Dissertation

submitted to the

Combined Faculty of Mathematics, Engineering and Natural Sciences

of Heidelberg University, Germany

for the degree of

Doctor of Natural Sciences

Put forward by

M.Sc. Toni Peter

born in: Ludwigsburg

Oral examination: April 23, 2024

Understanding the Era of Reionization via Numerical Methods for Radiative Transfer

Referees: Prof. Dr. Ralf Klessen

Prof. Dr. Luca Amendola

Abstract

The goal of this thesis are numerical studies of the era of reionization, which took place at about a 150 million years to a billion years after the Big Bang. Since reionization is a process driven by radiation, a major fraction of this work is dedicated to numerical methods for radiative transfer. In particular, we develop the Sweep method, which allows us to study reionization within large cosmological simulations. We begin by introducing the basics of the Sweep algorithm and its implementation in the simulation code Arepo. We discuss the motivation behind it, how it integrates with the rest of Arepo and perform a number of tests to assess its performance and physical accuracy. We find that the Sweep method does not only produce physically accurate results, but does so in a very efficient manner, even when applied to large simulations on a large number of processors. We then proceed by introducing the standalone radiative transfer postprocessing code Subsweep in which we add a variety of improvements to the original Sweep method, in particular the addition of sub-timesteps. We perform a number of additional tests to verify that Subsweep correctly solves a number of physical problems and show that sub-timesteps can drastically improve performance of the Sweep algorithm when applied to problems with heterogeneous environments without sacrificing accuracy. Finally, we apply Subsweep to the cosmological simulation suite TNG in order to recreate the era of reionization in the TNG universe. We find that Subsweep allows us to study the spatial structure of reionization in detail and that we can reproduce the observational constraints on the history of reionization reasonably well.

Zusammenfassung

Das Ziel dieser Arbeit ist die numerische Untersuchung, des Zeitalters der Reionisierung, das etwa 150 Millionen bis eine Milliarde Jahre nach dem Urknall stattfand. Da die Reionisierung ein Prozess ist, der durch Strahlung vorangetrieben wird, widmet sich ein wesentlicher Teil dieser Arbeit der Entwiklung numerischer Methoden für den Strahlungstransport.

Insbesondere entwickeln wir die Sweep-Methode, die es uns ermöglicht, die Reionisierung innerhalb großer kosmologischer Simulationen zu untersuchen. Wir beginnen mit einer Einführung in die Grundlagen des Sweep-Algorithmus und seiner Implementierung im Simulationscode Arepo. Wir erläutern die Motivation hinter der Methode, wie sie sich mit dem Rest von Arepo integriert und führen eine Reihe von Tests durch, um ihre Effizienz und physikalische Genauigkeit zu bewerten. Wir stellen fest, dass die Sweep-Methode nicht nur physikalisch genaue Ergebnisse liefert, sondern dies auch auf sehr effiziente Weise tut, selbst wenn sie auf großen Simulationen mit einer Vielzahl von Prozessoren angewendet wird.

Daraufhin stellen wir den eigenständigen Strahlungstransport Code Subsweep vor, in dem wir eine Vielzahl von Verbesserungen gegenüber der originalen Sweep-Methode hinzufügen, insbesondere die Einführung einer Zeitschritt-Hierarchie. Wir führen eine Reihe zusätzlicher Tests durch, um zu überprüfen, dass Subsweep eine Reihe von physikalischen Problemen korrekt löst, und zeigen, dass die Zeitschritthierarchie die Effizient des Sweep-Algorithmus erheblich verbessern kann, wenn sie auf Probleme mit heterogenen Umgebungen angewendet werden, ohne dabei die Genauigkeit zu beeinträchtigen.

Schließlich wenden wir Subsweep auf die kosmologische Simulationsreihe TNG an um das Zeitalter der Reionisierung in dem TNG-Universum nachzubilden. Wir stellen fest, dass Subsweep es uns erlaubt die räumliche Struktur der Reionisierung im Detail nachzuvollziehen und das wir Daten aus Beobachtungen reproduzieren können.

Contents

Chapter 1

Introduction

1.1 Motivation

Radiation emitted from stars, accretion disks around black holes, supernovae and other astronomical objects is an important ingredient in understanding the structure of the universe. Not only is radiation the primary phenomenon by which we can observe the universe, but it also plays an important role in the evolution of the universe itself. The earliest light we are able to observe stems from the time of recombination and is measured today as the so-called cosmic microwave background which has given us detailed insight into the structure of the very early universe. Hundreds of millions of years later, the first stars formed and the light they emitted began to gradually affect and shape the universe around it and played a major role in creating the complex, heterogeneous universe we see today.

One of the processes which is believed to be caused by the light emitted from the first stars is the reionization of the intergalactic medium. After recombination, the universe consisted primarily of neutral hydrogen. As the first stars form, the high-energy light they emit begins to slowly ionize the hydrogen around them, forming ionized bubbles. As time passes and more and more stars and other radiating objects form, the bubbles continue growing in size until they begin to overlap and eventually cover essentially the entire universe. While we do have a basic understanding of how this process took place, there are still a number of open questions about reionization. For example, the timeline of reionization is still relatively unconstrained with estimates of the beginning ranging between $z = 15 - 40$ [Abel et al., 2002, Iliev et al., 2008, Wise and Abel, 2008] and the end being estimated to somewhere between $z \approx 5.5 - 8$. There is also some uncertainty as to how much reionization is driven by stars and how much of it is caused by other radiation sources, such as the accretion disks of active galactic nuclei.

Observations of the early universe play a crucial role in improving our understanding of reionization quantitatively and qualitatively. However, numerical simulations of these non-linear processes are another important tool that can not only provide valuable information about the process itself, but can also tell us whether our understanding of the effects causing it is correct.

In this thesis, we will focus on studying reionization using numerical methods. The numerical treatment of the equations that describe the transport of radiation from the sources to the gas surrounding it is a challenging problem. Therefore, this work focuses to a large degree on the numerical methods themselves. In particular, we introduce the Sweep method for radiative transfer, verify its correctness in a number of test problems and describe its advantages. We first discuss the original implementation of the Sweep method in the simulation code Arepo [Springel, 2010b]. Afterwards, we extend the discussion to the standalone postprocessing code Subsweep which includes an extended Sweep method which introduces a method for performing sub-timesteps in coupled systems of radiative transfer and radiation chemistry.

After the discussion of the Sweep method and of related numerical methods, we will bring our focus to the application of Subsweep in order to study Reionization on the TNG simulations [Marinacci et al., 2018, Naiman et al., 2018, Nelson et al., 2018, 2019, 2021, Pillepich et al., 2018, 2019, Springel et al., 2018].

1.2 Outline

This work is split into six chapters.

In this first chapter, we give the motivation behind this work and briefly present the overall structure of the thesis.

The second chapter is concerned with the available theories and methods underlying this work. In the first half of the second chapter, we give a broad overview of the theoretical background of cosmology in general and reionization in particular. We briefly discuss the observational evidence and the leading cosmological model ΛCDM. The second half discusses some commonly used simulation methods in Astrophysics for numerical studies of Gravitation, Hydrodynamics and Radiative Transfer.

The third chapter is based on a publication on the Sweep algorithm for Astrophysics. It introduces the concept of transport sweeps as an effective solution for Radiative Transfer and discusses our implementation of the method in the simulation code Arepo. We show a number of tests performed with the method in order to show that it produces physically accurate results efficiently, even for large problem sets.

The fourth chapter introduces the standalone code Subsweep for radiative transfer

postprocessing and discusses its implementation details as well the sub-timestepping procedure, which is the major improvement to the Sweep method that Subsweep introduces. We test Subsweep in a variety of physical setups and find that it produces accurate results and that sub-timesteps can drastically improve performance, particularly for large simulations with highly heterogeneous media.

In the fifth chapter, we employ Subsweep in order to study the era of reionization in a set of postprocessing simulations performed on results of the TNG simulation suite. We find that our code reproduces the history of reionization reasonably well and compare the results to a number of observational constraints.

Finally, the sixth chapter concludes this thesis and provides a brief discussion of interesting directions for future research.

Chapter 2

Theory

In this chapter, we give a brief, general overview of the theoretical framework of cosmology in context and then focus in particular on the era of reionization.

2.1 Cosmology

Physical cosmology is the study of the evolution of the universe from its very early beginnings to the present time and beyond. In the last century, this field has made a lot of advances culminating in what is now seen as the standard model of cosmology, ΛCDM.

2.1.1 ΛCDM

One of the most fundamental realizations in physical cosmology is the observation that galaxies appear to move away from us, with more distant galaxies moving away faster than those nearer to us [Hubble, 1929, Lemaître, 1927]. Under the assumption of the cosmological principle, which states that the we humans do not live in a particularly special place with respect to the fundamental laws of physics, the remaining, natural interpretation of this observation is that the universe is continuously expanding, such that even objects that are at rest will be observed as receding from us.

This expansion is expressed as the time evolution of the scale factor $a(t)$, which determines physical distance. Given a comoving distance D, that is a distance that is not subject to the expansion of the universe, we find that the corresponding physical distance is $d = Da(t)$. The scale factor is defined such that at the time of the Big Bang t_{B}, $a(t_{\mathrm{B}}) = 0$ whereas at the present time ($t = 0$), $a(0) = 1$. The Hubble parameter H is the quantity describing the rate of expansion and is defined by $H(t) = \frac{\dot{a}(t)}{a(t)}$. We also define the redshift $z = \frac{1}{a} - 1$, which is a convenient quantity to use as a measure

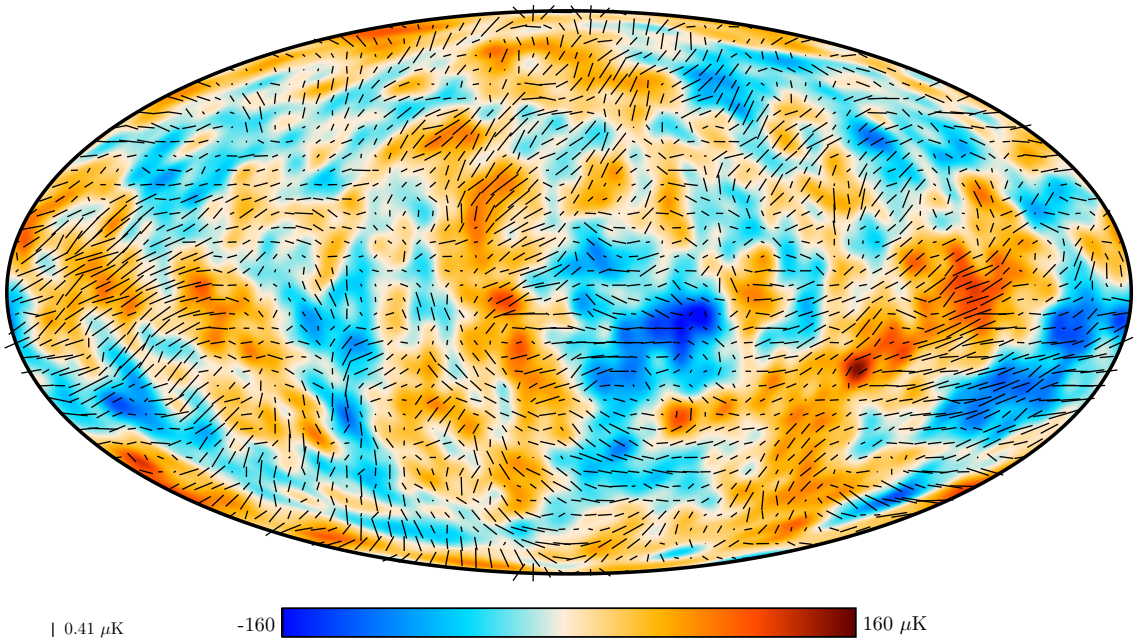

Figure 2.1. From Collaboration [2020]: The 2018 Planck map of the temperature anisotropies of the CMB. Color scale shows the magnitude of the anisotropy, while the black lines show the polarization.

of cosmological time, since it can be directly observed and does not depend on the parameters of the underlying model.

The current leading theory of cosmology is the Lambda-Cold Dark Matter model (ΛCDM), in which the universe is described as flat and homogeneous (on large scales) and with its behavior driven by the primary constituents of dark energy (Λ), cold dark matter (CDM) and ordinary, baryonic matter.

In ΛCDM, the evolution of the Hubble parameter is given by

$$H(t)^2 = H(0)^2 \left(\Omega_\Lambda a^{-3} + \Omega_\mathrm{k} a^{-2} + (\Omega_\mathrm{m} + \Omega_\mathrm{b}) a^{-3} + \Omega_\mathrm{rad} a^{-4}\right), \tag{2.1}$$

where $\Omega_{\Lambda,0}$ is the dark energy density parameter, $\Omega_{\mathrm{k},0}$ is the curvature parameter, $\Omega_{\mathrm{m},0}$ is the matter energy density and $\Omega_{\mathrm{rad},0}$ is the radiation energy density parameter.

The predictions made by this equation have made it possible to strongly constraint the cosmological parameters via observations. For example, the most recent observations of the angular power spectrum of fluctuations in the Cosmic Microwave Background (CMB) by the Planck missions [Collaboration, 2020] give values of $H_0 = (67.4 \pm 0.5)\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$, $\Omega_\mathrm{m} = 0.315 \pm 0.007$, $\Omega_\mathrm{lambda} = 0.679 \pm 0.013$, $\Omega_\mathrm{b} = 0.049\,00 \pm 0.000\,48$, and $\sigma_8 = 0.811 \pm 0.006$, where H_0 is the Hubble constant, Ω_m is the matter density parameter, Ω_Λ is the dark energy density parameter, Ω_b is the baryon mass density parameter and σ_8 is the matter fluctuation parameter.

Other methods for constraining the cosmological parameters include

1. The evolution of the supernova luminosity as a function of time [Perlmutter et al., 1999].

2. Baryonic acoustic oscillations [Eisenstein and Hu, 1998, Viel et al., 2002] used as a standard ruler.

3. Comparison of theoretical predictions and observations of the abundance of massive haloes [Peebles et al., 1989].

4. The relative mass fractions of species in Big Bang Nucleosynthesis [Cyburt et al., 2016, Walker et al., 1991].

2.1.2 Evolution of the universe in ΛCDM

In the early stages, the ΛCDM universe consisted primarily of low-velocity dark matter, with density perturbations described by the primordial power spectrum which describes the density fluctuations as a function of length scale and is characterized by the slope $n_s = 0.962$. The present day universe is much more heterogeneous, with most of the objects having been aggregated onto the filaments that describe the large scale structure of the universe. At the intersection of filaments are galaxy clusters, which themselves contain hundreds or thousands of galaxies, each of which contains, on average, hundreds of billions of stars. A major ongoing research question is understanding the formation of these large scale structures. The currently prevailing picture is that large scale structures form from the bottom up - from smaller scale structures.

In the very early stages of the universe, the expansion of space causes a gradual, adiabatic cooling of the hot matter that is distributed homogeneously throughout space. At t = 400.000 years, it has cooled enough for radiation to decouple from matter, thus allowing protons and electrons to recombine into hydrogen atoms. The radiation originating from this process is responsible for the cosmic microwave background which we observe today, redshifted by a factor of z = 1100. After this point, the universe contains no luminous sources and is therefore dark, an era called the Dark Ages. Small quantum fluctuations in the matter distribution which formed after the Big Bang lead to small overdensities, which become larger and larger over time due to gravity pulling more and more mass into overdense regions. These overdensities will eventually become large enough to form the first galaxies and galaxy clusters. The stars in these galaxies create light, ending the Dark Ages and beginning the process of reionization.

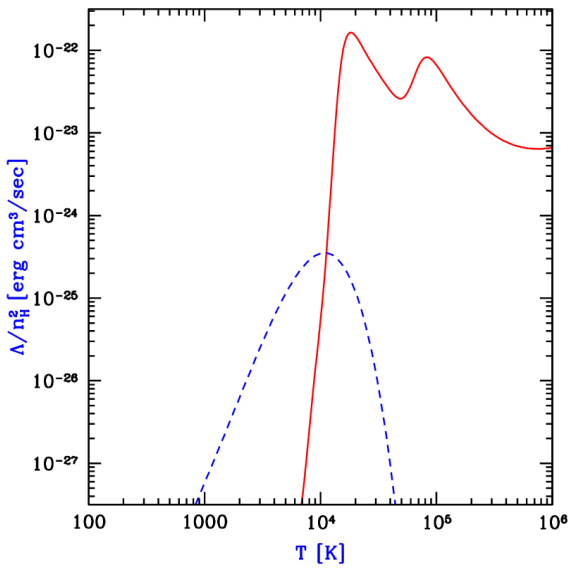

Figure 2.2. From [Barkana and Loeb, 2001]: Cooling rates in erg cm^3 s^{-1} for atomic hydrogen (solid, red line) and molecular hydrogen (dashed, blue line) as a function of temperature.

2.2 Star formation

Understanding star formation is an active area of research and is difficult even in the present-day universe. Our current understanding of star formation in the early universe is based on the theory of present-day star formation. The currently prevailing theory is that stars form when large molecular clouds, which were previously stabilized against gravitational collapse by their own gas pressure exceed a critical mass and begin collapsing. Since gas pressure increases with the temperature of the gas, efficient cooling is an essential ingredient of star formation. As shown in Fig. 2.2, in pristine gas, the primary coolants are molecular and atomic hydrogen for $T < 1 \times 10^4\,\mathrm{K}$ and $T > 1 \times 10^4\,\mathrm{K}$ respectively, and there are no efficient coolants for gas below $T < 1 \times 10^3\,\mathrm{K}$, which prevents clouds below virial temperatures of $1 \times 10^3\,\mathrm{K}$ from collapsing. When the gas is enriched with metals, more cooling processes become available, which in turn enables smaller and smaller clouds to collapse, increasing star formation efficiency.

This is the underlying reason for the observation that the first generations of stars are much more massive than average stars today - the extremely low metallicities in the primordial gas prevented smaller stars from forming.

From cosmological simulations, we estimate the redshift at which the first stars form to be around $z = 20$ to $z = 30$ [Yoshida, 2019].

2.3 Reionization

As the first stars form, their photons begin ionizing the surrounding neutral hydrogen in the IGM. Over time, the entire universe becomes ionized, in a process called reionization. It is still an open question when reionization took place, how it proceeded and which radiation sources were responsible for it. Predictions for the beginning of reionization range from $z \approx 30 - 40$ [Iliev et al., 2008, Wise and Abel, 2008] to $z \approx 15 - 20$ [Abel et al., 2002]. Reionization initially forms isolated, ionized bubbles which then expand and eventually cover the entire universe [Iliev et al., 2008, Lidz et al., 2007]. These timings are based on predictions made by cosmological simulations and therefore depend quite strongly on model specific details and parameters.

2.3.1 Observational evidence

Gunn-Peterson Troughs

Observational evidence of reionization comes from multiple sources. One example are Gunn-Peterson Troughs [Gunn and Peterson, 1965] in the spectra of high redshift quasars. As light from a quasar encounters neutral hydrogen, an absorption line is created at the Lyman-α wavelength. Since the spectrum is continuously redshifted as it travels, each cloud of neutral hydrogen causes an absorption line at a different wavelength, creating a feature that is called a Lyman-α forest. From this spectrum, it can be inferred at which redshift the light from the quasar stopped encountering neutral hydrogen clouds, which in turn provides an estimate for the timing of the end of reionization.

The Gunn-Peterson optical depth to Lyman-α photons is given by

$$\tau = 4.9 \times 10^5 \left(\frac{\Omega_0 h^2}{0.13}\right)^{-0.5} \left(\frac{\Omega_\mathrm{b} h^2}{0.02}\right) \frac{1+z}{7} x_\mathrm{HI}, \tag{2.2}$$

where Ω_0 is the matter density of the universe and $h = \frac{H}{100\,\mathrm{km\,s^{-1}\,Mpc^{-1}}}$. Even for small x_HI, the optical depth becomes large, so that the signal is fully absorbed. This means that the optical depth can only be used to probe the final stages of reionization, where the gas is mostly ionized already.

Results using this method estimate the end of reionization at around $z = 5.5$ to 6 [Becker et al., 2001, Fan et al., 2002, 2006a,b].

Cosmic Microwave Background

The Cosmic Microwave Background (CMB) is a remnant of recombination. Before recombination, photons had low mean free path lengths due to Thomson Scattering

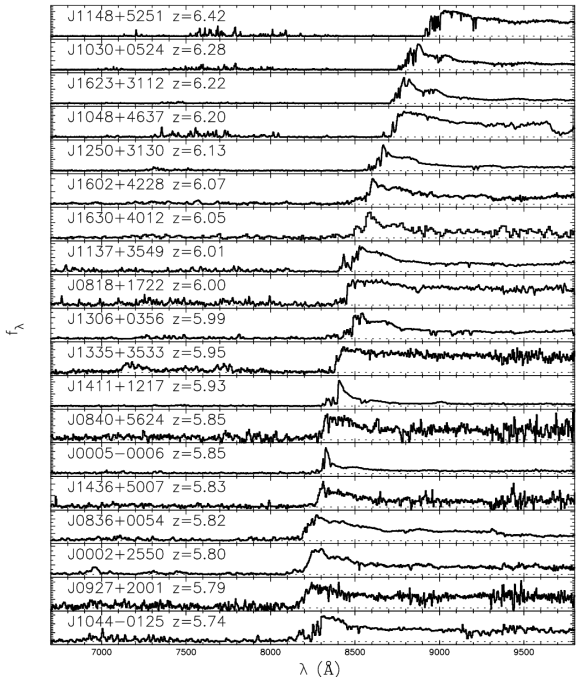

Figure 2.3. From Fan et al. [2006b]: Spectra of nineteen quasars between $5.74 < z < 6.42$, clearly exhibiting Gunn-Peterson troughs at different wavelengths.

on the free electrons of the ionized hydrogen. After recombination, hydrogen was neutral so that photons could travel freely. The photons of the CMB follow a black body spectrum with temperature $T = 2.725(1 + z)$ K with small fluctuations on the order of 1×10^{-5} K.

If electrons are created due to reionization, those electrons can cause Thomson scattering of photons which in turn causes a damping proportional to $e^{-\tau_{\mathrm{es}}}$ where τ_{es} is the Thomson scattering optical depth. Via this effect, the patchy nature of ionized regions during reionization cause a measurable effect on the anisotropy of the CMB [Doré et al., 2007, Santos et al., 2003].

Another effect of Thomson Scattering off of free electrons is the polarization of the CMB. There are two modes of polarization, E-mode and B-mode, in analogy to electrostatics. With respect to reionization, the important mode is the E-mode which has a vanishing curl and is created through Thomson scattering. The polarization introduces anisotropies at large angular scales in the power spectrum which can be used to estimate the timing of reionization.

Combining these CMB observations leads to estimates of the redshift of reionization of z=7.68+-0.79 [Planck Collaboration, 2014], where it is assumed that reionization takes place instantly everywhere.

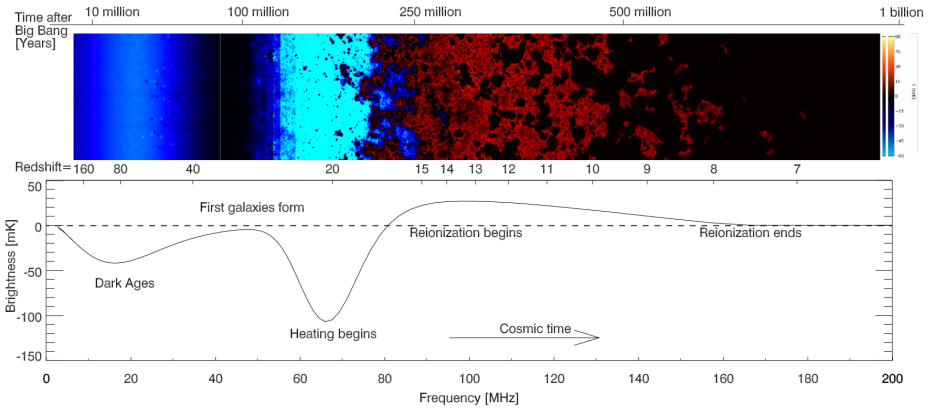

Figure 2.4. From Pritchard and Loeb [2012]: The upper panel shows representative slices through a cosmological simulation for the corresponding time. The bottom panel shows the 21-cm signal shown as the brightness temperature relative to the CMB temperature as a function of frequency at which the signal is received.

21cm Line

Possibly the most accurate observational evidence of reionization is provided by the 21cm line. This line corresponds to a hyperfine spin-flip transition in the ground state of neutral hydrogen with a energy differential of $E = \frac{hc}{21\,\mathrm{cm}} = 5.9\,\mu\mathrm{eV}$. Since the signal is at radio wavelengths, it can be observed from the ground. Using this signal, a 3D map of the neutral hydrogen distribution can be created by mapping the 21cm line according to redshift as $\lambda(z) = 21\,\mathrm{cm}(1+z)$.

The brightness temperature T_B of the 21cm line follows the proportionality

$$T_\mathrm{B} \propto 7\,\mathrm{mK}(1+\delta)x_\mathrm{HI}\left(1-\frac{T_\mathrm{CMB}}{T_\mathrm{S}}\right)(1+z)^{1/2}, \tag{2.3}$$

where T_CMB is the CMB temperature and T_S is the spin temperature of hydrogen. Its complicated evolution is shown in Fig. 2.4. Initially, at very high redshifts $z > 200$, the spin temperature is coupled to the CMB temperature, so that the brightness temperature is 0 and no 21cm signal is detected. At lower redshifts, $200 > z > 40$, the hydrogen cools and the spin temperature decreases, so that the cold gas can be seen in absorption relative to the CMB. The first stars create Lyman-α photons which lead to a strong coupling between the spin temperature and the hydrogen gas temperature. Initially, this leads to an absorption signal, but as the gas heats, it becomes an emission signal.

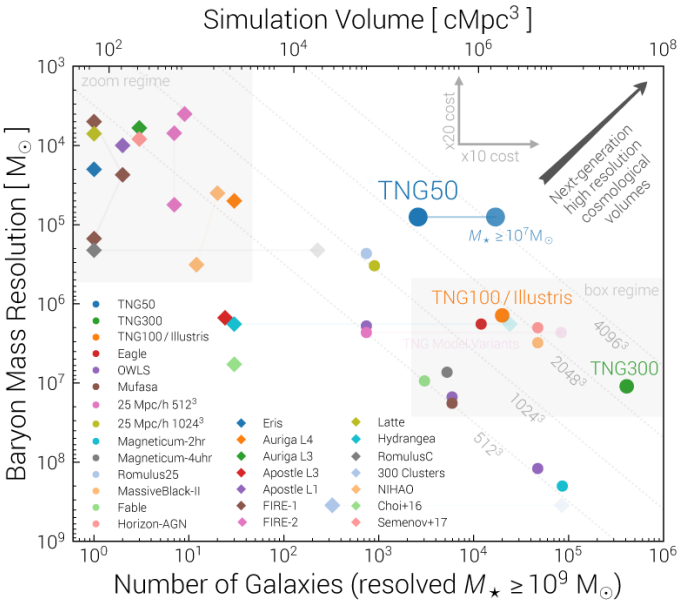

Figure 2.5. From Nelson et al. [2019]: Comparison of a variety of large cosmological simulations along two primary axes: The cosmological volume / the number of resolved galaxies, as well as the mass resolution.

2.4 Simulating Reionization

One of the best sources of constraints for reionization is given by cosmological simulations. Here, we are going to discuss some of the challenges that arise in these simulations.

2.4.1 Size and Resolution

On one hand, the size of a cosmological simulation needs to be sufficiently large in order to be representative for the whole universe and to reduce the error introduced through cosmic variance. The typical order of magnitude for a desirable box size is about 100 Mpc [Iliev et al., 2014, Shin et al., 2008]. A look at reionization in particular yields a similar estimate, with sizes of ionized bubbles being about 100 Mpc at $z = 6$.

On the other hand, as shown in Fig. 2.5, if computational cost is kept fixed, increasing the size of the box implies higher mass of the individual elements in the simulation and therefore worse resolution of the smallest structures in the simulation (typically galaxies), which implies an overall reduction of accuracy of the simulation.

2.4.2 Escape Fractions

Due to the limited ability of even modern computing hardware to simultaneously accurately resolve the interstellar medium at comparatively small length scales while still simulating cosmologically relevant box sizes, many relevant physical processes, such as star formation and supernovae are implemented by sub-grid models (see, for example, Springel and Hernquist [2003]) which provide an effective way to represent the physics of processes taking place at length scales below the resolution limit.

One such effect that is particularly relevant to understanding reionization is the absorption of UV radiation before it leaves a galaxy, giving rise to the escape fraction f_{esc} defined as the fraction of photons leaving the galaxy to the total number of photons that were emitted originally. The value of the escape fraction depends on galaxy properties, due to different dust opacities and densities and strongly constraining its value is difficult. Observational evidence on Lyman-α radiation gives an approximate dependence on the redshift that behaves as $f_{\mathrm{esc}} \propto (1+z)^{2.6\pm0.2}$ [Hayes et al., 2011] between $0.3 < z < 6.0$. In cosmological simulations, it is often taken to be between 10% and 20% [Finkelstein et al., 2019, Robertson et al., 2015].

2.5 Simulation methods

2.5.1 Introduction

One of the most useful tools for understanding the evolution of the early universe are computer simulations. In these, the equations describing the relevant physical phenomena such as gravity and hydrodynamics are solved in a discretized way and the system is evolved for the desired amount of time. Even simulations which contain only dark matter, that is, matter that does not interact with itself other than through gravity, replicate the large-scale structure of the universe well and have proved invaluable in aiding our understanding of cosmology [Efstathiou et al., 1985, Jenkins et al., 2001, Navarro et al., 1997]. However, if more accuracy and smaller length scales are desired, including the physics of baryons, such as hydrodynamics and radiation is necessary. In this section we will give a brief overview over some of the approaches to solving the different equations.

2.5.2 Gravity

Numerical techniques for simulations in which point-like objects interact via gravity, often called N-body simulations have a long history. The primary objective of all

these techniques is to compute the force on the j-th particle F_j, given by

$$\boldsymbol{F}_j = \sum_{i=1}^{N} \boldsymbol{F}_{ij}, \tag{2.4}$$

where F_{ij} is the gravitational force exerted by particle j on the particle i, which is given by the Newtonian force

$$\boldsymbol{F}_{ij} = -Gm_i m_j \frac{\boldsymbol{r}_i - \boldsymbol{r}_j}{|\boldsymbol{r}_i - \boldsymbol{r}_j|^3}, \tag{2.5}$$

where G is the gravitational constant, m_i and m_j are the masses and $\boldsymbol{r}_i$, $\boldsymbol{r}_j$ are the positions of the respective particles.

It should be noted that for our purposes, we will limit ourselves to low enough densities, such that any effects of general relativity can be neglected and we are allowed to treat gravity in a purely Newtonian framework.

