h_{\mathrm{eff}}, \mathbf{v}_L, d_{\mathrm{obs}}) \propto \mathcal{N}_{v_r}\left(\mathbf{v}_L(\mathbf{r})\cdot\hat{\mathbf{r}}\ ,\ \sigma_{cz}^2(1+\bar{z})^{-2} + \sigma_{\mathrm{NL}}^2\right), \tag{9.31}$$

$$P(d_{\mathrm{obs}} \mid h_{\mathrm{eff}}, \sigma_{\mathrm{NL}}, \mathbf{v}_L) \propto P(d)\mathcal{N}_\mu\left(5\log_{10}\frac{h_{\mathrm{eff}} d_{\mathrm{obs}}}{10\mathrm{pc}}\ ,\ \sigma_\mu^2\right) \mathcal{N}_{v_r}\left(\mathbf{v}_L(\mathbf{r})\cdot\hat{\mathbf{r}}\ ,\ \sigma_{cz}^2(1+\bar{z})^{-2} + \sigma_{\mathrm{NL}}^2\right). \tag{9.32}$$

The sampling of all the previous probabilities is repeated until enough non correlated steps have been done. In an effort to reduce the correlation length of the Markov chain, the probability of h_{eff} and σ_{NL} are collapsed against the linear velocity field due to the strong correlation between

---

[6] The random realization is done with my own version of `MUSIC` in order to output the overdensity.

the variables:

$$P(h_{\mathrm{eff}} \mid \sigma_{\mathrm{NL}}, d_{\mathrm{obs}}) = \int P(h_{\mathrm{eff}} \mid \sigma_{\mathrm{NL}}, \mathbf{v}_L, d_{\mathrm{obs}}) \mathrm{d}\mathbf{v}_L \tag{9.33}$$

$$= \mathcal{N}_\mu(\cdots) \int \mathcal{N}_{v_r}(\cdots) \mathrm{d}\mathbf{v}_L = \mathcal{N}_\mu(\cdots) N_{v_r}(\mathbf{0}, C) \tag{9.34}$$

where C is the correlation defined in Equation 9.11 with $\epsilon = \sigma_{cz}^2(1+\bar{z})^{-2} + \sigma_{\mathrm{NL}}^2$. It is worth to mention that C includes the correlation between all the observations and not simply a single pair of galaxies i, j. Therefore, its inversion is computationally expensive. The same can be done with σ_{NL} and the only difference is the missing probability for μ as it is a known value during the evaluation. With this technique, the correlation length of the Markov Chain is roughly 100 steps (or 100 sampling of each variables). Here I have a small difference with Graziani et al., 2019 as they do not include the probability function $P(\mu \mid d_{\mathrm{obs}}, h_{\mathrm{eff}})$. According to R. Graziani (private discussion), this is only a question of considering a sampling with respect to μ or d_{obs}. From my tests, I noticed no impact from this difference.

Collapsing the probabilities introduces an additional inversion of the correlation matrix. As usual for a 1D distribution, the sampling of such probabilities is done through an interpolation of the distributions. Both distributions are strongly peaked, thus a uniform mesh is strongly inefficient for the interpolation. A possibility is to use the Brent method to find the peak and then sample from it. The technique is fully described in the Appendix VI. In the near future, a different type of sampling than the Gibbs one will be applied and should avoid this heavy computation. This technique is called the Hamiltonian sampling and can be seen as a sampling through a dynamical evolution comparable to solving a Hamiltonian system (private conversation with R. Graziani).

Once enough steps are done (above ~ 1500), all the steps are analyzed and the parameters of the most probable one are selected as the correct candidates. In Graziani et al., 2019, they are using the average of all the solutions for the linear field. While this method is providing an excellent representation of the large scale structures, it also destroys part of the power spectrum and thus is not a correct approach for the initial conditions.

9.4.4 Constrained Realization at z=0

In Figure 9.6 and 9.7, an example of constrained realization at z=0 done with the method presented before. In the first figure, the overdensity is shown for the best step. The dots represent the observations and are colored according to their peculiar velocities. The galaxies moving towards (away from) us are in blue (red) and the galaxies that do not exhibit a strong direction in green. As mentioned before, in Graziani et al., 2019, they are not interested in producing initial conditions and thus do not conserve the power spectrum. Therefore, this figure is not comparable to the one in the previous paper. In the second figure, the standard deviation for the velocities of all the realizations is shown and represents the strength of the constraints. In the center, where many galaxies are observed, the initial conditions are well constrained by the observations, but,

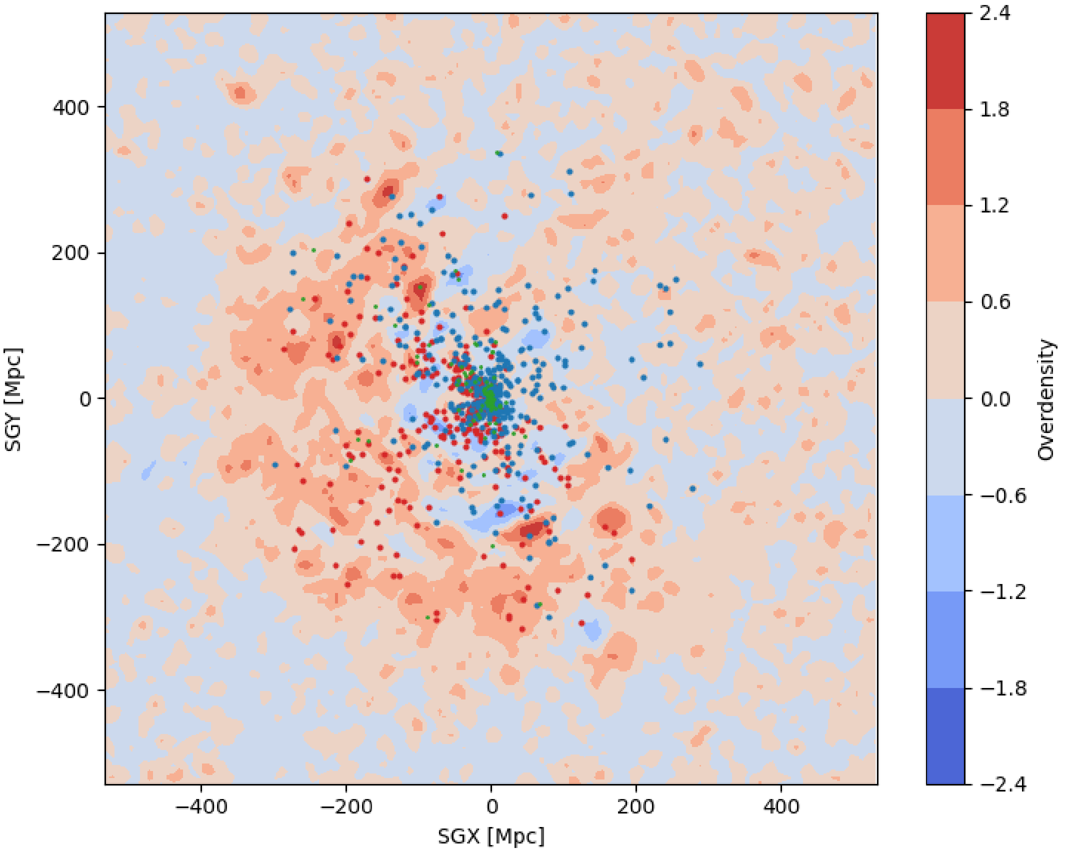


Figure 9.6 – Example of constrained realization at z=0 along the Supergalactic plane XY. The overdensity computed with the WF/CR is shown in the background and the galaxies taken from CF-3 are plotted on top of it. The colors represent if a galaxy is moving away from (red), towards (blue) us or do not show a strong direction (green). Some areas are below an overdensity of -1 and indicates the presence of a bug within the code that I am currently tracking.

at large distances, they are constrained only by the power spectrum.

9.5 Reverse Zel'dovich Approximation

The constrained realizations (CR) generated by the Gibbs sampling provide a realization at z=0, and therefore they need to be rewound to higher redshift. Doumler et al., 2013 developed a technique to do so using the Reverse Zel'dovich Approximation:

$$x_{\text{init}} = r - \frac{v}{H_0 f} \tag{9.35}$$

where x is the position at high redshift and r, v are the position and linear velocity at z=0. The idea is to compute a first CR, use the linear velocity field to rewind the position of the observed

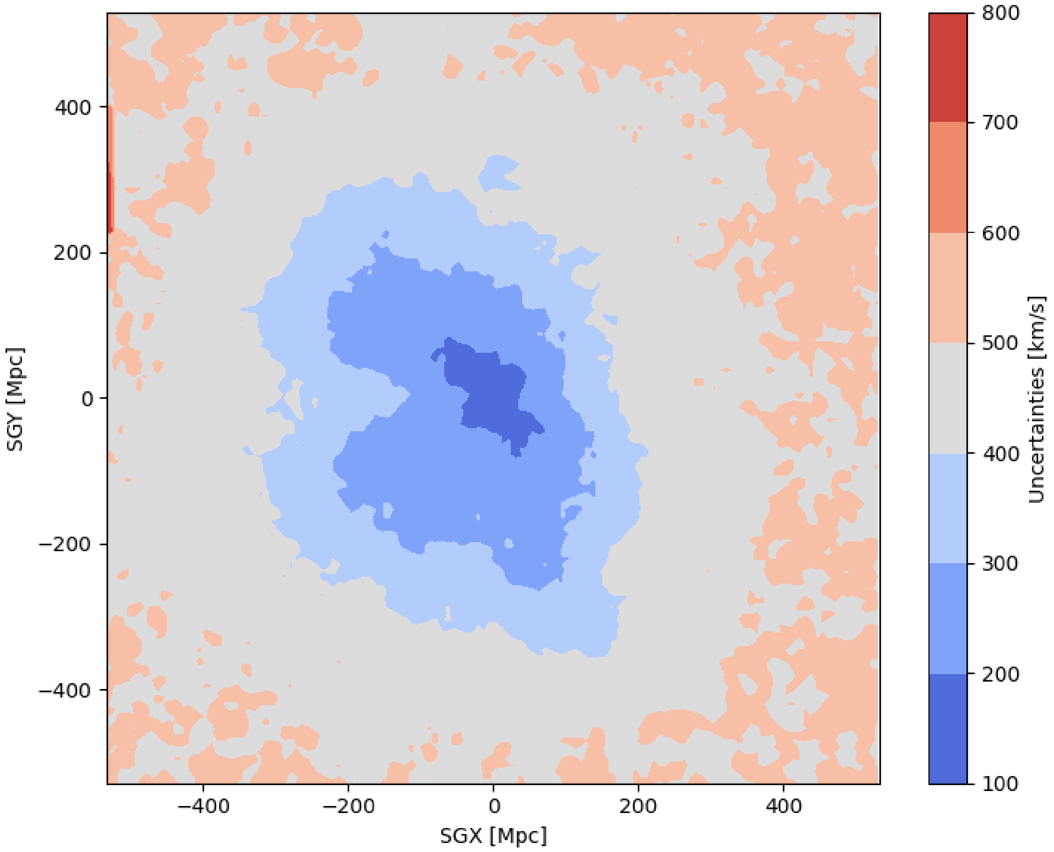


Figure 9.7 – Strength of the constraints for Figure 9.6 as function of the position. At the center, where a large quantity of observations exist, the initial conditions are well constrained, but at larger radius, they are only constrained by the power spectrum.

galaxies to their approximate position at high redshift (without any modification of the velocity). From them, a final CR (without any Gibbs sampling) is computed in order to provide the ICs. While this technique works, it has been further improved by Sorce et al., 2014. Their idea is to follow the same approach but, for the second CR, they use the full 3D velocity field computed from the first and not just the radial component taken from the observations. As the linear velocity field is curl free and thus should be completely defined by a scalar such as the radial velocity, the first approach should be enough. But as any observations contain some uncertainties, the method presented in Doumler et al., 2013 is not describing such field and is not sufficient to represent the full 3D field. Therefore, the second approach compensates this error by keeping the 3D velocities computed from the previous CR that are guaranteed to be curl free. It is worth to mention that in this step, the velocities are considered exact and no errors are added to the correlation matrix. As it was shown in Sorce et al., 2014, the second technique provides a higher precision and reliability and thus was chosen in my code. Unfortunately, this approach multiplies by 3 the number of constraints and requires a larger matrix inversion that does not fit on current GPUs. This computation is done on CPUs and thus is currently the slowest part of my code.

As mentioned earlier, `MUSIC` can read the random numbers from a file. From the constrained realization at high redshift and the power spectrum, the numbers are computed using equation 9.1. In order to ensure that we are respecting the condition derived in section 9.1, the random numbers are modified in order to obtain a null average and a standard deviation of 1. Once the random number file generated, zoom initial conditions can be generated with a large scale field corresponding to the constrained realization along with the small scales defined with the seed parameters in `MUSIC`. Therefore the heavy computations done with the Bayesian approach and rewinding are required only once and can then be used to generate a large quantity of different zoom ICs. The complete method implemented is given by the following steps:

- Generating the linear velocity field $\mathbf{v_L}$ from the observations v_r at position $\mathbf{r}$ with the Gibbs samplign,
- Interpolating $\mathbf{v_L}$ at $\mathbf{r}$,
- Computing the position of the observations at high redshift ($\mathbf{r^z}$) with the reverse Zel'dovich approximation (Equation 9.35) based on $\mathbf{v_L}$,
- Interpolating the velocities at high redshift ($\mathbf{v_r^z}$ here in 3D) from $\mathbf{v_L}$ at position $\mathbf{r^z}$,
- Computing the linear velocity field at high redshift ($\mathbf{v_L^z}$) with the WF/CR (Equation 9.8) from $\mathbf{v_r^z}$ and $\mathbf{r^z}$ assuming no uncertainties,
- Writing $\mathbf{v_L^z}$ as random numbers for `MUSIC` (Equation 9.1).

To conclude this chapter, the method described is able to generate initial conditions for cosmological simulations. To fully show it, it would require to run a cosmological simulation, extract the galaxies and use them as constraints for a second simulation. By comparing the large scale structures between the two simulations, the quality of the constrained initial conditions could be asserted. As the code still contains an error, this was not possible in time for this thesis. Anyway, thanks to the constraints from a large quantity of observed galaxies, the large scale environment of the Milky Way can be included within the ICs. A complete summary of the required steps is provided in Figure 9.8. The method consists in producing a realization constrained with the galaxies' positions and velocities at redshift 0 (left side of the graph), and then the realization is rewound to higher redshift in order to obtain the initial conditions (right side of the graph). From a single constrained realization at high redshift, a large set of zoom simulations at high resolution can be created with the same large scale environment but different small scales.

Finally, I would like to thank Romain Graziani for his help to understand the details of the method and thus the development of my code.

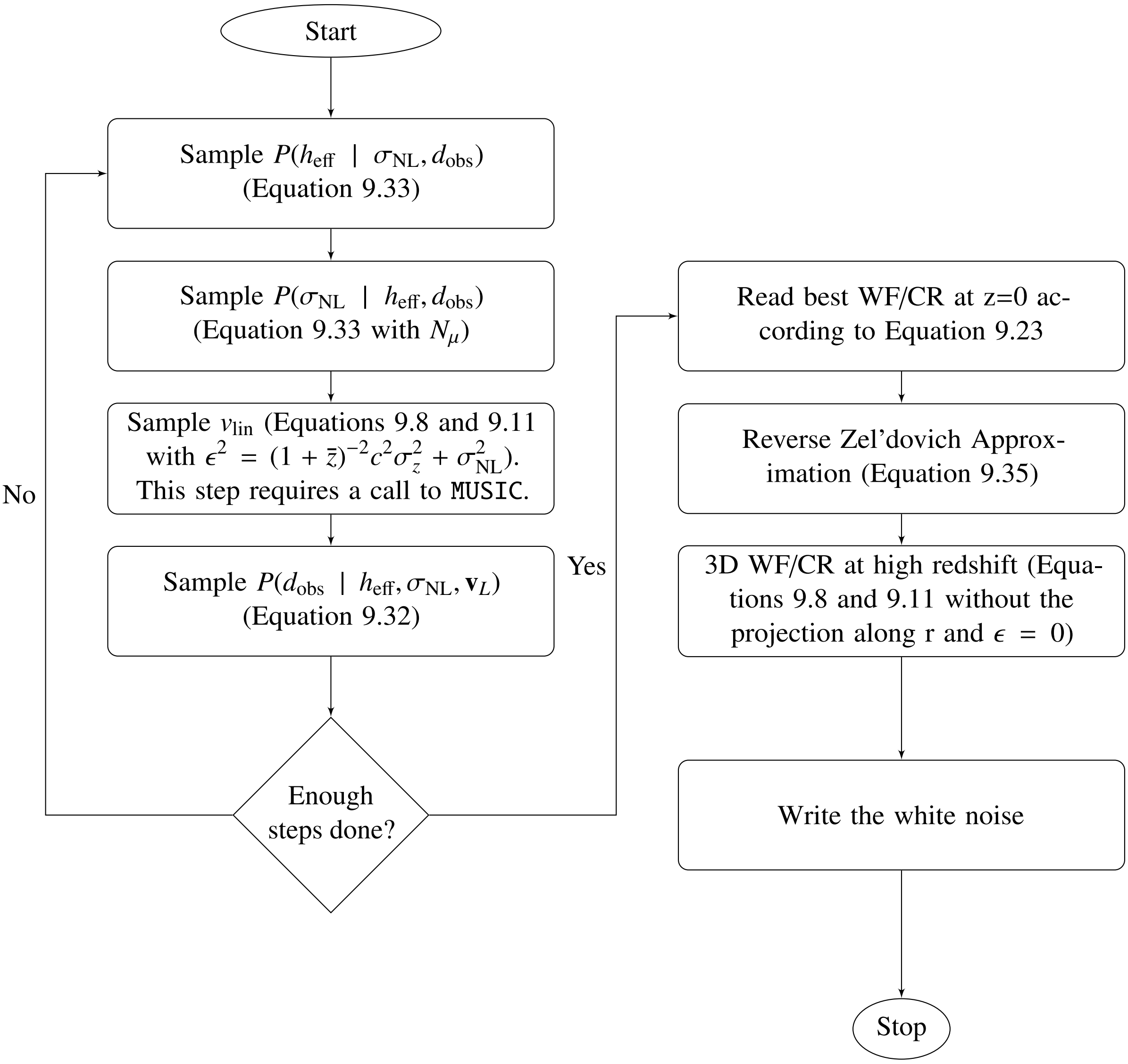


Figure 9.8 – Summary of the steps required for the generation of initial conditions. The left part corresponds to the Gibbs sampling and the right part is rewinding the best initial conditions to higher redshift. The white noise can be provided to `MUSIC` in order to generate the final ICs.