The trivial and exact method for solving Equation 2.5 in which each force component F_{ij} is computed individually is simple to implement but prohibitively expensive for anything but the smallest number of particles, due to its computational cost scaling with $\mathcal{O}(n^2)$.

In practice, approximate methods are used in which the force terms are not computed exactly. It can be shown that errors introduced by such approximations can be contained reasonably well [Hernquist et al., 1993].

A well-known class of methods for the approximation of the force terms in Equation 2.5 is given by hierarchical multipole methods that are based on spatial tree structures. The basic idea here is to observe that the force exerted on a particle by a group of particles can be expressed in terms of a multipole expansion (up to a desired order) and that the error from this approximation can be kept under a desired threshold as long as the opening angle (the angle under which the single particle sees the group) is small enough. With this idea, large groups of particles can be grouped together and computed at once, instead of having to compute the force terms individually for each particle in the group.

In order to make use of this approximation, one often employs hierarchical decomposition of the space, using, for example oct-trees, as in the original Barnes-Hut tree paper [Barnes and Hut, 1986], but other tree structures such as KD-trees [Stadel, 2001] can be used as well.

In another class of methods [Hockney and Eastwood, 2021], an auxiliary mesh is constructed. Mesh cells are then assigned a density via a variety of assignment schemes, such as nearest-grid-point, cloud-in-cell, etc. which vary primarily in the number of cells that the mass is distributed into and the weights given to each cell

involved in the assignment. Using the density field obtained in this way, the Poisson equation for the gravitational potential

$$\boldsymbol{\nabla}^2\Phi = 4\pi G\rho, \tag{2.6}$$

is then solved.

Methods for solving the Poisson equation fall into two main categories. The first category is given by iterative solvers in which the problem is solved in real space, such as the simple Gauss-Seidel method. Often, one employs strategies such as the Multigrid method [Brandt, 1977], in which the equation is solved on a number of different refinement levels of the grid in order to deal with the long-ranged nature of the Poisson equation. Intuitively, the coarser refinement levels in a Multigrid method help with quickly approximating the large-scale solution of the equation, while the finest levels solve the short length scales.

The second category of solvers for the Poisson equations solve the problem by performing a Fourier transform. The intuition behind this idea is given by the fact that the solution to the Poisson equation can be expressed as a convolution of the Greens function with the gravitational potential. In Fourier-space, the expensive operation of a convolution turns into multiplication, which can be efficiently calculated. In order to make the Fourier transform itself fast enough, algorithms such as the fast Fourier Transform (FFT) are used [Cooley and Tukey, 1965].

As a last note on this topic, it should be stated that many state-of-the-art codes for solving the gravitational N-body problem combine these approaches, such that each approach can be used in the area where it performs best. One famous example of this is the TreePM method that combines Tree based solvers for the evaluation of short-range forces of highly clustered particles with the extremely high performance of Fourier-transform-based Particle-Mesh for evaluation of the long-ranged forces [Bagla, 2002, Xu, 1995].

2.5.3 Hydrodynamics

For our purposes, the equations of hydrodynamics are the Euler equations given by three equations that describe the conservation of mass, momentum and energy respectively. Written in vector notation, the equations read

$$\frac{\partial}{\partial t}\begin{pmatrix}\rho\\ \rho\boldsymbol{v}\\ \rho\end{pmatrix} + \boldsymbol{\nabla}\begin{pmatrix}\rho\boldsymbol{v}\\ \rho\boldsymbol{v}\boldsymbol{v}^T + p\\ (\rho e + p)\boldsymbol{v}\end{pmatrix} = 0 \tag{2.7}$$

where ρ is the density of the fluid, $\boldsymbol{v}$ is the velocity field, $e = u + \frac{\boldsymbol{v}^2}{2}$ is the total energy per mass of the fluid with the thermal energy per mass u and p is the pressure of the gas which can be derived by an equation of state, for example

$$p = (\gamma - 1)\rho u \tag{2.8}$$

where γ is the adiabatic index defined by the ratio of specific heats $\gamma = \frac{c_\mathrm{p}}{c_\mathrm{v}}$.

Traditionally, there are two main categories of approaches to solving Equation 2.7.

In the Eulerian approach, the equations are discretized onto a fixed grid that does not move with the fluid. In order to allow the mesh to adapt to large variations in the density, a technique called adaptive mesh refinement is often used, in which the mesh is refined locally wherever the density exceeds a certain threshold. Eulerian approaches are often simple to implement and yield relatively good results. However, they suffer from their lack of Galilean invariance, such that results change depending on whether a bulk velocity is present or not [Tasker et al., 2008, Wadsley et al., 2008], as well as from their inherent mixing behavior, which introduces unphysical entropy into the system [Trac et al., 2007] and the structure of the grid can introduce artificial, undesired symmetries into the system.

The other main category of solvers uses a Lagrangian approach, in which the fluid is discretized by a set of particles which follow the flow velocity. Lagrangian approaches have a major advantage in that they adapt to the density variations inherent in cosmological simulations and automatically refine regions of high densities. However, they struggle with resolving shocks and can suppress fluid instabilities [Agertz et al., 2007].

The simulation code Arepo [Springel, 2010a,b] takes a hybrid approach to the problem of fluid dynamics in which the equations are solved in the Eulerian way on an unstructured Voronoi mesh which is generated by a set of Lagrangian points which move along the velocity field. This method, while difficult to implement, combines the benefits of Eulerian and Lagrangian approaches and resolves shocks well without introducing artificial symmetries.

2.5.4 Radiative transfer

In cosmological simulations, the proper treatment of radiation emitted from stars, galaxies, AGN, etc. is an important aspect for two reasons. Firstly, radiation is one of our best observational probes into the early universe and being able to incorporate radiation into the simulations facilitates comparison between simulations and observations. Secondly, the radiation and its interaction with matter plays a key role in the formation of many astrophysical objects, such as stars, black holes and galaxies.

However, computationally, solving radiative transfer in cosmological simulations remains a challenging problem for a variety of reasons. The most fundamental reason is the high dimensionality of the quantity of interest, specific intensity, and the partial differential equation describing its behavior, with three spatial dimensions, one temporal dimension, two angular dimensions (neglecting polarization), and a frequency dimension, adding up to a total of seven dimensions. Another reason is that the radiative transfer equation (RTE) itself changes its nature from being an elliptical partial differential equation in regions in which scattering dominates, to a hyperbolic equation in regions with long mean free paths. This makes development of a numerical solver which can efficiently solve the equation in all physical regimes particularly challenging. Moreover, most of the time, we are not just interested in the radiation field, but in its effect on the matter in our simulations. In practice, this means that we do not need to solve only the RTE, but the coupled system of the RTE and the radiation chemistry equations which describe how the photons are absorbed, scattered and created by the gas and in turn ionize, heat or otherwise modify the state of the gas.

The relevant quantity in radiative transfer is the specific radiative intensity $I_\nu(\boldsymbol{r}, t, \hat{\Omega})$, with frequency ν, spatial position $\boldsymbol{r}$, time t and solid angle $\hat{\Omega}$ given in units of $\mathrm{W\,m^{-2}\,sr^{-1}\,Hz^{-1}}$. The RTE is given by [Rybicki and Lightman, 1985]

$$\frac{1}{c}\frac{\partial}{\partial t} I_\nu + \hat{\boldsymbol{\Omega}} \cdot \boldsymbol{\nabla} I_\nu = j_\nu - (\bar{k}_{\nu,\mathrm{s}} + \bar{k}_{\nu,\mathrm{a}}) I_\nu + \frac{1}{4\pi} \int_S k_{\nu,\mathrm{s}}(\boldsymbol{\Omega'}) I_\nu \mathbf{d}\boldsymbol{\Omega'}. \tag{2.9}$$

Here, c is the speed of light, j_ν is the emission coefficient, $\bar{k}_{\nu,\mathrm{a}}$ is the absorption coefficient, $\int_S$ denotes the integral over the unit sphere with $\boldsymbol{\Omega'}$ being the solid angle relative to $\hat{\boldsymbol{\Omega}}$. The equation equates the rate of change of the radiative intensity along the direction $\hat{\boldsymbol{\Omega}}$ to a number of local processes which affect the intensity, namely absorption, scattering and sources. The total scattering coefficient $\bar{k}_{\nu,\mathrm{s}}$ is defined via the angle-dependent scattering coefficient $k_{\nu,\mathrm{s}}(\boldsymbol{\Omega'})$ as $\bar{k}_{\nu,\mathrm{s}} = \int_S k_{\nu,\mathrm{s}}(\boldsymbol{\Omega'}) \mathbf{d}\boldsymbol{\Omega'}$. For simplicity, we will assume isotropic scattering such that $k_{\nu,\mathrm{s}}(\boldsymbol{\Omega}) = \bar{k}_{\nu,\mathrm{s}}$. If the material coefficients (the source term and the absorption and scattering coefficients) change on timescales which are smaller than the typical light crossing time of the system, an often made assumption is the so-called infinite speed of light approximation in which we drop the first term of Eq. 2.9, obtaining

$$\hat{\boldsymbol{\Omega}} \cdot \boldsymbol{\nabla} I_\nu = j_\nu - (\bar{k}_{\nu,\mathrm{s}} + \bar{k}_{\nu,\mathrm{a}}) I_\nu + \frac{\bar{k}_{\nu,\mathrm{s}}}{4\pi} \int_S I_\nu \mathbf{d}\boldsymbol{\Omega'}. \tag{2.10}$$

A common approximation method in order to make the RTE more tractable is the so-called source iteration, in which the RTE is solved first under the assumption of

no scattering, i.e.

$$\hat{\mathbf{\Omega}} \cdot \boldsymbol{\nabla} I_\nu^0 = j_\nu^0 - (\bar{k}_{\nu,\mathrm{a}} + \bar{k}_{\nu,\mathrm{s}}) I_\nu^0, \tag{2.11}$$

where $j_\nu^0 = j_\nu$.

We then use the obtained intensity I_ν^0 to compute the local scattering terms given by the integral on the right-hand side of Eq. 2.10, which we then re-introduce as an effective source term $j_\nu^1 = j_\nu^0 + \frac{\bar{k}_{\nu,\mathrm{s}}}{4\pi} \int_S I_\nu \mathbf{d\Omega'}$. Given this new source term, we then solve eq. Eq. 2.11 again, to obtain I_ν^1, which will be used to compute j_ν^2, and so on. This process is iterated until the intensity converges.

Using source iteration, the remaining problem is solving the scattering-less RTE. In the following, we will discuss some common methods for solving this problem.

Long characteristics

One of the most intuitive methods for solving radiative transfer is the long-characteristics method, which is also often called raytracing [Abel and Wandelt, 2002, Abel et al., 1999a, Mihalas and Weibel-Mihalas, 1999]. Here, the intensity is computed by explicitly integrating the RTE along rays from each source to each cell. This method has the advantage of being relatively straightforward to implement and highly accurate. However, in practice it becomes expensive quite quickly, since the runtime of the algorithm depends linearly on the number of sources and it is difficult to parallelize since tracing a ray through the system requires non-local information.

A modification of this method, the short-characteristics method traces rays only up towards a certain, reasonable distance which makes the method easier to parallelize and cuts the cost of tracing the rays by a large amount.

Moment-based methods

In moment-based methods, the RTE isn't solved directly. Instead, a number of moments of the equations are computed. For example, the first moment of the time-dependent RTE is obtained by integrating Eq. 3.2 over a frequency range ν_1 to ν_2 and over $\mathbf{\Omega}$, and given by

$$\frac{\partial E}{\partial t} + \nabla \boldsymbol{F} = J - K \rho c E, \tag{2.12}$$

where $E = \int_S I_\nu(r, t, \mathbf{\Omega}) \mathrm{d}\mathbf{\Omega}$ is the zeroth moment of intensity, namely the radiation energy density, $\boldsymbol{F}$ is the radiative flux, J is the source energy density and K is the mean opacity between v_1 and v_2 and ρ is the local gas mass density. In order to solve the resulting moment equations, a closure relation is needed which relates two

of these moments. For example, in the flux limited diffusion approach [Levermore and Pomraning, 1981], the closure relation relates the flux to the energy density via $\boldsymbol{F} = \frac{-c\boldsymbol{\nabla}E}{3K\rho}$. While flux limited diffusion works well in optically thick media, it does not conserve photon direction in optically thin media and therefore also fails to form shadows behind dense objects. Another commonly used method, the M1 method [Dubroca and Feugeas, 1999, Levermore and Pomraning, 1981, Ripoll et al., 2001] assumes a different closure relation between the second moment (the radiation pressure tensor $\mathbb{P}$ and the energy density given by

$$\mathbb{P} = E\left(\frac{1-\chi}{2}\mathbb{I} + \frac{3\chi-1}{2}\boldsymbol{n}\otimes\boldsymbol{n}\right), \tag{2.13}$$

where $\boldsymbol{n} = \frac{\boldsymbol{F}}{|\boldsymbol{F}|}$, $\chi = \frac{3+4f^2}{5+2\sqrt{4-3f^2}}$ and $f = \frac{|\boldsymbol{F}|}{cE}$.

This closure relation is a good trade off between requiring only local quantities to be computed, while still conserving photon transport direction reasonably well and has been used successfully in many applications [Bieri et al., 2017, Costa et al., 2018, Rosdahl and Teyssier, 2015].

One problem with moment based methods in general is that information travels only one or a few cells during each timestep. While this is completely acceptable in many scenarios, it does cause problems in media with very long mean free paths and high resolution, where photons might have to travel many cells before they are absorbed, so that solving the system becomes computationally expensive.

Simplex

The SimpleX algorithm [Kruip et al., 2010, Ritzerveld and Icke, 2006] is another approach which can be considered similar to a short-characteristics method. In this method, radiative transfer is performed on a Delaunay triangulation generated from points which are sampled such that they are equidistant in mean free paths, which simplifies the transport step. In this method, photons are transported as discrete packages which travel from cell to cell via a set of transport rules that are chosen depending on the setup of the simulation.

For our purposes, the Simplex methods suffers from a similar problem to that of moment based methods, in that transporting photons in situations with long mean free paths can become prohibitively expensive.

Monte-Carlo methods

Another class of methods is given by Monte-Carlo methods [Dullemond et al., 2012, Noebauer and Sim, 2019, Oxley and Woolfson, 2003]. Here, the RTE is not discretized

but instead the radiation is represented by a number of test particles (often called photon packets) which are introduced into the simulation. Individual test particles can move, scatter, and be absorbed. Which of these processes takes place is determined stochastically by choosing from an appropriate probability distribution, such that the resulting bulk behavior of the photons represented by these test particles replicates the solution to the radiative transfer equations. A primary benefit of such methods is that they are often very intuitive to think about and implement. In particular, scattering is a process that is very easily handled by Monte-Carlo methods while it can represent a massive challenge for other methods.

Monte-Carlo methods also have the benefit of being easy to parallelize. As long as the spatial domain decomposition assigns an even distribution of radiative transfer work to each processor, the most intuitive approach to parallelization is also a good one - distribute photon packets according to the spatial decomposition and transfer photon packets that leave the boundaries of the simulation box to the processor responsible for the position they moved to.

The primary downside of the Monte-Carlo approach is not specific to radiative transfer - as a statistical method, it naturally suffers from statistical noise. Reducing the noise in order to obtain an accurate solution is the main challenge of any such method. Increasing the number of photon packets n which are tracked during the simulation will increase the signal-to-noise ratio as $\sqrt{n}$, but of course it will increase the computational cost of the simulation linearly with n.

2.5.5 Comparison of radiative transfer methods

Understanding the advantages of drawbacks of different methods of radiative transfer, in particular with respect to astrophysical problems has been an ongoing challenge. There have been efforts to compare a large number of astrophysical radiative transfer codes with each other [Iliev et al., 2006, 2009], but the question of which method to choose for a given problem is still a difficult one to answer.

Evaluating the accuracy of the solutions can already be comparatively difficult, since not too many physical scenarios exist in which analytical solutions are known. This means that often, one has to compare different numerical methods with each other in order to assess the accuracy.

Another aspect that one might compare between different radiative transfer codes is their computational performance, as well as their parallel scaling, which can vary drastically depending on the numerical method chosen.

Due to the nature of the radiative transfer equation, different methods often excel at different physical scenarios. While moment based methods might shine in optically

thick environments with lots of scattering, they might have difficulties in settings with predominantly optically thin gas.

Moreover, while some codes might naturally work with an arbitrary number of different photon frequencies, others choose to track only few or just a single photon frequency, and make up for the loss in accuracy by a drastic increase in performance.

Finally, codes may be differentiated by the ease with which they integrate with the other parts of a cosmological simulation, such as hydrodynamics and star formation. On one end of the spectrum is radiative transfer post-processing, in which only radiative transfer and radiation chemistry are solved for a given simulation. This can provide many benefits such as the simplicity of the implementation and the fact that one can optimize the code for radiative transfer without regarding the other computationally intensive processes. However, it misses any feedback effect that the radiation can have on the gas. On the other side, there are fully integrated radiative transfer implementations which run simultaneously with hydrodynamics, star formation, gravity and all other processes modeled in the simulation, which increases the physical consistency of the result at an often large cost of computational resources.

Chapter 3

Sweep: The method

This chapter is based on the publication (Peter et al. 2023) published by the Monthly Notices of the Royal Astronomical Society in 2023. I am the first author and wrote the majority of the paper with editing contributions from R.S. Klessen, Guido Kanschat, S.C.O. Glover and Peter Bastian. The code of the sweep module in Arepo (written in C) was written by me. The sweep code interfaces with the SPRAI module which was written by Ondrej Jaura. In this chapter, we describe the basic structure of the sweep method for radiative transfer and how it integrates into Arepo and interfaces with the chemistry code. The code is tested in a number of standard tests in which we study the expansion of an HII region and test the performance of the code in scaling tests.

3.1 Introduction

The era of reionization is an important period in the history of the universe, during which the composition of the intergalactic medium transitioned from mostly neutral to highly ionized. This period marks an important transition between the early universe which was largely homogeneous with small fluctuations and the highly structured and complex universe we see at present days [see e.g. Loeb and Barkana, 2001, Wise, 2019, Zaroubi, 2013].

One way to understand the process of reionization is with numerical simulations. However, modeling reionization is a numerically challenging problem. Whereas the physics of the early universe was dominated by gravity, reionization is driven by the first stars and galaxies. In order to understand reionization, it is necessary to accurately model the formation and feedback processes of these small objects. The small dwarf galaxies which are believed to be the dominant sources of ionizing photons only have sizes of $\sim 1\,\mathrm{kpc}$ in size, whereas in order to obtain representative samples, the simulated volume of space needs to be sufficiently large, with lengths on the order

of hundreds of Mpc [Iliev et al., 2014]. This implies a vast range of length scales that need to be represented in any numerical model. The need to simulate such large volumes of space also implies that we must be able to follow the effects of ionizing radiation from a very large number of sources. Together, these requirements strongly constrain our choice of algorithm for modeling the transport of ionizing photons in the early Universe. or example, ray tracing with long characteristics [Abel et al., 1999b, Mihalas and Weibel-Mihalas, 1999, Whalen and Norman, 2006], a method which has been used with great success to model individual HII regions in the local Universe [e.g. Kim et al., 2018, Peters et al., 2010] is completely impractical in this context, as its computational cost scales as the product of the number of ionizing sources and the number of resolution elements in the simulation, $N_{\mathrm{source}} \times N_{\mathrm{cell}}$. This motivates the search for approaches that are independent of the number of ionizing sources.

In this paper, we focus on the astrophysical simulation package Arepo [Springel, 2010b]. Arepo solves the gravitational equation and the magneto-hydrodynamical equations for a magnetized gas on a co-moving Voronoi grid. It also has different physics modules, including treatments of stellar feedback (supernovae, radiation) and non-equilibrium chemistry. The main goal of this project is optimizing the performance of radiative transfer in Arepo.

Radiative transfer is an especially challenging problem for numerical simulations for a number of reasons [Mihalas and Weibel-Mihalas, 1999]. The first is the high dimensionality of the relevant physical quantity: radiation intensity, which depends on three spatial, two directional, one temporal and one frequency dimension leading to a total of seven dimensions. Furthermore, the properties of the local medium, such as the emissivity, absorptivity and fraction of scattered photons are important for the solution of the radiative equation while at the same time being dependent on the radiation, thus creating a need for iterative schemes to obtain solutions of the full equations. In addition, the radiative transfer equation changes its mathematical properties from being elliptical in optically thick regions to being hyperbolic in optically thin regions, thus making it difficult to choose a specialized solver suited for a particular type of equation that works across all scales of optical depth.

One class of methods with this property are moment-based methods, where one solves the moments of the radiative transfer equation with some approximate closure relation. This can lead to drastically improved performance at the cost of precision. A number of moment-based methods exists, which differ primarily in the choice of closure relation, which is typically given by an approximate expression for the Eddington tensor. One example of a moment-based method is the flux limited diffusion approach [Levermore and Pomraning, 1981, Whitehouse and Bate, 2004] in which the

closure relation is derived under the assumption of slowly varying intensity and the purpose of the flux limiter is to ensure that changes in the radiation field cannot propagate faster than the speed of light. Flux-limited diffusion has been applied to various astrophysical problems [e.g. Boss, 2008, Krumholz et al., 2007], but the high diffiusivity of the method and its consequent inability to properly account for shadowing [see e.g. Hayes and Norman, 2003] make it a poor choice for modeling ionizing radiation.

A moment-based method with a different closure relation is given by the optically thin variable Eddington tensor method in which the Eddington tensor is calculated by assuming that all lines of sight to the sources in the simulation are optically thin [Gnedin and Abel, 2001b]. This algorithm is efficient, but its accuracy is highly problem-dependent.

The radiative transfer equation can also be solved by Monte Carlo methods, in which rays are represented by photon packets [Dullemond et al., 2012, Oxley and Woolfson, 2003]. Each photon packet is appropriately sampled from the distribution of sources which then interact with the gas according to their properties. This approach has the advantage of requiring few approximations to the equations themselves, so that the quality of the results is determined primarily by the number of photon packets emitted. A disadvantage of this approach is the presence of statistical noise, with a signal to noise ratio that scales as SNR $\propto \sqrt{n}$, where n is the number of photon packets. In addition, this method is difficult to parallelize in situations where duplicating the entire grid structure on every processor is impractical owing to the memory requirements, a situation we often find ourselves in when simulating e.g. cosmic reionization.

In this paper, we focus on the astrophysical simulation package Arepo [Springel, 2010b]. Arepo solves the gravitational equation and the magneto-hydrodynamical equations for a magnetized gas on a co-moving Voronoi grid. It also has different physics modules, including treatments of stellar feedback (supernovae, radiation) and non-equilibrium chemistry. The main goal of this project is optimizing the performance of radiative transfer in Arepo.

Some of the currently available methods for radiative transfer in Arepo that have computational costs that are largely independent of the number of sources are the M1 method, which is a moment-based method based on the M1 closure relation [Kannan et al., 2019], the Monte-Carlo radiation hydrodynamics method MCRT [Smith et al., 2020] and the SimpleX method [Chang et al., 2020, Jiang et al., 2014, Ritzerveld and Icke, 2006]. While the M1 method is comparatively fast, it suffers from numerical problems inherent to moment-based methods, such as the two-beam instability [Ros-

dahl et al., 2013]. The MCRT method employs a number of techniques to improve upon Monte-Carlo radiative transfer. Currently, it is not viable to perform simulations of galaxy formation with this approach but improvements to the method are still in active development. The original SimpleX method is similar to a short-characteristics scheme and does not require angular discretization. However, it suffers from numerical diffusion, which was the reason for the development of SimpleX2 [Paardekooper et al., 2010] and its implementation in Arepo, SPRAI [Jaura et al., 2018, 2020]. In these methods, angular discretization is introduced, effectively making them discrete ordinate methods. Discrete ordinates methods have the advantage that they do not require any physically motivated approximations such that in principle, any numerical artifacts can be reduced by an increase in the resolution.

Simplex2 and SPRAI work as follows. At the beginning of every time step, photons are created at source cells and distributed equally into all directional bins. Then, in every iteration, photons from a directional bin are transported from a Voronoi cell to its d most straightforward neighbors along that direction. The photon density is then used to update the local chemistry of the cell and some of the photons are scattered by re-distributing into the other directional bins. This process is iterated until all of the photons have been absorbed. This method performs well in optically thick regions in which the mean free path is short. However, in optically thin regions, this method requires many iterations, increasing computation times drastically.

Our proposed change to this algorithm is based on transport sweeps [Koch et al., 1991]. The idea is that, for a given direction, a cell is only solved once all its upwind neighbors along that direction have been solved. The main benefit of this method is that, in the absence of scattering, such a re-ordering allows us to obtain the full photon density field in a single sweep. In order to incorporate scattering, the sweep needs to be iterated.

The drawback of this method is that it induces an ordering on the cells due to the dependencies of cells on their upwind neighbors. While the dependency graph is trivial for regular grids, this is not the case for a Voronoi grid. At the same time, the code needs to be parallelized. Our current solution to these problems is task-based parallelism [Zeyao and Lianxiang, 2004] in which a task is a pair of a Voronoi cell and a given directional bin. For each task, we keep track of the number of unsolved upwind neighbors and only solve those tasks for which this number is zero. In this way, the dependency graph is never explicitly constructed but we still obtain a topological ordering of the cells.

This paper is structured as follows. In Section 3.2 we discuss the problem of radiative transfer in general (Section 3.2.2) before the concept of radiative transport

sweeps (Section 3.2.5) and the concrete implementation details which allow the code to run on large numbers of processors in parallel (Section 3.2.4) are introduced. We also discuss how to handle problems with periodic boundaries in the concept of transport sweeps (Section 3.2.7). In Section 3.3, we present a number of tests in order to demonstrate that our code reproduces physically correct results (Section 3.3.1, 3.3.2, 3.3.3, 3.3.4). We also study the computational performance of the code, especially in respect to its parallelization in Section 3.3.5 and 3.3.6. Finally, we conclude this paper and present some potential extensions of the code as well as possible applications in Section 3.4.

3.2 Methods

3.2.1 Structure of the code

In this paper we discuss an implementation of radiative transfer for the astrophysical simulation code Arepo [Springel, 2010b]. The structure of our code is based very closely on SPRAI, an existing radiative transfer module for Arepo whose design and operation is described in Jaura et al. [2018, 2020]. Indeed, the code shares SPRAI's interface between radiative transfer and the SGChem chemistry module.[1] We therefore do not discuss this aspect of the code here and refer the reader interested in details of this coupling to Jaura et al. [2018, 2020].

The defining characteristic of Arepo is that the hydrodynamical equations are solved on a Voronoi grid which is generated by points that are co-moving with the gas instead of a Eulerian grid with an adaptive mesh refinement scheme. This has the benefit of avoiding numerical artifacts caused by the structure of the grid while simultaneously adapting the grid to the gas density automatically.

The goal of the radiative transfer code is to solve the radiative transfer equation to obtain the radiative fluxes in all cells. These fluxes are then passed to the chemistry module which requires the fluxes to calculate the detailed chemical composition of the medium as well as the corresponding heating and ionization rates.

Arepo uses an adaptive timestepping approach in which regions that require higher accuracy are solved with a smaller timestep. Due to this, the full Voronoi grid is only available on the synchronization timesteps, i.e. the timesteps during which every cell is updated. Our current implementation of the radiative transfer method

[1] SGChem implements various different chemical networks. In this paper, we use its primordial chemistry network, first implemented in Arepo by Hartwig et al. [2015] and more recently updated by Schauer et al. [2019]

requires the full grid, so that we can only perform radiative transfer calculations during those synchronization steps. For the substeps, the radiative fluxes between cells are assumed to remain constant, keeping the value of the previous synchronization step. The validity of this approximation depends on the ratio of the lowest hydrodynamical/gravity timestep to the full synchronization timestep as well as on the ratio of the timescale at which hydrodynamics and gravity take place to the timescale of radiative transfer and the photo-chemistry. At the cost of code performance, the effect of the approximation can be reduced by limiting the highest timestep. For the tests performed in this paper, this approximation has been acceptable. In the future, extensions to the implementation can be considered in which radiative transfer takes place on the substeps as well, which would require adjusting the cell timestep criterion to take radiative transfer into account.

3.2.2 Radiative transfer

The quantity of interest in the problem of radiative transfer is the specific radiative intensity $I_\nu(\boldsymbol{r}, t, \hat{\Omega})$, with frequency ν, spatial position $\boldsymbol{r}$, time t and solid angle $\hat{\Omega}$ given in units of $\mathrm{W\,m^{-2}\,sr^{-1}\,Hz^{-1}}$. The radiative transfer equation is given by [Rybicki and Lightman, 1985].

$$\frac{1}{c}\frac{\partial}{\partial t} I_\nu + \hat{\mathbf{\Omega}} \cdot \boldsymbol{\nabla} I_\nu = j_\nu - (\bar{k}_{\nu,\mathrm{s}} + \bar{k}_{\nu,\mathrm{a}}) I_\nu + \frac{1}{4\pi} \int_S k_{\nu,\mathrm{s}}(\mathbf{\Omega'}) I_\nu \mathbf{d\Omega'}. \tag{3.1}$$

It relates the rate of change of the radiative intensity along the line with solid angle $\hat{\mathbf{\Omega}}$. Here, c is the speed of light, j_ν is the emission coefficient, $\bar{k}_{\nu,\mathrm{a}}$ is the absorption coefficient, $\int_S$ denotes the integral over the unit sphere with $\mathbf{\Omega'}$ being the solid angle relative to $\hat{\mathbf{\Omega}}$. The total scattering coefficient $\bar{k}_{\nu,\mathrm{s}}$ is defined via the angle-dependent scattering coefficient $k_{\nu,\mathrm{s}}(\mathbf{\Omega'})$ as $\bar{k}_{\nu,\mathrm{s}} = \int_S k_{\nu,\mathrm{s}}(\mathbf{\Omega'}) \mathbf{d\Omega'}$. From now on, we will assume isotropic scattering such that $k_{\nu,\mathrm{s}}(\mathbf{\Omega}) = \bar{k}_{\nu,\mathrm{s}}$. If the timescales on which the material coefficients (the source term and the absorption and scattering coefficients) change are all smaller than the typical light crossing time of the system, we can safely make the so-called infinite speed of light approximation in which we drop the first term of Eq. 3.1.

With these two assumptions we obtain

$$\hat{\mathbf{\Omega}} \cdot \boldsymbol{\nabla} I_\nu = j_\nu - (\bar{k}_{\nu,\mathrm{s}} + \bar{k}_{\nu,\mathrm{a}}) I_\nu + \frac{\bar{k}_{\nu,\mathrm{s}}}{4\pi} \int_S I_\nu \mathbf{d\Omega'}. \tag{3.2}$$

In order to solve this equation numerically, we need to find a discretization scheme, of which there are many for the radiative transfer problem. Here, we will focus on the discrete ordinate method in which the equation is discretized in all variables: time

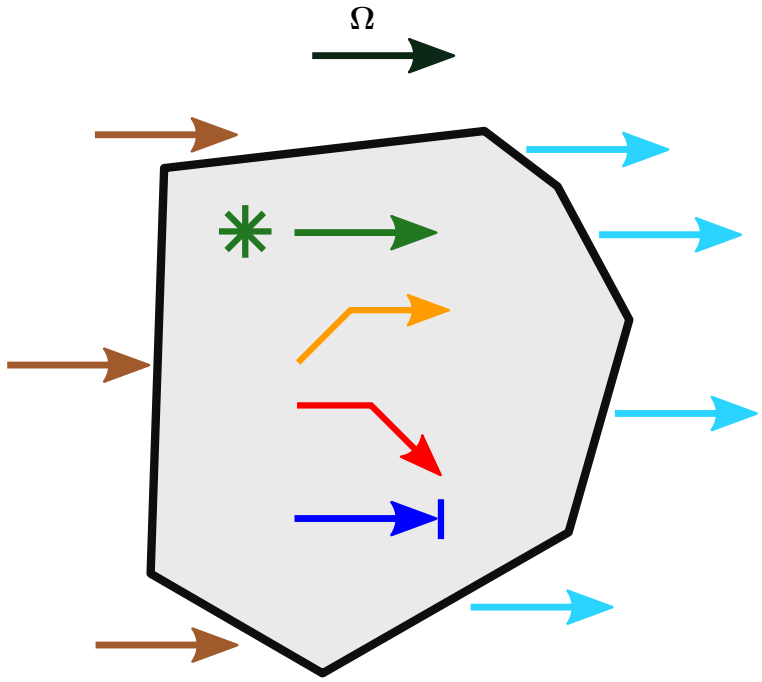

Figure 3.1. Illustration of the radiative processes described by Eq. 3.2 for a single grid cell: Incoming (brown) and outgoing (teal) radiation, sources (green), absorption (blue) and scattering (into considered solid angle: orange, out of it: red)

t, position $\boldsymbol{r}$, frequency, ν and the angular component Ω. The physical intuition behind Eq. 3.2 when applied to a small volume is illustrated in Fig. 3.1. The sources of radiation in this cell are through incoming radiation from cells to the left (brown arrows), the source term j directly (green arrow) or scattering into $\boldsymbol{\Omega}$ from a different $\boldsymbol{\Omega}'$ (orange arrow). Radiation from the cell is either scattered out of this $\boldsymbol{\Omega}$ (red arrow), absorbed (blue arrow) or leaves the cell towards the right (teal arrows). Thus, the neighboring cells fall into two categories: Cells upwind along $\boldsymbol{\Omega}$ (brown arrows) – in order to solve the local equations, we require the incoming specific intensities from those cells. Cells downwind along $\boldsymbol{\Omega}$ (teal arrows) – these depend on the local solution of the intensity for their own solution.

The discretized radiative transfer equation takes the form of a large, coupled system of equations. There are many different approaches to solving this problem. Which of these methods is the best strongly depends on the physical nature of the simulation. In very optically thick media, scattering dominates, which means the equations are elliptical and thus diffusive in nature. On the other hand, in optically thin media, the equations become hyperbolic and long-ranged.

3.2.3 Source iteration

An obvious approach to solve the resulting equations is to construct the full matrix describing the system and to apply an iterative solver such as the Generalized minimal residual method [Saad and Schultz, 1986] until convergence is reached. Due to the high dimensionality of the equation (three spatial, two angular and one frequency dimension), this quickly becomes infeasible due to the sheer size of the resulting matrix.

A different, well-known approach is known as source iteration which is given by Algorithm 1. Here, convergence of I_ν can be defined in a number of ways. The definition we choose is given by the condition

$$\forall \boldsymbol{r} \forall \boldsymbol{\Omega} : \frac{|I_\nu^i(\boldsymbol{r}, \boldsymbol{\Omega}) - I_\nu^{i-1}(\boldsymbol{r}, \boldsymbol{\Omega})|}{I_\nu^{i-1}(\boldsymbol{r}, \boldsymbol{\Omega})} < \epsilon, \tag{3.3}$$

where ϵ is a free parameter and should be chosen to be small.

1: Guess initial intensity I_ν^0. For example: $I_\nu^0 = 0$.
2: **while** I_ν^i not converged **do**
3: Compute source terms (using I_ν^i for scattering).
4: Solve Eq. 3.2 to obtain I_ν^{i+1} using source terms for each Ω and each ν.

Algorithm 1: Source iteration

The idea is to use an iterative scheme in which scattering is treated as a constant source term. This is still an iterative method, as the scattering source terms are recomputed after every iteration. This approach is suited best for optically thin media where scattering is not dominant and the source iteration converges quickly.

The main benefit of this is that it removes the coupling between the terms of different Ω, so that instead of solving one large coupled system of equations, Step 4 of Algorithm 1 only requires us to solve one smaller system of equations for each Ω. In the following, we will show that there is an efficient way to solve such a system of equations under certain conditions. In principle, we would like to simply iterate over the grid once and solve Eq. 3.2 for each grid cell to obtain an exact solution. However, as discussed previously, the equation gives rise to local dependencies that require us to solve upwind cells before their downwind neighbors.

As illustrated in Fig. 3.2, we can understand these local dependencies as a directed graph in which the nodes are the grid cells and an edge from the cell c_1 to the cell c_2 corresponds to a dependency of c_2 on c_1. Under the assumption that the graph is acyclic (which we can easily prove to be true for the induced dependency graph

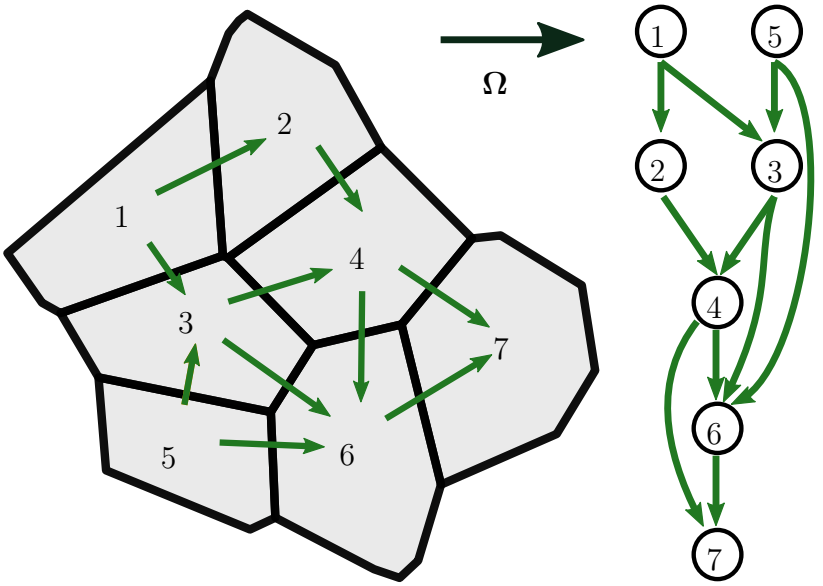

Figure 3.2. Left: Illustration of a 2D Voronoi grid and the dependencies induced by the sweep ordering for a sweep towards the right. Right: The directed, acyclic graph corresponding to the dependencies.

of a Voronoi grid, such as the one used in Arepo; see Appendix A.1), there is a topological ordering of the grid cells, such that any cell in the ordering only depends on the cells that come before it. This is analogous to a re-ordering of the cells of the matrix describing the system of equations (in which each non-zero entry corresponds to an edge in the dependency graph) such that the matrix becomes lower triangular and can be solved in one pass through the matrix. Such a pass through the cells of the system is called a transport sweep.

3.2.4 Parallelization

The sweep is clearly the most computationally intensive part of the source iteration algorithm (Algorithm 1). In order to apply this algorithm to large systems on modern hardware, we require some form of parallelization. The easiest way of parallelizing this algorithm would be to distribute the solution of different frequencies ν and angles Ω onto the processors. One problem with this method is that for very large numbers of processors there might simply not be enough different ν and Ω to efficiently employ all of them. Moreover, parallelizing over Ω and ν requires the information about the grid to be present on every processor, which, due to memory requirements, quickly becomes infeasible for large simulations. Due to these concerns, we choose to use a spatial decomposition of the grid.

For fully structured, euclidean grids, a sweeping algorithm and a domain decom-

position that optimizes the parallel performance of the sweep is given by the Koch-Baker-Alcouffe algorithm (Baker [2016], Koch et al. [1991]). This algorithm assumes that the number of processors N can be factorized as $N = N_{\mathrm{x}} \cdot N_{\mathrm{y}}$. The domain is then subdivided N_{x} times along the x axis and N_{y} times along the y axis, resulting in a decomposition of the domain into N columns. Each processor is then assigned one of the columns. For any direction Ω, any column has at most three faces with upwind dependencies (it will have fewer dependencies only if Ω is aligned with one of the coordinate axes). If those upwind dependencies are fulfilled, the column can be solved in its entirety without further communication. Of particular importance is that the solution of the upwind columns does not depend on incoming fluxes of the downwind column. For unstructured grids, the latter statement does not hold, and a sweep can require many cycles of back-and-forth communication between neighboring columns. This makes the problem of finding the optimal domain decomposition for unstructured grids much harder.

In this work, we decide to use the already available domain decomposition in Arepo (which is used for example for the hydrodynamics and gravity solvers) in order to simplify the problem and to reduce memory requirements. The domain decomposition employed in Arepo is based on the space-filling-curve approach. The idea of this approach is to simplify the optimization problem by arranging all the cells of the three dimensional computational domain on a one dimensional line and then dividing that line into a number of domains with approximately equal estimated workload. The advantage of using a space filling curve (such as the Peano-Hilbert curve) for this 1D to 3D mapping is that it results in reasonably localized domains (since the space filling curve maps points that are close in 1D to points that are close in 3D), thus reducing the amount of communication required. In Arepo, the estimated workload of a cell is given by a sum of the estimated work required for the gravitational and hydrodynamical calculations. In principle, this estimate could be extended to include the workload due to radiative transfer, thus possibly reducing the overall time to solution by accelerating radiative transfer at the cost of a reduction in load balance for gravity and hydrodynamics. However, for the sake of simplicity we choose not to do this in this work.

The remaining problem is to find an algorithm that performs the sweep across the entire grid which itself is distributed on different processors. One challenge in this is that it is infeasible to calculate the topological ordering of the global dependency graph because this would require gathering the necessary information onto a single core or employing a parallel algorithm for topological sorting. In the following section, we discuss our strategy for dealing with this problem, in which the topological ordering is

never explicitly computed but instead implicitly adhered to by a task-based parallelism approach. This method is based on Pautz [2002].

3.2.5 The sweep algorithm

In the following, we define a task as a tuple $(c, \mathbf{\Omega})$ of a cell c and a sweeping direction $\mathbf{\Omega}$. Solving a task means solving Eq. 3.2 in the cell c for the direction $\mathbf{\Omega}$ for all frequencies ν. Note that we have excluded frequency from the definition of a task because we choose to solve all available frequencies at once whenever we solve a task. For transport sweeps on structured grids, it is common to group the directions (for example into octants for a Cartesian grid) such that directions in the same group have the same dependency graph. On an unstructured grid, two directions that are almost parallel can still have different dependency graphs, so we choose to do no grouping of the directions.

For any task $t = (c, \mathbf{\Omega})$, we can define $d(t)$ / $u(t)$ to be the set of cells which are downwind / upwind of c with respect to $\mathbf{\Omega}$. For a given grid cell at $\boldsymbol{r}$, both $d(t)$ and $u(t)$ can easily be obtained in a single loop through the neighbors, by counting a neighbor at position $\boldsymbol{r}_\mathrm{n}$ as downwind if $(\boldsymbol{r}_\mathrm{n} - \boldsymbol{r}) \cdot \mathbf{\Omega} > 0$ and as upwind otherwise. This operation can be done without any communication to other processors, since Arepo ensures that grid cells belonging to other processors that are neighbors of any local cell are always present as local ghost particles and that the positions of the ghost particle is equivalent to the position of the corresponding cell on the other processor. Crucially, this ensures that the downwind/upwind neighbor relationship is always symmetric, even across processor boundaries.

With this, the unparallelized version of the algorithm to solve Step 4 in Algorithm 1 is given by 2. Note that this algorithm requires non-periodic boundary conditions, which guarantees that at least one cell has $u(t) = 0$. Extensions to periodic boundaries will be discussed in Section 3.2.7. Since the dependency graph is acyclic, this algorithm will always terminate.

For cells on the boundary, which have no upwind dependencies, the incoming fluxes are obtained from the boundary conditions. Fixed boundary conditions in which the value of the incoming radiation is $I_\nu = 0$ represent the simplest case. In Arepo, fixed boundaries are represented by cells with a connection to the first tetrahedron from which the grid was constructed and which encompasses the entire computational domain. A discussion of periodic boundaries (represented by ghost cells which stand for a particle on the other side of the boundary) will follow in Section 3.2.7.

The exact way in which the radiative intensity is calculated from the upwind fluxes in Step 7 will be discussed in Section 3.2.6.

initialize task queue $q \leftarrow \{\}$
for all $\mathbf{\Omega}$ and all cells c in grid **do**
    count number of required upwind fluxes $n(c, \mathbf{\Omega}) \leftarrow |u(t)|$
    if $n(c, \mathbf{\Omega}) = 0$ **then** add task $(c, \mathbf{\Omega})$ to q
while q not empty **do**
    get first task $t = (c, \mathbf{\Omega})$ from q
    solve t using upwind fluxes
    for downwind neighbor c_d in $d(t)$ **do**
        reduce missing upwind flux count $n(c_\mathrm{d}, \mathbf{\Omega})$ by 1.
        if $n(c_\mathrm{d}, \mathbf{\Omega}) = 0$ **then** add task $(c_\mathrm{d}, \mathbf{\Omega})$ to q.

Algorithm 2: Single-core sweep

initialize task queue $q \leftarrow \{\}$
initialize send queues for each processor i holding downwind neighbors of any of the cells in the domain of the current processor: $s_i \leftarrow \{\}$
for all $\mathbf{\Omega}$ and all cells c in grid **do**
    count number of required upwind fluxes $n(c, \mathbf{\Omega}) \leftarrow u(t)$
    if $n(c, \mathbf{\Omega}) = 0$ **then** add task $(c, \mathbf{\Omega})$ to q
while any cell unsolved or any s_i not empty **do**
    for each incoming message (flux f along $\mathbf{\Omega}$ into cell c) **do**
        reduce missing upwind flux count $n(c, \mathbf{\Omega})$ by 1.
        if $n(c, \mathbf{\Omega}) = 0$ **then** add task $(c, \mathbf{\Omega})$ to q.
    while q nonempty **do**
        get first task $t = (c, \mathbf{\Omega})$ from q
        solve t using upwind fluxes
        for downwind neighbor c_d in $d(t)$ **do**
            if c_d is remote cell on processor i **then**
                add flux to send queue s_i
            else
                reduce missing upwind flux count $n(c_\mathrm{d}, \mathbf{\Omega})$ by 1.
                if $n(c_\mathrm{d}, \mathbf{\Omega}) = 0$ **then** add task $(c_\mathrm{d}, \mathbf{\Omega})$ to q.
    send all messages in s_i

Algorithm 3: Parallel sweep

What we have described so far only works on a single processor. In order to parallelize, we introduce Algorithm 3, in which we communicate fluxes across processor domain boundaries. Here, we had to make an implementation choice regarding the communication scheme. The arguably simplest approach would be to send each flux immediately as we encounter it in Step 15. The benefit of this is that any downwind processor depending on the flux of this cell would be able to immediately obtain the required flux, thus potentially avoiding idle time. In practice however, we found this approach to be too inefficient because of the communication delays it causes. Therefore, we chose to buffer the fluxes in send queues and only send messages when there is nothing left to solve with the flux information we currently have. This reduced the delays due to communication significantly and improved the scaling behavior in the idealized test cases.

Note that Algorithm 3 solves the sweep for different directions $\mathbf{\Omega}$ concurrently. This is intentional, since it improves the parallel efficiency of the code. If sweeps for different directions were performed in serial, processors with domains that are downwind in the direction of the sweep will be idle in the beginning of the sweep, while processors with domains that are upwind will be idle at the end.

Note that a similar problem appears despite the parallel execution of different directions. It is called pipe fill or pipe drain [Vermaak et al., 2020], and appears when the number of domains becomes large enough that there are inner regions which cannot start sweeping until outer regions are resolved. For an illustration of this effect, see Fig. 3.3, which shows a simplified case of a square-shaped domain decomposed into 16 subdomains. As the figure shows, both the first and the last two directional sweeps will be performed while the central cores are idle, which reduces parallel efficiency. As the number of cores grows, so does the duration of the pipe fill/drain phenomenon. Note, that in Algorithm 3, a partial sweep in a single direction is not necessarily finished before one in another direction is started, thus exacerbating the problem, compared to the scenario depicted in Fig. 3.3. In addition to this, the domain decomposition and the dependency graph in Fig. 3.3 is much simpler than in an actual run of our code, due to the unstructured grid and the fact that the domain decomposition in our case has to be done with an eye towards the gravity and hydrodynamics solvers.

One such problem which arises due to the unstructured grid is what we call re-entry dependencies. They appear when the sweep direction is close to being aligned with the boundary between two domains. In such cases, the dependencies can form a zig-zag pattern, such as the one depicted in Fig. 3.4. In such a scenario, the number of cells which can be solved locally before communication to the neighboring domain is required is very low. In the extreme, but not unrealistic, case depicted in Fig. 3.4,

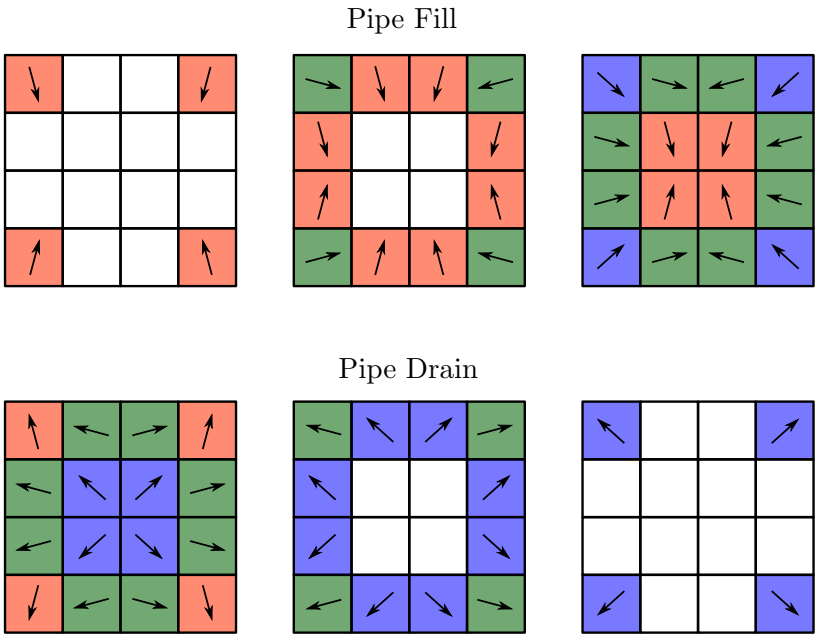

Figure 3.3. Illustration of the pipe fill/drain phenomenon. Each square denotes a computational domain belonging to a single processor. The arrows denote the direction of the sweep performed in that processor, while colors correspond to the (relative) time at which the sweep in that direction was first started, with red being before green which in turn denotes a time before blue. A white square without arrow means that the processor is currently idle.

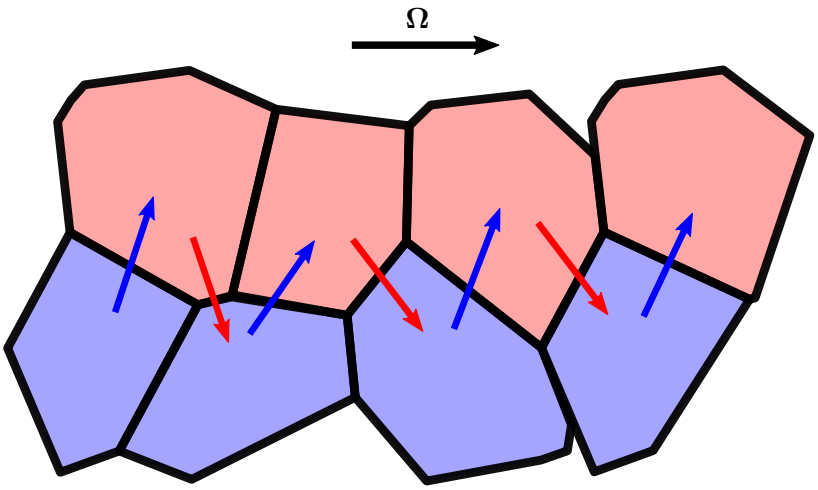

Figure 3.4. Illustration of re-entry dependencies arising in scenarios where the sweep direction is aligned with the boundary between two domains. The red/blue color of the cells corresponds to the domain in which they reside. Blue arrows denote a dependency requiring inter-processor communication from the blue domain to the red, whereas red arrows denote communication from red to blue.

each processor can solve only one cell before having to communicate the resulting flux. While the effect is slightly alleviated by the fact that processors are not required to finish the solution of one sweep direction before starting the next, this still slows down the code significantly, mainly due to the additional delay each communication introduces.

The problems described above can be solved partially by improved scheduling and communication strategies. The main goal of such strategies is for the processors in the outer regions to solve the tasks required by those in the center as quickly as possible [Adams et al., 2019] and to communicate the resulting fluxes immediately. Such prioritization can greatly improve the parallel efficiency of this code by reducing the pipe-fill/drain effect. For the sake of simplicity and to check whether this sweeping approach is feasible for the radiative transfer in Arepo, in this paper, we use a very simple prioritization strategy which prioritizes finishing one sweep direction before starting another one.

Another possible optimization is to intentionally omit certain fluxes between cells, thus removing dependencies from the graph. Doing so means that the result of the transport sweep is only an approximation and obtaining the solution would require iterating over a number of sweeps. However, if the right dependencies are removed (e.g. the re-entry dependencies discussed above), the performance improvement can potentially be large enough to offset the additional cost due to the iteration [Lucero Lorca, 2018].

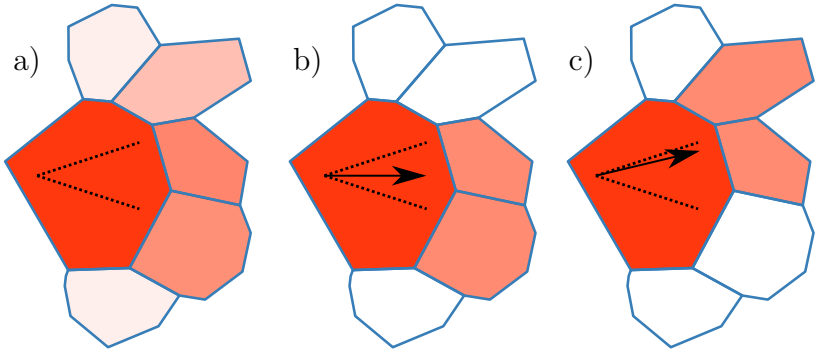

Figure 3.5. Illustration of the three different transport schemes. The black arrow represents the sweeping direction $\mathbf{\Omega}$. The dotted lines represent the solid angle corresponding to the direction $\mathbf{\Omega}$. The shading of the downwind cells represents the flux that the cells would receive, with white meaning no incoming flux and red meaning a high amount of incoming flux. a) Distribution proportional to fraction of area of the cell interfaces to the total area; b) Choosing the n most straightforward neighbors; c) Choosing the n most straightforward neighbors along a random vector in the solid angle corresponding to the direction.

3.2.6 Transport methods

In order to calculate the (downwind) fluxes out of a cell, given the source terms, absorption coefficients and the incoming (upwind) fluxes, we need to decide on a transport scheme with which we can solve Eq. 3.2.

Three such schemes are depicted in Fig. 3.5. Note that the solid angle corresponding to a given direction is given by $\frac{4\pi}{N_{\text{dir}}}$ where N_{dir} is the number of directions in our discretization.

In the first scheme, in Panel a), the outgoing flux F_i of radiation for the direction $\mathbf{\Omega}$ to a downwind neighbor i is given according to the distribution

$$F_i = F \frac{A_i \left(\boldsymbol{n}_i \cdot \mathbf{\Omega}\right)}{\sum_{j=1}^{N} A_j \left(\boldsymbol{n}_j \cdot \mathbf{\Omega}\right)}, \tag{3.4}$$

where F is the total outgoing flux (given by the sum of non-absorbed incoming radiation and the radiation created by the source term of this cell), $\boldsymbol{n}_j$ is the normal of the Voronoi face connecting the cell and the neighbor j and A_j is the area of that face. The Simplex2 method [Kruip et al., 2010, Paardekooper et al., 2010], which is the basis for the SPRAI implementation in Arepo [Jaura et al., 2018], introduces an additional transport method (called direction-conserving transport) in which the incoming flux is distributed equally onto n neighbors with the most straight-forward face normals along $\mathbf{\Omega}$, see Panel b) in Fig. 3.5. The authors showed that $n = n_{\text{dim}}$, with n_{dim} being the number of spatial dimensions is the optimal choice for direction-conserving trans-

port. The idea of this scheme is to reduce numerical diffusion. However, this comes at the cost of amplifying the effect that the angular discretization into a number of discrete directions introduces, namely that radiation is transported along preferential directions [Jaura et al., 2018], something that becomes very apparent in optically thin media where the mean free path is long. In principle, this behavior could be alleviated by increasing the number of directions. However, this increases memory requirements and computation time. In SPRAI this problem is solved in two ways.

Firstly, a slightly modified version of the direction conserving transport is employed in which the direction in which radiation will be transported is decided on a cell-by-cell basis. For each cell, instead of transporting radiation along $\mathbf{\Omega}_i$, a vector $\mathbf{\Omega}'_i$ is chosen randomly, with the only condition being that $\mathbf{\Omega}'_i$ is closer to $\mathbf{\Omega}_i$ than any of the angles $\mathbf{\Omega}_j$ for $i \neq j$ (in other words, $\mathbf{\Omega}'_i$ should be within the solid angle that $\mathbf{\Omega}_i$ corresponds to). This method is illustrated in Panel c) in Fig. 3.5. We choose not to implement this transport method for sweep, because it would require us to implement the random choice of $\mathbf{\Omega}'$ in a deterministic fashion, in order to allow us to properly count the number of upwind/downwind dependencies. The drawback of this is that our results will not agree exactly with those of SPRAI, even in the absence of any scattering, because of the different choice of transport method.

The second way in which SPRAI reduces preferential directions is that any radiative transfer step may be subdivided into N_{rot} substeps, each with the source terms reduced by a factor of $1/N_{\text{rot}}$. For each step, the radiation chemistry is updated according to the resulting intensity field. After every substep, the directions $\mathbf{\Omega}_i$ are rotated to new directions $\mathbf{\Omega}_i = \boldsymbol{R}(\theta, \phi) \cdot \mathbf{\Omega}'_i$ where $\boldsymbol{R}(\theta, \phi)$ is the rotation matrix and the spherical coordinate-angles θ and ϕ are randomly chosen as $\theta \in [0, \pi]$, $\phi \in [0, 2\pi]$. The remapping between angle-dependent quantities, such as the intensity is then done via $I_\nu(\boldsymbol{r}, \mathbf{\Omega}'_i) = \sum_{j=1}^{N_{\text{dir}}} c_{ij} I_\nu(\boldsymbol{r}, \mathbf{\Omega}')$ where N_{dir} is the number of discrete directions and the interpolation coefficients c_{ij} depend on the choice of interpolation and obey $\forall i : \sum_{j=1}^{N_{\text{dir}}} c_{ij} = 1$ For simplicity, we choose $c_{ij} = \frac{\Delta S_{ij}}{\Delta S_i}$, where ΔS_{ij} is the solid angle that $\mathbf{\Omega}_i$ and $\mathbf{\Omega}_j$ share and ΔS_i is the solid angle corresponding to any direction Ω_i. This random rotation of the directions effectively smears out preferential directions at the cost of additional computation time.

In SPRAI, radiation travels one cell at a time before the scattering terms are recomputed. This process is repeated until all photons are absorbed. Throughout this, SPRAI needs the directions to remain constant (otherwise, direction would not be conserved for more than a cell length). In the sweep method, the directions only need to remain constant throughout one single sweep. This means we can combine the source iteration (Algorithm 1) and the rotation of the directions, potentially saving

many iterations.

In all our tests, we use the first transport method in which outgoing fluxes are simply assigned via the geometry of the cell. A potential benefit of the direction conserving transport method is that it reduces the average number of downwind dependencies per cell from $15.54/2 \approx 7.8$ (since the average number of neighbors in a 3D-Voronoi grid is 15.54) to the number of dimensions, $n_{\mathrm{dim}} = 3$, thus making the dependency graph thinner.

3.2.7 Periodic Boundary Conditions

In simulations of the period of reionization, the simulated volume is often selected as a box which is supposed to be representative of the universe. During a normal simulation of such a box using gravity and hydrodynamics, periodic boundary conditions are employed to effectively model the influence of the adjacent regions of space without having to simulate those regions explicitly. The same idea applies to the radiative transfer. Using periodic boundaries, any photons leaving the box can re-enter it from the opposing side. If the box is large enough to be statistically representative, then this re-entry models the light from the neighboring regions.

In order to introduce periodic boundary conditions in simulations, the standard approach is to add a mirror image of each boundary cell on the other side of the grid, i.e. to add the same cell with its position shifted by the box size L. These mirror images are called ghost cells in Arepo. Fluxes going into such a ghost cell will then be treated as incoming fluxes into the corresponding *normal* cell.

In transport sweeps, introducing such ghost cells at the boundaries poses an additional challenge. As illustrated in Fig. 3.6, after the inclusion of the ghost cells, the induced dependency graph becomes cyclic. Clearly, no topological ordering exists for cyclic graphs. Simply applying Algorithm 3 to a grid with periodic boundaries thus cannot work - the algorithm would never terminate.

In order to solve this, we use an iterative approach, similar to Alg. 1. Any radiation going out of the boundaries of the computational domain is added to the effective source term of its periodic ghost cell for the next iteration. This breaks the cyclic dependencies induced by the periodic boundary condition but still ensures that any outgoing radiation is re-introduced into the box.