Conclusion Part V

10 Summary, Conclusion and Outlook

Don’t adventures ever have an end? I suppose not. Someone else always has to carry on the story.
— J. R. R. Tolkien, The Fellowship of the Ring

I conclude this thesis by providing a summary of my work. Then I discuss its impacts on the future of the group `SWIFT-GEAR`. Finally, I finish with an overview on the future of numerical astrophysics related to my work.

10.1 Summary and Conclusion

The first part of my work consisted in results obtained with the code `GEAR`. My main contribution to astrophysics has been through the study of ram pressure (RP) stripping using the moving box technique. This technique simulates both the tidal forces and hydrodynamics interactions between an evolving host and one of its dwarf satellite, at high resolution. Using initial conditions from cosmological simulations, I have shown that the thermal pressure plays an important role in the efficiency of the RP and UV background (UVB). I have described how the hot halo, at high temperature, impacts directly the stripping of the cold gas. This stripping becomes inefficient and the quenching of dwarfs is only done through starvation from the reserve of hot gas. The thermal pressure also compresses the cold gas, thus enables it to become optically thick through the H shielding and far more resistant to the UVB. As the increase in density and quantity of cold gas directly impact the star formation, the role of thermal pressure can be seen with a boost in the star formation.

My second contribution made with `GEAR` was the collaboration with the `AGORA` group. Our aim is to compare simulations produced by different codes. The objective of the comparisons is to try to improve the reproducibility of the simulations and understand the impact of the different implementations (e.g. AMR vs SPH). In this project, we produced the first set of cosmological simulations including a treatment of baryonic physics and stars within the `AGORA` project. The simulations will be the basis of a few different papers in the near future. While the first publication presented the calibration process and the first images of the simulations, it also hinted to large differences in metallicity within the circumgalactic and intergalactic medium that could help to constrain the stellar feedback models.

The second part of my thesis consisted in replacing `GEAR` with the new code named `SWIFT`. This is an important contribution to numerical astrophysics and it resulted in a full implementation of `GEAR`'s model and the addition of some missing modules in `SWIFT` (e.g. stellar feedback tasks). Through a comparison with the results of Revaz and Jablonka, 2018, I have shown that the code reproduces the simulations of `GEAR`. As `SWIFT` aims towards reaching the exascale simulations, it uses a large number of optimizations that are not included in `GEAR` making it far more efficient on large simulations (factor of 7.65 for our simulations). As in the case of `GADGET-2`, the code was initially designed for large and uniform volumes and, in a second time, started to be optimized for zoom simulations. Therefore, we can expect to see a larger speedup in the years to come.

In the last part of this thesis, I have presented my most recent and ongoing works. Firstly, another important contribution to numerical astrophysics has been through the development of the Continuous Simulation Data Stream (CSDS). I have designed and implemented within `SWIFT` a new output system that could complement the snapshots. Through a clever usage of the individual time steps within the output, the disk space required is reduced by one order of magnitude for the same time accuracy than snapshots. Secondly, I have implemented the tasks required for a new star formation scheme that will allow `SWIFT-GEAR`'s group to move from a stellar population point of view towards an individual modelling of stars. While this model concerns mostly the stellar evolution and feedback, the hardest part to implement is the star formation as it requires to form a single star from multiple gas particles and not simply from a single gas particle, as before. This new scheme uses sink particles to group gas particles together and creates stars when enough mass is accumulated within the sink particles. Finally, I have developed a code producing initial conditions constrained with observations in order to reproduce the large scale structure of the Local Group. Such initial conditions increase the quality of comparisons between our environment and simulations.

10.2 Perspective for `SWIFT-GEAR`

Two different approaches will be explored in the future by the group `SWIFT-GEAR` and will both directly benefit from my work. The approaches will be the low resolution and the high resolution ones. The low resolution (e.g. comparable to Revaz and Jablonka, 2018 or slightly lower) approach will be on understanding the impact of the environment on dwarf galaxies. Simulations will be done within a fully cosmological context and will not require the usage of a technique such as the moving box to simulate the environment. It will allow to better explore the impact of the thermal pressure presented in this thesis.

As simulations will have a large number of particles, `GEAR` is not efficient enough to run them. The improved performances of my implementation within `SWIFT` will help to keep the computation time under reasonable limits. Moreover, thanks to `SWIFT`'s modularity, it will be possible to easily see the impact of different SPH flavors (e.g. using the energy diffusion implemented within `SPHENIX`). It could be also possible to see the impact of the host's UV through a treatment of radiative transfer. Finally, a last interesting topic is the impact of the host's active galactic nucleus (AGN) on the evolution of dwarfs. This could be easily done thanks to the implementation of AGN in `SWIFT` by the `EAGLE` team.

As we aim to see the impact of the environment and compare with observations done within the Local Group, its special environment should be taken into account for any comparisons. Obtaining such environment, without constrained initial conditions, is complex and generally requires a large number of simulations in order to obtain a single setup corresponding approximately to our environment. With my code, the large scale structure can be easily recovered and thus a smaller set of simulations will be required.

My last project contributing to this approach is `AGORA`. As the collaboration studies Milky Way like galaxies, it help us to understand how our models behave when applied to larger galaxies. Such galaxies have not been investigated for a long time with `GEAR` and will be present in the low resolution simulations. With the upcoming comparisons, this project will tell us if our models are correctly reproducing larger galaxies or if their design is too focused on dwarf galaxies and should be improved.

In opposition, the high resolution approach will focus on ultra faint dwarf galaxies and requires, obviously, higher resolution than those of current simulations. Therefore a new star formation and stellar model are needed as the stellar population approximation is not sufficient at such resolution. The new models will be based on my sink particles. As before, this increase in resolution will require more particles and thus a better scaling than `GEAR`. This approach will allow the examination of the low end of the galaxy luminosity distribution function and of the missing satellites on its most troublesome side. It will also increase our knowledge on the production of elements and their diffusion (e.g. first stars or r-process). With this increased resolution, the supernovae explosions will also be resolved sufficiently to reduce the impact of the metal and energy deposition scheme and should produce results that are less dependent on subgrid models. Finally, my work on the CSDS will enable our group to follow more accurately the evolution of the fastest phenomena without excessively increasing the output size for both the low and high resolution approaches.

10.3 Perspective for Numerical Astrophysics

At larger scale than our research group, the domain of numerical astrophysics will see many improvements in the next decade. My contribution that will impact the largest community, for many years to come, is my work within `SWIFT`. Through the collaboration with the `SWIFT` team, we have been able to implement a code that will achieve the required performances, on the software side, for the exascale simulation. Indeed, as shown in Borrow et al., 2020, `SWIFT` is 36 times faster than `GADGET`-2 and is slightly faster than the required performances [1]. While the code has been mostly implemented with the idea of reproducing the EAGLE simulation at the same resolution but with a volume 27x larger, the code already has good performances on zoom simulations. In the next years, we can expect a stronger focus on zoom simulations that will increase our performances as it has been the case between `GADGET`-2 and `GADGET`-3.

In the case of hydrodynamics, `SWIFT` already contains a large set of schemes based on either the meshless method or the traditional SPH. Through its modular approach, `SWIFT` encourages the implementation of new methods. Among the new possible methods, the moving mesh has always been of particular interest but was only implemented in 1 and 2 dimensions within `SWIFT`. Some early tries in 3 dimensions have been done but required far too much memory, in the current architecture, for the generation of the mesh. Recently, additional work has been made to the 3D moving mesh through a new design of the memory management for the mesh. Thus `SWIFT` might propose soon a moving mesh model and becomes the first code, to my knowledge, to have simultaneously SPH, meshless and moving mesh methods. Furthermore, comparison between the different methods have been made, but never all together (SPH, moving mesh and meshless) within the exact same code and operations. Such studies will certainly improve our understanding of hydrodynamics in a cosmological context.

The last important upgrade of `SWIFT` will come from the implementation of the radiative transfer within the task system. This will allow the addition of stellar radiations. It will impact the gas through both the ionization and the radiation pressure. This additional heating of the gas will certainly reduce the impact of the delayed cooling and give a more realistic model for the feedback.

With the additional performances of the new codes (e.g. `SWIFT`, `CHANGA` or the future version `RAMSES`), more and more simulations will use a radiative cooling without the simplification of equilibrium as done in Lupi, 2019. This assumption especially breaks around stars. Removing it will allow a better description of supernovae and star forming gas. By combining both the non equilibrium radiative cooling and the radiative transfer, it will also be possible to have a better description of the self shielding that again will improve our simulations around stars. On the analysis side, this improvement will allow to directly compare the full sky map of the 21 cm line from SKA with maps of HI from simulations.

As mentioned earlier, moving towards higher resolution will render the smoothed metal inefficient

---

[1] Assuming an equal speedup from software and hardware, it corresponds to $\sqrt{1000} \sim 32$

at propagating the metals at large scale and will require the implementation of the diffusion equation for the metals in SPH. Some numerical solutions exist already but are not yet sufficient for cosmological simulations due to the extremely small time steps required (Shen et al., 2010) or the stability (Greif et al., 2009). Theses methods will certainly receive improvements from the community soon to make them more stable and efficient. This diffusion coupled with a resolution at the individual star level will produce chemo-dynamical simulations at an unprecedented level of accuracy. It means that the details of the chemical enrichment from the first stars and the r-process will be resolved enough to fully understand how they impact the next generation of stars and the gas. The modeling of supernovae will also be improved in this model as the individual stars will have their own positions and can create even more complex features in the interstellar medium through their explosion at different localization and thus densities. All the previous improvements will reduce the impact of the subgrid models and should increase our trust into the simulations.

The generation of initial conditions is a domain that will have a small improvement too. As the growth of cosmological structures is really sensitive to any type of perturbations, the initial conditions should have as little perturbations as possible. `MUSIC` relies on a mesh to construct them and places the particles on it (other codes also usually relies on this approach), and thus creates preferential directions for gravity. This can bias the final state of a simulation. A solution to this issue is to use a glass configuration where all the particles are approximately at the same distance to each others (Sirko, 2005). Such configurations can be easily obtained by reversing the sign of gravity and evolving a system until reaching an equilibrium. According to my sources, `MUSIC` is currently under development to include this new feature.

With the end of the last calibration step within the `AGORA` collaboration, a regain of motivation around the project has been clearly seen with the proposal of new ideas that will certainly enhance the quality of our simulations. The first publication using our calibration process will certainly study the circumgalactic medium but other projects are already in the minds of the `AGORA` participants. For example, we discussed about comparing shocks, dwarf galaxies, satellites, outflows or even black hole accretion. The production of comparable simulations was a hard but necessary work that sets the ground to new interesting studies. I am convinced that my work will result in the `AGORA` community publishing in a more regular basis.

Bibliography

Abadi, M. G. et al. (July 2003). “Simulations of Galaxy Formation in a Λ Cold Dark Matter Universe. I. Dynamical and Photometric Properties of a Simulated Disk Galaxy”. In: *The Astrophysical Journal* 591, pp. 499–514. DOI: 10.1086/375512.

Abbasi, H. et al. (Aug. 2009). “Extending I/O through high performance data services”. In: *2009 IEEE International Conference on Cluster Computing and Workshops*. ISSN: 2168-9253, pp. 1–10. DOI: 10.1109/CLUSTR.2009.5289167.

Abbott, B. P. et al. (Oct. 2017). “Multi-messenger Observations of a Binary Neutron Star Merger”. In: *The Astrophysical Journal* 848. ADS Bibcode: 2017ApJ...848L..12A, p. L12. DOI: 10.3847/2041-8213/aa91c9.

Abel, T. et al. (Aug. 1997). “Modeling primordial gas in numerical cosmology”. In: *New Astronomy* 2.3, pp. 181–207. DOI: 10.1016/S1384-1076(97)00010-9.

Adamek, J. et al. (Apr. 2016). “General relativity and cosmic structure formation”. en. In: *Nature Physics* 12.4. Number: 4 Publisher: Nature Publishing Group, pp. 346–349. DOI: 10.1038/nphys3673.

Agertz, O. et al. (Sept. 2007). “Fundamental differences between SPH and grid methods”. In: *Monthly Notices of the Royal Astronomical Society* 380, pp. 963–978. DOI: 10.1111/j.1365-2966.2007.12183.x.

Amaro-Seoane, P. et al. (May 2013). “eLISA: Astrophysics and cosmology in the millihertz regime”. In: *GW Notes, Vol. 6, p. 4-110* 6, pp. 4–110.

Aoki, W. et al. (Aug. 2009). “Chemical composition of extremely metal-poor stars in the Sextans dwarf spheroidal galaxy”. en. In: *Astronomy and Astrophysics, Volume 502, Issue 2, 2009, pp.569-578* 502.2, p. 569. DOI: 10.1051/0004-6361/200911959.

Aoyama, S. et al. (Aug. 2018). “Cosmological simulation with dust formation and destruction”. In: *Monthly Notices of the Royal Astronomical Society* 478, pp. 4905–4921. DOI: 10.1093/mnras/sty1431.

Applebaum, E. et al. (Feb. 2020). “A stochastically sampled IMF alters the stellar content of simulated dwarf galaxies”. In: *Monthly Notices of the Royal Astronomical Society* 492, pp. 8–21. DOI: 10.1093/mnras/stz3331.

Applebaum, E. et al. (Jan. 2021). “Ultrafaint Dwarfs in a Milky Way Context: Introducing the Mint Condition DC Justice League Simulations”. In: *The Astrophysical Journal* 906. ADS Bibcode: 2021ApJ...906...96A, p. 96. DOI: 10.3847/1538-4357/abcafa.

Arnett, W. D. and A. G. W. Cameron (1967). "Supernova hydrodynamics and nucleosynthesis". In: *Canadian Journal of Physics* 45, p. 2953. doi: 10.1139/p67-243.

Arnett, W. D. et al. (1989). "Supernova 1987A". In: *Annual Review of Astronomy and Astrophysics* 27, pp. 629–700. doi: 10.1146/annurev.aa.27.090189.003213.

Arraki, K. S. et al. (Feb. 2014). "Effects of baryon removal on the structure of dwarf spheroidal galaxies". In: *Monthly Notices of the Royal Astronomical Society* 438, pp. 1466–1482. doi: 10.1093/mnras/stt2279.

Aubert, D. and R. Teyssier (2010). "Reionization Simulations Powered by Graphics Processing Units. I. On the Structure of the Ultraviolet Radiation Field". en. In: *The Astrophysical Journal* 724.1, p. 244. doi: 10.1088/0004-637X/724/1/244.

Aubert, D. et al. (Apr. 2018). "The Inhomogeneous Reionization Times of Present-day Galaxies". In: *The Astrophysical Journal* 856, p. L22. doi: 10.3847/2041-8213/aab14d.

Bagla, J. S. and S. Ray (Sept. 2003). "Performance characteristics of TreePM codes". en. In: *New Astronomy* 8.7, p. 665. doi: 10.1016/S1384-1076(03)00056-3.

Bahé, Y. M. et al. (Aug. 2012). "The competition between confinement and ram pressure and its implications for galaxies in groups and clusters". In: *Monthly Notices of the Royal Astronomical Society* 424. ADS Bibcode: 2012MNRAS.424.1179B, pp. 1179–1186. doi: 10.1111/j.1365-2966.2012.21292.x.

Bahl, H. and H. Baumgardt (Mar. 2014). "A comparison of the distribution of satellite galaxies around Andromeda and the results of ΛCDM simulations". In: *Monthly Notices of the Royal Astronomical Society* 438, pp. 2916–2923. doi: 10.1093/mnras/stt2399.

Balsara, D. S. (1995). "von Neumann stability analysis of smooth particle hydrodynamics–suggestions for optimal algorithms". In: *Journal of Computational Physics* 121, pp. 357–372. doi: 10.1016/S0021-9991(95)90221-X.

Barbosa, C. E. et al. (Jan. 2021). "What does (not) drive the variation of the low-mass end of the stellar initial mass function of early-type galaxies". In: *Astronomy and Astrophysics* 645, p. L1. doi: 10.1051/0004-6361/202039810.

Bate, M. R., I. A. Bonnell, and N. M. Price (Nov. 1995). "Modelling accretion in protobinary systems". In: *Monthly Notices of the Royal Astronomical Society* 277, pp. 362–376. doi: 10.1093/mnras/277.2.362.

Battaner, E. and E. Florido (2000). "The Rotation Curve of Spiral Galaxies and its Cosmological Implications". In: *Fundamentals of Cosmic Physics* 21, pp. 1–154.

Bauer, A. and V. Springel (July 2012). "Subsonic turbulence in smoothed particle hydrodynamics and moving-mesh simulations". In: *Monthly Notices of the Royal Astronomical Society* 423, pp. 2558–2578. doi: 10.1111/j.1365-2966.2012.21058.x.

Belyanin, A. A., V. V. Kocharovsky, and V. V. Kocharovsky (Dec. 1996). "Gamma-ray bursts from the final stage of primordial black hole evaporation". In: *Monthly Notices of the Royal Astronomical Society* 283, p. 626. doi: 10.1093/mnras/283.2.626.

Benson, A. J. et al. (June 2002). "The effects of photoionization on galaxy formation - I. Model and results at z=0". In: *Monthly Notices of the Royal Astronomical Society* 333, pp. 156–176. doi: 10.1046/j.1365-8711.2002.05387.x.

Bertschinger, E. (1998). “Simulations of Structure Formation in the Universe”. In: *Annual Review of Astronomy and Astrophysics* 36, pp. 599–654. doi: 10.1146/annurev.astro.36.1.599.

Bertschinger, E. (Nov. 2001). “Multiscale Gaussian Random Fields and Their Application to Cosmological Simulations”. In: *The Astrophysical Journal Supplement Series* 137. ADS Bibcode: 2001ApJS..137....1B, pp. 1–20. doi: 10.1086/322526.

Bigiel, F. et al. (Nov. 2008). “THE STAR FORMATION LAW IN NEARBY GALAXIES ON SUB-KPC SCALES”. en. In: *The Astronomical Journal* 136.6. Publisher: IOP Publishing, pp. 2846–2871. doi: 10.1088/0004-6256/136/6/2846.

Binney, J. and S. Tremaine (1987). “Galactic dynamics”. In: *Princeton, NJ, Princeton University Press, 1987, 747 p.*

Bland-Hawthorn, J., R. Sutherland, and D. Webster (July 2015). “Ultrafaint Dwarf Galaxies—the Lowest-mass Relics from Before Reionization”. In: *The Astrophysical Journal* 807, p. 154. doi: 10.1088/0004-637X/807/2/154.

Bleuler, A. and R. Teyssier (Dec. 2014). “Towards a more realistic sink particle algorithm for the RAMSES CODE”. In: *Monthly Notices of the Royal Astronomical Society* 445, pp. 4015–4036. doi: 10.1093/mnras/stu2005.

Blok, W. J. G. de et al. (Dec. 2008). “High-Resolution Rotation Curves and Galaxy Mass Models from THINGS”. In: *The Astronomical Journal* 136, pp. 2648–2719. doi: 10.1088/0004-6256/136/6/2648.

Boer, T. J. L. de et al. (Aug. 2012a). “The star formation and chemical evolution history of the Fornax dwarf spheroidal galaxy”. en. In: *Astronomy and Astrophysics* 544, A73. doi: 10.1051/0004-6361/201219547.

Boer, T. J. L. de et al. (Mar. 2012b). “The star formation and chemical evolution history of the sculptor dwarf spheroidal galaxyâ”. In: *Astronomy and Astrophysics* 539, A103. doi: 10.1051/0004-6361/201118378.

Boer, T. J. L. de et al. (Dec. 2014). “The episodic star formation history of the Carina dwarf spheroidal galaxy”. en. In: *Astronomy and Astrophysics* 572, A10. doi: 10.1051/0004-6361/201424119.

Bonifacio, P. et al. (Apr. 2018). “TOPoS. IV. Chemical abundances from high-resolution observations of seven extremely metal-poor stars”. In: *Astronomy and Astrophysics* 612, A65. doi: 10.1051/0004-6361/201732320.

Bonvin, V. et al. (Mar. 2017). “H0LiCOW - V. New COSMOGRAIL time delays of HE 0435-1223: H0 to 3.8 per cent precision from strong lensing in a flat ΛCDM model”. In: *Monthly Notices of the Royal Astronomical Society* 465, pp. 4914–4930. doi: 10.1093/mnras/stw3006.

Borrow, J. et al. (July 2018). “SWIFT: Maintaining weak-scalability with a dynamic range of 10^4 in time-step size to harness extreme adaptivity”. In: *arXiv e-prints* 1807, arXiv:1807.01341.

Borrow, J. et al. (Dec. 2020). “Sphenix: Smoothed Particle Hydrodynamics for the next generation of galaxy formation simulations”. In: *arXiv:2012.03974 [astro-ph]*. arXiv: 2012.03974.

Bovill, M. S. and M. Ricotti (Mar. 2009). “Pre-Reionization Fossils, Ultra-Faint Dwarfs, and the Missing Galactic Satellite Problem”. In: *The Astrophysical Journal* 693, pp. 1859–1870. doi: 10.1088/0004-637X/693/2/1859.

Boyarsky, A., O. Ruchayskiy, and M. Shaposhnikov (2009). “The Role of Sterile Neutrinos in Cosmology and Astrophysics”. In: *Annual Review of Nuclear and Particle Science* 59.1. _eprint: https://doi.org/10.1146/annurev.nucl.010909.083654, pp. 191–214. doi: 10.1146/annurev.nucl.010909.083654.

Bromm, V., P. S. Coppi, and R. B. Larson (Jan. 2002). “The Formation of the First Stars. I. The Primordial Star-forming Cloud”. In: *The Astrophysical Journal* 564, pp. 23–51. doi: 10.1086/323947.

Brooks, A. M. and A. Zolotov (May 2014). “Why Baryons Matter: The Kinematics of Dwarf Spheroidal Satellites”. In: *The Astrophysical Journal* 786. ADS Bibcode: 2014ApJ...786...87B, p. 87. doi: 10.1088/0004-637X/786/2/87.

Brooks, A. M. et al. (Mar. 2013). “A Baryonic Solution to the Missing Satellites Problem”. In: *The Astrophysical Journal* 765, p. 22. doi: 10.1088/0004-637X/765/1/22.

Brown, T. M. et al. (July 2012). “The Primeval Populations of the Ultra-faint Dwarf Galaxies”. In: *The Astrophysical Journal* 753. ADS Bibcode: 2012ApJ...753L..21B, p. L21. doi: 10.1088/2041-8205/753/1/L21.

Buck, T., A. A. Dutton, and A. V. Macciò (Aug. 2016). “Simulated ΛCDM analogues of the thin plane of satellites around the Andromeda galaxy are not kinematically coherent structures”. In: *Monthly Notices of the Royal Astronomical Society* 460, pp. 4348–4365. doi: 10.1093/mnras/stw1232.

Bullock, J. S. and M. Boylan-Kolchin (Aug. 2017). “Small-Scale Challenges to the ΛCDM Paradigm”. In: *Annual Review of Astronomy and Astrophysics* 55, pp. 343–387. doi: 10.1146/annurev-astro-091916-055313.

Bullock, J. S., A. V. Kravtsov, and D. H. Weinberg (Aug. 2000). “Reionization and the Abundance of Galactic Satellites”. In: *The Astrophysical Journal* 539, pp. 517–521. doi: 10.1086/309279.

Carlberg, R. G. (Nov. 2009). “Star Stream Folding by Dark Galactic Subhalos”. In: *The Astrophysical Journal Letters* 705, pp. L223–L226. doi: 10.1088/0004-637X/705/2/L223.

Carr, B. and F. Kühnel (2020). “Primordial Black Holes as Dark Matter: Recent Developments”. In: *Annual Review of Nuclear and Particle Science* 70.1. _eprint: https://doi.org/10.1146/annurev-nucl-050520-125911, pp. 355–394. doi: 10.1146/annurev-nucl-050520-125911.

Cescutti, G. and C. Chiappini (Jan. 2013). *Galactic chemical evolution: The role of the first stars*. Tech. rep. Publication Title: arXiv e-prints ADS Bibcode: 2013arXiv1301.1908C Type: article.

Chan, J. H. H. et al. (Aug. 2018). “How do stars affect ψDM haloes?” In: *Monthly Notices of the Royal Astronomical Society* 478, pp. 2686–2699. doi: 10.1093/mnras/sty900.

Chan, T. K. et al. (Sept. 2019). “Cosmic ray feedback in the FIRE simulations: constraining cosmic ray propagation with GeV γ-ray emission”. In: *Monthly Notices of the Royal Astronomical Society* 488, pp. 3716–3744. doi: 10.1093/mnras/stz1895.

Cloet-Osselaer, A. et al. (Aug. 2014). “Numerical simulations of dwarf galaxy merger trees”. In: *Monthly Notices of the Royal Astronomical Society* 442. ADS Bibcode: 2014MNRAS.442.2909C, pp. 2909–2925. doi: 10.1093/mnras/stu1071.

Cohen, J. G. and W. Huang (Aug. 2009). “The Chemical Evolution of the Draco Dwarf Spheroidal Galaxy”. In: *The Astrophysical Journal* 701, pp. 1053–1075. DOI: 10.1088/0004-637X/701/2/1053.

Collaboration, P. et al. (Sept. 2020). “Planck 2018 results. VI. Cosmological parameters”. en. In: *Astronomy & Astrophysics, Volume 641, id.A6, <NUMPAGES>67</NUMPAGES> pp.* 641, A6. DOI: 10.1051/0004-6361/201833910.

Corbelli, E. and P. Salucci (Jan. 2000). “The extended rotation curve and the dark matter halo of M33”. In: *Monthly Notices of the Royal Astronomical Society* 311, pp. 441–447. DOI: 10.1046/j.1365-8711.2000.03075.x.

Cormier, D. et al. (Apr. 2014). “The molecular gas reservoir of 6 low-metallicity galaxies from the Herschel Dwarf Galaxy Survey. A ground-based follow-up survey of CO(1-0), CO(2-1), and CO(3-2)”. In: *Astronomy and Astrophysics* 564, A121. DOI: 10.1051/0004-6361/201322096.

Creasey, P. et al. (June 2017). “Spreading out and staying sharp - creating diverse rotation curves via baryonic and self-interaction effects”. In: *Monthly Notices of the Royal Astronomical Society* 468, pp. 2283–2295. DOI: 10.1093/mnras/stx522.

Cullen, L. and W. Dehnen (Oct. 2010). “Inviscid smoothed particle hydrodynamics”. In: *Monthly Notices of the Royal Astronomical Society* 408, pp. 669–683. DOI: 10.1111/j.1365-2966.2010.17158.x.

Dalla Vecchia, C. and J. Schaye (July 2008). “Simulating galactic outflows with kinetic supernova feedback”. In: *Monthly Notices of the Royal Astronomical Society* 387, pp. 1431–1444. DOI: 10.1111/j.1365-2966.2008.13322.x.

Dashyan, G. et al. (Aug. 2019). “AGN-driven quenching of satellite galaxies”. In: *Monthly Notices of the Royal Astronomical Society* 487, pp. 5889–5901. DOI: 10.1093/mnras/stz1697.

Davé, R. et al. (June 2019). “SIMBA: Cosmological simulations with black hole growth and feedback”. In: *Monthly Notices of the Royal Astronomical Society* 486, pp. 2827–2849. DOI: 10.1093/mnras/stz937.

Dehnen, W. (June 2001). “Towards optimal softening in three-dimensional N-body codes - I. Minimizing the force error”. In: *Monthly Notices of the Royal Astronomical Society* 324, pp. 273–291. DOI: 10.1046/j.1365-8711.2001.04237.x.

Dehnen, W. (June 2002). “A Hierarchical O(N) Force Calculation Algorithm”. In: *Journal of Computational Physics* 179. ADS Bibcode: 2002JCoPh.179...27D, pp. 27–42. DOI: 10.1006/jcph.2002.7026.

Dehnen, W. (Sept. 2014). “A fast multipole method for stellar dynamics”. In: *Computational Astrophysics and Cosmology* 1. ADS Bibcode: 2014ComAC...1....1D, p. 1. DOI: 10.1186/s40668-014-0001-7.

Dehnen, W. and H. Aly (Sept. 2012). “Improving convergence in smoothed particle hydrodynamics simulations without pairing instability”. In: *Monthly Notices of the Royal Astronomical Society* 425, pp. 1068–1082. DOI: 10.1111/j.1365-2966.2012.21439.x.

Del Popolo, A. and M. Le Delliou (Feb. 2017). “Small Scale Problems of the ΛCDM Model: A Short Review”. en. In: *Galaxies* 5.1, p. 17. DOI: 10.3390/galaxies5010017.

Di Cintio, A. et al. (Jan. 2014). “The dependence of dark matter profiles on the stellar-to-halo mass ratio: a prediction for cusps versus cores”. In: *Monthly Notices of the Royal Astronomical Society* 437, pp. 415–423. DOI: 10.1093/mnras/stt1891.

Doumler, T. (June 2012). “Constrained local universe simulations from galaxy peculiar velocities”. en. PhD thesis. Université Claude Bernard - Lyon I.

Doumler, T. et al. (Apr. 2013). “Reconstructing cosmological initial conditions from galaxy peculiar velocities - I. Reverse Zeldovich Approximation”. In: *Monthly Notices of the Royal Astronomical Society* 430, pp. 888–901. DOI: 10.1093/mnras/sts613.

Dubrulle, B. (May 2019). “Beyond Kolmogorov cascades”. In: *Journal of Fluid Mechanics* 867. ADS Bibcode: 2019JFM...867P...1D, P1. DOI: 10.1017/jfm.2019.98.

Durier, F. and C. Dalla Vecchia (Jan. 2012). “Implementation of feedback in smoothed particle hydrodynamics: towards concordance of methods”. In: *Monthly Notices of the Royal Astronomical Society* 419. ADS Bibcode: 2012MNRAS.419..465D, pp. 465–478. DOI: 10.1111/j.1365-2966.2011.19712.x.

Eckart, C. (May 1960). “Variation Principles of Hydrodynamics”. In: *Physics of Fluids* 3, pp. 421–427. DOI: 10.1063/1.1706053.

Einasto, J. et al. (Nov. 1974). “Missing mass around galaxies - Morphological evidence”. In: *Nature* 252, pp. 111–113. DOI: 10.1038/252111a0.

Einstein, A. (1915a). “Erklarung der Perihelionbewegung der Merkur aus der allgemeinen Relativitatstheorie”. In: *Sitzungsber. preuss.Akad. Wiss., vol. 47, No.2, pp. 831-839, 1915* 47, pp. 831–839.

Einstein, A. (1915b). “Die Feldgleichungen der Gravitation”. In: *Sitzungsberichte der Königlich Preußischen Akademie der Wissenschaften (Berlin), Seite 844-847*.

Eisenstein, D. J. and W. Hu (Mar. 1998). “Baryonic Features in the Matter Transfer Function”. In: *The Astrophysical Journal* 496, pp. 605–614. DOI: 10.1086/305424.

Emerick, A. et al. (Aug. 2016). “Gas Loss by Ram Pressure Stripping and Internal Feedback from Low-mass Milky Way Satellites”. In: *The Astrophysical Journal* 826, p. 148. DOI: 10.3847/0004-637X/826/2/148.

Escala, I. et al. (Feb. 2018). “Modelling chemical abundance distributions for dwarf galaxies in the Local Group: the impact of turbulent metal diffusion”. In: *Monthly Notices of the Royal Astronomical Society* 474. ADS Bibcode: 2018MNRAS.474.2194E, pp. 2194–2211. DOI: 10.1093/mnras/stx2858.

Euclid Collaboration (Oct. 2011). “Euclid Definition Study Report”. In: *arXiv e-prints* 1110, arXiv:1110.3193.

Evans, C. et al. (Jan. 2015). *The Science Case for Multi-Object Spectroscopy on the European ELT*. Tech. rep. Publication Title: arXiv e-prints ADS Bibcode: 2015arXiv150104726E Type: article.

Evrard, A. E. et al. (July 2002). “Galaxy Clusters in Hubble Volume Simulations: Cosmological Constraints from Sky Survey Populations”. In: *The Astrophysical Journal* 573, pp. 7–36. DOI: 10.1086/340551.

Farber, R. et al. (Apr. 2018). “Impact of Cosmic-Ray Transport on Galactic Winds”. In: *The Astrophysical Journal* 856, p. 112. DOI: 10.3847/1538-4357/aab26d.

Federrath, C. et al. (Apr. 2010). “Modeling Collapse and Accretion in Turbulent Gas Clouds: Implementation and Comparison of Sink Particles in AMR and SPH”. In: *The Astrophysical Journal* 713, pp. 269–290. DOI: 10.1088/0004-637X/713/1/269.

Ferland, G. J. et al. (Oct. 2017). “The 2017 Release Cloudy”. In: *Revista Mexicana de Astronomia y Astrofisica* 53. ADS Bibcode: 2017RMxAA..53..385F, pp. 385–438.

Fillingham, S. P. et al. (Dec. 2016). “Under pressure: quenching star formation in low-mass satellite galaxies via stripping”. In: *Monthly Notices of the Royal Astronomical Society* 463. ADS Bibcode: 2016MNRAS.463.1916F, pp. 1916–1928. DOI: 10.1093/mnras/stw2131.

Fitts, A. et al. (Nov. 2017). “fire in the field: simulating the threshold of galaxy formation”. In: *Monthly Notices of the Royal Astronomical Society* 471, pp. 3547–3562. DOI: 10.1093/mnras/stx1757.

Frebel, A. and V. Bromm (Nov. 2012). “Chemical Signatures of the First Galaxies: Criteria for One-shot Enrichment”. In: *The Astrophysical Journal* 759, p. 115. DOI: 10.1088/0004-637X/759/2/115.

Freiburghaus, C., S. Rosswog, and F.-K. Thielemann (Nov. 1999). “R-Process in Neutron Star Mergers”. In: *The Astrophysical Journal Letters* 525, pp. L121–L124. DOI: 10.1086/312343.

Friedmann, A. (Dec. 1924). “Über die Möglichkeit einer Welt mit konstanter negativer Krümmung des Raumes”. In: *Zeitschrift fur Physik* 21, pp. 326–332. DOI: 10.1007/BF01328280.

Fritz, T. K. et al. (Nov. 2018). “Gaia DR2 proper motions of dwarf galaxies within 420 kpc. Orbits, Milky Way mass, tidal influences, planar alignments, and group infall”. en. In: *Astronomy & Astrophysics, Volume 619* 619, A103. DOI: 10.1051/0004-6361/201833343.

Fromang, S., P. Hennebelle, and R. Teyssier (Oct. 2006). “A high order Godunov scheme with constrained transport and adaptive mesh refinement for astrophysical magnetohydrodynamics”. In: *Astronomy and Astrophysics* 457, pp. 371–384. DOI: 10.1051/0004-6361:20065371.

Fukugita, M., C. J. Hogan, and P. J. E. Peebles (Aug. 1998). “The Cosmic Baryon Budget”. In: *The Astrophysical Journal* 503, pp. 518–530. DOI: 10.1086/306025.

Gaia Collaboration (Nov. 2016). “The Gaia mission”. In: *Astronomy and Astrophysics* 595, A1. DOI: 10.1051/0004-6361/201629272.

Gaitskell, R. J. (Nov. 2004). “Direct detection of dark matter”. In: *Annual Review of Nuclear and Particle Science* 54.1. Publisher: Annual Reviews, pp. 315–359. DOI: 10.1146/annurev.nucl.54.070103.181244.

Gardner, J. P. et al. (Apr. 2006). “The James Webb Space Telescope”. In: *Space Science Reviews* 123, pp. 485–606. DOI: 10.1007/s11214-006-8315-7.