There are numerous choices for how to define convergence for this iterative method. We use the relative difference in the source terms j_i^n (where i denotes the cell and n denotes the iteration number) as an error

$$E_i^n = \frac{j_i^n - j_i^{n-1}}{j_i^n + j_i^{n-1}}. \tag{3.5}$$

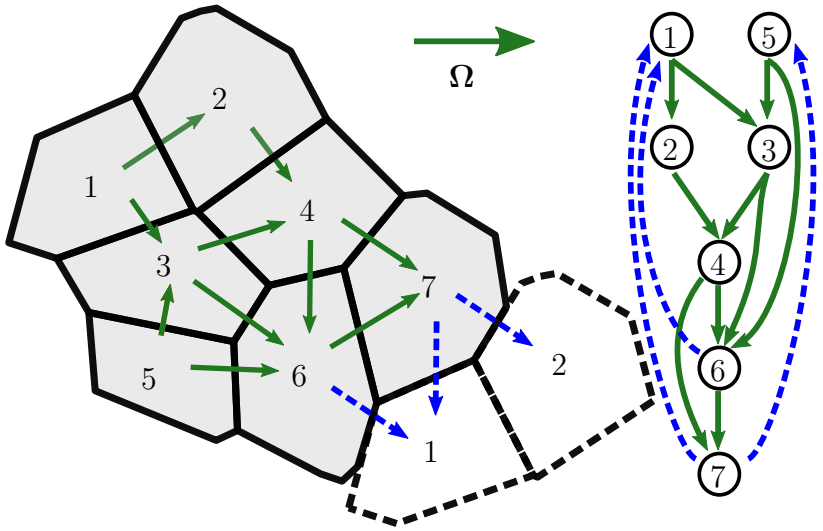

Figure 3.6. Left: Illustration of a 2D Voronoi grid and the dependencies induced by the sweep ordering for a sweep towards the right under the assumption of periodic boundary conditions. Solid boundaries and gray background represent normal cells, dashed boundaries and white background represent periodic ghost cells. Right: The directed, cyclic graph corresponding to the dependencies.

The iteration is stopped if $\forall i : E_i < \epsilon$, where ϵ is the convergence threshold which can be chosen by the user. Additionally, we define a maximum number of iterations after which the algorithm will terminate, even if there are still cells which exceed the error threshold.

Clearly, if the number of iterations needed in order to reach convergence in the source terms of the periodic boundary iteration is $n_{\text{it,pbc}}$, while the number of iterations needed to relax the terms introduced due to scattering (Algorithm 1) is $n_{\text{it,scat}}$, this iterative scheme increases the overall runtime of the algorithm by a factor of $n_{\text{it,pbc}} n_{\text{it,scat}}$ compared to the runtime of a single sweep. In order to improve on this, an interesting approach could be to combine the two iterative schemes, such that the source terms due to scattering and the periodic boundary conditions are calculated at the same time. If doing so does not change the behavior of the individual schemes, this would reduce the runtime overhead to $\max(n_{\text{it,pbc}}, n_{\text{it,scat}})$.

As a way to reduce $n_{\text{it,pbc}}$, we tried an approach we call "warm starting" in which the final values of the source terms obtained in a previous RT step are used as an initial guess in the next step, instead of using $j_i^n = 0$ as a guess. This is made technically challenging by the fact that the grid might change between one RT step and the next, for example by removing cells from the computation or by introducing new ones in adaptive refinement schemes. As a guess for the source term for any newly created cell we use $j_i^n = 0$.

3.3 Tests

For all the test simulations we make the following choices regarding the parameters of the numerical discretization. For the frequency discretization, we choose a single frequency bin corresponding to photons in the range $(13.6\,\mathrm{eV}, \infty)$, i.e. with enough energy to ionize hydrogen.[2] For the angular discretization, we use 84 directions isotropically distributed over the unit sphere generated by simulated annealing [Jaura et al., 2018]. The code supports other numbers of directions but we choose 84 as a compromise between lower numbers which reduce the accuracy of the solution and higher numbers which increase memory consumption and overall runtime and to be consistent with the results of Jaura et al. [2018], where the number of directions was also chosen to be 84.

We use $N_{\mathrm{rot}} = 5$ random rotations of the directions in every time step. We find that this number of rotations is sufficient to smooth out any obvious preferential directions in the results and still small enough to keep the run time reasonable.

For all tests, we fix the hydrogen ionization cross section at $\sigma_{\mathrm{H}} = 5.38 \times 10^{-18}\,\mathrm{cm}^2$, corresponding to the value for a 14.4 eV photon, and the case B recombination rate coefficient to a constant value $\alpha_{\mathrm{B}} = 2.59 \times 10^{-13}\,\mathrm{cm}^3\,\mathrm{s}^{-1}$.

3.3.1 Expansion tests

We consider first the idealized scenario of an ionizing source surrounded by a homogeneous distribution of neutral atomic hydrogen. Here, the source will form a spherical region of ionized hydrogen around it, known as an HII region. Strömgren [1939] showed that in ionization equilibrium, the radius of this region is given by the Strömgren radius,

$$R_{\mathrm{St}} = \left(\frac{3 N_\gamma}{4\pi \alpha_{\mathrm{B}} n_{\mathrm{e}} n_{\mathrm{H}^+}} \right)^{1/3}, \tag{3.6}$$

where $\alpha_{\mathrm{B}}(T)$ is the case B recombination coefficient of hydrogen and n_{e} is the electron number density. Given that the medium inside the spherical region is highly ionized, it follows that $n_{\mathrm{e}} \approx n_{\mathrm{H}^+}$.

In the initial phase of the evolution, the expansion is simply driven by radiation which ionizes the neutral gas just beyond the ionization front (I-front). It takes place at very high velocities, compared to the speed of sound in the ionized gas c_{s}, so that the hydrodynamical response of the gas is irrelevant for the movement of the ionization front. This initial, rapid expansion is called the R-type expansion (R=rarefied).

[2] Note that Sweep can readily deal with multiple energy bins; we make this choice here purely for simplicity.

Under the assumption that the density of the gas remains constant, the time-evolution of the radius of the ionization sphere is given by

$$R_{\rm r}(t) = R_{\rm St}\left(1 - e^{-t/t_{\rm rec}}\right)^{1/3}, \tag{3.7}$$

where $t_{\rm rec} = (\alpha_{\rm B} n_{\rm H})^{-1}$ is the recombination time.

Once the radius of the sphere reaches the Strömgren radius, the second phase of the evolution, called D-type (D=dense) begins. In this phase, the expansion of the sphere is driven by a pressure gradient between the ionized, inner region and the neutral, outer region. This pressure gradient is caused by the large temperature difference between the two regions. In this second phase, the I-front is preceded by a shock front since it moves at velocities that are supersonic in the neutral medium but subsonic in the ionized medium. An analytical expression for the radius of the sphere as a function of time was first derived by Spitzer [1978] and is given by

$$R_{\rm d}(t) = R_{\rm St}\left(1 + \frac{7}{4}\frac{c_{\rm s} t}{R_{\rm St}}\right)^{4/7}, \tag{3.8}$$

where $t = 0$ here corresponds to the time at which the ionization front transitions from R-type to D-type.

R-type expansion

As a first test of the radiative transfer code, we study the R-type expansion of a HII region. The following tests are performed at 3 different resolutions of 32^3, 64^3, and 128^3 cells. We use the same initial conditions as those in the R-type expansion test in Jaura et al. [2018] and Baczynski et al. [2015], in order to compare our results. The simulation box is a cube with side length $L = 12.8\,\mathrm{kpc}$. At the center of the box is a idealized point source which emits photons at a rate of $\dot{N}_\gamma = 1 \times 10^{49}\,\mathrm{s}^{-1}$. The box is initialized with a gas with homogeneous number density $n_{\rm H} = 10^{-3}\,\mathrm{cm}^{-3}$. With these parameters, we find values of $R_{\rm St} = 6.79\,\mathrm{kpc}$ for the Strömgren radius and $t_{\rm rec} = 122.4\,\mathrm{Myr}$ for the recombination time. We initialize the gas as being purely neutral hydrogen (i.e. $x_{\rm H} = 1$, $x_{\rm H+} = 0$). Since the density response of the gas is irrelevant for the R-type expansion, we run the simulation without hydrodynamics, so that only radiative transfer and ionization chemistry take place.

In order to compare the time evolution of the radius of the ionized sphere to the analytical prediction, we need to define the radius of the sphere. In the simple analytical model, there is a sharp transition between the ionized and the non-ionized regions. However, in our simulation, due to the limited resolution of the grid, the transition region has a finite size. This means that a different definition of the radius

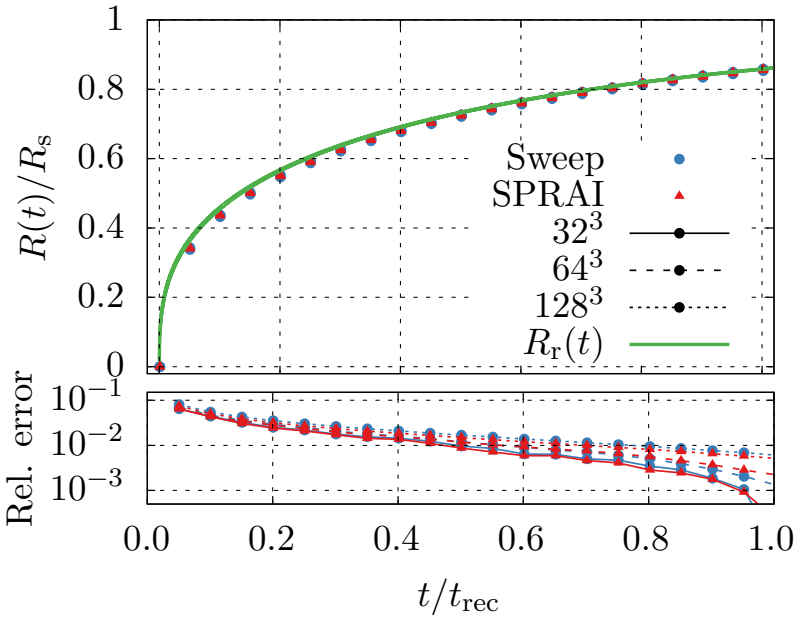

Figure 3.7. R-type expansion of a ionization front in a uniform medium. Top panel: Radius $R(t)$ of the ionization sphere normalized by the Strömgren radius $R_{\rm St}$ as a function of time t, normalized by the recombination time $t_{\rm rec}$. Blue dots: Numerical results for Sweep. Red triangles: Numerical results for SPRAI, Solid lines: Results for 32^3 particles, Dashed lines: Results for 64^3 particles, Dotted lines: Results for 128^3 particles. Green line: Analytical prediction $R_{\rm r}(t)$ given by Eq. 3.7. Bottom panel: Relative error $|R(t) - R_{\rm r}(t)|\,/R_{\rm r}(t)$

of the sphere is required. Here, we define the radius $R(t)$ as the radius at which the average ionization is $x_{\rm H+} = x_{\rm H} = 0.5$, i.e.

$$\int_{\boldsymbol{S}(R)} {\rm d}\boldsymbol{r}\, x_{\rm H}(\boldsymbol{r}) = 0.5, \tag{3.9}$$

where $\boldsymbol{S}(R)$ denotes the surface of the sphere of radius R around the origin. To calculate the value of this integral in practice, we average the HII abundance over a spherical shell of a given thickness $\Delta \ll R$.

In the upper panel of Fig. 3.7, the radius $R(t)$ (normalized by the Strömgren radius $R_{\rm St}$) of the ionized sphere is shown as a function of time (normalized by the recombination time $t_{\rm rec}$). In the lower panel, the relative error of the results compared to the analytical prediction is shown as a function of time. The results are shown for the three resolutions. For each resolution, we also show a comparison to the results obtained by performing the same simulation with the SPRAI code, as well as to the analytical prediction given by Eq. 3.7.

The comparison of analytical prediction and the simulation results shows that, while after the first timestep, the error is on the order of $\approx 8\%$, it decreases with

time and drops below 1% for all resolutions towards the end of the simulation. In contrast to our expectations, the agreement with the analytical prediction decreases with increasing resolution. While the simulation with 128^3 particles shows a relative error of $\approx 0.8\%$ at the end of the simulation, the simulation with 32^3 particles drops to an error of $\approx 0.1\%$ at the same time. We do not have an intuitive explanation for these results. However, we emphasize that the analytical prediction assumes a perfectly sharp boundary, which does not exist in practice, where the boundary has an associated thickness. Due to this, the value of the radius depends quite strongly on the definition of the radius in Eq. 3.9. While the choice of a ionization threshold 0.5 is intuitive, a different value will give rise to different radii and therefore change the dependence of the relative error on the resolution of the simulation.

For all resolutions, the results of Sweep and SPRAI agree very well, which increases our confidence in the numerical results. There is no clear difference between the relative errors of the two codes. While the relative error is slightly lower for Sweep at 128^3, the exact opposite is visible at 64^3 where SPRAI shows slightly lower errors. At 32^3, the results of both codes agree well with the analytical prediction and show virtually no difference in the relative error.

D-type expansion

Our D-type expansion test is set up very similar to the R-type test. The main qualitative difference between the two setups is that we need to take hydrodynamics into account, since the D-type expansion is due to the gas response driven by the thermal pressure between the inner, ionized region and the outer, neutral region. As in the R-type test, we chose our parameters as in Jaura et al. [2018], in order to facilitate comparison. We perform the D-type expansion for the 128^3 resolution case.

In order to use the analytical prediction given by Eq. 3.8, we need to obtain a value for the speed of sound in the ionized medium c_s. In principle, one could obtain the speed of sound using the temperature of the ionized medium via

$$c_\mathrm{s} = \sqrt{\frac{\gamma k_\mathrm{B} T_\mathrm{avg}}{\mu m_\mathrm{H}}}, \tag{3.10}$$

where $\gamma = \frac{5}{3}$ is the adiabatic index, μ is the mean molecular weight (in atomic units) and m_H is the atomic mass of hydrogen.

However, this is difficult in practice, since the temperature is not constant inside the ionized sphere. Therefore, we treat the speed of sound c_s as a fit parameter to our data, which is consistent with the approach in Jaura et al. [2018]. In doing so, we obtain a value of $12.8\,\mathrm{km\,s^{-1}}$ corresponding to an average temperature $T_\mathrm{avg} = 11\,914\,\mathrm{K}$.

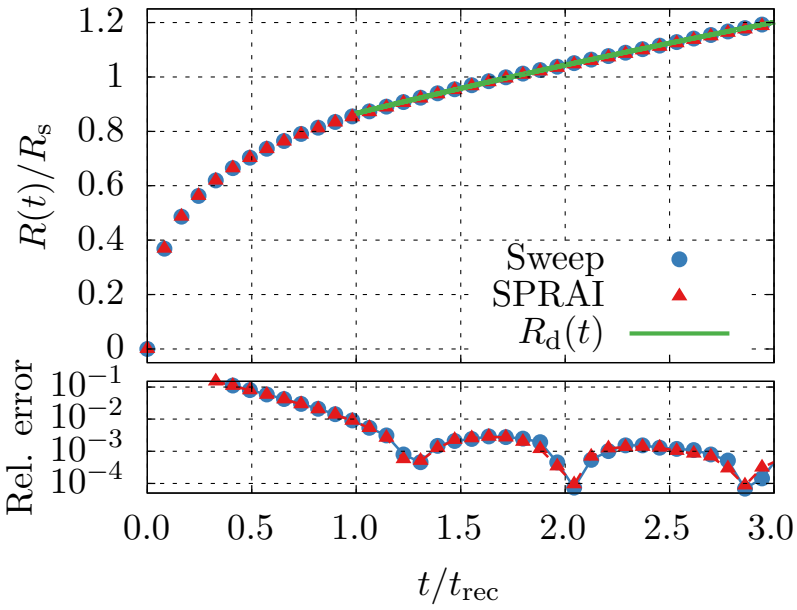

Figure 3.8. D-type expansion of a ionization front in a uniform medium. Top panel: Radius $R(t)$ of the ionization sphere normalized by the Strömgren radius R_{St} as a function of time t, normalized by the recombination time t_{rec}. Blue dots: Numerical results for Sweep. Red triangles: Numerical results for SPRAI, Green line: Analytical prediction $R_d(t)$ given by Eq. 3.8. Bottom panel: Relative error $|R(t) - R_d(t)|/R_d(t)$

In the top panel Fig. 3.8, the dependence of the radius of the ionization sphere, normalized by the Strömgren radius R_{St} is shown as a function of the time, normalized by the recombination time t_{rec} as well as the analytical prediction given by Eq. 3.8. The prediction describes the behavior for $R(t) > R_{St}$, but we also display the solution at lower times, starting at $t = t_{rec}$ and find that it describes the data quite well even in this range. This is confirmed by the relative error of the data with respect to the analytical prediction, shown in the lower panel of Fig. 3.8. Beginning at $t = t_{rec}$ the error never exceeds 1%. We find no discernible difference in the relative error between the results of Sweep and SPRAI, solidifying the fact that Sweep produces physically correct results.

Since this is the only test involving hydrodynamics, we will also discuss the relative performance of radiative transfer compared to the other parts of the code here, even though it should be noted that such a performance comparison is problem dependent. In a run using Sweep, the radiative transfer takes up approximately 75% of the total computation time, with Voronoi grid construction (12%) and hydrodynamics (10%) using up most of the remaining time. While this implies that in this test radiative transfer is by far the most expensive part of the code, Sweep still vastly outperforms SPRAI (which takes up $\approx 98\%$ of the total run time) by a factor of ≈ 16.

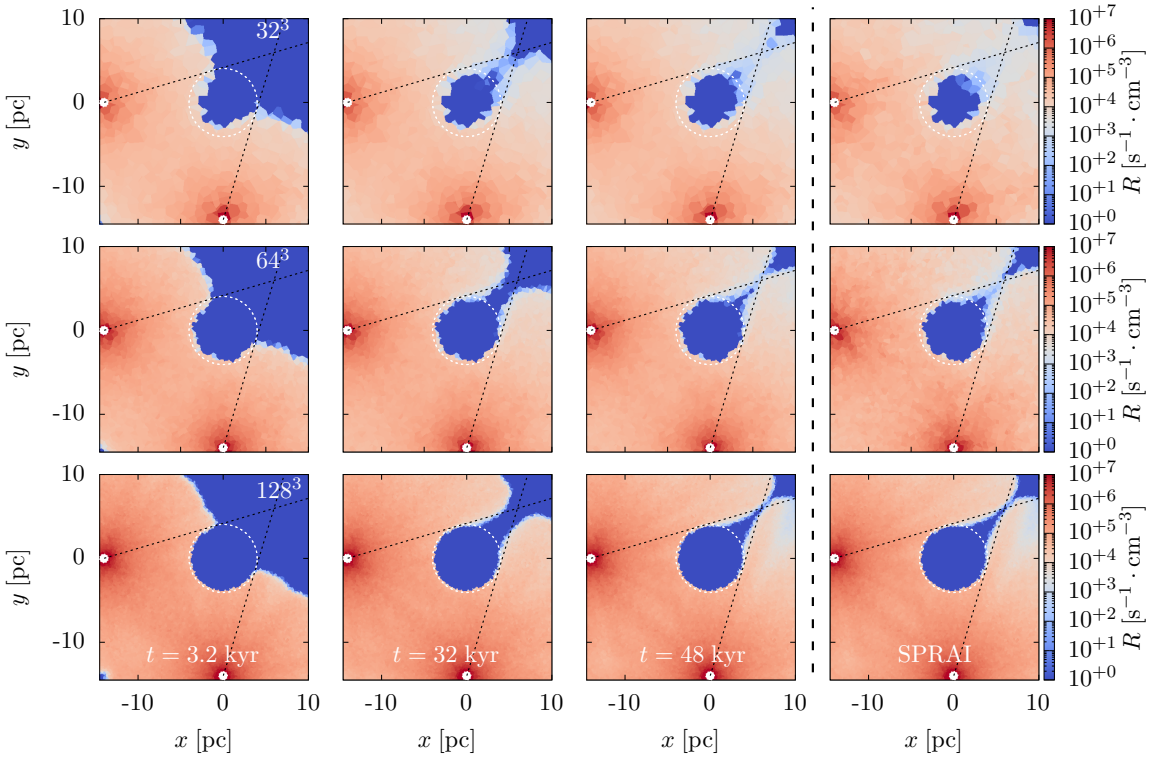

Figure 3.9. The photon rate R in a slice through the $z = 0$-plane of the simulation box. First row: 32^3 particles, Second row: 64^3 particles, Third row: 128^3. First column: Sweep at $t = 3.2\,\mathrm{kyr}$, second column: Sweep at $t = 32\,\mathrm{kyr}$, third column: $t = 48\,\mathrm{kyr}$, last column: SPRAI at $t = 48\,\mathrm{kyr}$. The white dashed line represents the over-dense clump. White solid circles represent the position of the sources. The black dashed lines delineate the shape of an ideal shadow behind the clump.

3.3.2 Shadowing behavior of radiation field behind a clump

The previous tests have established that the Sweep method replicates results obtained with SPRAI. However, due to the spherical symmetry, the directional dependence of the radiation is not tested in the expansion tests. To do so, we perform another test in which we study how well dense objects cast shadows behind them.

This test is set up in the same way as the corresponding test in Jaura et al. [2018] and we will use the results obtained by the SPRAI method as a basis for comparison. The simulation takes place in a box of side length $L = 32\,\mathrm{pc}$, filled with neutral hydrogen at a number density of $n_\mathrm{H} = 1\,\mathrm{cm}^{-3}$ everywhere except in the center of the box where a dense clump at number density $n_\mathrm{H} = 1000\,\mathrm{cm}^{-3}$ and radius $R = 4\,\mathrm{pc}$ is placed. The temperature of the gas is set to $T = 1000\,\mathrm{K}$. Two point sources are placed at $\boldsymbol{r}_1 = (-14, 0, 0)$ pc and $\boldsymbol{r}_2 = (0, -14, 0)$ pc, both emitting photons at a rate of $N_\gamma = 1.61 \times 10^{48}\,\mathrm{s}^{-1}$. The time-step of the simulation is $\Delta t = 0.32\,\mathrm{kyr}$.

An analysis of this test, which includes hydrodynamics and discusses the tempera-

ture, pressure, and density response has been performed in the original SPRAI paper [Jaura et al., 2018]. Since the code coupling the radiative transfer to the hydrodynamics of Arepo is the same as the one used in SPRAI, any results obtained there are also valid for our method. Since we are interested only in the photon rate field resulting from the simulations, we perform these simulations without hydrodynamics. Here, the photon rate $R(\boldsymbol{r}, t)$ is defined as the number of photons in the frequency bin corresponding to the ionization of hydrogen at 13.6 eV per unit time per unit volume, i.e. as $R(\boldsymbol{r}, t) = \frac{1}{13.6\,\mathrm{eV}} \int_{\boldsymbol{\Omega}} I_\nu(\boldsymbol{r}, \boldsymbol{\Omega}, t) \mathrm{d}\boldsymbol{\Omega}$. In Figure 3.9, the photon rate R is shown as a slice through the simulation box along the x-y plane for different times (columns) and resolutions (rows). For each resolution, the result obtained with SPRAI is shown for the last time (i.e. $t = 48\,\mathrm{kyr}$).

It is clear that the over-dense clump acts as an obstacle and initially prevents photons from entering its shadow. However, due to numerical diffusion, the shadow is not as sharp as expected in the exact solution. As time progresses, the photon rate in the (theoretical) shadow behind the clump increases, because the regions between the sources and the shadow have become ionized and stopped absorbing photons. With increasing resolution, the effect of numerical diffusion decreases and the shadow becomes more defined.

In order to quantify the shadowing behavior and to compare Sweep and SPRAI as well as the quality of the shadow at different resolutions, we calculate the mass averaged fraction of ionized hydrogen in the volume of the shadow. The volume is given by the intersection of two (infinitely extended) cones, with their tips at $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ respectively and their base determined by the great circle lying in the over-dense clump. In the 2D slice shown in Fig. 3.9, this volume V_S corresponds to the area between the white dashed circle and the black dashed lines. This fraction $\bar{x_\mathrm{H}}$ is given by

$$\overline{x_\mathrm{H}} = \frac{\int_{V_\mathrm{S}} x_\mathrm{H}(\boldsymbol{r}) \rho(\boldsymbol{r}) \mathrm{d}V}{\int_{V_\mathrm{S}} \rho(\boldsymbol{r}) \mathrm{d}V}, \tag{3.11}$$

where $x_\mathrm{H}(\boldsymbol{r})$ is the abundance of ionized hydrogen at position $\boldsymbol{r}$ and $\rho(\boldsymbol{r})$ denotes the mass density at position $\boldsymbol{r}$.

In Figure 3.10, $\overline{x_\mathrm{H}}$ is shown as a function of time. Neither Sweep nor SPRAI form a perfect shadow, demonstrated by the fact that the ionization fraction begins to increase at $t \approx 20\,\mathrm{kyr}$. Before this time, the ionization front has not reached the region behind the over-dense clump. Clearly, the shadowing behavior improves with higher resolutions. This is in line with the explanation that the protrusion of the ionization front into the shadow is due to numerical diffusion, since higher resolutions decrease the effect of numerical diffusion. We also find that Sweep forms a slightly

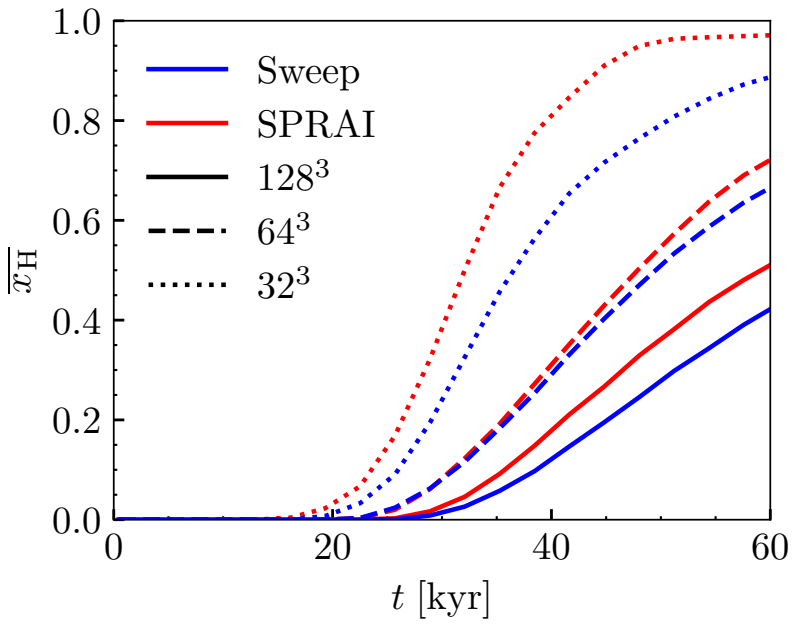

Figure 3.10. The average hydrogen abundance $\overline{x_{\mathrm{H}}}$ (see Eq. 3.11) in the shadow volume as a function of time for both Sweep (blue) and SPRAI (red) for three different resolutions: 128^3 (solid line), 64^3 (dashed line) and 32^3 (dotted line),

more defined shadow, with the ionization fraction strictly below the values for SPRAI for all times and resolutions.

3.3.3 Scattering

In order to test the source iteration scheme described in Section 3.2.3, we test a setup similar to the illumination of a dense clump described in Section 3.3.2. The only difference is that this test setup will only use one source positioned at $\boldsymbol{r}_1 = (-4.8, 0, 0)$ pc, which is very close to the dense clump positioned in the center (which has radius $r = 4\,\mathrm{pc}$), creating a large shadow behind the clump.

In order to test that our implementation of the source iteration reproduces scattering in a physical manner, we perform a number of simulations in which we vary only the effective scattering cross section. For simplicity, we choose a model in which the scattering coefficient is entirely independent of the chemical composition of the gas, with the scattered fraction of the intensity in a cell given by

$$(\mathrm{d}I_\nu)_{\mathrm{s}} = I_\nu \left(1 - e^{-\mathrm{d}n_{\mathrm{nucleons}}\sigma_{\mathrm{s}}}\right), \tag{3.12}$$

where $_\nu$ is the incoming intensity, n_{nucleons} is the column density of nucleons in the cell, σ_{s} is the effective cross section of the scattering. The column density is calculated as

$$\mathrm{d}n_{\mathrm{nucleons}} = n\langle \mathrm{d}r \rangle, \tag{3.13}$$

with the number density of nucleons n and the mean distance traveled in the cell $\langle \mathrm{d}r \rangle$. For more details of these calculations see Section 2.2 in Jaura et al. [2018]

For the resolution of the tests, we chose $n = 128^3$ particles for all test simulations. We vary the scattering cross section as $\sigma_\mathrm{s} = 0\,\mathrm{cm}^{-2}, 5 \times 10^{-22}\,\mathrm{cm}^{-2}, 1 \times 10^{21}\,\mathrm{cm}^{-2}$, and $1 \times 10^{20}\,\mathrm{cm}^{-2}$. In addition, we vary the number of source iterations performed as $n_\mathrm{it,scat} = 2, 3, 4$ in order to check the convergence of the method.

We intuitively expect the shadow to become less and less prominent as the scattering cross section increases, due to the influx of scatter light on the low density gas.

The results of these tests at $t = 40\,\mathrm{kyr}$ are shown in Fig. 3.11. Even for $\sigma_\mathrm{s} = 0\,\mathrm{cm}^{-2}$, the ionized regions protrude substantially beyond the ideal shadow. This is the same numerical diffusion we already observed in Section 3.3.2. However, the shadow volume clearly decreases for increasing values of the scattering cross sections until the shadow vanishes almost entirely at $\sigma_\mathrm{s} = 5 \times 10^{21}\,\mathrm{cm}^{-2}$. We also note that the number of iterations barely affects the result after $n_\mathrm{it,scat} = 2$, implying that the method converges rather quickly in this test case.

3.3.4 Periodic Boundary Conditions

As discussed in Section 3.2.7, Sweep handles periodic boundary conditions by an iterative scheme. In order to show that this scheme produces physical results, we perform a test similar to the R-type expansion in Section 3.3.1. We chose the case with a resolution of $n = 32^3$. The primary difference in this new test is the position of the point source, which we move to $\boldsymbol{r} = (6.336, 0, 0)$ kpc. Since the same box size of $L = 12.8\,\mathrm{kpc}$ is used, that corresponds to a source located very close to the right boundary of the simulation box.

In Figure 3.13, the mean relative error given by Eq. 3.5 is shown as a function of the number of periodic boundary iterations at different times with and without warm starting. The first clear trend that can be seen is that while the initial error remains roughly constant throughout time, the speed of the convergence decreases drastically. While it takes $N_\mathrm{it} = 6$ iterations to reach an error of $E < 10^{-10}$ for the first timestep at $t = 14.5\,\mathrm{Myr}$, it takes $N_\mathrm{it} = 14$ iterations to reach the same threshold at $t = 43.5\,\mathrm{Myr}$.