Garrison-Kimmel, S. et al. (Oct. 2017). “Not so lumpy after all: modelling the depletion of dark matter subhaloes by Milky Way-like galaxies”. In: *Monthly Notices of the Royal Astronomical Society* 471. ADS Bibcode: 2017MNRAS.471.1709G, pp. 1709–1727. DOI: 10.1093/mnras/stx1710.

Garrison-Kimmel, S. et al. (July 2019). “The Local Group on FIRE: dwarf galaxy populations across a suite of hydrodynamic simulations”. In: *Monthly Notices of the Royal Astronomical Society* 487. ADS Bibcode: 2019MNRAS.487.1380G, pp. 1380–1399. DOI: 10.1093/mnras/stz1317.

Geha, M. et al. (Sept. 2017). “The SAGA Survey. I. Satellite Galaxy Populations around Eight Milky Way Analogs”. In: *The Astrophysical Journal* 847. ADS Bibcode: 2017ApJ...847....4G, p. 4. doi: 10.3847/1538-4357/aa8626.

Gillet, N. et al. (Feb. 2015). “Vast Planes of Satellites in a High-resolution Simulation of the Local Group: Comparison to Andromeda”. In: *The Astrophysical Journal* 800, p. 34. doi: 10.1088/0004-637X/800/1/34.

Godoy, W. F. et al. (July 2020). “ADIOS 2: The Adaptable Input Output System. A framework for high-performance data management”. In: *SoftwareX* 12, p. 100561. doi: 10.1016/j.softx.2020.100561.

Gong, H. and E. C. Ostriker (Jan. 2013). “Implementation of Sink Particles in the Athena Code”. In: *The Astrophysical Journal Supplement Series* 204, p. 8. doi: 10.1088/0067-0049/204/1/8.

Gorski, K. (Sept. 1988). “On the pattern of perturbations of the Hubble flow”. In: *The Astrophysical Journal Letters* 332, pp. L7–L11. doi: 10.1086/185255.

Governato, F. et al. (May 2012). “Cuspy no more: how outflows affect the central dark matter and baryon distribution in Λ cold dark matter galaxies”. In: *Monthly Notices of the Royal Astronomical Society* 422, pp. 1231–1240. doi: 10.1111/j.1365-2966.2012.20696.x.

Grand, R. J. J. et al. (May 2021). *Determining the full satellite population of a Milky Way-mass halo in a highly resolved cosmological hydrodynamic simulation*. Tech. rep. Publication Title: arXiv e-prints ADS Bibcode: 2021arXiv210504560G Type: article.

Graziani, R. et al. (Oct. 2019). “The peculiar velocity field up to z $\tilde$ 0.05 by forward-modelling Cosmicflows-3 data”. In: *Monthly Notices of the Royal Astronomical Society* 488, pp. 5438–5451. doi: 10.1093/mnras/stz078.

Graziani, R. (Sept. 2018). “Modélisation Bayésienne des mesures de vitesses particulières dans le projet CosmicFlows”. fr. PhD thesis. Université de Lyon.

Grcevich, J. and M. E. Putman (May 2009). “H I in Local Group Dwarf Galaxies and Stripping by the Galactic Halo”. In: *The Astrophysical Journal* 696. ADS Bibcode: 2009ApJ...696..385G, pp. 385–395. doi: 10.1088/0004-637X/696/1/385.

Greif, T. H. et al. (Feb. 2009). “Chemical mixing in smoothed particle hydrodynamics simulations”. In: *Monthly Notices of the Royal Astronomical Society* 392, pp. 1381–1387. doi: 10.1111/j.1365-2966.2008.14169.x.

Grudić, M. Y. et al. (Sept. 2021). “STARFORGE: Towards a comprehensive numerical model of star cluster formation and feedback”. In: *Monthly Notices of the Royal Astronomical Society* 506. ADS Bibcode: 2021MNRAS.506.2199G, pp. 2199–2231. doi: 10.1093/mnras/stab1347.

Haardt, F. and P. Madau (Feb. 2012). “Radiative Transfer in a Clumpy Universe. IV. New Synthesis Models of the Cosmic UV/X-Ray Background”. In: *The Astrophysical Journal* 746. ADS Bibcode: 2012ApJ...746..125H, p. 125. doi: 10.1088/0004-637X/746/2/125.

Haemmerlé, L. et al. (Apr. 2020). “Formation of the First Stars and Black Holes”. In: *Space Science Reviews* 216, p. 48. doi: 10.1007/s11214-020-00673-y.

Hahn, O. and T. Abel (Aug. 2011). “Multi-scale initial conditions for cosmological simulations”. In: *Monthly Notices of the Royal Astronomical Society* 415. ADS Bibcode: 2011MNRAS.415.2101H, pp. 2101–2121. doi: 10.1111/j.1365-2966.2011.18820.x.

Hahn, O., C. Rampf, and C. Uhlemann (May 2021). “Higher order initial conditions for mixed baryon-CDM simulations”. In: *Monthly Notices of the Royal Astronomical Society* 503. ADS Bibcode: 2021MNRAS.503..426H, pp. 426–445. DOI: 10.1093/mnras/staa3773.

Hairer, E., C. Lubich, and G. Wanner (Mar. 2010). *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*. English. 2nd ed. 2006. 2nd printing 2010 edition. Heidelberg ; New York: Springer.

Harvey, D. et al. (Nov. 2018). “The impact of cored density profiles on the observable quantities of dwarf spheroidal galaxies”. In: *Monthly Notices of the Royal Astronomical Society* 481, pp. L89–L93. DOI: 10.1093/mnrasl/sly159.

Hausammann, L., Y. Revaz, and P. Jablonka (Apr. 2019). “Satellite dwarf galaxies: stripped but not quenched”. In: *Astronomy and Astrophysics* 624, A11. DOI: 10.1051/0004-6361/201834871.

Henderson, B. and K. Bekki (May 2016). “Significant Enhancement of H2 Formation in Disk Galaxies under Strong Ram Pressure”. In: *The Astrophysical Journal Letters* 822, p. L33. DOI: 10.3847/2041-8205/822/2/L33.

Hesthaven, J. S. (Feb. 2018). *Numerical Methods for Conservation Laws*. Computational Science & Engineering. Society for Industrial and Applied Mathematics. DOI: 10.1137/1.9781611975109.

Hillebrandt, W. and J. C. Niemeyer (Sept. 2000). “Type Ia Supernova Explosion Models”. In: *Annual Review of Astronomy and Astrophysics* 38.1. Publisher: Annual Reviews, pp. 191–230. DOI: 10.1146/annurev.astro.38.1.191.

Hirai, Y., M. S. Fujii, and T. R. Saitoh (Aug. 2021). “SIRIUS project. I. Star formation models for star-by-star simulations of star clusters and galaxy formation”. In: *Publications of the Astronomical Society of Japan* 73. ADS Bibcode: 2021PASJ...73.1036H, pp. 1036–1056. DOI: 10.1093/pasj/psab038.

Hoffman, Y. and E. Ribak (Oct. 1991). “Constrained realizations of Gaussian fields - A simple algorithm”. In: *The Astrophysical Journal Letters* 380, pp. L5–L8. DOI: 10.1086/186160.

Hopkins, P. F. (Feb. 2013). “A general class of Lagrangian smoothed particle hydrodynamics methods and implications for fluid mixing problems”. In: *Monthly Notices of the Royal Astronomical Society* 428, pp. 2840–2856. DOI: 10.1093/mnras/sts210.

Hopkins, P. F. (June 2015). “A new class of accurate, mesh-free hydrodynamic simulation methods”. In: *Monthly Notices of the Royal Astronomical Society* 450. ADS Bibcode: 2015MNRAS.450...53H, pp. 53–110. DOI: 10.1093/mnras/stv195.

Hopkins, P. F. and M. Y. Grudić (Mar. 2019). “Numerical problems in coupling photon momentum (radiation pressure) to gas”. In: *Monthly Notices of the Royal Astronomical Society* 483. ADS Bibcode: 2019MNRAS.483.4187H, pp. 4187–4196. DOI: 10.1093/mnras/sty3089.

Hopkins, P. F., E. Quataert, and N. Murray (Oct. 2011). “Self-regulated star formation in galaxies via momentum input from massive stars”. en. In: *Monthly Notices of the Royal Astronomical Society* 417.2, pp. 950–973. DOI: 10.1111/j.1365-2966.2011.19306.x.

Hubber, D. A., S. Walch, and A. P. Whitworth (Apr. 2013). “An improved sink particle algorithm for SPH simulations”. In: *Monthly Notices of the Royal Astronomical Society* 430, pp. 3261–3275. DOI: 10.1093/mnras/stt128.

Hubble, E. (Mar. 1929). “A Relation between Distance and Radial Velocity among Extra-Galactic Nebulae”. In: *Proceedings of the National Academy of Science* 15, pp. 168–173. DOI: 10.1073/pnas.15.3.168.

Hulse, R. A. and J. H. Taylor (Jan. 1975). “Discovery of a pulsar in a binary system”. In: *The Astrophysical Journal Letters* 195, pp. L51–L53. DOI: 10.1086/181708.

Hummel, J. A., A. Stacy, and V. Bromm (Aug. 2016). “The first stars: formation under cosmic ray feedback”. In: *Monthly Notices of the Royal Astronomical Society* 460, pp. 2432–2444. DOI: 10.1093/mnras/stw1127.

Ibata, R. A. et al. (June 2002). “Uncovering cold dark matter halo substructure with tidal streams”. In: *Monthly Notices of the Royal Astronomical Society* 332, pp. 915–920. DOI: 10.1046/j.1365-8711.2002.05358.x.

Irwin, M. J. et al. (Feb. 2007). “Discovery of an Unusual Dwarf Galaxy in the Outskirts of the Milky Way”. In: *The Astrophysical Journal Letters* 656, pp. L13–L16. DOI: 10.1086/512183.

Jeon, M. et al. (Mar. 2021). “The role of faint population III supernovae in forming CEMP stars in ultra-faint dwarf galaxies”. In: *Monthly Notices of the Royal Astronomical Society* 502, pp. 1–14. DOI: 10.1093/mnras/staa4017.

Ji, A. P. et al. (Jan. 2016). “High-resolution Spectroscopy of Extremely Metal-poor Stars in the Least-evolved Galaxies: Bootes II”. In: *The Astrophysical Journal* 817, p. 41. DOI: 10.3847/0004-637X/817/1/41.

Jiang, F. et al. (Mar. 2021). “SatGen: a semi-analytical satellite galaxy generator - I. The model and its application to Local-Group satellite statistics”. In: *Monthly Notices of the Royal Astronomical Society* 502, pp. 621–641. DOI: 10.1093/mnras/staa4034.

Jönsson, H. et al. (Sept. 2020). “APOGEE Data and Spectral Analysis from SDSS Data Release 16: Seven Years of Observations Including First Results from APOGEE-South”. In: *The Astronomical Journal* 160, p. 120. DOI: 10.3847/1538-3881/aba592.

Jungman, G., M. Kamionkowski, and K. Griest (Mar. 1996). “Supersymmetric dark matter”. In: *Physics Reports* 267, pp. 195–373. DOI: 10.1016/0370-1573(95)00058-5.

Kapferer, W. et al. (May 2009). “The effect of ram pressure on the star formation, mass distribution and morphology of galaxies”. In: *Astronomy and Astrophysics* 499, pp. 87–102. DOI: 10.1051/0004-6361/200811551.

Katz, N. (June 1992). “Dissipational galaxy formation. II - Effects of star formation”. In: *The Astrophysical Journal* 391, pp. 502–517. DOI: 10.1086/171366.

Katz, N., D. H. Weinberg, and L. Hernquist (July 1996). “Cosmological Simulations with TreeSPH”. In: *The Astrophysical Journal Supplement Series* 105, p. 19. DOI: 10.1086/192305.

Keller, B. W. et al. (Jan. 2019). “Chaos and variance in galaxy formation”. In: *Monthly Notices of the Royal Astronomical Society* 482, pp. 2244–2261. DOI: 10.1093/mnras/sty2859.

Kennicutt Jr., R. C. (May 1998). “The Global Schmidt Law in Star-forming Galaxies”. In: *The Astrophysical Journal* 498, pp. 541–552. DOI: 10.1086/305588.

Kim, J.-h. et al. (Jan. 2014). “The AGORA High-resolution Galaxy Simulations Comparison Project”. In: *The Astrophysical Journal Supplement Series* 210. ADS Bibcode: 2014ApJS..210...14K, p. 14. DOI: 10.1088/0067-0049/210/1/14.

Kim, J.-h. et al. (Dec. 2016). “The AGORA High-resolution Galaxy Simulations Comparison Project. II. Isolated Disk Test”. In: *The Astrophysical Journal* 833. ADS Bibcode: 2016ApJ...833..202K, p. 202. DOI: 10.3847/1538-4357/833/2/202.

King, S. F. (Feb. 2004). “Neutrino mass models”. In: *Reports on Progress in Physics* 67, pp. 107–157. DOI: 10.1088/0034-4885/67/2/R01.

Kirby, E. N. et al. (Dec. 2010). “Multi-element Abundance Measurements from Medium-resolution Spectra. II. Catalog of Stars in Milky Way Dwarf Satellite Galaxies”. In: *The Astrophysical Journal Supplement Series* 191, pp. 352–375. DOI: 10.1088/0067-0049/191/2/352.

Kirby, E. N. et al. (Dec. 2013). “The Universal Stellar Mass-Stellar Metallicity Relation for Dwarf Galaxies”. In: *The Astrophysical Journal* 779, p. 102. DOI: 10.1088/0004-637X/779/2/102.

Kleyna, J. T. et al. (Sept. 2005). “Ursa Major: A Missing Low-Mass CDM Halo?” In: *The Astrophysical Journal Letters* 630, pp. L141–L144. DOI: 10.1086/491654.

Kobayashi, C., T. Tsujimoto, and K. Nomoto (Aug. 2000). “The History of the Cosmic Supernova Rate Derived from the Evolution of the Host Galaxies”. In: *The Astrophysical Journal* 539. ADS Bibcode: 2000ApJ...539...26K, pp. 26–38. DOI: 10.1086/309195.

Koudmani, S., N. A. Henden, and D. Sijacki (May 2021). “A little FABLE: exploring AGN feedback in dwarf galaxies with cosmological simulations”. In: *Monthly Notices of the Royal Astronomical Society* 503. ADS Bibcode: 2021MNRAS.503.3568K, pp. 3568–3591. DOI: 10.1093/mnras/stab677.

Kronberger, T. et al. (Apr. 2008). “On the influence of ram-pressure stripping on the star formation of simulated spiral galaxies”. In: *Astronomy and Astrophysics* 481, pp. 337–343. DOI: 10.1051/0004-6361:20078904.

Kroupa, P. (Apr. 2001). “On the variation of the initial mass function”. en. In: *Monthly Notices of the Royal Astronomical Society* 322.2, pp. 231–246. DOI: 10.1046/j.1365-8711.2001.04022.x.

Krumholz, M. R., R. I. Klein, and C. F. McKee (July 2012). “Radiation-hydrodynamic Simulations of the Formation of Orion-like Star Clusters. II. The Initial Mass Function from Winds, Turbulence, and Radiation”. In: *The Astrophysical Journal* 754, p. 71. DOI: 10.1088/0004-637X/754/1/71.

Krumholz, M. R., C. F. McKee, and R. I. Klein (Aug. 2004). “Embedding Lagrangian Sink Particles in Eulerian Grids”. In: *The Astrophysical Journal* 611, pp. 399–412. DOI: 10.1086/421935.

Lavaux, G. (Mar. 2016). “Bayesian 3D velocity field reconstruction with VIRBIUS”. In: *Monthly Notices of the Royal Astronomical Society* 457, pp. 172–197. DOI: 10.1093/mnras/stv2915.

Lemaître, G. (1927). “Un Univers homogène de masse constante et de rayon croissant rendant compte de la vitesse radiale des nébuleuses extra-galactiques”. In: *Annales de la Société Scientifique de Bruxelles* 47, pp. 49–59.

Li, H. et al. (July 2021). “Gaia EDR3 Proper Motions of Milky Way Dwarfs. I. 3D Motions and Orbits”. In: *The Astrophysical Journal* 916. ADS Bibcode: 2021ApJ...916....8L, p. 8. DOI: 10.3847/1538-4357/ac0436.

Li, Y.-S., G. De Lucia, and A. Helmi (Jan. 2010). “On the nature of the Milky Way satellites”. In: *Monthly Notices of the Royal Astronomical Society* 401, pp. 2036–2052. DOI: 10.1111/j.1365-2966.2009.15803.x.

Libeskind, N. I. et al. (Sept. 2015). “Planes of satellite galaxies and the cosmic web”. In: *Monthly Notices of the Royal Astronomical Society* 452, pp. 1052–1059. DOI: 10.1093/mnras/stv1302.

Libeskind, N. I. et al. (Oct. 2020). “The HESTIA project: simulations of the Local Group”. In: *Monthly Notices of the Royal Astronomical Society* 498. ADS Bibcode: 2020MNRAS.498.2968L, pp. 2968–2983. DOI: 10.1093/mnras/staa2541.

LIGO Scientific Collaboration and VIRGO Collaboration (Feb. 2016). “Observation of Gravitational Waves from a Binary Black Hole Merger”. In: *Physical Review Letters* 116, p. 061102. DOI: 10.1103/PhysRevLett.116.061102.

Livermore, R. C., S. L. Finkelstein, and J. M. Lotz (Feb. 2017). “Directly Observing the Galaxies Likely Responsible for Reionization”. In: *The Astrophysical Journal* 835, p. 113. DOI: 10.3847/1538-4357/835/2/113.

Lobanov, A. P. (Jan. 2012). “The SKA and "High-Resolution" Science”. In: *Astrophysics and Space Science Proceedings* 25. ADS Bibcode: 2012ASSP...25...75L, p. 75.

LSST Science Collaboration (Dec. 2009). “LSST Science Book, Version 2.0”. In: *arXiv e-prints* 0912, arXiv:0912.0201.

Lupi, A. (Apr. 2019). “H2 chemistry in galaxy simulations: an improved supernova feedback model”. In: *Monthly Notices of the Royal Astronomical Society* 484, pp. 1687–1701. DOI: 10.1093/mnras/stz100.

Lüttgau, J. et al. (Nov. 2018). “Toward Understanding I/O Behavior in HPC Workflows”. In: *2018 IEEE/ACM 3rd International Workshop on Parallel Data Storage Data Intensive Scalable Computing Systems (PDSW-DISCS)*, pp. 64–75. DOI: 10.1109/PDSW-DISCS.2018.00012.

Lynden-Bell, D. (Mar. 1976). “Dwarf galaxies and globular clusters in high velocity hydrogen streams”. In: *Monthly Notices of the Royal Astronomical Society* 174, pp. 695–710. DOI: 10.1093/mnras/174.3.695.

Ma, X., J. Lee, and M. Winslett (Mar. 2006). “High-level buffering for hiding periodic output cost in scientific simulations”. In: *IEEE Transactions on Parallel and Distributed Systems* 17.3. Conference Name: IEEE Transactions on Parallel and Distributed Systems, pp. 193–204. DOI: 10.1109/TPDS.2006.36.

Macciò, A. V. et al. (Dec. 2017). “The edge of galaxy formation - I. Formation and evolution of MW-satellite analogues before accretion”. In: *Monthly Notices of the Royal Astronomical Society* 472. ADS Bibcode: 2017MNRAS.472.2356M, pp. 2356–2366. DOI: 10.1093/mnras/stx2048.

Maeder, A. (2009). “Physics, Formation and Evolution of Rotating Stars”. In: *Physics, Formation and Evolution of Rotating Stars: , Astronomy and Astrophysics Library. ISBN 978-3-540-76948-4. Springer Berlin Heidelberg, 2009.* DOI: 10.1007/978-3-540-76949-1.

Marchesini, D. et al. (Aug. 2002). “Hα Rotation Curves: The Soft Core Question”. In: *The Astrophysical Journal* 575, pp. 801–813. DOI: 10.1086/341475.

Marinacci, F. et al. (Nov. 2018). “First results from the IllustrisTNG simulations: radio haloes and magnetic fields”. In: *Monthly Notices of the Royal Astronomical Society* 480. ADS Bibcode: 2018MNRAS.480.5113M, pp. 5113–5139. DOI: 10.1093/mnras/sty2206.

Martel, H. and P. R. Shapiro (June 1998). “A convenient set of comoving cosmological variables and their application”. In: *Monthly Notices of the Royal Astronomical Society* 297, pp. 467–485. DOI: 10.1046/j.1365-8711.1998.01497.x.

Martin, N. F. et al. (Sept. 2007). “A Keck/DEIMOS spectroscopic survey of faint Galactic satellites: searching for the least massive dwarf galaxies”. In: *Monthly Notices of the Royal Astronomical Society* 380, pp. 281–300. DOI: 10.1111/j.1365-2966.2007.12055.x.

Martin, N. F. et al. (Dec. 2016). “The PAndAS View of the Andromeda Satellite System. II. Detailed Properties of 23 M31 Dwarf Spheroidal Galaxies”. In: *The Astrophysical Journal* 833, p. 167. DOI: 10.3847/1538-4357/833/2/167.

Mashchenko, S., J. Wadsley, and H. M. P. Couchman (Jan. 2008). “Stellar Feedback in Dwarf Galaxy Formation”. In: *Science* 319, p. 174. DOI: 10.1126/science.1148666.

Massari, D. et al. (Nov. 2018). “Three-dimensional motions in the Sculptor dwarf galaxy as a glimpse of a new era”. In: *Nature Astronomy* 2. ADS Bibcode: 2018NatAs...2..156M, pp. 156–161. DOI: 10.1038/s41550-017-0322-y.

Mayer, L. et al. (July 2006). “Simultaneous ram pressure and tidal stripping; how dwarf spheroidals lost their gas”. In: *Monthly Notices of the Royal Astronomical Society* 369. ADS Bibcode: 2006MNRAS.369.1021M, pp. 1021–1038. DOI: 10.1111/j.1365-2966.2006.10403.x.

McConnachie, A. W. (July 2012). “The Observed Properties of Dwarf Galaxies in and around the Local Group”. In: *The Astronomical Journal* 144. ADS Bibcode: 2012AJ....144....4M, p. 4. DOI: 10.1088/0004-6256/144/1/4.

McConnachie, A. W. and K. A. Venn (Sept. 2020). “Revised and New Proper Motions for Confirmed and Candidate Milky Way Dwarf Galaxies”. In: *The Astronomical Journal* 160, p. 124. DOI: 10.3847/1538-3881/aba4ab.

McGaugh, S. S., F. Lelli, and J. M. Schombert (Nov. 2016). “Radial Acceleration Relation in Rotationally Supported Galaxies”. In: *Physical Review Letters* 117, p. 201101. DOI: 10.1103/PhysRevLett.117.201101.

McKinnon, R. et al. (June 2017). “Simulating the dust content of galaxies: successes and failures”. In: *Monthly Notices of the Royal Astronomical Society* 468, pp. 1505–1521. DOI: 10.1093/mnras/stx467.

McWilliam, A. (1997). “Abundance Ratios and Galactic Chemical Evolution”. en. In: *Annual Review of Astronomy and Astrophysics* 35, pp. 503–556. DOI: 10.1146/annurev.astro.35.1.503.

Mitra, S. et al. (Sept. 2005). “An Efficient, Nonintrusive, Log-Based I/O Mechanism for Scientific Simulations on Clusters”. In: *2005 IEEE International Conference on Cluster Computing*. ISSN: 2168-9253, pp. 1–10. DOI: 10.1109/CLUSTR.2005.347041.

Moe, M., K. M. Kratter, and C. Badenes (Apr. 2019). “The Close Binary Fraction of Solar-type Stars Is Strongly Anticorrelated with Metallicity”. In: *The Astrophysical Journal* 875, p. 61. DOI: 10.3847/1538-4357/ab0d88.

Monaghan, J. J. and R. A. Gingold (Nov. 1983). “Shock Simulation by the Particle Method SPH”. In: *Journal of Computational Physics* 52, pp. 374–389. DOI: 10.1016/0021-9991(83)90036-0.

Morris, J. P. and J. J. Monaghan (Sept. 1997). “A Switch to Reduce SPH Viscosity”. In: *Journal of Computational Physics* 136, pp. 41–50. DOI: 10.1006/jcph.1997.5690.

Morselli, L. et al. (June 2020). “A panchromatic spatially resolved analysis of nearby galaxies – II. The main sequence – gas relation at sub-kpc scale in grand-design spirals”. In: *Monthly Notices of the Royal Astronomical Society* 496, pp. 4606–4623. DOI: 10.1093/mnras/staa1811.

Müller, O. et al. (Nov. 2016). “Testing the two planes of satellites in the Centaurus group”. en. In: *Astronomy & Astrophysics, Volume 595, id.A119, <NUMPAGES>9</NUMPAGES> pp.* 595, A119. DOI: 10.1051/0004-6361/201629298.

Navarro, J. F., C. S. Frenk, and S. D. M. White (May 1996). “The Structure of Cold Dark Matter Halos”. In: *The Astrophysical Journal* 462, p. 563. DOI: 10.1086/177173.

Navarro, J. F. et al. (Feb. 2010). “The diversity and similarity of simulated cold dark matter haloes”. In: *Monthly Notices of the Royal Astronomical Society* 402, pp. 21–34. DOI: 10.1111/j.1365-2966.2009.15878.x.

Newman, A. B. et al. (Mar. 2013). “The Density Profiles of Massive, Relaxed Galaxy Clusters. II. Separating Luminous and Dark Matter in Cluster Cores”. In: *The Astrophysical Journal* 765, p. 25. DOI: 10.1088/0004-637X/765/1/25.

Newton, O. et al. (Sept. 2018). “The total satellite population of the Milky Way”. In: *Monthly Notices of the Royal Astronomical Society* 479, pp. 2853–2870. DOI: 10.1093/mnras/sty1085.

Nichols, M., Y. Revaz, and P. Jablonka (Apr. 2014). “Gravitational tides and dwarf spheroidal galaxies”. In: *Astronomy and Astrophysics* 564, A112. DOI: 10.1051/0004-6361/201322710.

Nichols, M., Y. Revaz, and P. Jablonka (Oct. 2015). “The post-infall evolution of a satellite galaxy”. en. In: *Astronomy & Astrophysics, Volume 582, id.A23, <NUMPAGES>26</NUMPAGES> pp.* 582, A23. DOI: 10.1051/0004-6361/201526113.

Nishimura, S. et al. (May 2006). “r-Process Nucleosynthesis in Magnetohydrodynamic Jet Explosions of Core-Collapse Supernovae”. In: *The Astrophysical Journal* 642, pp. 410–419. DOI: 10.1086/500786.

Okamoto, T. et al. (Nov. 2005). “Effects of feedback on the morphology of galaxy discs”. In: *Monthly Notices of the Royal Astronomical Society* 363. ADS Bibcode: 2005MNRAS.363.1299O, pp. 1299–1314. DOI: 10.1111/j.1365-2966.2005.09525.x.

Oñorbe, J. et al. (Dec. 2015). “Forged in FIRE: cusps, cores and baryons in low-mass dwarf galaxies”. In: *Monthly Notices of the Royal Astronomical Society* 454, pp. 2092–2106. DOI: 10.1093/mnras/stv2072.

Padmanabhan, T. (Dec. 2002). “Theoretical Astrophysics - Volume 3, Galaxies and Cosmology”. In: *Theoretical Astrophysics - Volume 3, Galaxies and Cosmology, by T. Padmanabhan, pp. 638. Cambridge University Press, December 2002. ISBN-10: 0521562422. ISBN-13: 9780521562423.* Num Pages: 638. DOI: 10.2277/0521562422.

Pagel, B. E. J. (Jan. 2009). “Nucleosynthesis and Chemical Evolution of Galaxies”. In: *Nucleosynthesis and Chemical Evolution of Galaxies, by Bernard E. J. Pagel, Cambridge, UK: Cambridge University Press, 2009.*

Pawlowski, M. S. (Feb. 2018). “The planes of satellite galaxies problem, suggested solutions, and open questions”. In: *Modern Physics Letters A* 33, p. 1830004. DOI: 10.1142/S0217732318300045.

Peebles, P. J. E. (1980). “The large-scale structure of the universe”. In: *Research supported by the National Science Foundation. Princeton, N.J., Princeton University Press, 1980. 435 p.*

Peebles, P. J. E. (Dec. 1982). “Large-scale background temperature and mass fluctuations due to scale-invariant primeval perturbations”. In: *The Astrophysical Journal* 263. ADS Bibcode: 1982ApJ...263L...1P, pp. L1–L5. doi: 10.1086/183911.

Peebles, P. J. E. (Jan. 1993). *Principles of Physical Cosmology*. Publication Title: Principles of Physical Cosmology by P.J.E. Peebles. Princeton University Press ADS Bibcode: 1993ppc..book.....P.

Pen, U.-L. (Dec. 1997). “Generating Cosmological Gaussian Random Fields”. en. In: *The Astrophysical Journal, Volume 490, Issue 2, pp. L127-L130.* 490.2, p. L127. doi: 10.1086/311042.

Peñarrubia, J., J. F. Navarro, and A. W. McConnachie (Jan. 2008). “The Tidal Evolution of Local Group Dwarf Spheroidals”. In: *The Astrophysical Journal* 673, pp. 226–240. doi: 10.1086/523686.

Persic, M., P. Salucci, and F. Stel (July 1996). “The universal rotation curve of spiral galaxies — I. The dark matter connection”. In: *Monthly Notices of the Royal Astronomical Society* 281, pp. 27–47. doi: 10.1093/mnras/278.1.27.

Poirier, S. (Jan. 2004). “Etude de l’évolution chimique et dynamique d’objets proto-galactiques : Application à l’évolution des galaxies spirales”. thesis. Université Louis Pasteur (Strasbourg).

Pontzen, A. et al. (Feb. 2021). “EDGE: a new approach to suppressing numerical diffusion in adaptive mesh simulations of galaxy formation”. In: *Monthly Notices of the Royal Astronomical Society* 501. ADS Bibcode: 2021MNRAS.501.1755P, pp. 1755–1765. doi: 10.1093/mnras/staa3645.

Porter, T. A., R. P. Johnson, and P. W. Graham (2011). “Dark Matter Searches with Astroparticle Data”. In: *Annual Review of Astronomy and Astrophysics* 49.1. _eprint: https://doi.org/10.1146/annurev-astro-081710-102528, pp. 155–194. doi: 10.1146/annurev-astro-081710-102528.

Price, D. J. (Feb. 2012a). “Resolving high Reynolds numbers in smoothed particle hydrodynamics simulations of subsonic turbulence”. en. In: *Monthly Notices of the Royal Astronomical Society: Letters* 420.1. Publisher: Oxford Academic, pp. L33–L37. doi: 10.1111/j.1745-3933.2011.01187.x.

Price, D. J. (Feb. 2012b). “Smoothed particle hydrodynamics and magnetohydrodynamics”. In: *Journal of Computational Physics* 231. ADS Bibcode: 2012JCoPh.231..759P, pp. 759–794. doi: 10.1016/j.jcp.2010.12.011.

Price, D. J. et al. (Sept. 2018). “Phantom: A Smoothed Particle Hydrodynamics and Magnetohydrodynamics Code for Astrophysics”. en. In: *Publications of the Astronomical Society of Australia* 35, e031. doi: 10.1017/pasa.2018.25.

Prunet, S. et al. (Oct. 2008). “Initial Conditions For Large Cosmological Simulations”. en. In: *The Astrophysical Journal Supplement Series, Volume 178, Issue 2, pp. 179-188 (2008).* 178.2, p. 179. doi: 10.1086/590370.

Rahmati, A. et al. (Apr. 2013). “On the evolution of the H I column density distribution in cosmological simulations”. In: *Monthly Notices of the Royal Astronomical Society* 430, pp. 2427–2445. doi: 10.1093/mnras/stt066.

Raiteri, C. M., M. Villata, and J. F. Navarro (Nov. 1996). "Simulations of Galactic chemical evolution. I. O and Fe abundances in a simple collapse model." In: *Astronomy and Astrophysics* 315, pp. 105–115.

Ramos, P. et al. (June 2020). "Full 5D characterisation of the Sagittarius stream with Gaia DR2 RR Lyrae". In: *Astronomy and Astrophysics* 638, A104. DOI: 10.1051/0004-6361/202037819.

Read, J. I., O. Agertz, and M. L. M. Collins (July 2016). "Dark matter cores all the way down". In: *Monthly Notices of the Royal Astronomical Society* 459, pp. 2573–2590. DOI: 10.1093/mnras/stw713.

Revaz, Y. and P. Jablonka (Feb. 2012). "The dynamical and chemical evolution of dwarf spheroidal galaxies with GEAR". en. In: *Astronomy & Astrophysics, Volume 538, id.A82, <NUMPAGES>28</NUMPAGES> pp.* 538, A82. DOI: 10.1051/0004-6361/201117402.

Revaz, Y. et al. (July 2009). "The dynamical and chemical evolution of dwarf spheroidal galaxies". en. In: *Astronomy and Astrophysics, Volume 501, Issue 1, 2009, pp.189-206* 501.1, p. 189. DOI: 10.1051/0004-6361/200911734.

Revaz, Y. and P. Jablonka (Aug. 2018). "Pushing back the limits: detailed properties of dwarf galaxies in a ΛCDM universe". en. In: *Astronomy & Astrophysics, Volume 616, id.A96, <NUMPAGES>18</NUMPAGES> pp.* 616, A96. DOI: 10.1051/0004-6361/201832669.

Revaz, Y. et al. (Apr. 2016). "Computational issues in chemo-dynamical modelling of the formation and evolution of galaxies". In: *Astronomy and Astrophysics* 588, A21. DOI: 10.1051/0004-6361/201526438.

Riess, A. G. et al. (Sept. 1998). "Observational Evidence from Supernovae for an Accelerating Universe and a Cosmological Constant". In: *The Astronomical Journal* 116, pp. 1009–1038. DOI: 10.1086/300499.

Rigopoulos, G. and W. Valkenburg (Jan. 2015). "On the accuracy of N-body simulations at very large scales". In: *Monthly Notices of the Royal Astronomical Society* 446, pp. 677–682. DOI: 10.1093/mnras/stu2070.

Robertson, B. E. and A. V. Kravtsov (June 2008). "Molecular Hydrogen and Global Star Formation Relations in Galaxies". In: *The Astrophysical Journal* 680, pp. 1083–1111. DOI: 10.1086/587796.

Robertson, B. E. et al. (Feb. 2010). "Computational Eulerian hydrodynamics and Galilean invariance". In: *Monthly Notices of the Royal Astronomical Society* 401, pp. 2463–2476. DOI: 10.1111/j.1365-2966.2009.15823.x.

Robertson, H. P. (Nov. 1935). "Kinematics and World-Structure". In: *The Astrophysical Journal* 82, p. 284. DOI: 10.1086/143681.

Robertson, H. P. (Apr. 1936a). "Kinematics and World-Structure II." In: *The Astrophysical Journal* 83. ADS Bibcode: 1936ApJ....83..187R, p. 187. DOI: 10.1086/143716.

Robertson, H. P. (May 1936b). "Kinematics and World-Structure III." In: *The Astrophysical Journal* 83. ADS Bibcode: 1936ApJ....83..257R, p. 257. DOI: 10.1086/143726.

Roca-Fàbrega, S. et al. (Aug. 2021). "The AGORA High-resolution Galaxy Simulations Comparison Project. III. Cosmological Zoom-in Simulation of a Milky Way-mass Halo". In: *The Astrophysical Journal* 917. ADS Bibcode: 2021ApJ...917...64R, p. 64. DOI: 10.3847/1538-4357/ac088a.

Rosdahl, J. et al. (Dec. 2013). “RAMSES-RT: radiation hydrodynamics in the cosmological context”. In: *Monthly Notices of the Royal Astronomical Society* 436. ADS Bibcode: 2013MNRAS.436.2188R, pp. 2188–2231. DOI: 10.1093/mnras/stt1722.

Ross, R. B. et al. (July 2008). “Visualization and parallel I/O at extreme scale”. en. In: *Journal of Physics: Conference Series* 125, p. 012099. DOI: 10.1088/1742-6596/125/1/012099.

Rosswog, S. et al. (Aug. 2000). “Merging neutron stars: asymmetric systems”. In: *Astronomy and Astrophysics* 360, pp. 171–184.