We believe that this effect is partially due to re-entry dependencies - a cell very close the the right boundary at $x = 6.4\,\mathrm{kpc}$ will often have downwind dependencies at the left side of the boundary at $x = -6.4\,\mathrm{kpc}$, especially for a sweep direction which is close to being contained within the y-z plane. The cells on the other side of the boundary will then often have downwind neighbors on the right side of the boundary. The effective distance traveled of photons along such re-entry dependencies is strongly

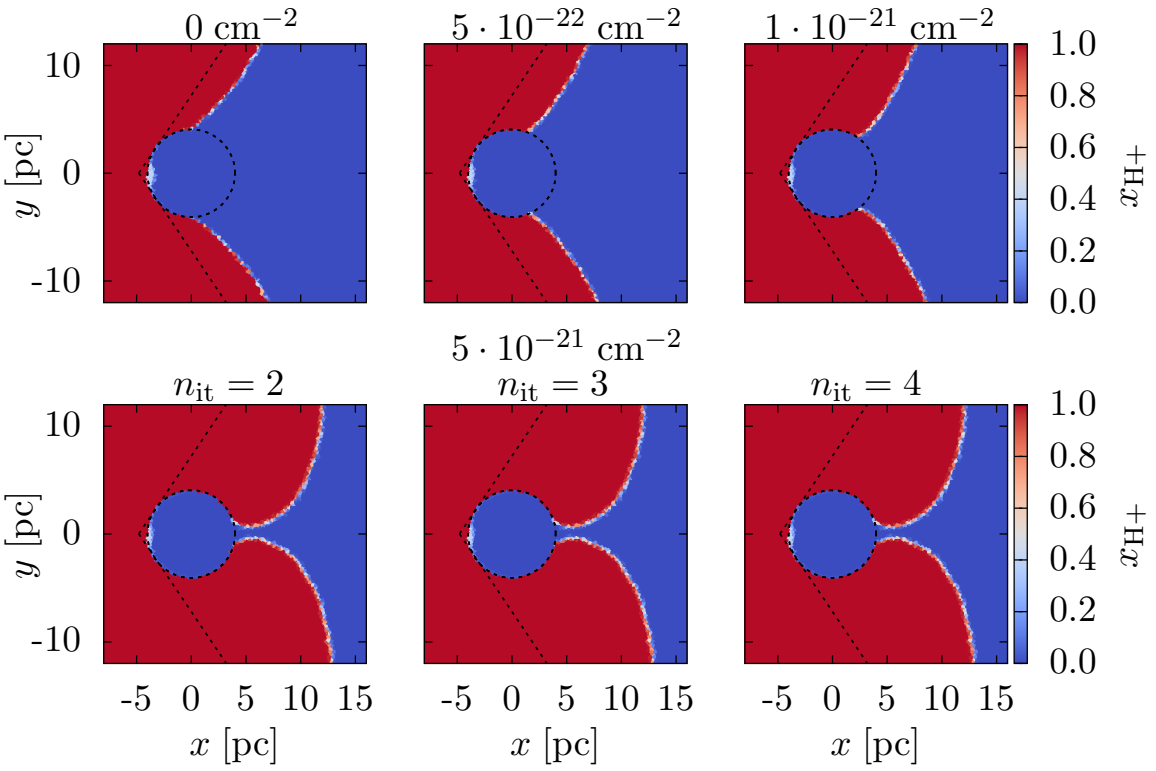

Figure 3.11. The abundance of ionized hydrogen $x_{\mathrm{H+}}$ at $t = 40\,\mathrm{kyr}$ in a slice through the z-plane of the simulation box. The x and y axis show the x and y position in the box respectively. The dark dashed circle indicates the position of the over-dense clump. The dashed lines indicate indicate the boundaries of a hypothetical, perfectly sharp shadow. Top left: $\sigma_{\mathrm{s}} = 0\,\mathrm{cm}^{-2}$, Top center: $\sigma_{\mathrm{s}} = 5 \times 10^{-22}\,\mathrm{cm}^{-2}$, Top right: $\sigma_{\mathrm{s}} = 1 \times 10^{-21}\,\mathrm{cm}^{-2}$, Bottom: $\sigma_{\mathrm{s}} = 5 \times 10^{21}\,\mathrm{cm}^{-2}$, Bottom left: $n_{\mathrm{it,scat}} = 2$, Bottom center: $n_{\mathrm{it,scat}} = 3$, Bottom right: $n_{\mathrm{it,scat}} = 4$,

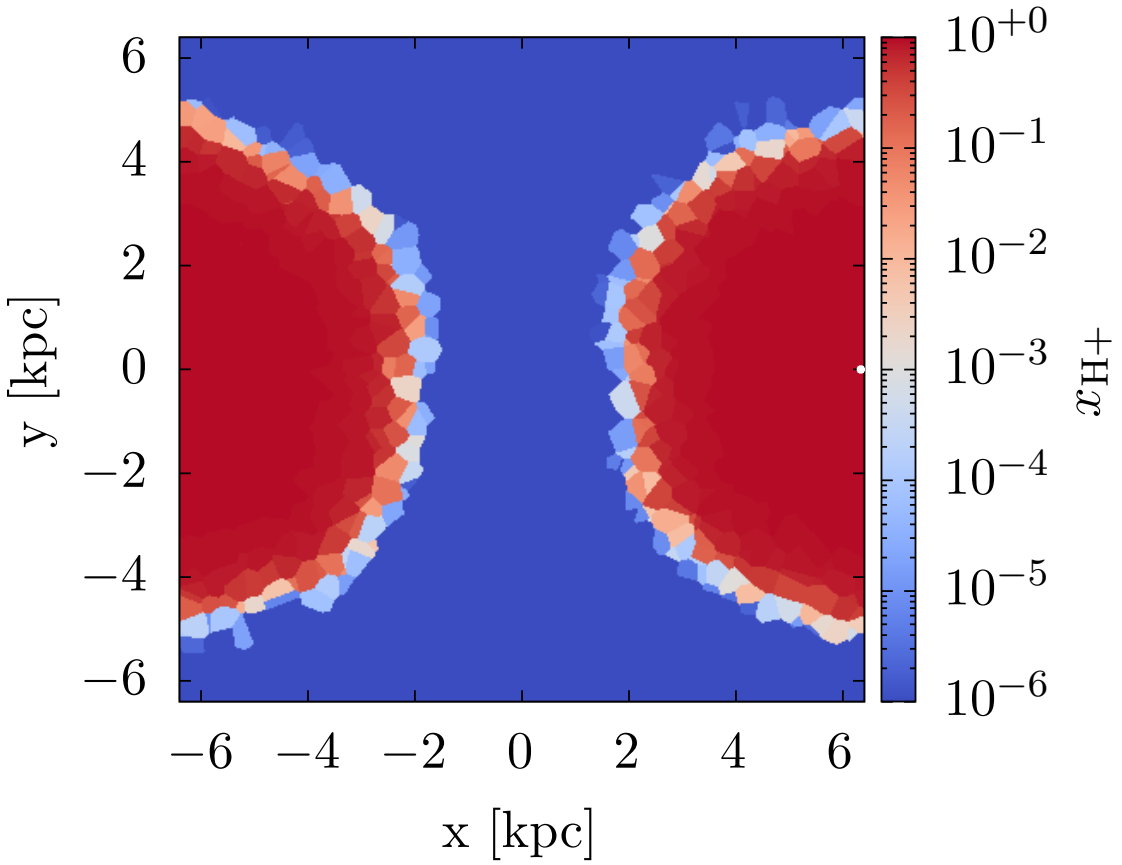

Figure 3.12. A slice through the simulation box of the test described in Section 3.3.4 at $z = 0$ and $t = 58\,\mathrm{Myr} \approx 0.473 \cdot t_{\mathrm{rec}}$. The x and y axis show the x and y position in the box respectively. The color scale indicates the abundance of ionized hydrogen $x_{\mathrm{H+}}$, with blue being neutral and red being fully ionized. The small white dot indicates the position of the source in the box.

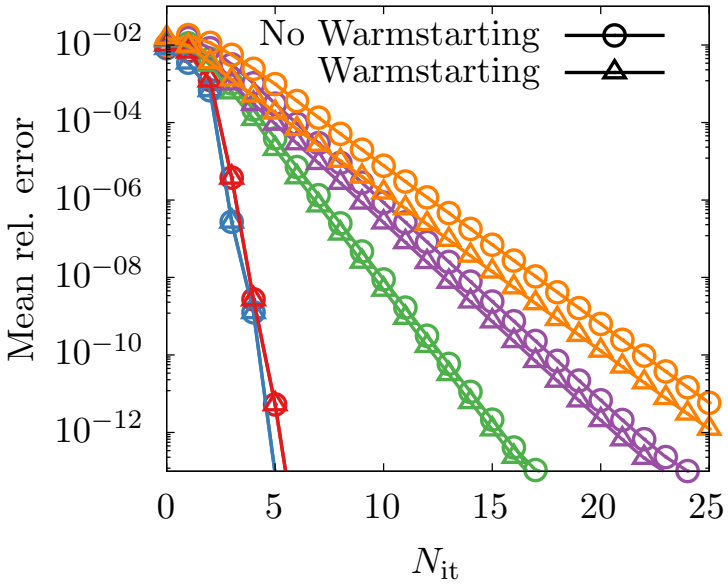

Figure 3.13. The mean relative error in the source terms (given by Eq. 3.5) as a function of the number of iterations N_{it} for the periodic boundary conditions test described in Section 3.3.4. The colors correspond to different time steps. Blue: $t = 14.5\,\mathrm{Myr}$. Red: $t = 29\,\mathrm{Myr}$. Green: $t = 43.5\,\mathrm{Myr}$. Purple: $t = 58\,\mathrm{Myr}$. Orange: $t = 72.5\,\mathrm{Myr}$. Circles: Without warmstarting Triangles: With warmstarting

determined by the number of iterations N_{it} since it takes one full iteration for the information about those photons to travel one cell.

This effect is exacerbated due to the location of the source in the test setup described above, since it is located very close to the boundary. This means that a high number of photons will be traveling along the boundary in a direction parallel to the y-z plane.

In Figure 3.13, the mean relative error given by Eq. 3.5 is shown as a function of the number of periodic boundary iterations at different times with and without warmstarting.

As another test of the convergence of the iterative scheme, we calculate the radius of the ionized bubble as a function of time and compare the result to the analytical prediction. The simulations in this test are equal to those in Section 3.3.1, with the only difference being the position of the source in the box, requiring the proper treatment of periodic boundary conditions in order to reproduce the behavior of the R-type expansion. We choose the box with 32^3 particles and perform simulations with iteration counts $1 \leq n_{\mathrm{it,pbc}} \leq 20$. All other parameters are chosen equal to those in Section 3.3.1.

Figure 3.14 shows the relative error between the radius of the ionized sphere and the

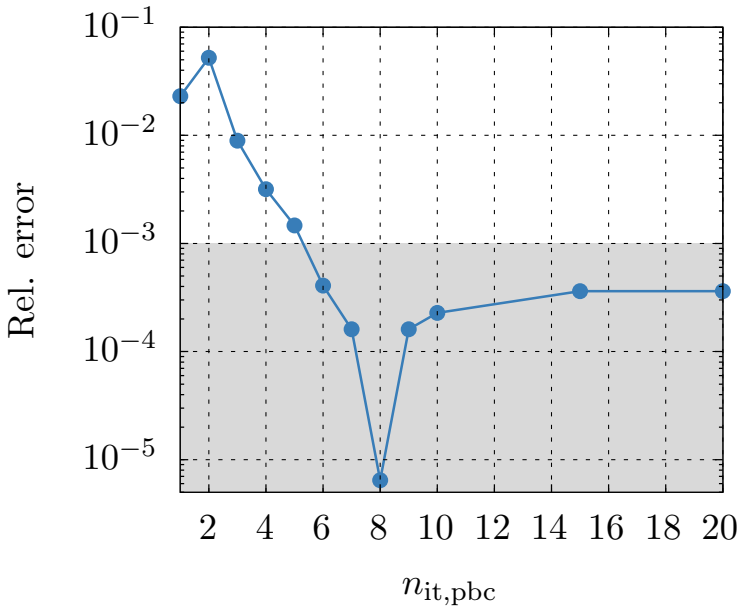

Figure 3.14. Relative error of the radius of the ionized sphere in the R-type expansion in a uniform medium with a source located at the boundary of the box as a function of the number of periodic boundary iterations $n_{\mathrm{it,pbc}}$. The gray area signifies the approximate level of error expected due to the difference between numerical results and analytical prediction for a R-type expansion in the absence of periodic boundary conditions.

analytical prediction as a function of the number of periodic iterations. As expected, the error decreases with the number of periodic iterations. After approximately 5 iterations, the error reaches values below 10^{-3}, at which point it is indistinguishable from the error between the analytical prediction and the numerical results (see Fig. 3.7) which means that any discussion of the exact behavior of the error below that point is futile.

3.3.5 Strong Scaling

After the physical tests, we now discuss the scaling behavior of the Sweep method. We begin by studying the strong scaling, i.e. the dependence of the time to solution T of a problem of fixed time on the number n of computing cores. It is customary to study the scaling behavior of the code by comparing the time to solution $t(n)$ for a run on n cores to the time to solution t_{base} for a base case at n_{base} (typically, $n_{\mathrm{base}} = 1$) cores. The time to solution of an ideally parallelized code t_{ideal} decreases as

$$t_{\mathrm{ideal}}(n) = \frac{t_{\mathrm{base}}}{n/n_{\mathrm{base}}}. \tag{3.14}$$

The parallel speedup S is defined as

$$S(n) = \frac{t_{\text{base}}}{t(n)}, \tag{3.15}$$

and it follows from Eq. 3.14 that the speedup of an ideally parallelized code S_{ideal} is given by

$$S_{\text{ideal}}(n) = \frac{n}{n_{\text{base}}}. \tag{3.16}$$

We also define the parallel efficiency $\epsilon(n)$ as the fraction of the achieved speedup:

$$\epsilon(n) = \frac{S(n)}{S_{\text{ideal}}} = \frac{S(n)}{n/n_{\text{base}}}. \tag{3.17}$$

For these tests, we use the same simulation setup as in the shadowing test described in Section 3.3.2. We study three different fixed problem sizes with 32^3, 256^3 and 512^3 Voronoi cells respectively. For each problem, we perform simulations for different numbers of cores. In the case of 32^3 particles, we use a range from $n = 1$ to $n = 512$. For 256^3 we use $n = 96$ to $n = 2048$ and for 512^3, we perform runs from $n = 2048$ to $n = 8192$ cores. For the smallest case of 32^3 particles, we compare our results to the SPRAI code. We did not include a comparison to SPRAI for the larger problem sizes, since the run-time grew too large.

In Fig. 3.15, the parallel speedup is shown as a function of the number of compute cores in comparison to the ideal behavior given by Eq. 3.16 for each of the three problem sizes. For low core numbers, both SPRAI and Sweep scale well with the number of cores. At $n = 96$ cores the speedup of Sweep is $S(96) \approx 40$ with SPRAI being slightly faster at $S(96) \approx 48$, corresponding to parallel efficiencies of $\epsilon(96) = 42\%$ and $\epsilon(96) = 50\%$ respectively. At higher core numbers, the rate of increase in the speedup declines, the parallel efficiency drops to $\epsilon(512) = 28\%$ for SPRAI and 20% for Sweep. This behavior is to be expected, since the ratio of the required inter-process communication to communicate the fluxes crossing processor domains to the amount of cells to solve locally decreases as the number of cores increases.

For the higher resolution runs, the rate of decrease in the efficiency of Sweep is lower. Comparing the run with 256^3 particles at $n = 96$ to that with $n = 2048$ shows a decrease in parallel efficiency to $\approx 60\%$. For the run with 512^3 particles, the parallel efficiency increases beyond 1. Such a result may initially seem counter-intuitive, but can be explained by the fact that for some numbers of cores the domain decomposition turns out to be particularly unfortunate, decreasing the efficiency of Sweep due to worse scheduling behavior or similar effects. If such a case is used as the reference simulation to which simulations at higher core numbers are compared the result are parallel efficiencies larger than 1.

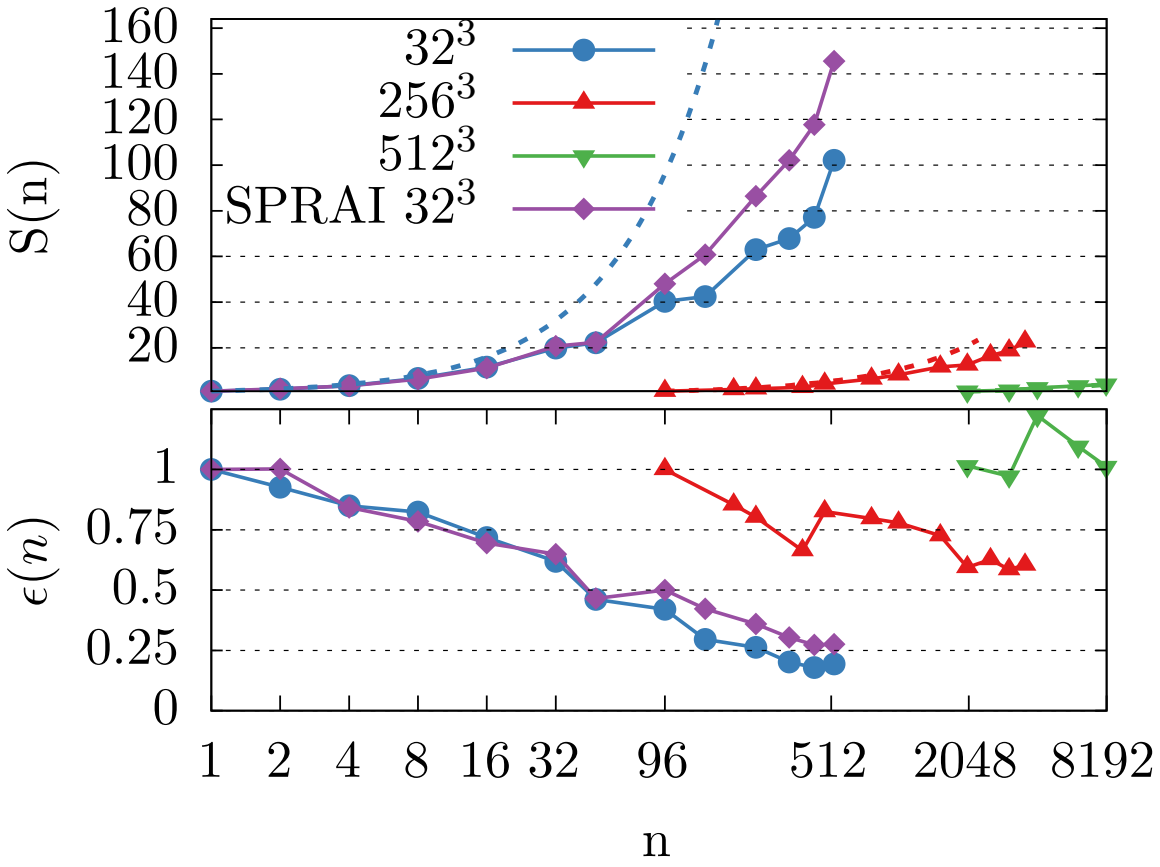

Figure 3.15. Top: The parallel speedup of $S(n)$ as a function of the number of cores n. For Sweep, three problem sizes are shown: 32^3 (blue circles), 256^3 (red triangles), 512^3 (green triangles). For SPRAI, we show the problem size 32^3 (purple diamonds). For each problem size, the ideal, linear scaling behavior with respect to the base cases $n_{\text{base}} = 1$ for 32^3, $n_{\text{base}} = 96$ for 256^3 and $n_{\text{base}} = 2048$ for 512^3 is given by Eq. 3.16 is shown as the dashed line. Bottom: The parallel efficiency (defined in Eq. 3.17) as a function of the number of cores for the same configurations.

This, along with the fact that the real run-time of the code is hidden, highlights the fundamental problem with simply comparing the speedup of two codes without comparing their respective run times, since the parallel efficiency improves as the single-core performance of the parallel part of the code decreases.

Therefore, it is important to show the run time of the code. In order to compare the run times of different problem sizes in a reasonable manner, we define the time per task as

$$t_{\text{task}}(n) = \frac{nt(n)}{N_{\text{dir}} N_{\text{cells}} N_{\text{freq}}} \,, \tag{3.18}$$

where N_{dir} is the number of directions (84 in our case), N_{cells} is the number of cells (32^3, 256^3 and 512^3, depending on the problem size) and N_{freq} is the number of frequencies. This is the effective time it takes a single core to solve a single cell in a single direction for a single frequency. For an ideally parallelized code, t_{task} is independent of the number of cores. This definition allows a comparison across different problem sizes by looking at the effective loss in performance given by $t_{\text{task}}(1)/t_{\text{task}}(n)$, which we believe is a realistic assessment of the performance of the code between runs of vastly different numbers of cores.

In Fig. 3.16, the time per task is shown as a function of the number of cores for the three different problem sizes. The figure shows that the two codes obtain very different run times on this particular test. Whereas the scaling behavior of the two codes are very similar, Sweep outperforms SPRAI by a factor of ≈ 20 at $n = 1$ cores. It is important to note, however, that this result does not hold for any kind of simulation, since Sweep is written with a focus on simulations of reionization, where mean free path lengths are potentially high, while SPRAI performs comparatively well in dense media.

The lower panel of Fig. 3.16 shows the effective performance $t_{\text{task}}(1)/t_{\text{task}}(n)$. This demonstrates that, while the effective performance of Sweep decreases with the number of cores, it only decreases to $\approx 20\%$ at $n = 8192$ cores.

3.3.6 Weak Scaling

As another test of the parallel efficiency of Sweep with increasing number of cores, we perform a weak scaling test by increasing the problem size in proportion to the number of cores, thus keeping the number of cells per core constant. The speedup of an ideally parallelized algorithm in the weak scaling case is given by

$$S(n) = 1. \tag{3.19}$$

As a base case, we choose the $n = 1$ case with a resolution of 32^3 cells, identical to the corresponding $n = 1$ simulation in the strong scaling test. In addition to the

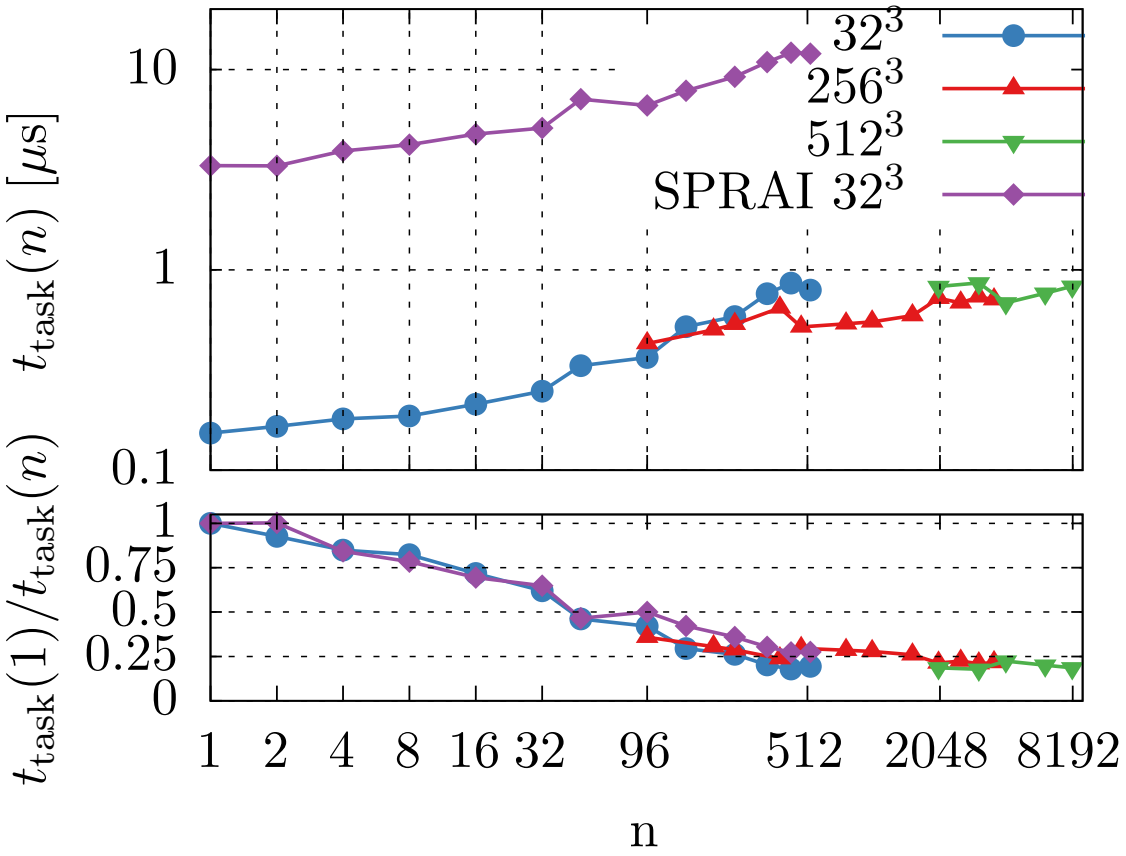

Figure 3.16. Top: The time per task t_{task} (see Eq. 3.18) as a function of the number of cores n for three problem sizes: 32^3 (blue), 256^3 (red), 512^3 (green). Bottom: The performance loss $t_{\text{task}}(1)/t_{\text{task}}(n)$ as a function of the number of cores for the same three problems.

base case we perform simulations for $n = 8, 48, 480$ (528 for SPRAI, due to memory requirements) and 4096 cores with resolutions of 64^3, 128^3, 256^3 and 512^3 particles respectively[3]. We do not include the case of 4096 cores on 512^3 particles for SPRAI, due to slightly increased memory requirements making a run on this number of cores difficult.

Figure 3.17 shows the speedup as a function of the number of cores n, which, in the case of weak scaling is equivalent to the parallel efficiency. The speedup initially drops quite quickly, to values of $\sim 22\,\%$ for Sweep and $\sim 9\,\%$ for SPRAI at $n = 48$ cores. However, the speedup does not decrease further and remains at similar values until $n = 4096$ cores.

We believe that the initial decrease in efficiency is due to the overhead in communication compared to the base case of $n = 1$ cores. In particular, the re-entry dependencies discussed in Section 3.2.5 significantly slow down performance due to the amount of communication in which very few fluxes are exchanged.

While some parallel efficiency will be lost due to the increasing amount of communication for higher number of cores, another effect diminishing the parallel efficiency is the "pipe fill" described in Section 3.2.5, since cores whose domains lie in the inner regions of the simulation box cannot begin solving before those with domains in the outer regions have finished their sweep. At low numbers of cores ($n = 1$ or $n = 8$), no such domains exist, since every domain is adjacent to a boundary of the simulation box. As the number of cores increases, the number of inner regions increases and parallel efficiency decreases. In order to check whether this effect is already affecting our results and decreasing the parallel efficiency significantly, we generated program output which displayed the timing at which the first task is solved for each core. This allowed us to estimate the amount of performance lost due to idle time. We found that for $n = 4096$, this delay is still insignificant compared to the communication overhead.

3.4 Conclusion

In this paper, we introduced a sweep-based radiative transfer method which we implemented for the moving-mesh hydrodynamics code Arepo. The method solves the radiative transfer equation under the assumption of an infinite speed of light and a steady state solution. As a first test of our implementation of the code, we studied the expansion of an HII region around a point source for the R-type and the D-type

[3]Note that the ratio of particles to cores does not remain exactly constant because of the number of cores was required to be divisible by 48

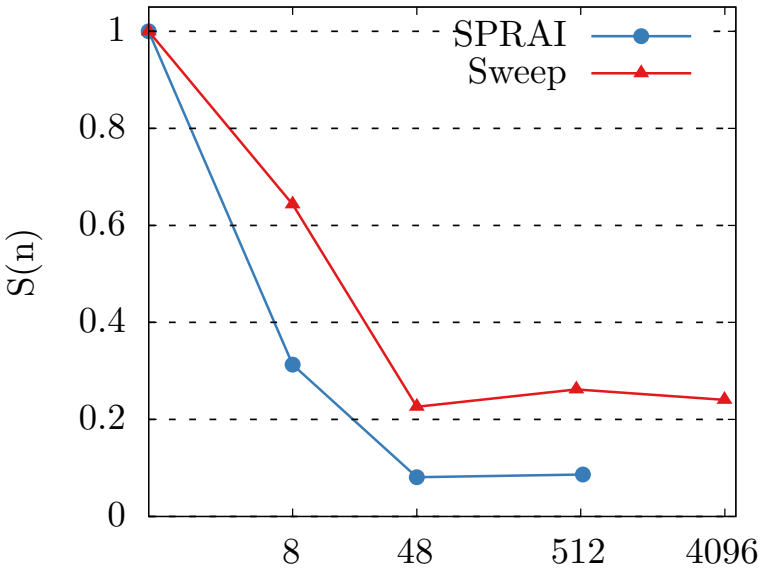

Figure 3.17. The parallel speedup $S(n)$ as a function of the number of cores n for a problem which scales in size with the number of cores for both Sweep (red) and SPRAI (blue). The ideal, linear scaling behavior given by Eq. 3.19 is shown in green.

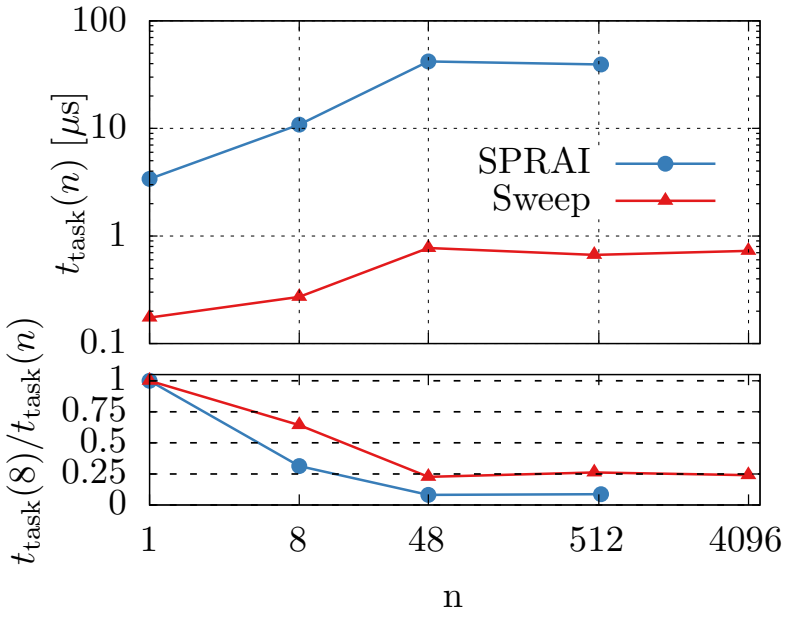

Figure 3.18. Top: The time per task t_{task} (see Eq. 3.18) as a function of the number of cores n in a weak scaling test where the problem size scales with the number of cores from 32^3 at $n = 1$ to 512^3 at $n = 4096$. Bottom: The performance loss $t_{\text{task}}(1)/t_{\text{task}}(n)$ as a function of the number of cores. Blue line: SPRAI, red line: Sweep.

regime and compared the results to the analytical predictions as well as results obtained with the SPRAI code. We also performed a test which allowed us to study the shadowing behavior behind a dense blob of gas. For all the tests, we find good agreement with our results and the results obtained via SPRAI.

In addition, we performed tests to better understand whether the source iteration method employed correctly deals with scattering. To ensure that the code can also handle periodic boundary conditions we performed a series of tests similar to the R-type expansion but with a source very close to the boundaries of the box. We find that after $\sim$5 iterations of the periodic boundary sources, the results are virtually indistinguishable from those of the standard R-type test.

We also analyze the parallel efficiency of our code in order to assess whether large-scale simulations would be feasible with this method. To this end, we perform strong and weak scaling tests. For the strong scaling, we find similar scaling behavior between SPRAI and Sweep for the smallest test case (32^3 particles), with Sweep outperforming SPRAI by a factor of $\sim$10 in the actual runtime. We find a constant, slow decrease in the parallel efficiency down to $\sim$20% at 512 cores for the smallest test case, however comparing the run time per cell between the large test cases (512^3 particles) at 8192 cores and the smallest at 1 cores, we find that Sweep still operates at $\sim$20% efficiency. In the case of weak scaling, the parallel efficiency decreases quite significantly with the number of cores. However, Sweep still performs better than SPRAI by a factor of 4 at $n = 512$ with a parallel efficiency of $\sim$25% at 4096 cores. We expect the reduction in efficiency to be due to the fact that the domains of some cores are located in the inner region of the entire computational domain.

One possible measure to improve parallel efficiency is to change the domain decomposition. This is made more complicated due to the fact that sweep is intended to run in parallel to gravitation and hydrodynamics. In addition, estimating the amount of computational work for a single Voronoi cell is straightforward for gravitation and hydrodynamics, but difficult in general for radiative transfer, where assigning the cell to a certain core not only increases the total work load of that core but also affects the global scheduling problem.

Finally, we note that although we have developed Sweep with the goal of modeling cosmological reionization, the algorithm itself is far more general than this and could readily be adapted for use in other applications in which it is advantageous to have a method for modeling radiation transfer that is independent of the number of sources.

Chapter 4

Subsweep: Subsweep: A code for radiative transfer postprocessing based on the Sweep method

In this chapter we describe the Subsweep code for radiative transfer postprocessing. Subsweep is a standalone code that takes outputs from hydrodynamical simulations and produces the dynamically evolving solution to radiative transfer of the simulation. We also introduce the sub-timestepping improvement to the Sweep algorithm described in Chapter 3, which allows cell timesteps to adapt to the local properties of the medium, improving performance of the algorithm drastically.

Here, we describe the motivation behind the code, describe the relevant algorithms for domain decomposition, grid construction and radiative chemistry as well as the sub-timestepping improvement and its implementation. Eventually, we test the code on a variety of physical setups, such as the expansion of a HII region, its shadowing behavior, its ability to treat periodic boundary conditions and test the performance of the sub-timestepping algorithm.

4.1 Introduction

Reionization is the process in which the universe shifts from being fully neutral to being almost completely ionized everywhere. This is an important part of the transition from the primordial, homogeneous universe to the present-day universe which is full of heterogeneous, complex structures [see e.g. Loeb and Barkana, 2001, Wise, 2019, Zaroubi, 2013].

We want to understand reionization by performing a set of simulations in which radiative transfer and radiation chemistry are evolved on large cosmological simula-

tions such as the TNG simulations [Marinacci et al., 2018, Naiman et al., 2018, Nelson et al., 2018, 2019, 2021, Pillepich et al., 2018, 2019, Springel et al., 2018].

Here, we introduce a new code Subsweep which aims to solve radiative transfer and radiation chemistry on large inputs, such as those from cosmological simulations.

Radiative transfer is the physical theory that describes the propagation of radiation and its interactions with matter, such as absorption and scattering [Mihalas and Weibel-Mihalas, 1999]. There are multiple reasons why radiative transfer is a particularly challenging numerical problem, beginning with the high dimensionality of the quantity of interest: specific intensity which depends on three spatial, two directional, one temporal and one frequency dimension leading to a total of seven dimensions. In addition, the radiative transfer equation can be seen as an elliptical equation in optically thick media and as a hyperbolic equation in optically thin media, making it particularly difficult to select a single solution method that works well across the entire parameter space. Furthermore, we are primarily interested in physical scenarios in which the properties of the medium in which the radiation is transported change due to the influence of the radiation. At the same time, the radiation transport itself is dependent on the properties of the underlying medium (such as emissivity and opacity), calling for a solution to the coupled equations of radiative transfer and radiation chemistry.

Moment-based methods, where one solves the moments of the radiative transfer equation with some approximate closure relation, as opposed to the full radiative transfer equation, are a leading class of methods. This approximation can lead to drastically improved performance while decreasing the overall accuracy of the result. All moment-based methods need to choose a closure relation, which is typically given in terms of an expression for the Eddington tensor. One of the first examples of a moment-based method is the flux limited diffusion approach [Levermore and Pomraning, 1981, Whitehouse and Bate, 2004] in which one assumes that the intensity varies sufficiently slowly, and introduces a flux limiter to ensure that the signal speed of the radiation field remains lower than the speed of light. Flux-limited diffusion has been successfully applied in astrophysics [e.g. Boss, 2008, Krumholz et al., 2007], but its main drawback is its diffusive nature which results in a lack of proper shadow formation [see e.g. Hayes and Norman, 2003].

Another example of a moment-based method is the optically thin variable Eddington tensor method in which the Eddington tensor is computed under the assumption that all lines of sight to the sources in the simulation are optically thin [Gnedin and Abel, 2001b]. While efficient, this algorithm is applicable only to a comparatively narrow range of problems.

The M1 method is another moment-based method based on the M1 closure relation [Kannan et al., 2019]. While it is comparatively fast, it suffers from numerical problems inherent to moment-based methods, such as the two-beam instability [Rosdahl et al., 2013].

Another class of methods is given by Monte Carlo approaches, in which radiation is represented by individual photon packets [Dullemond et al., 2012, Oxley and Woolfson, 2003]. Packets are created by sampling them according to a probability distribution and interacting with the matter following statistical predictions. These distributions are determined by the properties of the gas. A primary advantage of these methods is that their accuracy is determined by the number of emitted photon packets, making it easily tunable. Inherent to the nature of the method is statistical noise, with a signal to noise ratio that scales as $\mathrm{SNR} \propto \sqrt{n}$, where n is the number of photon packets.

In Peter et al. [2023], we introduced the Sweep method, which is a very efficient way of computing the exact solution to the scattering-less radiative transfer equation in parallel on large inputs based on the concept of transport sweeps [Koch et al., 1991, Zeyao and Lianxiang, 2004]. Transport sweeps are a subclass of discrete ordinate methods, in which the radiative transfer equation is solved simply by discretizing it in all of the available variables (time, space, frequency and angle). During a sweep, scattering is assumed to be negligible, such that the radiative transfer equations for different angles decouple. In order to solve the resulting equations efficiently, the algorithm computes an ad-hoc topological sorting of the grid with respect to the direction of the sweep, resulting in a method that computes the exact solution to the equation in a single pass through the grid. We have found the resulting method to be very performant and accurate at the same time.

Our previous implementation of Sweep within the cosmological simulation code Arepo [Springel, 2010b] worked well on medium sized inputs, but a major problem was the fact that it could only perform global operations in which radiative transfer is performed on the entire box, before a global chemistry update is performed. This limitation makes large runs prohibitively expensive since the need for a small timestep in one of the cells of the entire simulation will imply a small global timestep everywhere.

In this paper, we introduce the standalone radiative transfer postprocessing code Subsweep, which deals with this problem by introducing a substepping procedure, similar to the sub-timestepping in modern hydrodynamical codes. This method works by assigning grid cells individual timesteps, which are chosen from a power-of-two hierarchy and adapted to the local, physical timescales of processes relevant to radiative transfer. Transport sweeps are then performed according to this timestep assignment in a physically consistent way, resulting in a computation in which cells with very

low desired timesteps can be evolved accurately without sacrificing performance by strictly adhering to a global, low timestep.

We also find that this new method drastically alleviates the computational cost of incorporating periodic boundary conditions, one of the main challenges for the initial version of the Sweep algorithm. Previously, periodic boundary conditions were implemented by source iteration - photon fluxes leaving the simulation box are introduced as a source term for a subsequent sweep, until convergence is reached. In the new substepping approach, we use the concept of Warmstarting - periodic source term of the previous iteration are used as a guess for the the new timestep. By applying this concept also to the sub-timestep sweeps, we find that sufficient accuracy for our applications is reached without performing any additional source iterations.

This paper is structured as follows. In Section 4.2, we discuss the implementation details of Subsweep, with a particular focus on the spatial domain decomposition (Section 4.2.2), the construction of the Voronoi grid (Section 4.2.3) before we focus on radiative transfer in general and the Sweep method in particular (Section 4.2.4) before we introduce the substepping approach (Section 4.2.5) and end with the details of our radiation chemistry solver (Section 4.2.9). Afterwards, we perform a number of tests of our code (Section 4.3), showing the physical accuracy of the results in an R-Type expansion test in the normal case (Section 4.3.1) and the case of the expansion happening across a periodic boundary (Section 4.3.2) and perform a test to study the shadowing behavior of the code (Section 4.3.3). We assess the performance of the substepping method by performing a one-dimensional R-Type expansion on a large span of parameters (Section 4.3.4) and perform a brief series of tests for the radiation chemistry solver (Section 4.2.9). Finally, we conclude this paper and discuss future extensions of the code as well as possible applications in Section 4.4.

4.2 Methods

4.2.1 General structure of the code

In this paper we discuss the Subsweep simulation code[1] which is a standalone code for postprocessing of large cosmological simulations. Currently, the code works with outputs of Arepo implementations, but extensions for output formats of other simulation codes are possible. The code requires data specifying the coordinates, temperatures and chemical compositions of a number of cells. Source terms can either be explicitly specified by the user or will be computed from a set of source cells (such as star

[1]Source code publicly available at `https://github.com/tehforsch/subsweep`

particles in the case of Arepo) which also needs to be present in the inputs. It will then distribute the data onto the desired number of cores (which we briefly discuss in 4.2.2), construct a the Voronoi grid (discussed in 4.2.3) and solve the radiative transfer equation coupled to radiation chemistry and write out the intermediate and final results of the computation. Documentation for the usage of the code is available alongside the source code. The code is written with a particular focus on the post-processing of high-redshift cosmological simulations and reionization, but extensions incorporating present-day chemistry into the code should be straightforward.

4.2.2 Domain decomposition

In order to run our code in parallel, we have to distribute the available data over multiple cores. We choose to use a simplified version of a standard Peano-Hilbert space-filling curve approach to spatial domain decompositioning which we will briefly describe in the following.

The first goal that the domain decomposition tries to solve is to distribute the particles onto the n cores $1 \dots n$ in such a way that the total runtime of the program is minimized. Since this is a very difficult optimization problem to solve in general, we make it more concrete by defining two primary goals of the domain decomposition. The first goal is the minimization of the load imbalance $\frac{\max\{L_i\}-\min\{L_i\}}{\max\{L_i\}}$ where the load L_i on core i can be defined in a variety of ways, which we will discuss later.

The second goal is to keep the total time spent communicating as low as possible. This requirement is almost equivalent to minimizing the surface area of the intersection between the domains, because shared interfaces are where communication needs to take place in order to solve them.

A third priority that is specific to transport sweep algorithms is that even if goals 1 and 2 are fulfilled optimally, the resulting sweep can still be slow if the cells are arranged in such a way that not all cores can work simultaneously due to the task dependencies that need to be fulfilled (see the discussion in 3.2.5).

For structured grids, a domain decomposition that optimizes the parallel performance of the transport sweep is given by the Koch-Baker-Alcouffe algorithm [Baker and Koch, 1998, Koch et al., 1991]. For unstructured grids, optimizing the performance by the domain decomposition is difficult in general, which has been discussed in detail [Adams et al., 2019, Vermaak et al., 2020].

Here, we will briefly discuss our implementation of a well-known, approach based on space-filling curves that can solve requirements 1 and 2 simultaneously. For now, we find that even though we do not optimize explicitly for the third goal, i.e. the sweep scheduling, the resulting performance is enough for our purposes.

In our case, a space-filling curve is given by a mapping f and its inverse f^{-1} between a one-dimensional interval and all the possible floating point positions in the three-dimensional simulation box with side length L

$$f : [c_{\min}, c_{\max}] \to [0, L]^3, \tag{4.1}$$

$$f^{-1} : [0, L]^3 \to [c_{\min}, c_{\max}], \tag{4.2}$$

where $c_{\min}$ and $c_{\max}$ are the minimum and maximum values of the domain of the space filling curve respectively. We call $f(\boldsymbol{r})$ the key of a particle at position $\boldsymbol{r}$. The basic idea of a domain decomposition using such a space filling curve is to move the three-dimensional optimization problem of distributing a set of points $\{p_j \in [0, L]^3\}$ onto n cores $1 \dots n$ to a more tractable, one-dimensional problem. In our case, this one-dimensional problem is the problem of finding cut-offs s_i for $i = 1 \dots n-1$ so that the load balance is minimized if each core i gets assigned the points $\{p_j \mid s_{i-1} < f^{-1}(p_j) < s_i\}$, where we take $s_0 = c_{\min}$ and $s_n = c_{\max}$. If the space-filling curve is chosen such that it maps close-by points on the interval $[c_{\min}, c_{\max}]$ to close-by points in three dimensional space, the resulting distribution of points will form reasonably compact domains. A common choice for such a curve is the Hilbert curve.

In order to execute the domain decomposition using the Hilbert curve, we require the load function $L_i(c_1, c_2)$ which computes the total load of the particles on core i between the keys c_1 and c_2. Here, we assume that the load can be computed as a sum over the load for each particle.

1: **procedure** FIND s_i
2: Initial guess: $s_i \leftarrow s_{i-1} + \frac{c_{\max} - s_{i-1}}{n-i}$
3: **for** $d \leftarrow 1, d_{\max}$ **do**
4: For each rank k, compute $L_k(s_{i-1}, s_i)$.
5: Compute $L \leftarrow \sum_{k=1}^{n} L_k(s_{i-1}, s_i)$ via a global sum.
6: **if** $L = L_{\text{local}}$ **then return** s_i
7: **else if** $L < L_{\text{desired}}$ **then**
8: $s_i \leftarrow \frac{s_i + c_{\max}}{2}$
9: **else if** $L > L_{\text{local}}$ **then**
10: $s_i \leftarrow \frac{s_i + s_{i-1}}{2}$
return s_i

Algorithm 4: Cut-off search

In order to find the distributions of the keys s_i we proceed as follows:

1. For each core, compute the keys for all local particles and sort them, so that

computing the load function $L_i(c_1, c_2)$ becomes a cheap operation for any keys c_1, c_2.

2. Compute the total load of the entire simulation $L_{\text{total}} = \sum_{i=1}^{n} L_i(c_{\min}, c_{\max})$ (via a global sum).

3. Compute the desired load on each core as $L_{\text{desired}} = \frac{L_{\text{total}}}{n}$.

4. Using this, compute the cutoffs s_i, starting with s_1 using the parallel search described in Alg. 4.

4.2.3 Construction of the Voronoi Grid

In order to perform the Sweep algorithm over a set of points, we need to construct a Mesh, so that we can determine the connectivity of cells. In order to avoid any additional numerical artifacts, we decide to use a similar mesh as the one that was used in the code which generated the outputs which we are trying to post-process using Subsweep. Since we are mostly interested in postprocessing simulation outputs of Arepo, we choose to use a Voronoi grid, which is the mesh that Arepo is based on.

There are many different algorithms for constructing Voronoi grids. For simplicity, the one here is based closely on the method described in Springel [2010b]. The method is based on incremental insertion [Bowyer, 1981, Watson, 1981], extended to allow construction of the grid for a point set distributed onto multiple cores.

Construction of the Local Delaunay triangulation

The Voronoi grid is constructed from its dual, the Delaunay triangulation. The serial incremental insertion algorithm for constructing the Delaunay triangulation proceeds as follows: Given a set of N mesh-generating points $\{p_i \mid 1 \leq i \leq N\}$, begin with an all-encompassing tetrahedron, i.e. one that is large enough to contain all points p_i. Now, for every point p, locate the tetrahedron in the triangulation which contains p. How exactly this is done in a performant way is described in Section 4.2.3. Using p, we split the tetrahedron containing p into 4 new tetrahedra. After the split, the resulting triangulation is not necessary Delaunay. In order to restore Delaunayhood, we begin by putting each of the 4 newly formed tetrahedra on a stack. For each tetrahedron t in the stack, we find the face F which is opposite of p in t. We then find the tetrahedron t' which is on the other side of F, and locate the point p' which is opposite of F within t'. If p' is contained in the circumcircle around t, then the face F violates the Delaunay criterion and needs to be removed. To do so we perform a

flip orientation on the two tetrahedra t and t' which will result in a number of new tetrahedra, each of which will now have to be checked for Delaunayhood, so we put them on the stack as well. Once the stack is empty, Delaunayhood has been restored again and we can begin inserting the next point.

The flip operation between two tetrahedra t and t', their shared face F and the two points p and p' opposite of F in each of the tetrahedra respectively works as follows: Compute the intersection point q of the face F with the line between p and p'. If q lies inside F, we perform a 2-to-3-flip, in which the two tetrahedra are replaced by three. If the intersection point lies outside one of the edges of F, we take into account the neighboring tetrahedron along that edge and perform the opposite operation - a 3-to-2 flip - in which the three tetrahedra are converted to two. If the intersection point lies outside two edges, the flip can be skipped. It can be shown [Edelsbrunner and Shah, 1996] that flipping the remaining violating edges will restore Delaunayhood. For more information on this procedure see Springel [2010b].

Point location

While inserting a point p into the triangulation, we need to locate the tetrahedron containing p. This is performed by the simple "jump and walk" method. The method works by using a priority queue q. We initialize q as containing only the last tetrahedron that was inserted into the triangulation. Now we iteratively take the highest priority tetrahedron t out of the queue. If t contains p, return t. If t does not contain p, we add all the neighboring tetrahedra of t to q with their priority determined by their distance to p (so that tetrahedra closest to p are searched first). The method performs the best if the order of the points inserted into the triangulation is such that two points inserted after another are also at similar positions (which in turn makes the initial guess better). In order to achieve this, we begin the construction by sorting all points according to their Peano-Hilbert key.

Parallel Delaunay construction

In principle, we would like to construct the global Delaunay triangulation T_{global} on all of the points in the entire simulation. In practice, we are limited to those points that are available on each core. All we can do is to construct a local triangulation T_{local} over all of the local points. The goal of the triangulation is to provide connection information and in order for it to be useful, this connection information has to be consistent with what the other cores see. It is clear that in order to do so and preserve Delaunayhood we need to import points that lie on other cores, which we call halo points. More precisely, we we want to construct T_{local} in such a way that it is

consistent with T_{global}, in the sense that for every local point p, the set of tetrahedra $\{t|t \in T_{\text{local}}, p \in t\}$ is the same as $\{t|t \in T_{\text{global}}, p \in t\}$. Note that this requirement does not extend to halo points, allowing us to stop importing additional halo points once all local points are consistent in the above sense.

Here, we will describe the algorithm for the halo search. The goal here is to import every necessary halo point in order to reach a consistent local triangulation, while importing as few as possible superfluous points in order to speed up the grid generation and keep memory overhead as low as possible. The basic idea is that a tetrahedron t is consistent with the global triangulation T_{global} iff we have imported the set of all points $\{p|p \in C(t)\}$ from all other cores, where $C(t)$ is the circumcircle of the tetrahedron. To do so, we begin by constructing the Delaunay triangulation of all local points T^0_{local}. Initially, we flag every tetrahedron in the triangulation as "undecided" and then iterate on the following process:

For every undecided tetrahedron t, we compute the circumcircle $C(t) = (c, r)$ with center c and radius r. Given the circumcircle $C(t)$, we compute the search radius r'. Search other cores for all points p that are within r' distance of c which we have not imported locally yet. If there is no such point anywhere (which means we have imported all points that could be contained in the circumcircle of the tetrahedron), flag the tetrahedron as "decided". Otherwise, we add all points p to the list of newly imported points. Now, we construct T^{i+1}_{local} by inserting the set of all newly imported points into T^i_{local}. We flag any newly created tetrahedron which contains a local point as undecided.

Here, the search radius r' is computed as follows: If the radius of the circumcircle r is smaller than the average expected size of a Voronoi tetrahedron $\bar{l}$, then $r' = r$. Otherwise, we use $r' = \bar{l}$, unless we have previously performed a radius search for this tetrahedron before, in which case we use $r' = r'_{\text{previous}} * \alpha$ where $\alpha > 1$ is a free parameter. This is done because in the first few iterations of the triangulation, very large tetrahedra tend to form because we are not yet aware of the presence of very nearby points on other cores. If we blindly performed a radius search with the radius of the circumcircle r, we might unnecessarily import a large number of points from other cores. However, if the triangulation should contain this large tetrahedron, the exponentially increasing search radius will ensure that we perform a search with the proper radius within a reasonable number of iterations.

If periodic boundaries are desired, the same procedure described above, which imports halo points from other ranks can also import periodic haloes (i.e. points that represent a point shifted by a multiple of the box size along one or more axes) both from other cores and from the set of local points. All that is required is that

the radius search which searches for all points within a given radius of a given point takes periodic boundary conditions into account. Since constructing the distributed triangulation requires many radius searches, we need to perform the radius search quickly. To do so, we construct a standard Oct-tree on the set of all local points which reduces point search to a $\mathcal{O}(n \log n)$ operation.

Degeneracies

Another difficulty in creating Delaunay triangulation is how to deal with degenerate cases and those that are close to being degenerate. One solution to this problem is to perform all operations in arbitrary precision arithmetic. However, this will reduce the performance of the code drastically. In Subsweep we take an approach similar to the one in Springel [2010b] where we perform the critical checks (such as the checks that ask whether a point is contained in a tetrahedron or whether a tetrahedron is positively oriented) in floating point arithmetic first. If the result of the floating point operation is at risk of being qualitatively wrong due to numerical round-off errors, we perform it again in arbitrary precision arithmetic. In the current code, we do not deal with truly degenerate cases (for example, a point lying exactly on a face of a tetrahedron) because we find them to be extremely rare in practice, however it is possible to extend the method to account for degeneracies. For more information on this procedure, we refer to Springel [2010b].

4.2.4 The Sweep algorithm

We introduced the Sweep algorithm in the context of Astrophysics in 3.2.5. Here, we will briefly recap the basics of the algorithm in order to explain the required fundamentals for understanding the extensions we will introduce in later sections.

Given the specific radiative intensity $I_\nu(\boldsymbol{r}, t, \hat{\Omega})$, with frequency ν, spatial position $\boldsymbol{r}$, time t and solid angle $\hat{\Omega}$ given in units of $\mathrm{W\,m^{-2}\,sr^{-1}\,Hz^{-1}}$, the general Radiative Transfer Equation(RTE) reads [Rybicki and Lightman, 1985]

$$\frac{1}{c}\frac{\partial}{\partial t} I_\nu + \hat{\boldsymbol{\Omega}} \cdot \boldsymbol{\nabla} I_\nu = j_\nu - (\bar{k}_{\nu,\mathrm{s}} + \bar{k}_{\nu,\mathrm{a}}) I_\nu + \frac{1}{4\pi} \int_S k_{\nu,\mathrm{s}}(\boldsymbol{\Omega'}) I_\nu \mathbf{d}\boldsymbol{\Omega'}. \tag{4.3}$$

In the case of Subsweep, we assume that scattering terms are negligible and make the infinite speed of light assumption, so that we obtain

$$\hat{\boldsymbol{\Omega}} \cdot \boldsymbol{\nabla} I_\nu = j_\nu - \bar{k}_{\nu,\mathrm{a}} I_\nu. \tag{4.4}$$

For more details on these assumptions see Peter et al. [2023].

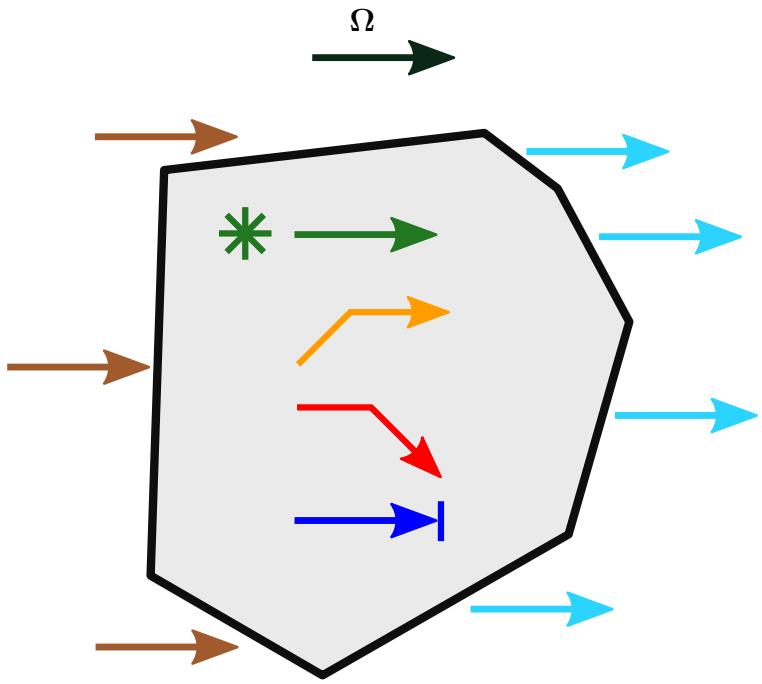

Figure 4.1. Illustration of the radiative processes described by Eq. 4.4 for a single grid cell: Incoming (orange) and outgoing (red) radiation, sources (green), absorption (blue)

The Sweep method is a discrete ordinate method, which means that it solves the RTE by discretizing it in every variable, that is in space, time, angle and frequency.

Equation 4.4 can be intuitively understood using the illustration in Fig. 4.1, which shows the processes affecting a small volume of space. Sources of radiation in this volume are through incoming radiation from cells to the left (brown arrows), the source term j directly (green arrow). Radiation from the cell is either absorbed (blue arrow) or leaves the cell towards the right (teal arrows). The neighboring cells therefore fall into two categories. Cells upwind of the cell along $\mathbf{\Omega}$ (brown arrows) need to have their solution computed before this cell, since we require the incoming fluxes from those cells to solve the local problem. Cells downwind of the cell require the outgoing fluxes of the local solution in order to be solved.

The crucial idea of the Sweep method is that it finds a topological sorting of the partial order induced by the upwind-downwind relation, such that the exact solution to the (scattering-less) RTE can be obtained in n_{dir} passes through the grid, where n_{dir} is the number of bins into which we choose discretize the angular directions. It is crucial that the upwind-downwind relation is transitive, so that it is a partial order (which is equivalent to there being no cycles in the directed graph induced by the order), because otherwise a topological sorting of the cells does not exist. We have shown in A.1 that this is always true for such an ordering induced by a Voronoi grid,

which is the only type of grid that we are going to work with in this paper. It should be noted that the ordering is trivially acyclic for Euclidean grids or grids generated by adaptive mesh refinement, so that the Sweep algorithm can also be used in another type of grid that is widely used in Astrophysics.

1: initialize task queue $q \leftarrow \{\}$
2: **for** all $\mathbf{\Omega}$ and all cells c in grid **do**
3: count number of required upwind fluxes $n(c, \mathbf{\Omega}) \leftarrow u(t)$
4: **if** $n(c, \mathbf{\Omega}) = 0$ **then** add task $(c, \mathbf{\Omega})$ to q
5: **while** q not empty **do**
6: get first task $t = (c, \mathbf{\Omega})$ from q
7: solve t using upwind fluxes
8: **for** downwind neighbor c_d in $d(t)$ **do**
9: reduce missing upwind flux count $n(c_\mathrm{d}, \mathbf{\Omega})$ by 1.
10: **if** $n(c_\mathrm{d}, \mathbf{\Omega}) = 0$ **then** add task $(c_\mathrm{d}, \mathbf{\Omega})$ to q.

Algorithm 5: Single-core sweep

The single-core sweep algorithm is described in Algorithm 5. In order to find the topological sorting, the Sweep algorithm starts the computation by computing an upwind count for each direction and each cell which is simply the number of cells upwind of the cell in the given direction. The idea is to keep track of the set of all (cell, direction) pairs which can currently be solved, which are those whose upwind neighbors have already been solved. Whenever we solve a task, we reduce the upwind count of all its downwind dependencies by 1. If the upwind count of this dependency is now zero, we put this dependency into the task queue. Once the queue is empty, we have solved all tasks and have obtained the solution to the RTE. The grid being acyclic guarantees that this algorithm always terminates.

In order to perform the algorithm described in Alg.5 in parallel on many cores with a spatial domain decomposition, a number of modifications need to be made to the algorithm. The basic idea of algorithm does not change in the parallelized version of the code. The main difference is that we need to take task dependencies between different cores into account. Previously, having solved a task meant that we could simply reduce the upwind count of all its downwind dependencies by one. Now, the downwind dependencies of a task might be on a different core. In this case, we send a message to that core consisting of the outgoing fluxes of the local cell, the id of the downwind cell and the direction of the task. As soon as the other core receives that message, it will reduce the upwind count for the corresponding cell

initialize task queue $q \leftarrow \{\}$
initialize send queues for each processor i holding downwind neighbors of any of the cells in the domain of the current processor: $s_i \leftarrow \{\}$
for all $\mathbf{\Omega}$ and all cells c in grid **do**
    count number of required upwind fluxes $n(c, \mathbf{\Omega}) \leftarrow u(t)$
    if $n(c, \mathbf{\Omega}) = 0$ **then** add task $(c, \mathbf{\Omega})$ to q
while any cell unsolved or any s_i not empty **do**
    for each incoming message (flux f along $\mathbf{\Omega}$ into cell c) **do**
        reduce missing upwind flux count $n(c, \mathbf{\Omega})$ by 1.
        if $n(c, \mathbf{\Omega}) = 0$ **then** add task $(c, \mathbf{\Omega})$ to q.
    $n_{\text{solved}} = 0$
    while q nonempty **and** $n_{\text{solved}} < N_{\max}$ **do**
        get first task $t = (c, \mathbf{\Omega})$ from q
        solve t using upwind fluxes
        n_{solved} += 1
        for downwind neighbor c_{d} in $d(t)$ **do**
            if c_{d} is remote cell on processor i **then**
                add flux to send queue s_i
            else
                reduce missing upwind flux count $n(c_{\text{d}}, \mathbf{\Omega})$ by 1.
                if $n(c_{\text{d}}, \mathbf{\Omega}) = 0$ **then** add task $(c_{\text{d}}, \mathbf{\Omega})$ to q.
    send all messages in s_i

Algorithm 6: Parallel sweep

by one (and add it to the solve queue if the upwind count is 0 at this point). In order to improve performance, messages are not sent immediately, since sending lots of small messages tends to increase communication overhead and reduce performance as a result. The opposite strategy of sending messages only after all tasks that are solvable locally have been solved also comes with performance drawbacks, since it can cause long waiting times on neighboring cores who cannot perform any work before receiving new incoming fluxes. In practice, we therefore use an intermediate approach where we solve at most $N_{\max}$ local tasks before new messages are sent and received. Here, $N_{\max}$ is a free parameter and the two extremes are recovered for $N_{\max} = 1$ and $N_{\max} = \infty$ respectively. In order for this parallel algorithm to terminate, it is crucial that all cores agree on the connections between their local cells. If this is not the case, cores can end up waiting for incoming messages that will never be sent, causing infinite deadlocks. The property that all cores agree on the interfaces between their boundary cells is ensured by the particular way in which grid construction is performed by the algorithm described in Section 4.2.3.