Ryan-Weber, E. V. et al. (Feb. 2008). “The Local Group dwarf Leo T: HI on the brink of star formation”. In: *Monthly Notices of the Royal Astronomical Society* 384, pp. 535–540. DOI: 10.1111/j.1365-2966.2007.12734.x.

Saitoh, T. R. and J. Makino (May 2013). “A Density-independent Formulation of Smoothed Particle Hydrodynamics”. In: *The Astrophysical Journal* 768. ADS Bibcode: 2013ApJ...768...44S, p. 44. DOI: 10.1088/0004-637X/768/1/44.

Sawala, T. et al. (Mar. 2010). “Formation of isolated dwarf galaxies with feedback”. In: *Monthly Notices of the Royal Astronomical Society* 402, pp. 1599–1613. DOI: 10.1111/j.1365-2966.2009.16035.x.

Sawala, T. et al. (Apr. 2016). “The APOSTLE simulations: solutions to the Local Group’s cosmic puzzles”. In: *Monthly Notices of the Royal Astronomical Society* 457. ADS Bibcode: 2016MNRAS.457.1931S, pp. 1931–1943. DOI: 10.1093/mnras/stw145.

Scalo, J. M. (May 1986). “The stellar initial mass function”. In: *Fundamentals of Cosmic Physics* 11, pp. 1–278.

Schaller, M. (2019a). “Cosmology equations in SWIFT”. In: *Not published*.

Schaller, M. (2019b). “FMM in SWIFT”. In: *Not published*.

Schaller, M. (2019c). “Full derivation of SPH equations in a co-moving frame”. In: *Not published*.

Schaller, M. and J. Borrow (2019). “SPH implementation in SWIFT”. In: *SWIFT repository*.

Schaller, M. et al. (2016). “SWIFT: Using task-based parallelism, fully asynchronous communication, and graph partition-based domain decomposition for strong scaling on more than 100,000 cores”. In: *arXiv:1606.02738 [astro-ph]*. arXiv: 1606.02738, pp. 1–10. DOI: 10.1145/2929908.2929916.

Schaye, J. et al. (Jan. 2015). “The EAGLE project: simulating the evolution and assembly of galaxies and their environments”. en. In: *Monthly Notices of the Royal Astronomical Society* 446.1, pp. 521–554. DOI: 10.1093/mnras/stu2058.

Schober, J., D. R. G. Schleicher, and R. S. Klessen (Dec. 2013). “Magnetic field amplification in young galaxies”. In: *Astronomy and Astrophysics* 560, A87. DOI: 10.1051/0004-6361/201322185.

Schroyen, J. et al. (Sept. 2011). “Simulations of the formation and evolution of isolated dwarf galaxies - II. Angular momentum as a second parameter”. In: *Monthly Notices of the Royal Astronomical Society* 416, pp. 601–617. DOI: 10.1111/j.1365-2966.2011.19083.x.

Shen, S., J. Wadsley, and G. Stinson (Sept. 2010). “The enrichment of the intergalactic medium with adiabatic feedback - I. Metal cooling and metal diffusion”. In: *Monthly Notices of the Royal Astronomical Society* 407, pp. 1581–1596. DOI: 10.1111/j.1365-2966.2010.17047.x.

Shetrone, M. et al. (Feb. 2003). “VLT/UVES Abundances in Four Nearby Dwarf Spheroidal Galaxies. I. Nucleosynthesis and Abundance Ratios”. In: *The Astronomical Journal* 125, pp. 684–706. doi: 10.1086/345966.

Shetrone, M. D., P. Côté, and W. L. W. Sargent (Feb. 2001). “Abundance Patterns in the Draco, Sextans, and Ursa Minor Dwarf Spheroidal Galaxies”. In: *The Astrophysical Journal* 548, pp. 592–608. doi: 10.1086/319022.

Simon, J. D. (Aug. 2018). “Gaia Proper Motions and Orbits of the Ultra-faint Milky Way Satellites”. In: *The Astrophysical Journal* 863. ADS Bibcode: 2018ApJ...863...89S, p. 89. doi: 10.3847/1538-4357/aacdfb.

Simon, J. D. (Aug. 2019). “The Faintest Dwarf Galaxies”. en. In: *Annual Review of Astronomy and Astrophysics, vol. 57, p.375-415* 57, p. 375. doi: 10.1146/annurev-astro-091918-104453.

Simon, J. D. and M. Geha (Nov. 2007). “The Kinematics of the Ultra-faint Milky Way Satellites: Solving the Missing Satellite Problem”. In: *The Astrophysical Journal* 670, pp. 313–331. doi: 10.1086/521816.

Simon, J. D. et al. (Mar. 2005). “High-Resolution Measurements of the Halos of Four Dark Matter-Dominated Galaxies: Deviations from a Universal Density Profile”. In: *The Astrophysical Journal* 621, pp. 757–776. doi: 10.1086/427684.

Simpson, C. M. et al. (July 2018). “Quenching and ram pressure stripping of simulated Milky Way satellite galaxies”. In: *Monthly Notices of the Royal Astronomical Society* 478. ADS Bibcode: 2018MNRAS.478..548S, pp. 548–567. doi: 10.1093/mnras/sty774.

Sirko, E. (Nov. 2005). “Initial Conditions to Cosmological N-Body Simulations, or, How to Run an Ensemble of Simulations”. In: *The Astrophysical Journal* 634, pp. 728–743. doi: 10.1086/497090.

Skidmore, W., TMT International Science Development Teams, and T. Science Advisory Committee (Dec. 2015). “Thirty Meter Telescope Detailed Science Case: 2015”. In: *Research in Astronomy and Astrophysics* 15, p. 1945. doi: 10.1088/1674-4527/15/12/001.

Smith, B., S. Sigurdsson, and T. Abel (Apr. 2008). “Metal cooling in simulations of cosmic structure formation”. en. In: *Monthly Notices of the Royal Astronomical Society* 385.3, pp. 1443–1454. doi: 10.1111/j.1365-2966.2008.12922.x.

Smith, B. D. et al. (Apr. 2017). “GRACKLE: a chemistry and cooling library for astrophysics”. In: *Monthly Notices of the Royal Astronomical Society* 466. ADS Bibcode: 2017MNRAS.466.2217S, pp. 2217–2234. doi: 10.1093/mnras/stw3291.

Smith, M. C., D. Sijacki, and S. Shen (July 2018). “Supernova feedback in numerical simulations of galaxy formation: separating physics from numerics”. In: *Monthly Notices of the Royal Astronomical Society* 478. ADS Bibcode: 2018MNRAS.478..302S, pp. 302–331. doi: 10.1093/mnras/sty994.

Sorce, J. G. (July 2015). “Minimization of biases in galaxy peculiar velocity catalogues”. In: *Monthly Notices of the Royal Astronomical Society* 450, pp. 2644–2657. doi: 10.1093/mnras/stv760.

Sorce, J. G. et al. (Feb. 2014). “Simulations of the Local Universe constrained by observational peculiar velocities”. In: *Monthly Notices of the Royal Astronomical Society* 437, pp. 3586–3595. doi: 10.1093/mnras/stt2153.

Springel, V. (Dec. 2005). “The cosmological simulation code GADGET-2”. In: *Monthly Notices of the Royal Astronomical Society* 364, pp. 1105–1134. doi: 10.1111/j.1365-2966.2005.09655.x.

Springel, V. (Jan. 2010). “E pur si muove: Galilean-invariant cosmological hydrodynamical simulations on a moving mesh”. In: *Monthly Notices of the Royal Astronomical Society* 401. ADS Bibcode: 2010MNRAS.401..791S, pp. 791–851. doi: 10.1111/j.1365-2966.2009.15715.x.

Springel, V. and L. Hernquist (July 2002). “Cosmological smoothed particle hydrodynamics simulations: the entropy equation”. In: *Monthly Notices of the Royal Astronomical Society* 333, pp. 649–664. doi: 10.1046/j.1365-8711.2002.05445.x.

Springel, V. and L. Hernquist (Feb. 2003). “Cosmological smoothed particle hydrodynamics simulations: a hybrid multiphase model for star formation”. In: *Monthly Notices of the Royal Astronomical Society* 339. ADS Bibcode: 2003MNRAS.339..289S, pp. 289–311. doi: 10.1046/j.1365-8711.2003.06206.x.

Springel, V. et al. (June 2005). “Simulations of the formation, evolution and clustering of galaxies and quasars”. In: *Nature* 435, pp. 629–636. doi: 10.1038/nature03597.

Springel, V. et al. (Mar. 2018). “First results from the IllustrisTNG simulations: matter and galaxy clustering”. In: *Monthly Notices of the Royal Astronomical Society* 475. ADS Bibcode: 2018MNRAS.475..676S, pp. 676–698. doi: 10.1093/mnras/stx3304.

Stadel, J. et al. (Sept. 2009). “Quantifying the heart of darkness with GHALO - a multibillion particle simulation of a galactic halo”. en. In: *Monthly Notices of the Royal Astronomical Society: Letters, Volume 398, Issue 1, pp. L21-L25.* 398.1, p. L21. doi: 10.1111/j.1745-3933.2009.00699.x.

Starkenburg, E. et al. (Nov. 2017). “The Pristine survey - I. Mining the Galaxy for the most metal-poor stars”. In: *Monthly Notices of the Royal Astronomical Society* 471, pp. 2587–2604. doi: 10.1093/mnras/stx1068.

Stinson, G. et al. (Dec. 2006). “Star formation and feedback in smoothed particle hydrodynamic simulations - I. Isolated galaxies”. In: *Monthly Notices of the Royal Astronomical Society* 373. ADS Bibcode: 2006MNRAS.373.1074S, pp. 1074–1090. doi: 10.1111/j.1365-2966.2006.11097.x.

Strigari, L. E. et al. (May 2008). “The Most Dark-Matter-dominated Galaxies: Predicted Gamma-Ray Signals from the Faintest Milky Way Dwarfs”. In: *The Astrophysical Journal* 678, pp. 614–620. doi: 10.1086/529488.

Tafelmeyer, M. et al. (Dec. 2010). “Extremely metal-poor stars in classical dwarf spheroidal galaxies: Fornax, Sculptor, and Sextans”. en. In: *Astronomy and Astrophysics, Volume 524, id.A58, <NUMPAGES>21</NUMPAGES> pp.* 524, A58. doi: 10.1051/0004-6361/201014733.

Teyssier, R. (Apr. 2002). “Cosmological hydrodynamics with adaptive mesh refinement. A new high resolution code called RAMSES”. In: *Astronomy and Astrophysics* 385, pp. 337–364. doi: 10.1051/0004-6361:20011817.

Teyssier, R. (Aug. 2015). “Grid-Based Hydrodynamics in Astrophysical Fluid Flows”. In: *Annual Review of Astronomy and Astrophysics* 53, pp. 325–364. doi: 10.1146/annurev-astro-082214-122309.

Teyssier, R. and B. Commerçon (July 2019). “Numerical Methods for Simulating Star Formation”. In: *Frontiers in Astronomy and Space Sciences* 6, p. 51. doi: 10.3389/fspas.2019.00051.

Teyssier, R. et al. (Mar. 2013). “Cusp-core transformations in dwarf galaxies: observational predictions”. In: *Monthly Notices of the Royal Astronomical Society* 429, pp. 3068–3078. doi: 10.1093/mnras/sts563.

Theler, R. et al. (Oct. 2020). “The chemical evolution of the dwarf spheroidal galaxy Sextans”. en. In: *Astronomy & Astrophysics, Volume 642, id.A176, <NUMPAGES>13</NUMPAGES> pp.* 642, A176. doi: 10.1051/0004-6361/201937146.

Thomas, N. et al. (May 2021). “The radio galaxy population in the SIMBA simulations”. In: *Monthly Notices of the Royal Astronomical Society* 503. ADS Bibcode: 2021MNRAS.503.3492T, pp. 3492–3509. doi: 10.1093/mnras/stab654.

Tinsley, B. M. (1980). “Evolution of the Stars and Gas in Galaxies”. In: *Fundamentals of Cosmic Physics* 5, pp. 287–388.

Tolstoy, E., V. Hill, and M. Tosi (Sept. 2009). “Star-Formation Histories, Abundances, and Kinematics of Dwarf Galaxies in the Local Group”. In: *Annual Review of Astronomy and Astrophysics* 47, pp. 371–425. doi: 10.1146/annurev-astro-082708-101650.

Tornatore, L. et al. (Dec. 2007). “Chemical enrichment of galaxy clusters from hydrodynamical simulations”. In: *Monthly Notices of the Royal Astronomical Society* 382. ADS Bibcode: 2007MNRAS.382.1050T, pp. 1050–1072. doi: 10.1111/j.1365-2966.2007.12070.x.

Tremmel, M. et al. (Sept. 2017). “The Romulus cosmological simulations: a physical approach to the formation, dynamics and accretion models of SMBHs”. In: *Monthly Notices of the Royal Astronomical Society* 470, pp. 1121–1139. doi: 10.1093/mnras/stx1160.

Tricco, T. S. (May 2015). *Simulating Astrophysical Magnetic Fields with Smoothed Particle Magnetohydrodynamics*. Tech. rep. Publication Title: arXiv e-prints ADS Bibcode: 2015arXiv150504494T Type: article.

Tsujimoto, T. et al. (Dec. 1995). “Relative frequencies of Type Ia and Type II supernovae in the chemical evolution of the Galaxy, LMC and SMC”. en. In: *Monthly Notices of the Royal Astronomical Society* 277.3, pp. 945–958. doi: 10.1093/mnras/277.3.945.

Tsujimoto, T. et al. (June 2015). “Chemical feature of Eu abundance in the Draco dwarf spheroidal galaxy†”. In: *Publications of the Astronomical Society of Japan* 67, p. L3. doi: 10.1093/pasj/psv035.

Tully, R. B., H. M. Courtois, and J. G. Sorce (Aug. 2016). “Cosmicflows-3”. In: *The Astronomical Journal* 152, p. 50. doi: 10.3847/0004-6256/152/2/50.

Valcke, S., S. de Rijcke, and H. Dejonghe (Sept. 2008). “Simulations of the formation and evolution of isolated dwarf galaxies”. In: *Monthly Notices of the Royal Astronomical Society* 389, pp. 1111–1126. doi: 10.1111/j.1365-2966.2008.13654.x.

Vandenbroucke, B. and K. Wood (Sept. 2019). “Radiation hydrodynamics simulations of the evolution of the diffuse ionized gas in disc galaxies”. In: *Monthly Notices of the Royal Astronomical Society* 488, pp. 1977–1986. doi: 10.1093/mnras/stz1841.

Vasiliev, E. and V. Belokurov (Oct. 2020). “The last breath of the Sagittarius dSph”. In: *Monthly Notices of the Royal Astronomical Society* 497, pp. 4162–4182. doi: 10.1093/mnras/staa2114.

Vazdekis, A. et al. (Oct. 1996). “A New Chemo-evolutionary Population Synthesis Model for Early-Type Galaxies. I. Theoretical Basis”. In: *The Astrophysical Journal Supplement Series* 106, p. 307. DOI: 10.1086/192340.

Venn, K. A. et al. (Apr. 2017). “Gemini/GRACES spectroscopy of stars in Tri II”. In: *Monthly Notices of the Royal Astronomical Society* 466, pp. 3741–3752. DOI: 10.1093/mnras/stw3198.

Walker, A. G. (1937). “On Milne’s Theory of World-Structure”. In: *Proceedings of the London Mathematical Society, (Series 2) volume 42, p. 90-127* 42, pp. 90–127. DOI: 10.1112/plms/s2-42.1.90.

Wall, J. E. et al. (Dec. 2019). “Collisional N-body Dynamics Coupled to Self-gravitating Magnetohydrodynamics Reveals Dynamical Binary Formation”. In: *The Astrophysical Journal* 887, p. 62. DOI: 10.3847/1538-4357/ab4db1.

Webster, D., R. Sutherland, and J. Bland-Hawthorn (Nov. 2014). “Ultrafaint Dwarfs—Star Formation and Chemical Evolution in the Smallest Galaxies”. In: *The Astrophysical Journal* 796, p. 11. DOI: 10.1088/0004-637X/796/1/11.

Weisberg, J. M., J. H. Taylor, and L. A. Fowler (Oct. 1981). “Gravitational waves from an orbiting pulsar”. In: *Scientific American* 245, pp. 74–82. DOI: 10.1038/scientificamerican1081-74.

Weisz, D. R. et al. (July 2014). “The Star Formation Histories of Local Group Dwarf Galaxies. I. Hubble Space Telescope/Wide Field Planetary Camera 2 Observations”. In: *The Astrophysical Journal* 789. ADS Bibcode: 2014ApJ...789..147W, p. 147. DOI: 10.1088/0004-637X/789/2/147.

Wetzel, A. R. et al. (Aug. 2016). “Reconciling Dwarf Galaxies with ΛCDM Cosmology: Simulating a Realistic Population of Satellites around a Milky Way-mass Galaxy”. In: *The Astrophysical Journal Letters* 827, p. L23. DOI: 10.3847/2041-8205/827/2/L23.

Whalen, D., R. M. Hueckstaedt, and T. O. McConkie (Mar. 2010). “Radiative and Kinetic Feedback by Low-Mass Primordial Stars”. In: *The Astrophysical Journal* 712, pp. 101–111. DOI: 10.1088/0004-637X/712/1/101.

Wheeler, C. et al. (Dec. 2019). “Be it therefore resolved: cosmological simulations of dwarf galaxies with 30 solar mass resolution”. In: *Monthly Notices of the Royal Astronomical Society* 490, pp. 4447–4463. DOI: 10.1093/mnras/stz2887.

Wiersma, R. P. C., J. Schaye, and B. D. Smith (Feb. 2009a). “The effect of photoionization on the cooling rates of enriched, astrophysical plasmas”. In: *Monthly Notices of the Royal Astronomical Society* 393, pp. 99–107. DOI: 10.1111/j.1365-2966.2008.14191.x.

Wiersma, R. P. C. et al. (Oct. 2009b). “Chemical enrichment in cosmological, smoothed particle hydrodynamics simulations”. In: *Monthly Notices of the Royal Astronomical Society* 399. ADS Bibcode: 2009MNRAS.399..574W, pp. 574–600. DOI: 10.1111/j.1365-2966.2009.15331.x.