4.2.5 Substepping

The main problem with the Sweep algorithm above is that the entire grid operates on the same timestep. This is not a problem for the RTE alone, since, given our assumption of infinite speed of light, it is independent of time and therefore reaches a steady-state solution immediately. However, we are interested in solutions of the RTE coupled to the radiation chemistry equations, which are manifestly time-dependent.

In practice, a large fraction of the cells in the simulation are either fully ionized or fully neutral and have settled into an equilibrium where their chemistry update could be performed at high timesteps. However, cells along ionization fronts require comparatively low timesteps in order to accurately integrate both the RTE and the chemical equations. With the previous sweep algorithm, the only option was to use a low value for the global timestep, which in turn means that solving the system for the desired amount of time requires a higher number of sweeps.

In the following, we will introduce a modification to the Sweep algorithm which allows cells to perform sub-timesteps, if required. Effectively, this solves the global timestep problem by letting cells choose their desired timestep. In the following, we will explain how this algorithm works in practice. An illustration of the algorithm is shown in Figure 4.2.

In order to do so, we introduce n different timestep levels. Each cell c is assigned to a timestep level $l(c)$. During a full timestep $\Delta t_{\max}$, a cell at timestep level 0 receives one update for a full timestep $\Delta t_{\max}$. cells at level l receive 2^l updates with

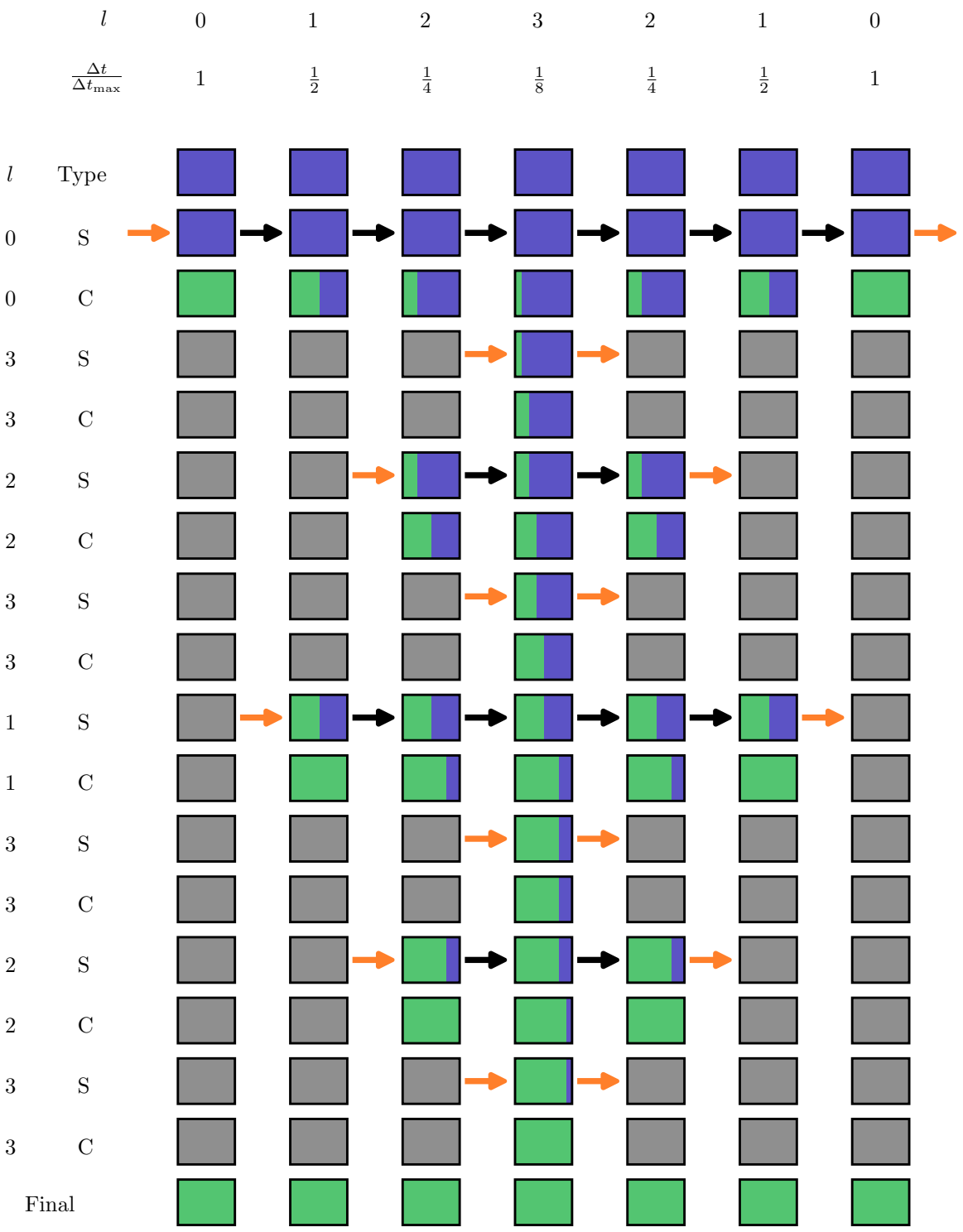

Figure 4.2. Illustration of the Sweep substepping procedure. The rectangles represent cells, with the color of the cell indicating how far the cell has been integrated. A fully blue cell has not been integrated at all, while a green cell has been integrated for a total of $\Delta t_{\max}$. Arrows represent fluxes going into the cell which are computed during the sweep steps. Black arrows denote normal fluxes, while orange arrows represent boundary terms for the sweep. Each row represents either a sweep (denoted by S) or a chemistry update (denoted by C) at the corresponding level l. The last row represents the final state with each cell having been fully integrated.

a timestep of $2^{-l}\Delta t_{\max}$. At the end of each full timestep, each cell computes the timescale at which its relevant quantities change. In our case, this timescale is given by the minimum $t_{\min}$ of the three timescales at which the temperature T, the ionized hydrogen fraction $x_{\mathrm{H\,II}}$ and the photon flux F change:

$$t_{\min} = \min\left[\left|\frac{T_i + T_{i-1}}{T_i - T_{i-1}}\right|, \left|\frac{x_{\mathrm{HII,i}} + x_{\mathrm{HII,i-1}}}{x_{\mathrm{HII,i}} - x_{\mathrm{HII,i-1}}}\right|, \left|\frac{F_i + F_{i-1}}{F_i - F_{i-1}}\right|, \right] \tag{4.5}$$

where $i-1$ and i refer to the values in the previous partial sweep and the current one respectively. This timescale is used to compute a desired timestep

$$\Delta t_{\mathrm{desired}} = x t_{\min} \tag{4.6}$$

where $x \in (0, 1]$ is a dimensionless free parameter which controls the accuracy of the integration. This desired timestep is then used to determine the timestep level l' of the cell for the next full timestep as

$$l' = \left\lceil \log_2 \frac{\Delta t_{\max}}{\Delta t_{\mathrm{desired}}} \right\rceil \tag{4.7}$$

for the entire next full timestep. In order to keep a fixed number of levels n, the value of l' will be reduced to at most $n-1$ and increased to at least 0. Modifications of this method where cells can change their timestep level in the middle of a full timestep are possible, but for reasons of simplicity, we have not implemented them at this point.

Given the distribution of the cells onto the n timestep levels $0 \ldots n-1$, we introduce the following terminology: A "partial sweep" at level l, or l-sweep is a sweep of all the cells which are at level l or higher. During a l-sweep, we call cells "active" if their timestep level l' fulfills $l' >= l$, i.e. if they are involved in the l-sweep. A "full sweep" is the procedure by which the system is integrated for a full timestep $\Delta t_{\max}$ and consists of 1 0-sweep, 1 1-sweeps, 2 2-sweeps, 4 3-sweeps, ..., and 2^{n-2} $(n-1)$-sweeps. The order in which they are performed is illustrated in Fig. 4.2.

Since only the cells at levels larger or equal than l participate in an l-sweep, we need to decide how to treat the incoming fluxes from cells which are at levels below l. In this method, the fluxes of all cells at timestep levels $0 \ldots l-1$ are kept constant and effectively treated as boundary conditions for the cells at higher levels. During the l-sweep, the fluxes going into each active cell c are corrected by computing

$$F'_{\mathrm{out}} - F_{\mathrm{out}} = f(F'_{\mathrm{in}}, c) - f(F_{\mathrm{in}}, c) \tag{4.8}$$

where F_{out} is the value of the flux before the partial sweep, F_{in} is the incoming flux term before the partial sweep, F'_{in} is the incoming flux in this partial sweep and f is

the function that computes the outgoing fluxes given the incoming fluxes and crucially depends on the chemical composition which might change during a chemistry update. This function depends on the implementation of the chemistry and its exact form in our hydrogen-only chemistry implementation will be discussed in Section 4.2.9. Outgoing fluxes of a cell are either used directly as input into other local cells, or communicated to other cores, as in the original sweep algorithm without substepping. Flux corrections are applied to cells whether or not the target cell itself is active.

Once the l-sweep is finished, all cells c have their chemistry updated by $\Delta t_{\max} 2^{-l(c)}$. This means that at this moment, cells at higher levels (and therefore lower timesteps) have experienced "less" time, than those at lower levels. This will be corrected by performing additional partial sweeps on the higher levels, so that at the end of a full sweep, each particle has been integrated for exactly $\Delta t_{\max}$. It should also be noted that consistency is guaranteed in the sense that the order in which the partial sweeps are performed guarantees that for any given partial sweep, all active cells have experienced the same amount of time. After every full sweep, the cells are moved onto their new timestep level, according to their desired timestep (see Eq. 4.7). Crucially, after the timestep levels have been updated, each core communicates the new timestep level of each of its cells which has a neighbor on another core to that core. This is important because all of the cores have to agree on which cells are active at each level. If they do not agree on this, one of the cores will expect incoming fluxes over the interface shared by the two cells while the other will not send those fluxes, resulting in a deadlock of the partial sweep.

4.2.6 Wind up

At the beginning of the simulation, we do not know how to distribute the cells onto the timestep levels, since we have no prior data on the timescales at which their relevant quantities will change. If we had to guess the timestep level of any cell, the only reasonable choice we can make in order to not violate the timestep criterion of any single cell, is to place all cells in the highest level (the lowest timestep). However, performing a full sweep in this setup would require a total of $2^{n+1} - 1$ sweeps of the entire grid, an extremely expensive operation. In order to avoid this, we compute the timescales of each cell by placing each cell in level $n - 1$ and beginning with a $(n - 1)$-sweep, i.e. a sweep at the smallest allowed timestep ($2^{-(n-1)} \Delta t_{\max}$). We then allow each cell to move one level down, if its desired timestep is large enough, and perform a $n - 2$ sweep, and so on. In the end, we have performed n partial sweeps and have simulated a total time of $\sum_{i=0}^{n} 2^{-i} \Delta t_{\max} = \left(1 - 2^{-(n-1)}\right) \Delta t_{\max}$. In order to align the time intervals with multiples of $\Delta t_{\max}$, we perform one more partial sweep

of all cells with timestep $2^{-(n-1)}\Delta t_{\max}$ which will bring the total simulated time to $\Delta t_{\max}$. From now on, every timestep will be performed by a full sweep, which totals $\Delta t_{\max}$.

4.2.7 Periodic Boundary Conditions

Periodic boundary conditions are an important tool to study cosmological volumes of space self-consistently, by allowing effects from the matter outside of the simulation box to be approximately modeled by the contents of the simulation box itself. In the case of radiative transfer, this means re-introducing photons that escape the box on one side to the mirrored position on the opposite side.

In 4.2.3, we discussed how periodic boundaries are taken into consideration during mesh construction. This means that each cell at the boundary of the box knows the location of its periodic neighbors. As discussed in 3.2.7, there is no obvious, self-consistent way of re-introducing outgoing photons within a single sweep. However, we can make use of the source iteration algorithm and treat fluxes that leave the boundaries of the simulation box as source terms for the next iteration of the algorithm. Each iteration then approximates the true periodic source terms until convergence is reached. However, applying this approach to a full sweep has the obvious drawback that every iteration takes exactly as long as the original sweep. Since a full sweep over the grid is an expensive operation, repeating it a number of times in order to reach an acceptable level of convergence can quickly become infeasible.

In 3.2.7 we discussed the concept of Warmstarting, where the resulting fluxes from previous sweeps are re-introduced in the next iteration in order to speed up convergence. Moreover, Warmstarting integrates extremely well with the sub-timestepping approach introduced in Subsweep. To do so, we use the outgoing periodic fluxes of every partial sweep as incoming fluxes into the corresponding cells for the next partial sweep. This has a number of benefits. Primarily, it changes the algorithm so that it does not require a global cost (re-running the full sweep) in order to fix an often local problem (convergence of the periodic fluxes in the cells with the most activity). Instead, the algorithm naturally adapts itself to the local requirements and decreases the timestep in cells with particularly bad convergence behavior with respect to periodic boundary conditions. It should be noted that this happens without requiring any timestep criterion specific to periodic boundary conditions - cells that have not converged to their true periodic fluxes will automatically reduce the timestep, since that derives (among other things) from the rate of change in the flux terms, as shown in Eq. 4.5. This combination of Warmstarting and substepping has proven so effective that we have chosen not to implement any global iteration on levels of full sweeps in

Subsweep.

4.2.8 Rotations

As discussed for the original Sweep implementation in 3.2.6, we perform rotations of the directional bins between transport sweeps in order to smooth out the effect that the discretization of the directional bins has on the result. The preferential directions introduced by this discretization can easily lead to very apparent star-shaped artifacts in the hydrogen ionization fraction around strong sources.

In Subsweep, we keep this approach to smoothing out preferential directions. Here, remapping the flux corrections from one timestep to the next becomes important. As in the original implementation, the directions $\boldsymbol{\Omega}_i$ are rotated to new directions $\boldsymbol{\Omega}_i = \boldsymbol{R}(\theta, \phi) \cdot \boldsymbol{\Omega}'_i$ where $\boldsymbol{R}(\theta, \phi)$ is the rotation matrix for the spherical coordinate-angles θ and ϕ. The angles are chosen from a uniform distribution of $\theta \in [0, \pi]$, $\phi \in [0, 2\pi]$. Remapping of the fluxes onto the new angular bins is then done via $F(\boldsymbol{r}, \boldsymbol{\Omega}'_i) = \sum_{j=1}^{N_{\text{dir}}} \frac{\Delta S_{ij}}{\Delta S_i} F(\boldsymbol{r}, \boldsymbol{\Omega}')$ where N_{dir} is the number of directional bins, the interpolation coefficients ΔS_{ij} are given by the solid angle that $\boldsymbol{\Omega}_i$ and $\boldsymbol{\Omega}_j$ share and ΔS_i is the solid angle corresponding to any direction Ω_i.

These rotations are performed only after every full Sweep and not after partial Sweeps. It is possible in principle to rotate the bins also after every partial Sweep, but doing so can have a very strong, discontinuous effect on the convergence timescale of some cells. In order to safely incorporate sub-timestep rotations into the substepping approach, we think it is necessary to introduce the ability for cells to change their desired timestep during partial sweeps, not only during full sweeps. Therefore, we have chosen not to introduce this additional complexity to the algorithm.

The drawback of this choice is that if the full-sweep timestep Δt_{max} is chosen to be large compared to the timescales at which ionization fronts move a large amount of cells (which is desirable in order to fully take advantage of the substepping approach), artifacts due to preferential directions can be visible. In order to avoid these artifacts, the full-sweep timestep has to be decreased, increasing computation time.

4.2.9 Radiation Chemistry

The implementation of the radiation chemistry in our code follows Rosdahl et al. [2013] closely. In the current form, the code only treats the ionization, heating and cooling of hydrogen in a primordial gas. We assume zero helium in our code. However, extensions to incorporate helium or more complex chemistry networks are possible and intended in the structure of the code. This includes adding more frequency bins for the

radiative transfer. In the current form, we use one frequency bin which incorporates all frequencies $f \geq 13.6\,\mathrm{eV}$.

The chemical state of a cell c is described by the state vector: $U = (T, x_{\mathrm{H\,II}})$ alongside its (constant) density ρ. The first thing that the implementation of the chemistry needs to provide is the function $f(F, c)$ discussed in Section 4.2.5. This function computes the outgoing photon flux of a cell c given the incoming flux F which depends on the chemical state U of the cell. For our hydrogen-only chemistry, this function is given by

$$f(F, c) = Fe^{-n_{\mathrm{H\,I}}\sigma d}, \tag{4.9}$$

where $n_{\mathrm{H\,I}}$ is the density of neutral hydrogen, σ is the number weighted average cross section for photons with $13.6\,\mathrm{eV} < \nu < \infty$ and $d = \sqrt[3]{\frac{3V}{4\pi}}$ is the approximate size of the cell (V is the volume of the cell). In principle we could be more consistent in our choice of cell size by computing the effective length of the cell along the given direction Ω of the sweep, but did not do so in order to keep this as simple as possible.

1: **procedure** UPDATE(ΔT)
2: Remember initial state $U_{\mathrm{init}} \leftarrow (T, x_{\mathrm{H\,II}})$
3: Compute $T' \leftarrow$ TEMPERATUREUPDATE(Δt).
4: **if** $\left|\frac{T'-T}{T}\right| > \epsilon$ **then**
5: $U \leftarrow U_{\mathrm{init}}$
6: UPDATE($\Delta T/2$)
7: UPDATE($\Delta T/2$)
8: **return**
9: **else**
10: $T \leftarrow T'$
11: Compute $x'_{\mathrm{H\,II}} \leftarrow$ IONIZATIONFRACTIONUPDATE(Δt).
12: **if** $\left|\frac{x'_{\mathrm{H\,II}}-x_{\mathrm{H\,II}}}{x_{\mathrm{H\,II}}}\right| > \epsilon$ **then**
13: $U \leftarrow U_{\mathrm{init}}$
14: UPDATE($\Delta T/2$)
15: UPDATE($\Delta T/2$)
16: **return**
17: **else**
18: $x_{\mathrm{H\,II}} \leftarrow x'_{\mathrm{H\,II}}$

Algorithm 7: Chemistry update

The basic chemistry update of a cell, given the incoming photon flux F of photons above 13.6 eV proceeds as in Algorithm 7.

Temperature update

The temperature update is performed by solving the equation

$$\frac{\partial T}{\partial t} = \frac{m_{\mathrm{p}}\mu\,(\gamma-1)}{\rho k_{\mathrm{B}}}\Lambda \tag{4.10}$$

where m_{p} is the mass of the proton, μ is the average mass of the particles in the gas in units of m_{p}, γ is the adiabatic index, k_{B} is the Boltzmann constant, ρ is the mass density of the gas and Λ is the total combined heating and cooling term. In our case, we assume that the gas consists only of hydrogen, so that $\mu = \frac{1}{1+x_{\mathrm{H\,II}}}$ where $x_{\mathrm{H\,II}}$ is the hydrogen ionization fraction.

Λ is given by a sum of the photo-heating term and the sum of all cooling processes

$$\Lambda = H_{\mathrm{photo}} + \left(\zeta(T) + \psi(T)\right) n_{\mathrm{e}} n_{\mathrm{H\,I}} + \left(\eta(T) + \Theta(T)\right) n_{\mathrm{e}} n_{\mathrm{H\,II}} + \bar{\omega}(T) n_{\mathrm{e}} \tag{4.11}$$

where H_{photo} describes photo-heating, n_{e}, $n_{\mathrm{H\,I}}$ and $n_{\mathrm{H\,II}}$ are the (number-)density of electrons, neutral hydrogen and ionized hydrogen respectively and the other terms describe cooling due to collisional ionization $\zeta(T)$, collisional excitation $\psi(T)$, recombination $\eta(T)$, Bremsstrahlung $\Theta(T)$ and Compton cooling $\bar{\omega}(T)$. We use the on-the-spot approximation in which we assume that every case-A recombination (that is, recombination to the ground state) will emit a photon which is immediately re-absorbed by the surrounding neutral atoms so that it results in no additional recombination. Therefore $\eta(T)$ denotes the cooling rate of case-B recombination only.

We solve Equation 4.10 by updating the temperature via a semi-implicit formulation given by

$$T^{t+\Delta t} = T^{t} + \frac{\mu\Lambda}{\frac{\rho k_{\mathrm{B}}}{(\gamma-1)m_{\mathrm{p}}\Delta t} - \Lambda'}. \tag{4.12}$$

Here, $\Lambda' = \frac{\partial \Lambda}{\partial T}$ is the derivative of the total heating rate with respect to temperature.

The full expression for all the heating and cooling terms is given in A.2.

The equation describing the evolution of $n_{\mathrm{H\,II}}$ is given by

$$\frac{\partial n_{\mathrm{H\,II}}}{\partial t} = n_{\mathrm{H\,I}}\left(\beta(T) n_{\mathrm{e}} + \Gamma\right) - \alpha(T) n_{\mathrm{H\,II}} n_{\mathrm{e}} \tag{4.13}$$

where $\beta(T)$ is the electron collisional ionization rate, $\alpha(T)$ is the case-B recombination rate and Γ is the photoionization rate, which is computed as $\sum_{i=1}^{n_{\mathrm{faces}}} \sum_{j=1}^{n_{\mathrm{dir}}} F_{i,j}$, where n_{faces} is the number of neighboring faces of the cell, n_{dir} is the number of discrete directions and $F_{i,j}$ is the incoming photon flux from a given neighbor in the given direction, where $F_{i,j} = 0$ if the neighbor is downwind in the given direction.

4.3 Tests

4.3.1 R-type expansion of a HII region

Here we study the R-type expansion of a HII-region. Strömgren [1939] showed that a point source in a medium consisting of hydrogen with uniform density will eventually create a spherical HII-region with radius $R_{\rm st} = \left(\frac{3\dot{N}}{4\pi\alpha_{\rm B}(T)n_{\rm e}^2}\right)$ where $\dot{N}$ is the rate of ionizing photons emitted from the source, $\alpha_{\rm B}$ is the case-B recombination coefficient as a function of temperature T and $n_{\rm e}$ is the electron density. If we assume that the gas inside the ionized region is fully ionized then $n_{\rm H} \approx n_{\rm e}$ and we find that the time evolution of the system is given by

$$R(t) = R_{St}\left(1 - e^{t/t_{\rm rec}}\right). \tag{4.14}$$

where the recombination time $t_{\rm rec}$ is given by $t_{\rm rec} = \frac{1}{n_{\rm H}\alpha_{\rm B}(T)}$. This test is set up in exactly the same way as in Jaura et al. [2018] and Baczynski et al. [2015], as well as the corresponding test in Section 3.3.1. We use a cubic simulation box with $L = 12.8\,{\rm kpc}$ filled with hydrogen with a homogeneous density of $n_{\rm H} = 1 \times 10^{-3}\,{\rm cm}^{-}3$ which is assumed to be fully neutral in the beginning. In the center of the box, we place a single source which emits ionizing photons (of energy $E > 13.6\,{\rm eV}$) at a rate of $\dot{N} = 1 \times 10^{49}\,1/{\rm s}$. During the test, we disable photo-heating and all cooling terms and keep the temperature at $T = 100\,{\rm K}$ everywhere and fix the case-B recombination coefficient to $\alpha_{\rm B} = 2.59 \times 10^{-13}\,{\rm cm}^3{\rm s}$. We perform this test for a number of different resolutions, $n = 32^3, 64^3, 128^3, 256^3$. The top panel of Figure 4.3 shows the numerical result for the radius of the ionized bubble as a function of time, compared to the analytical expression given by Equation 4.14. In the bottom panel, the relative error between the simulation result and the analytical prediction is shown. The radius of the ionized bubble is defined as the radius r at which a small spherical shell with radius r has an average ionization of $\overline{x_{\rm HII}} = 0.5$. For more details regarding the computation of this value, see Section 3.3.1.

The runs at all resolutions follow the analytical prediction closely with relative errors of below 2% for all values the initial phase of the expansion where $t < 0.1t_{\rm rec}$. It should be noted that the general trend is that error increases with increasing resolution, a result that we have also seen in the Sweep implementation of Arepo in Section 3.3.1. While this is counter-intuitive, we note that the analytical prediction assumes a perfectly sharp boundary, something that is clearly not the case in the numerical solution to the problem. This means that the value of the radius depends strongly on its definition, so that small deviations in the radius are not necessarily meaningful.

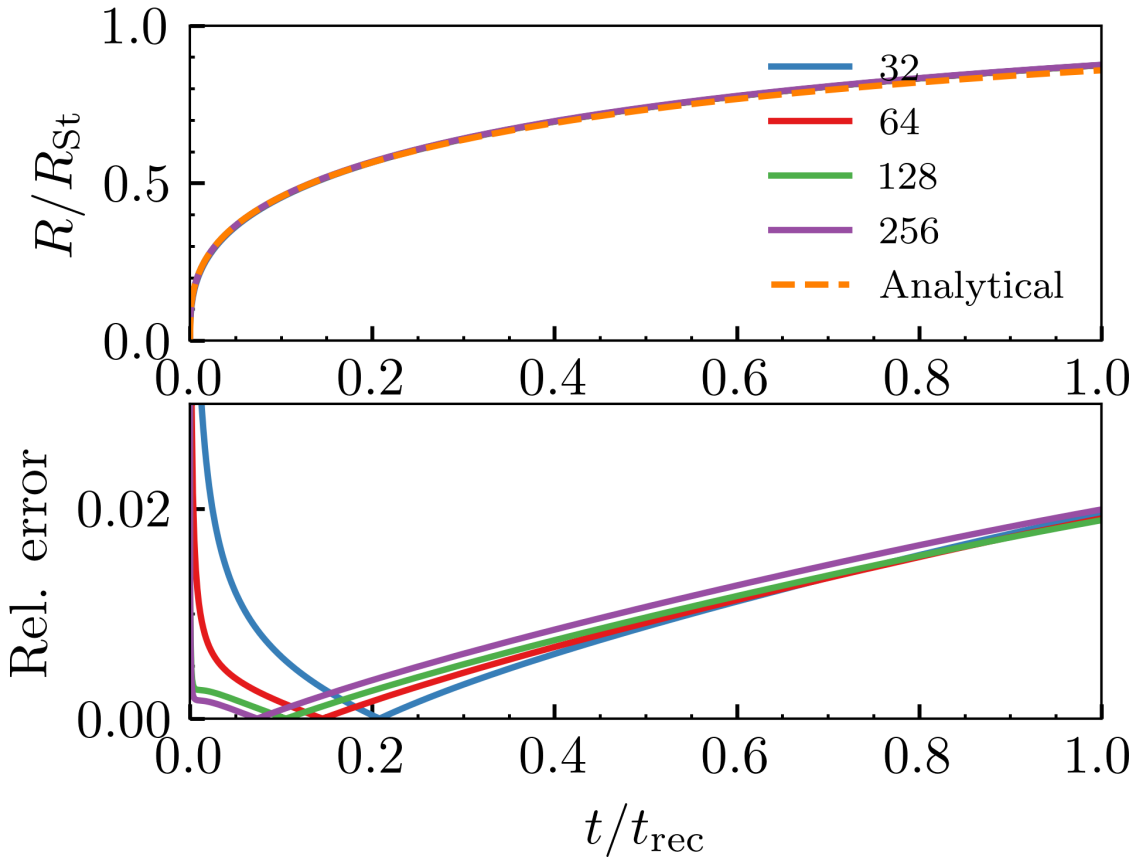

Figure 4.3. Top panel: The value of the radius of the ionized bubble at the center of the simulation box, normalized by the Strömgren radius $R_{\rm St}$ as a function of time normalized by the recombination time $t_{\rm rec}$. Different lines represent different resolutions 32^3 (blue), 64^3 (red), 128^3 (green), 256^3 (purple) with the orange, dashed line representing the analytical prediction given by Eq. 4.14. Bottom panel: The relative error $|R(t) - R_{\rm r}(t)|\,/R_{\rm r}(t)$ between the analytical prediction and the numerical results as a function of $t/t_{\rm rec}$.

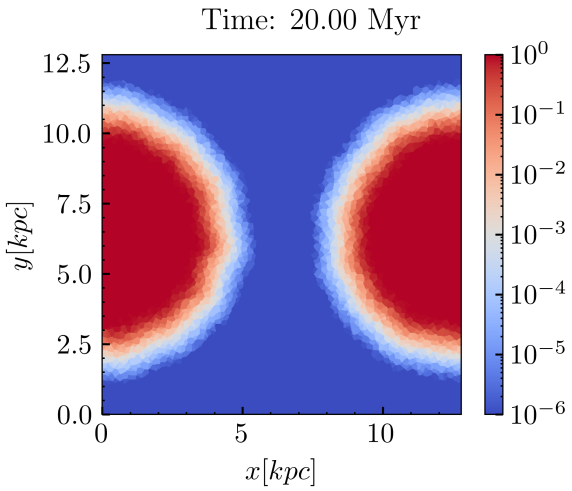

Figure 4.4. A slice through the box in the plane $z = 6.4\,\mathrm{kpc}$ at t=20 Myr. The color shows different values of the ionization fraction with blue being neutral and red being ionized.

4.3.2 Periodic test

In order to test the behavior of the algorithm in setups with periodic boundary conditions, we perform a test similar to the one in Section 3.3.4. In this test we perform a R-type expansion of a HII region as in the previous section (Section 4.3.1). The difference between the two simulations is that in this test, the source is placed not in the center box but right at the boundary in the x-direction at position $\boldsymbol{r} = (0, 6.4, 6.4)$ kpc. A slice through the box illustrating the setup of the test is shown in Fig. 4.4.

In the simulation there is no cell at exactly that position, so the source term will be introduced into a cell that is slightly to the right (i.e. at positive x) of $\boldsymbol{r}$, namely at $\boldsymbol{r} = (\epsilon, 6.4, 6.4)$ kpc, where $\epsilon > 0$ is small. Since the source is placed so close to the edge of the simulation box at $x = 0$, any photons originating at the source with a direction to the left must first pass through the (periodic) boundary before they re-enter from the other side and begin having an effect on the gas. Since the only symmetry breaking element in this setup is the simulation box itself, an accurate solver should produce a reasonably symmetric result, up to the precision determined by the resolution of the simulation. If photons exiting the boundary are not re-introduced on the other side consistently, we will notice a lack of ionization in the right side of the box, compared to the left side.