Williamson, D. and H. Martel (Nov. 2018). “Chemodynamics of Dwarf Galaxies under Ram Pressure”. In: *The Astrophysical Journal* 867, p. 72. DOI: 10.3847/1538-4357/aae538.

Wong, Y. Y. Y. (Nov. 2011). “Neutrino Mass in Cosmology: Status and Prospects”. In: *Annual Review of Nuclear and Particle Science* 61, pp. 69–98. DOI: 10.1146/annurev-nucl-102010-130252.

Woosley, S. E. et al. (Sept. 1994). “The r-process and neutrino-heated supernova ejecta”. In: *The Astrophysical Journal* 433, pp. 229–246. DOI: 10.1086/174638.

Wright, A. C. et al. (Jan. 2019). “Reignition of star formation in dwarf galaxies”. In: *Monthly Notices of the Royal Astronomical Society* 482. ADS Bibcode: 2019MNRAS.482.1176W, pp. 1176–1189. DOI: 10.1093/mnras/sty2759.

Xiao, Q., P. Shang, and J. Wang (June 2012). “Co-located Compute and Binary File Storage in Data-Intensive Computing”. In: *2012 IEEE Seventh International Conference on Networking, Architecture, and Storage*, pp. 199–206. DOI: 10.1109/NAS.2012.29.

Yoon, J. et al. (June 2019). “Origin of the CEMP-no Group Morphology in the Milky Way”. In: *The Astrophysical Journal* 878, p. 97. DOI: 10.3847/1538-4357/ab1ead.

Yozin, C. and K. Bekki (Oct. 2015). “The transformation and quenching of simulated gas-rich dwarf satellites within a group environment”. In: *Monthly Notices of the Royal Astronomical Society* 453, pp. 14–28. DOI: 10.1093/mnras/stv1593.

Zaroubi, S. et al. (Aug. 1995). “Wiener Reconstruction of the Large-Scale Structure”. In: *The Astrophysical Journal* 449, p. 446. DOI: 10.1086/176070.

Zaroubi, S., Y. Hoffman, and A. Dekel (Aug. 1999). “Wiener Reconstruction of Large-Scale Structure from Peculiar Velocities”. In: *The Astrophysical Journal* 520, pp. 413–425. DOI: 10.1086/307473.

Zel’dovich, Y. B. (Mar. 1970). “Gravitational instability: An approximate theory for large density perturbations.” In: *Astronomy and Astrophysics* 5, pp. 84–89.

Zheng, F. et al. (May 2013). “FlexIO: I/O Middleware for Location-Flexible Scientific Data Analytics”. In: *2013 IEEE 27th International Symposium on Parallel and Distributed Processing*. ISSN: 1530-2075, pp. 320–331. DOI: 10.1109/IPDPS.2013.46.

Zumalacárregui, M. and U. Seljak (Oct. 2018). “Limits on Stellar-Mass Compact Objects as Dark Matter from Gravitational Lensing of Type Ia Supernovae”. In: *Physical Review Letters* 121, p. 141101. DOI: 10.1103/PhysRevLett.121.141101.

Zwicky, F. (Jan. 2009). “Republication of: The redshift of extragalactic nebulae”. In: *General Relativity and Gravitation* 41, pp. 207–224. DOI: 10.1007/s10714-008-0707-4.

Avenue de la Rochelle 6
1008 Prilly
loic.hausammann@protonmail.ch
+41 78 843 68 20
Nationality: Swiss

Achievements

- Successfully developed a new and more efficient code for cosmological simulations (37x faster and output 20x smaller)
- Participated as a main contributor in an international collaboration regrouping the most renowned groups in astrophysics

Skills and knowledge

- Programming mainly with C/C++, CUDA and Python
- Parallel computing (MPI, pthreads, OpenMP, GPU)
- Data analysis and Machine Learning with Python
- GIT, Sphinx, Doxygen
- Linux
- Teaching
- Other languages: Matlab, Java, Fortran

Personal Interests

- Programming
- Video games (kinship officer in LotRo for ca. 35 players)
- Other Hobbies: Scuba diving, Board and Pen-and-paper role-playing games (as game master)

Languages

- French: Native fluency
- English: Professional working proficiency
- German: A1/A2 with a strong motivation to learn
- Italian: A1/A2 with a good comprehension

Loïc Hausammann

As a tenacious and curious problem solver with a hands-on, even elbow deep, programming experience and mindset, I leverage my strong abstract thinking skills and my drive to continuously measure my work, in order to achieve high quality results. A team player by nature, I strive in environments where collaboration is placed at the center of building sustainable solutions, such as it was in the 3 of international collaborations I participated in.

Professional Experience

October 2016 – now
Internship, then research assistant, then PhD researcher in Numerical Astrophysics *at Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland*

- Implemented core functionalities for the High Performance Computing (HPC) software SWIFT in C in a multinational team of ca. 10 persons improving the computation speed by a factor of 37x.
- Implemented a visualization tool for the task architecture in SWIFT.
- Designed and developed the new output system in SWIFT, achieved a division by 10 of required memory.
- Supervised classes of Bachelor and Master students in exercise sessions and personal projects.
- Conducted guided tour of the observatory of Geneva to non-scientific visitors.

September 2015 – July 2016
Swiss Civilian Service *in various locations*

- Developed an image recognition software in the Enterprise Resource Planning (ERP) and Customer Relationship Management (CRM) platform Odoo for Compassion Suisse, Yverdon, Switzerland (4 months)
- Ran the receptionist of the internal medicine at Hospital Samaritain, Vevey, Switzerland (6 months)

Education

February 2015 – July 2015
Master project *at Princeton University, United States of America*

- Developed a software in Python that models measurement on a Tokamak reactor from simulation data (plasma physics).

August 2010 – July 2015
Bachelor and Master in Physics, Minor in Computational Science and Engineering *at Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland*

- Focused on numerical and plasma physics.
- Taught general physics to the sections of medicine and life science as a teaching assistant.

Publications

The complete list is available on ADS.

Appendix Part VI

Generating a new Cooling Table

The generation of new cooling table is partially described in Smith et al., 2008 and Smith et al., 2017. The work is based on Abel et al., 1997 that defines the names of all the rates. Unfortunately, `GRACKLE` is not really respecting the different names but all the equations are described in the source code (calc_rates_g.f). Three different steps are required to generate the new tables: computing the cooling, computing the ionization rate and then merging everything into a `HDF5` cooling table.

For the first step, a possibility is to use the code `Cloudy` (Ferland et al., 2017) that is widely used in astrophysics. I developed a code based on the work of Smith et al., 2017 that simply runs `Cloudy` for each point in the parameter space (temperature, density, redshift). The script `cloudy_loop.py` is provided along with this thesis in the directory `scripts/cooling_table`[2] and needs the parameter files `cooling_metal.yml` and `cooling_no_metal.yml`.

The table in Cloudy 17.01 are not fully consistent therefore the code may crash at high or low densities. If the crash is due to negative population for a given element, you will either need to deactivate the corresponding energy level in `CLOUDY_ROOT/data/stout/*/*/*.nrg` or interpolate the missing points[3]. For my table, the elements that were causing problems are F_3, P_2, Ar_3, Ar_4 and Ni_3.

The second step is not documented in the previous papers and in fact do not need to be recomputed when adjusting the parameter space, but only when using a new UV background. As I have only copy and paste the UV background for my cooling tables, I am not going to explain how to generate them. For more information on the subject, I recommend Abel et al., 1997 and Haardt and Madau, 2012.

The final step simply consists in creating a table from all the files generated and can be done with the script `generates_hdf5_file.py` given in the same directory as the previous files. For the primordial cooling, it simply copies the values from the output files. For the metals cooling, only the contribution from the metals is required, therefore the primordial cooling is removed by subtracting the previous table. An existing cooling table is required for this script in order to copy and paste the ionization tables.

---

[2] Expect a few days of simulation in parallel.

[3] I followed the second approach for my table.

Examples

`SWIFT` includes some examples in its directory in order to verify the code. For `SWIFT-GEAR`, I have implemented mainly 3 examples: the Rayleight-Taylor instability, the `AGORA` disk and the zoom simulations. The first one tests the hydrodynamics while the two others tests the full physics implemented in `SWIFT-GEAR`. The `AGORA` simulation is an example out of the cosmological context, at low resolution and with a massive galaxy while the zoom simulation is a cosmological simulation of a dwarf galaxy at high resolution. It is worth to mention that the zoom simulation is the type of simulation that we aim for with `SWIFT-GEAR`.

Rayleigh-Taylor Instability

The Rayleigh-Taylor instability is one of the classical test in hydrodynamics. It consists in a high density fluid over a low density fluid at equilibrium. Under the effect of the gravity, the two fluids start to mix together and produce the typical mushroom clouds seen in the images of nuclear explosions.

In this example, I am following the implementation of Saitoh and Makino, 2013. The particles are set on an almost regular grid that is deformed along the y-axis in order to match the density profile and the instability is seeded with an initial vertical sinusoidal velocity. The density profile is computed in order to have a gas at equilibrium under the effect of the constant vertical gravity field. At the top and bottom of the box, some particles are kept fixed during the whole simulation (they are responsible for the bands in the figure) in order to set the boundary conditions and avoid the particles moving from the bottom to the top due to the periodic conditions. In Figure 1, the instability is shown at late time for `SPHENIX`.

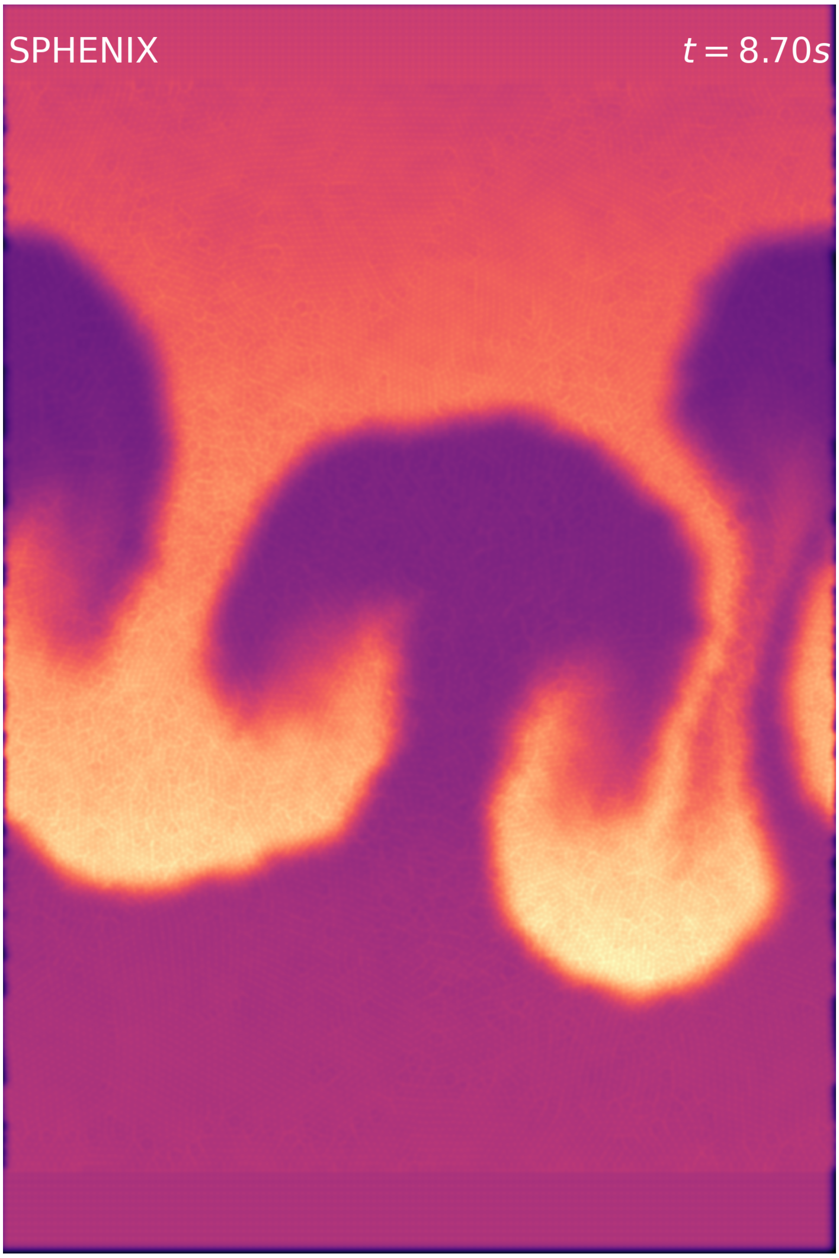


Figure 1 – Density of the gas in a Rayleigh-Taylor instability. The instability consists in the mixing of a gas of low density (bottom) and high density (top) in a constant gravitational field.

AGORA Disk

The `AGORA` disk simulation is described in Kim et al., 2016. The aim of the project is to compare different codes on the same problem. In this paper, the participating codes were slightly modified in order to simulate the same physics (e.g. same cooling function, stellar yields, ...). I decided to use the same simulation while keeping our standard physics to check the correctness of `SWIFT-GEAR` compared to `GEAR`. The initial conditions consist in an analytical model of an isolated disk galaxy with properties corresponding to a Milky Way at $z = 1$ and are fully described in Kim et al., 2016. In this example, we are using the low resolution initial conditions.

The simulation was run for 500 Myr with both `SWIFT` and `GEAR` including our full physics (without the cosmological expansion) and the results are plotted in Figures 2, 3 and 4. The figures shown here have been selected in order to give a good overview of the properties of the stars through the star formation rate profile (Figure 2) and the gas through the density and temperature projections (Figures 3 and 4). As the code for the figures is taken from the `AGORA` project (and slightly modified in order to be compatible with `SWIFT`), the figures can be directly compared with the published paper but the physics and the resolution used are different. Therefore, some differences are expected and can be seen (especially in the temperature projection). In the figures, some differences are seen between `GEAR` and `SWIFT-GEAR` due to the stochastic approach of the star formation and a different accumulation of the rounding errors (amplified by the chaotic behavior of our equations). Thanks to this first example, we can see that `SWIFT-GEAR` is able to reproduce the results of `GEAR` in a non-cosmological context and at low resolution.

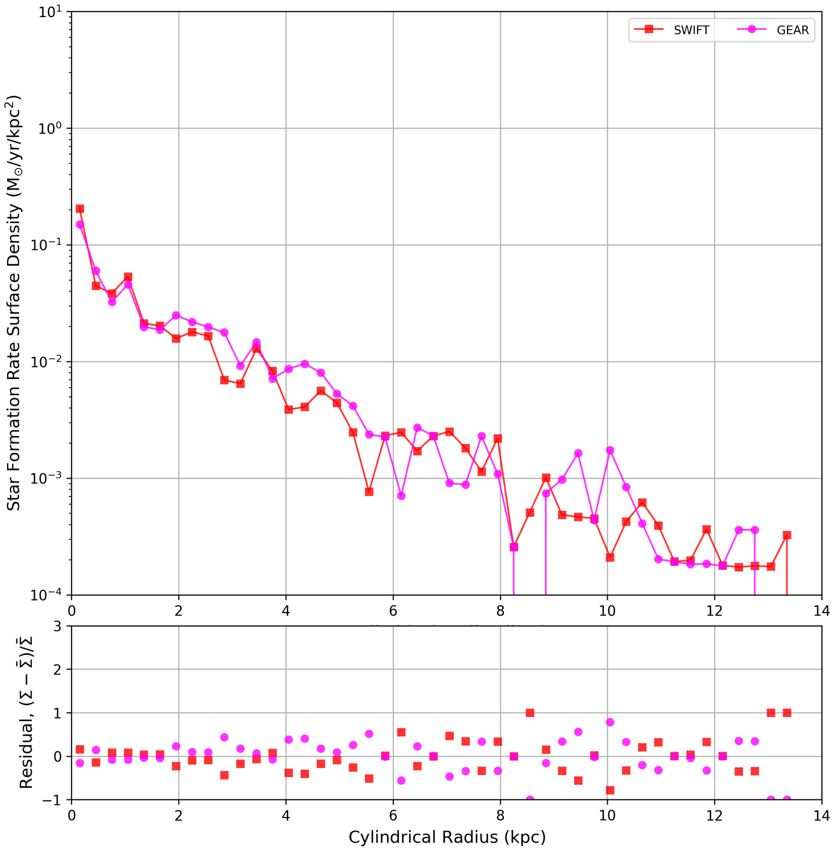


Figure 2 – Star formation rate profile for the low resolution `AGORA` disk simulation. This corresponds to Figure 27 in Kim et al., 2016 but with a slightly different physics. This figure shows that both codes produce the same star formation.

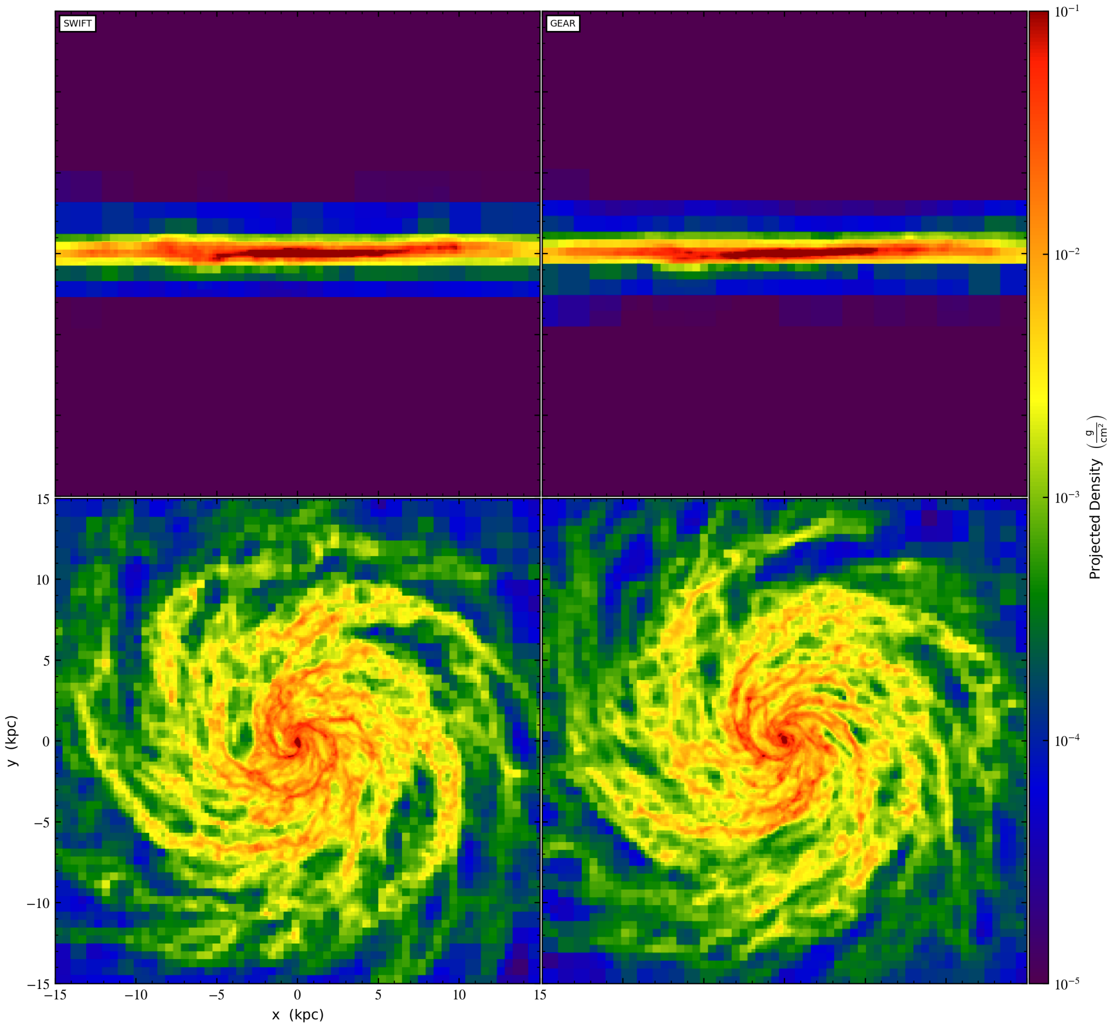


Figure 3 – Density projection of the low resolution `AGORA` disk simulation for both `GEAR` (right) and `SWIFT-GEAR` (left). This corresponds to Figure 3 in Kim et al., 2016 but with a slightly different physics. The global features of the disk in `GEAR` are reproduced by `SWIFT-GEAR`.

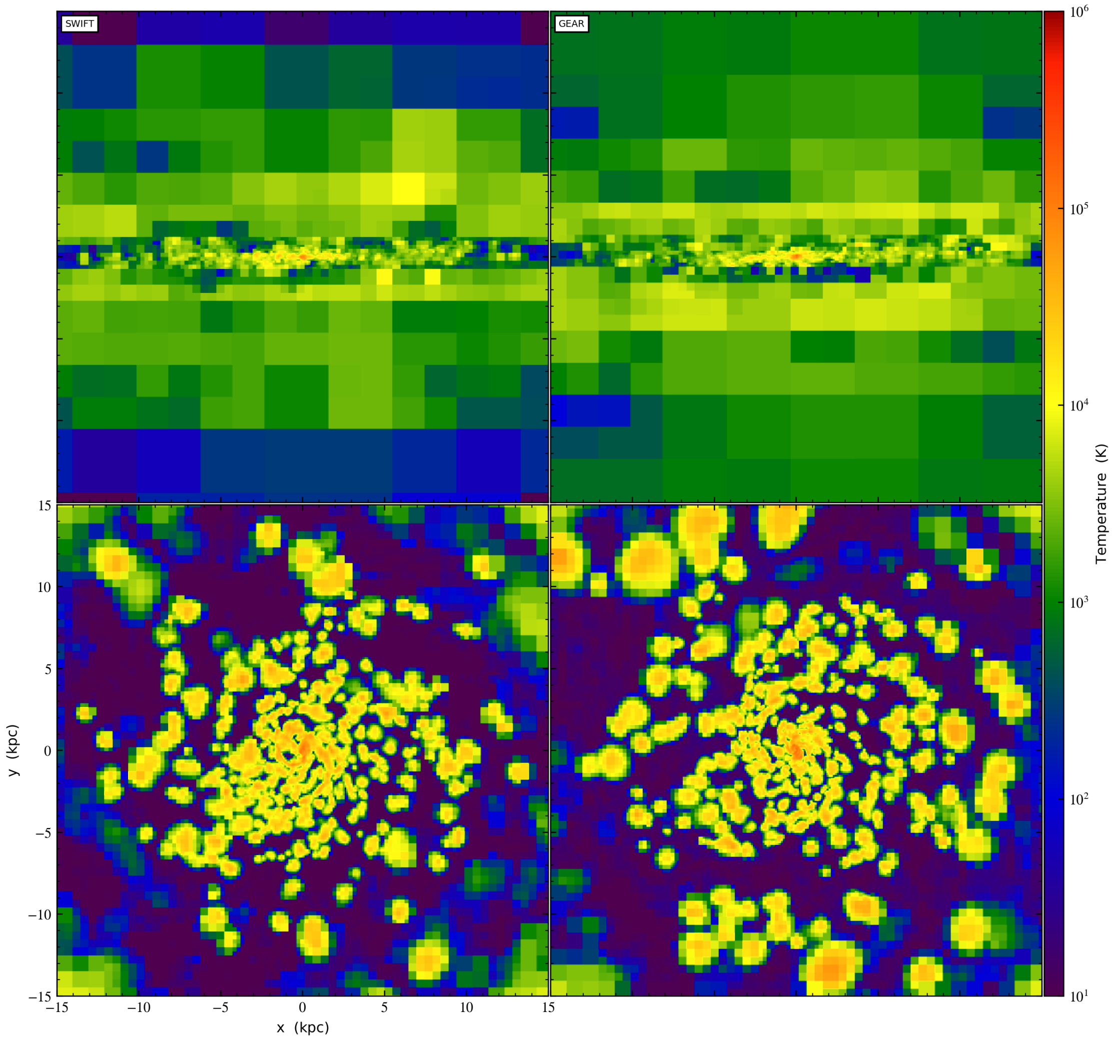


Figure 4 – Temperature projection of the low resolution `AGORA` disk simulation for both `GEAR` (right) and `SWIFT-GEAR` (left). This corresponds to Figure 15 in Kim et al., 2016 but with a slightly different physics. Due to the difference of physics and resolution, the temperature projection in this thesis and the previously cited paper present strong differences. The global features of the disk in `GEAR` are reproduced by `SWIFT-GEAR`.

Zoom Simulations

The zoom technique is a way to generate initial conditions at high resolution (fully described in the chapter 9). The key idea is to select a region of interest in the whole box and to increase the resolution only in this area. It means that the low resolution particles will act as boundary conditions for the high resolution area.

The example is based on the paper Revaz and Jablonka, 2018 and compares directly `SWIFT-GEAR` with

`GEAR`. It contains two different set of initial conditions that correspond to the galaxy 50 and 177 in the previous paper. In Figures 5 and 6, the star formation rate and the stellar abundances are compared between `GEAR` and `SWIFT-GEAR`. They both produce comparable results, but due to the chaotic behavior of the galaxy evolution, some differences can be observed.

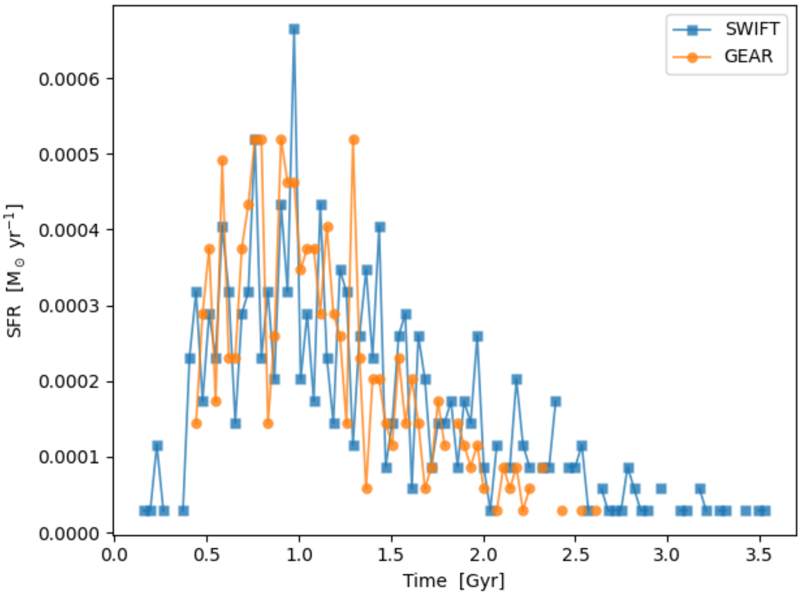


Figure 5 – Star formation history for the dwarf galaxy 177 from Revaz and Jablonka, 2018 done with both `SWIFT-GEAR` (in blue) and `GEAR`(in orange). Due to the low mass of the galaxy, the gas can easily escape due to the feedback of the stars and the UV background and stop forming stars at early time (below 4 Gyr). `SWIFT-GEAR` produces slightly less stars but the overall star formation histories agree well.

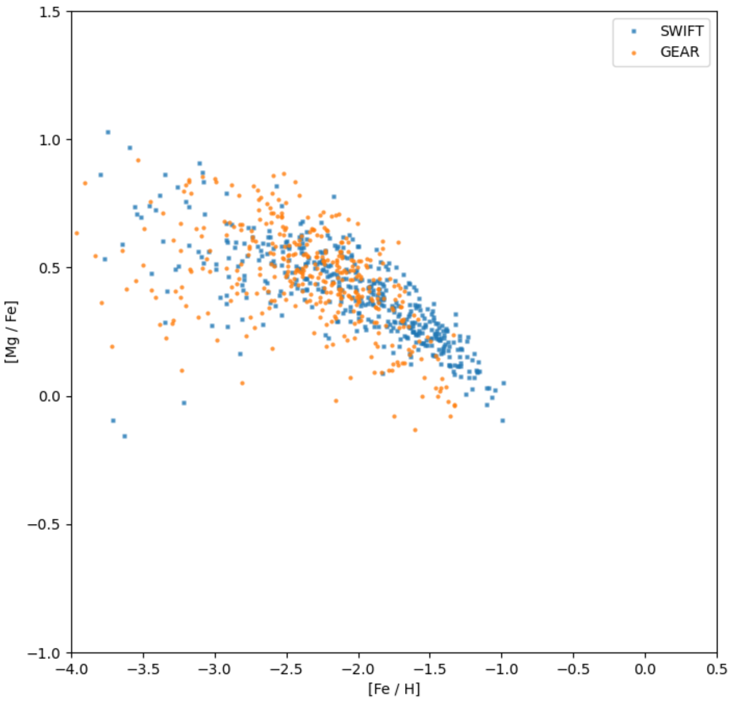


Figure 6 – Distribution of the stars for [Mg/Fe] vs [Fe/H] for both `SWIFT-GEAR` (in blue) and `GEAR` (in orange). At low metallicity, [Mg/Fe] forms a plateau at around 0.4 due to the SNII and then it decreases until almost 0 for this galaxy due to the SNIa. Both codes produce approximately the same distribution of stars.

Derivation of the IMF coefficients

The initial mass function (IMF) described in section 4.6.1 requires a normalization. First let's recall the function:

$$\xi(m) = B_i m^{-\alpha_i} \tag{1}$$

where

$$\begin{cases} \alpha_0 = 0.3, & 0.01 \leq m/M_\odot < 0.08, \\ \alpha_1 = 1.3, & 0.08 \leq m/M_\odot < 0.50, \\ \alpha_2 = 2.7, & 0.50 \leq m/M_\odot < 1.00, \\ \alpha_3 = 2.3, & 1.00 \leq m/M_\odot, \end{cases} \tag{2}$$

and B_i are the coefficients that we need to compute. As we wish the function to be continuous, a relation between the coefficients can be found from the boundaries between each part:

$$\lim_{x\to M_{i+1}^-} \xi(x) = B_i M_{i+1}^{-\alpha_i} = B_{i+1} M_{i+1}^{-\alpha_{i+1}} = \lim_{x\to M_{i+1}^+} \xi(x) \tag{3}$$

where M_i is the lower limit of the interval i and $M_i^\pm$ represents lower / upper limits. This results in the following equations: $B_{i+1} = B_i M_{i+1}^{\alpha_{i+1}-\alpha_i}$. We are still missing a last equation in order to solve this system. This equation is obtained from the normalization of the IMF:

$$\int_{M_0}^{M_{\max}} \xi(x)\mathrm{d}x = \sum_i \int_{M_i}^{M_{i+1}} B_i x^{-\alpha_i}\mathrm{d}x = 1 \tag{4}$$

where $M_{\max}$ (M_0) is the maximal (minimal) mass assumed for a star ($M_{\max} = 50M_\odot$ and $M_0 = 0.01M_\odot$ in `SWIFT-GEAR`). The final system of equations to solve is given by:

$$\begin{cases} B_{i+1} = B_i M_{i+1}^{\alpha_{i+1}-\alpha_i} \\ \displaystyle\sum_i \frac{B_i}{1-\alpha_i}\left(M_{i+1}^{1-\alpha_i} - M_i^{1-\alpha_i}\right) = 1 \end{cases} \tag{5}$$

and the solution is given in table 1. As the second equation is simply a normalization, this system can be easily solved in two steps. In the first one, we assume $B_0 = 1$ and find all the other coefficients. Then, using the second equation, all the coefficients are normalized. A script doing this computation is provided in the git repository hosting this thesis. In the case of the IMF in

Table 1 – Coefficients of the IMF presented in Kroupa, 2001.

Coefficient	Value for ξ	Value for ϕ
B_0	2.66	13.03
B_1	0.06	0.29
B_2	0.03	0.15
B_3	0.03	0.15

mass, the same computation can be done but with α_i replaced by $\alpha_i - 1$.

Brent Minimization

During the sampling of h_{eff} and σ_{NL} (Equation 9.33), we need to invert the correlation matrix (Equation 9.11 and corresponds to C in the previous equation) for each evaluation of the distribution. As the inversion is computationally heavy ($O(n^3)$ where n is the number of rows or columns in the matrix), the required number of inversion should be as low as possible. As both distributions are strongly peaked, it is enough to find the peak of the distribution and evaluate it at regular intervals until reaching a negligible probability. The final distribution can be interpolated from the computed data point.

The Brent minimization is an efficient way to find this peak (see Figure 7 for an example of minimization for a Gaussian in blue). The method consists of a mix between a parabolic interpolation and a golden section search. Each step starts with 3 points (1-3 in the figure) and computes the parabola passing through them (in black). If the maximum of the parabola is within the interval defined by 1 and 3, it replaces one of the boundary (4 replaces 3) otherwise a step of the golden search is performed. The golden search is very similar to a binary search. The idea is to use 4 points and not 3 to select the next interval as we are interested in finding the maximum and not a zero. This method takes its name due to the position of the next point within the selected interval ($d_{1-2}/d_{2-3} = \phi$ where d are the distance between two points and ϕ the golden number). This number arises from the restriction of having the same intervals when picking the next one in order to have the same convergence rate no matter the location of the peak. This is done until the difference between two successive points is low enough. The Brent minimization can accurately find the peak of our distributions in usually 5-6 evaluations of the distribution (relative error of 10^{-6}). It is worth to mention that all the evaluations are kept in memory and used for the final sampling.

The sampling is then completed by evaluating the function at fixed interval given by a fraction of the distribution's width. It is done from the peak until reaching a low probability defined as a given fraction of the peak (blue crosses). A quick and rough evaluation of the width is given by the standard deviation of the points used by the method. Finally, it is worth to mention that all the probabilities are computed in logarithm in order to suppress the overflows.

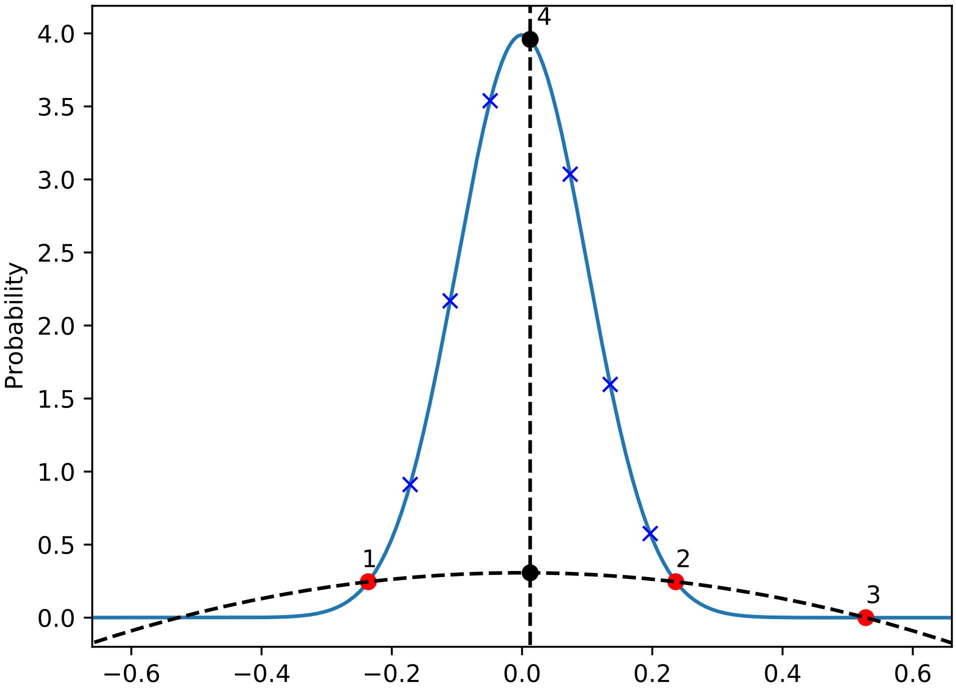


Figure 7 – Example of the Brent minimization for a Gaussian in blue. Starting from the interval $[-1, 1]$, the first 3 points are initialized using the golden ratio and then a parabolic search is done with the black lines. Thanks to the low accuracy required in this example, the point 4 is already close enough to the peak. Using the standard deviation of the x-coordinates of points 1 to 4, the size of the Gaussian is evaluated. This size is then used to sample the Gaussian on each side of 4 with the blue crosses until the probability is below a given fraction of the peak.