In order to quantify how well our solver deals with periodic boundary conditions, we compute the asymmetry a, defined as the relative difference between the average

ionization fraction in the left side of the box and the right side of the box given by

$$a = \left| \frac{\bar{x}_{\text{HII, left}} - \bar{x}_{\text{HII, left}}}{\bar{x}_{\text{HII, left}} + \bar{x}_{\text{HII, left}}} \right| \tag{4.15}$$

where the (volume-)averaged ionization fractions $\bar{x}_{\text{HII, left}}$ and $\bar{x}_{\text{HII, right}}$ are defined as

$$\bar{x}_{\text{HII, left}} = f(\epsilon, L/2 + \epsilon)$$

and

$$\bar{x}_{\text{HII, right}} = f(0, \epsilon) + f(L/2 + \epsilon, L),$$

with

$$f(x_1, x_2) = \frac{2}{V} \int_{x_1}^{x_2} \mathrm{d}x \int_{-L/2}^{L/2} \mathrm{d}y \int_{-L/2}^{L/2} \mathrm{d}z \; x_{\text{HII}}.$$

That is, the two averages are computed over the left- and right- halves of the simulation box as seen from the source at ϵ, which corresponds to the left- and right- halves of the box except for the small ϵ-sized sliver on the left.

From our implementation of periodic boundary conditions, we can expect that smaller timesteps will achieve more accurate results than larger timesteps, since the initial estimate of the photon fluxes is given simply by the fluxes from the previous timestep - if the timestep itself is small, the prediction will be more accurate. However, the main goal of this test is not to just test the behavior of the solver with respect to the timestep, but to check whether allowing the solver to perform sub-timesteps has a positive effect on the accuracy. In order to test this, we perform the simulation setup described above for a variety of timesteps and different numbers of sub-timestep levels n and compute the periodic asymmetry given by Eq. 4.15.

In Fig. 4.5 we show the asymmetry a as a function of the number of timestep levels n for different values of the maximum timestep Δt_{max}. Initially, it should be noted that the asymmetry is already quite small, with values below $a < 0.4\,\%$ even at a timestep of $\Delta t_{\text{max}} = 400\,\text{kyr}$, which corresponds to $\Delta t_{\text{max}} \approx \frac{3}{1000} t_{\text{rec}}$ (the recombination time in this test is the same as in 4.3.1). Despite this, we still see the expected overall trend, which is that the asymmetry decreases as the timestep decreases, in approximately linear fashion. Moreover, allowing sub-timestepping to use more levels also decreases the asymmetry, with a clear downwards trend for $n <= 5$. At $n = 6$ and $n = 7$ the asymmetry increases temporarily. While this may initially seem worrying, the magnitude of the asymmetry is already below the values that we can reasonably expect to resolve given the relatively low resolution of this test. We also note that the numerical result of this test is quite strongly dependent on the exact value of ϵ - while changing it slightly does not change the overall trend, it does change the absolute values of the asymmetry, especially for values below $a < 0.001$.

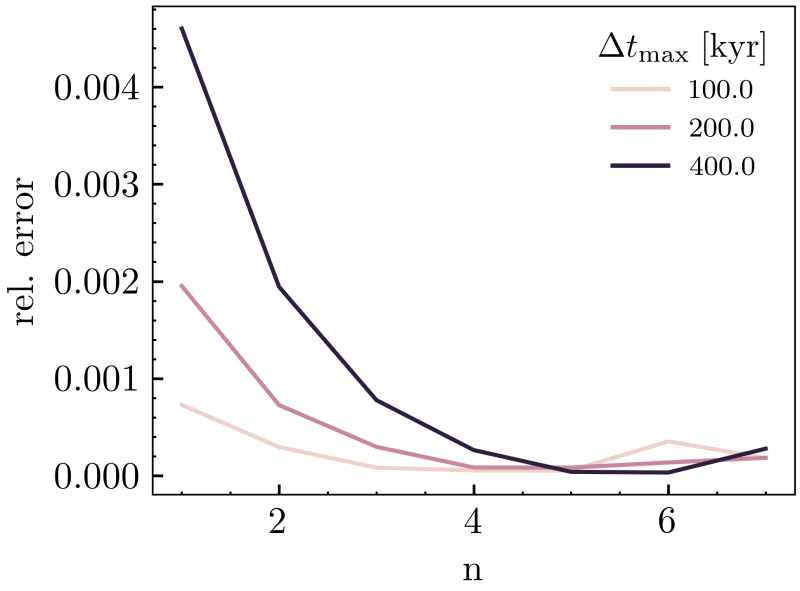

Figure 4.5. The asymmetry of the average ionization fraction (see Eq. 4.15) as a function of the number of timestep levels used for the test. The different lines are different values for the maximum timestep used.

4.3.3 Shadowing behavior behind an overdense clump

Due to the radial symmetry of the R-type expansion test shown in Section 4.3.1, it does not test the directional properties of the radiative transfer itself very well. In order to better test the behavior of sweep, we perform the following test which is also identical to the test setup in Jaura et al. [2018] and Baczynski et al. [2015] and in Section 3.3.2. In this test, we study the formation of a shadow behind an overdense clump. To do so, we set up a box of length $L = 32\,\mathrm{pc}$. The box is filled with hydrogen with a spatially varying density with

$$n_{\mathrm{H}}(\boldsymbol{x}) = \begin{cases} 1000\,\mathrm{cm}^{-}3 & \text{where } |x| < 4\,\mathrm{pc} \\ 1\,\mathrm{cm}^{-}3 & \text{else} \end{cases} \tag{4.16}$$

We place two point sources at $\boldsymbol{r}_1 = (-14, 0, 0)$ pc and $\boldsymbol{r}_2 = (0, -14, 0)$ pc, which emit photons at a rate of $\dot{N} = 1.61 \times 10^{48}\,\mathrm{s}^{-1}$. An analysis of this test, which includes hydrodynamics and discusses the temperature, pressure, and density response has been performed in the original SPRAI paper [Jaura et al., 2018].

In Figure 4.6, we show the rate of photons in units of $\mathrm{cm}^{-3}\mathrm{s}^{-1}$ in a slice through the simulation box along the x-y plane for different times (columns) and resolutions (rows). We find that the Sweep algorithm will correctly form a shadow behind the overdense clump. However, due to numerical diffusion, the shadow does not follow the theoretically expected form exactly. Because of this, the region behind the overdense

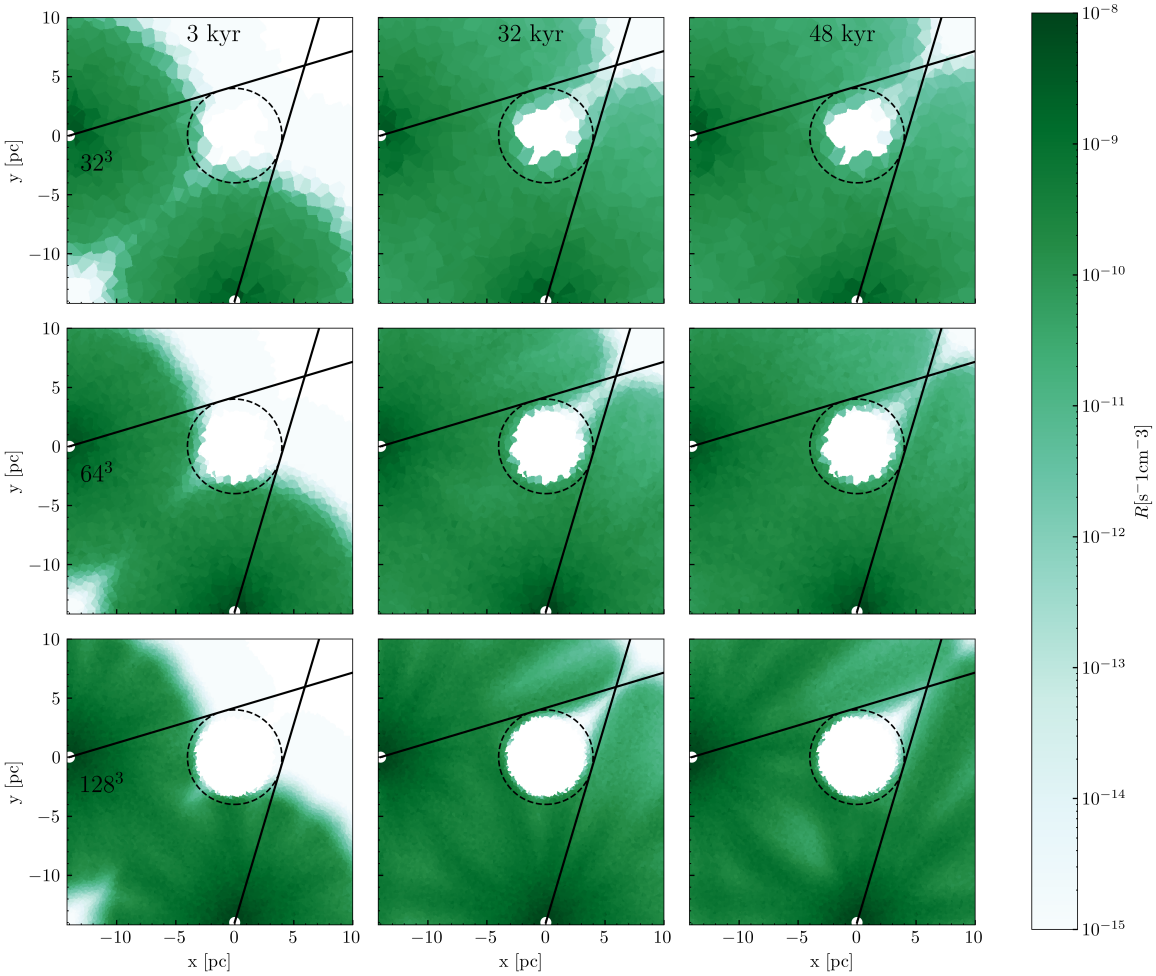

Figure 4.6. The photon rate R in a slice through the $z = 0$-plane of the simulation box. First row: 32^3 particles, Second row: 64^3 particles, Third row: 128^3. First column: $t = 3.0\,\mathrm{kyr}$, second column: $t = 32\,\mathrm{kyr}$, third column: $t = 48\,\mathrm{kyr}$, The black dashed circle represents the overdense clump. White solid circles represent the position of the sources. The black dashed lines delineate the shape of an ideal shadow behind the clump.

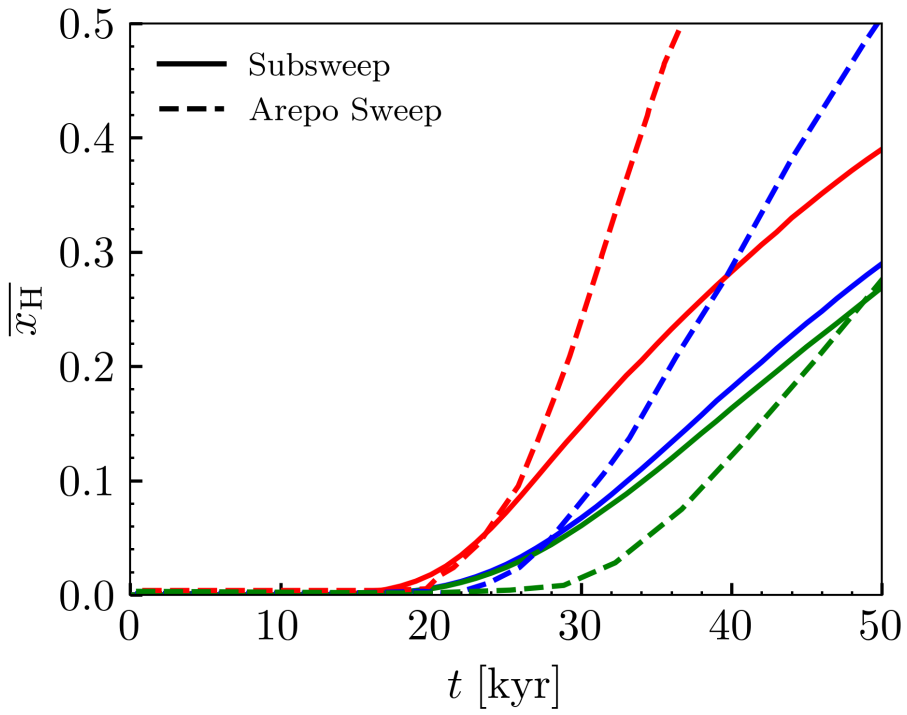

Figure 4.7. The average hydrogen abundance $\overline{x_{\rm H}}$ (see Eq. 4.17) in the shadow volume as a function of time for Subsweep (solid lines) and the original Sweep implementation in Arepo (dashed lines) for three different resolutions: 128^3 (green), 64^3 (blue) and 32^3 (red),

clump will slow begin to ionize. In order to quantify the shadowing behavior we calculate the mass averaged fraction of ionized hydrogen in the volume of the shadow. The volume is given by the intersection of two (infinitely extended) cones, with their tips at $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ respectively and their base determined by the great circle lying in the overdense clump. The overdense clump itself is excluded from the volume. In the 2D slice shown in Fig. 4.6, this volume $V_{\rm S}$ corresponds to the area between the black dashed circle and the black lines . The average ionization fraction in the shadow region $\bar{x_{\rm H}}$ is given by

$$\overline{x_{\rm H}} = \frac{\int_{V_{\rm S}} x_{\rm H}(\boldsymbol{r})\rho(\boldsymbol{r})\mathrm{d}V}{\int_{V_{\rm S}} \rho(\boldsymbol{r})\mathrm{d}V}, \tag{4.17}$$

where $x_{\rm H}(\boldsymbol{r})$ is the abundance of ionized hydrogen at position $\boldsymbol{r}$ and $rho(\boldsymbol{r})$ denotes the mass density at position $\boldsymbol{r}$.

Figure 4.7, $\overline{x_{\rm H}}$ is shown as a function of time. The ionization fraction begins to increase at $t \approx 20\,\mathrm{kyr}$, demonstrating that Sweep does not form a perfect shadow. The shadowing behavior improves going from lower resolution to higher resolution. This is in line with the explanation that the protrusion of the ionization front into the shadow is due to numerical diffusion, since higher resolutions decrease the effect of

numerical diffusion. We also find that for low resolutions (32^3, 64^3), Subsweep forms a better shadow than the Arepo Sweep implementation, especially at late times, while Arepo Sweep performs better in the high resolution case 128^3. It might be surprising that there are different results at all, considering the fact that the two implementations use the same algorithm for the radiation transport. The difference between the two implementations that is relevant to this test is that the chemistry updates are performed differently. The Arepo version of Sweep will perform a radiation chemistry update for each time a directional sweep encounters a cell. In Subsweep, cell abundances are fixed until the end of the sub-timestep and therefore remain the same for each directional sweep. This can result in different behavior at the ionization front.

4.3.4 1D R-type expansion

In order to test the behavior of the substepping algorithm, we perform a test in which we study the expansion of an ionization front in a one dimensional box filled with hydrogen of uniform number density $n = 1 \times 10^{-4}\,\mathrm{cm}^{-}3$. The medium extends from 0 to L. The gas is kept at a constant temperature $T = 100\,\mathrm{K}$. A source emitting a constant flux of ionizing photons of $1 \times 10^5\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$ in the direction towards the right is placed at $x = 0$. The time evolution of this system is characterized by the formation of an ionized region of all cells with $0 < x < R(t)$ where $R(t)$ is the length of this ionized region as a function of time and given by $R(t) = R_{\mathrm{st,1d}}\left(1 - e^{-t/t_{\mathrm{rec,1d}}}\right)$. Here $R_{\mathrm{st,1d}} = \frac{F}{\alpha_{\mathrm{B}} n^2}$ is the one-dimensional Strömgren length and $t_{\mathrm{rec},1d} = \frac{1}{\alpha_{\mathrm{B}} n}$ is the recombination time.

For the numerical simulation of the system, we divide the interval into N equidistant cells with width $\frac{L}{N}$ along the line. The leftmost cell contains the source. In this test, we only perform sweeps in one direction (pointing to the right). In practice, the ionization front will not be infinitely thin but extend over several cells. If N is large enough such that the ionization front is well-resolved, we can expect the analytical expression for the total ionized volume fraction $x_{\mathrm{analytical}} = \frac{R(t)}{L}$ to accurately predict the numerical result $x_{\mathrm{numerical}}$, so that we can define a simulation to have converged to the right result if

$$|x_{\mathrm{numerical}} - x_{\mathrm{analytical}}| < \epsilon \tag{4.18}$$

where $\epsilon > 0$ is the error tolerance which we choose to be $\epsilon = 1\%$. Our goal is to study the convergence behavior of the sweep algorithm in this system. In order to do so, we perform runs with different values for the maximum number of allowed timestep levels n, namely $n = 1,\ 2,\ 3,\ 4,\ 5$ and 6 as well as different numbers of cells $N = 160,\ 320,\ 640,\ 1280,\ 2560,\ 5120$ and $10\,240$. For each set of values of n and

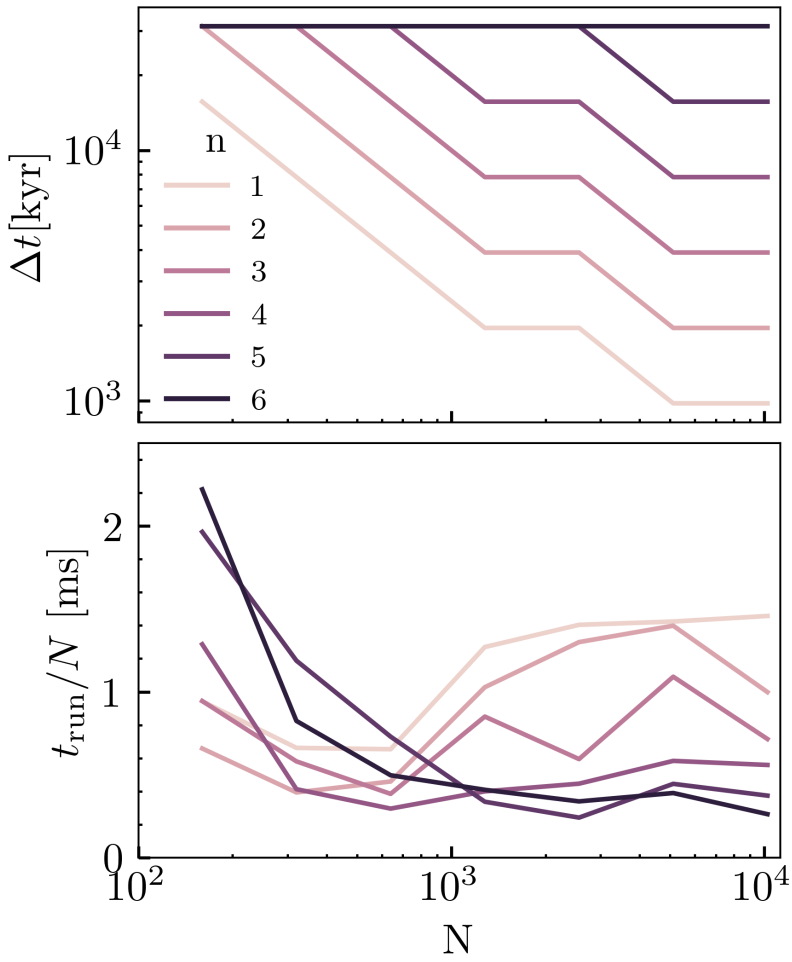

Figure 4.8. Top panel: The largest converging timestep Δt as a function of the number of particles N. The different lines represent runs with different number of allowed timestep levels n. Bottom panel: The total runtime t_{run} divided by the number of particles N.

N, our goal is to find the highest value of the timestep Δt for which the simulation reaches the correct result, i.e. where Eq. 4.18 holds.

In Figure 4.8, we show the maximum converging timestep Δt_{max} as a function of N for each value of n. For low values of N, adding more timestep levels does not result in a meaningfully different result. However, as N increases, the separation between runs at different n becomes clear, with higher values of n resulting in an increase in the highest possible converging timestep. This result clearly demonstrates that using substepping allows us to use higher timesteps while still converging to the physically correct result. In fact, each additional substepping level allows us to increase the maximum timestep Δt_{max} by a factor of two, which is the expected outcome.

It should be noted that this is a trivial implication if all cells were kept on the lowest timestep during the simulation, since in this case the subsweeping algorithm reduces

to performing 2^{n-1} sweeps of the full system with a timestep of $2^{-(n-1)}\Delta t_{\max}$ each. In order to demonstrate that this is not the case and that we have gained something from the subsweeping, the bottom panel of Fig. 4.8 shows the total runtime t_{run} of the corresponding simulation in the top panel, divided by the number of particles n. This clearly shows that, for large N, using more timestep levels alleviates the need to use a global, low timestep which in turn reduces simulation time significantly, while still producing physically accurate results. For small N, the substepping does not improve performance and at times will even decrease performance. One possible explanation for this could be that most if not all of the cells in the simulations are at a very low timestep. While having a lot of timestep levels will not change the numerical result of the simulation, it can decrease performance due to the additional computational overhead of communicating the levels of each of the cells multiple times for each timestep.

4.3.5 Tests of the radiation chemistry

In order to test the radiation chemistry, we perform a series of tests that follow those performed Rosdahl et al. [2013] as closely as possible. The difference between the two setups is that our radiation chemistry solver does not incorporate Helium, so some of the results can look different. However, we still expect the results to look qualitatively similar. For all these tests, we take a single cell which we initialize with a given temperature, density, fraction of ionized hydrogen and ionizing flux. We perform tests with all possible combinations of Densities between 1×10^{-8}, 1×10^{-6}, 1×10^{-4}, 1×10^{-2}, 1 and $1 \times 10^{2}\text{cm}^{-}3$, initial ionized hydrogen fractions between 0, 0.2, 0.5, 0.8 and 1.0, initial temperatures between 1×10^{3}, 1.6×10^{4}, 1.8×10^{5}, 3×10^{6} and 1×10^{8}K and either zero ionizing photon flux, or a ionizing photon flux of $1 \times 10^{5}\,\text{photons/s/cm}^2$. For all of these configurations, we perform normal time evolution tests in which we let the system evolve freely. For each test, we let the system evolve for a total time of 10 Gyr.

In Fig. 4.9 we show the hydrogen ionization fraction as a function of time. We find that most configurations eventually converge, but both the limit and the convergence time vary drastically between the different configuration.

In the case of zero ionizing flux (top panel of Fig. 4.9), the configurations that do not converge are those with very low densities ($n <= 1 \times 10^{-6}\,\text{cm}^{-3}$). There, recombination rates and collisional ionization rates are extremely low due to the n^2 dependence. The lack of ionizing flux results in zero photoionization, so that the ionization fraction in these cells remains constant over extremely long times. In all other cases, we find that the ionization fraction always converges either to 0 or 1. At

high enough densities ($n >= 1 \times 10^{-2}\,\mathrm{cm}^{-3}$), the cell always ends up fully neutral after long enough times. If the initial temperature is high enough and the density low enough, the cell can become fully ionized, although it might eventually become neutral again on extremely long timescales, but this would require running the tests for even longer timescales. Since timescales of longer than 10 Gyr are not relevant for our application we will refrain from running these tests for even longer.

If an ionizing flux of $F = 1 \times 10^5\,\mathrm{s}^{-1}\,\mathrm{cm}^{-2}$ is present (bottom panel of Fig. 4.9), virtually all configurations at low densities are immediately ionized. Only at comparatively high densities ($n >= 1\,\mathrm{cm}^{-3}$), recombination can dominate such that cells reach an equilibrium value (of approximately $x_{\mathrm{HII}} = 0.4$ for $n = 1\,\mathrm{cm}^{-3}$ and very close to $x_{\mathrm{HII}} = 0.0$ for $n = 1 \times 10^2\,\mathrm{cm}^{-3}$).

In Fig. 4.10 we show the temperature as a function of time for all test cases. As above, most configurations eventually converge, with the exception of zero ionizing flux in the case of very low densities. In the absence of an ionizing flux (top panel of Fig. 4.10), cells never heat and cool down on timescales determined by their densities. For the case of an ionizing flux of $F = 1 \times 10^5\,\mathrm{s}^{-1}\,\mathrm{cm}^{-2}$ (bottom panel of Fig. 4.10), equilibrium temperatures are on the order of $1 \times 10^4\,\mathrm{K}$, with convergence time being strongly affected by the cell density.

4.4 Conclusion

In this chapter, we discussed the radiative transfer postprocessing code Subsweep. Subsweep is a standalone code that takes input from astrophysical, hydrodynamics simulations codes (currently only Arepo is supported, but extensions are easily possible) and performs radiative transfer on the input. We discussed the choice and implementation of the domain decomposition as well as the algorithm for the Voronoi grid / Delaunay triangulation employed to efficiently construct a Voronoi grid in parallel.

We briefly discussed the original Sweep method, which is a discrete ordinate method which gives the exact solution to the scattering-less radiative transfer equation in a single pass over the grid in order to introduce the main feature of Subsweep - the extension of the transport sweep method to incorporate sub-timestep sweeps in order to solve the coupled system of radiative transfer and radiation chemistry more efficiently We described how the substepping is implemented by performing a hierarchy of timesteps in which individual cells are evolved according to their required timestep criterion as opposed to being evolved alongside all other cells in a global timestep.

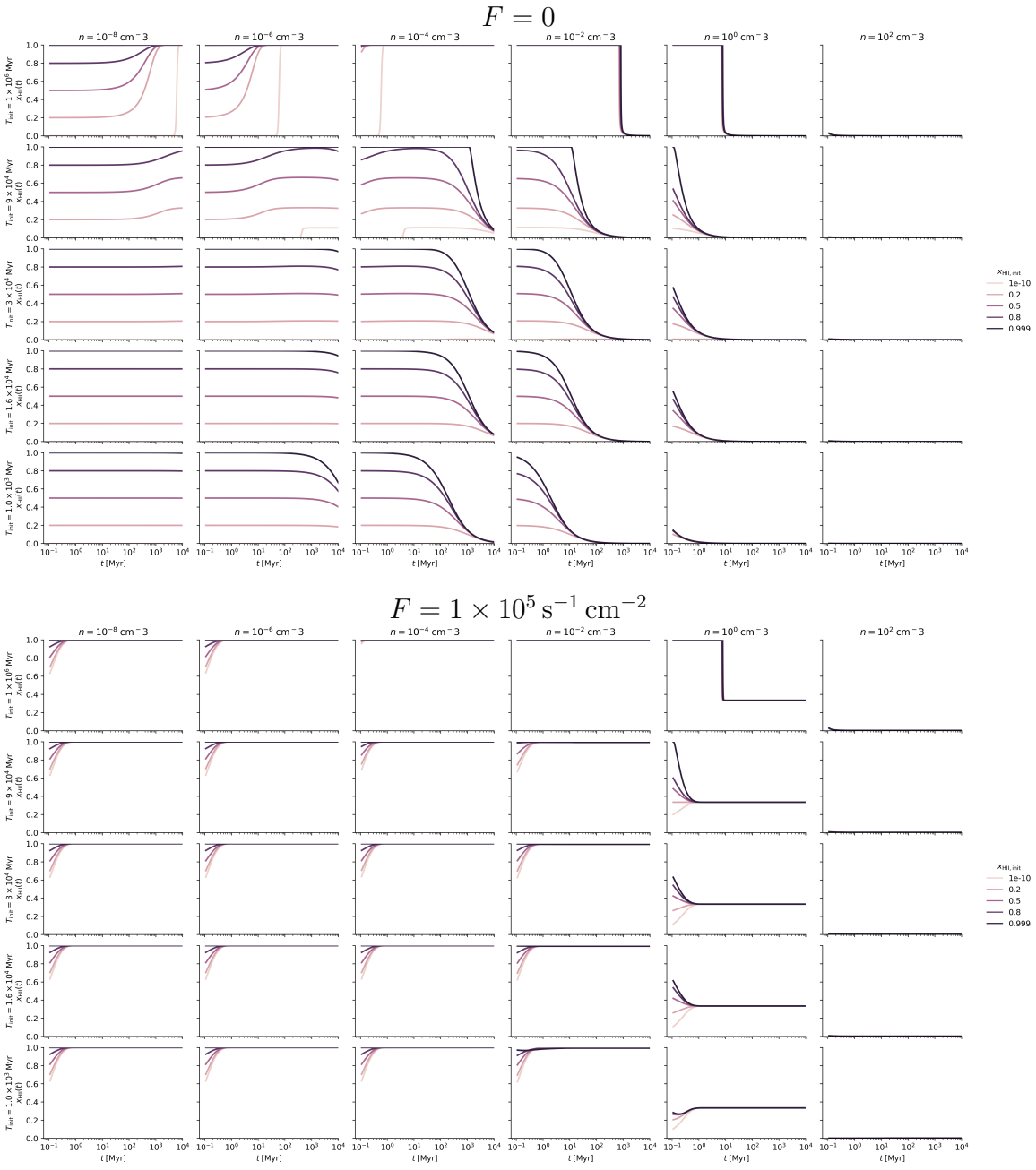

Figure 4.9. The time evolution of the ionized hydrogen ionization fraction for different values of the density in the cell (columns), different initial temperatures (rows) and different values of the initial ionized fraction (line colors). Top panel: no ionizing flux. Bottom panel: with an ionizing flux of $1 \times 10^5\,\mathrm{s}^{-1}\,\mathrm{cm}^{-2}$

Figure 4.10. The time evolution of the cell temperature for different values of the density in the cell (columns), different initial temperatures (rows) and different values of the initial ionized fraction (line colors). Top panel: no ionizing flux. Bottom panel: with an ionizing flux of $1 \times 10^5\,\mathrm{s^{-1}\,cm^{-2}}$

We tested the code on an R-type expansion of a ionized bubble in a medium of homogeneous density and found good agreement with the analytical prediction for this test. We also performed a test in which we studied the formation of a shadow behind an overdense clump. We found that the method performs well, but found significant differences in its behavior compared to the original Arepo implementation of Sweep. We believe that this difference is due to the different chemistry implementations as well as the slightly different treatment of photon fluxes between the two methods.

We investigated the performance of the sub-timestepping in a 1D-test in which we studied the equivalent of an R-type expansion. This allowed quickly varying parameters over a large range of values. We find that substepping does allow for larger maximum timesteps without sacrificing the accuracy of the solution, which in turns results in a reduction of the overall time to solution.

Our test of the R-type expansion around an ionizing source close to the boundary of the simulation box shows that sub-timesteps help significantly with reducing the overall cost in simulation time that a proper source iteration to convergence incurs. In fact, we find that for our application, the combination of substepping with Warm-starting (re-using the fluxes from a previous iteration) is enough to reach good levels of accuracy in simulations with periodic boundary conditions.

We also briefly discuss and test the implementation of radiation chemistry in Subsweep, which is a simple explicit solver with internal substepping that tracks hydrogen and the corresponding ionization and heating processes.

The primary extension to this method that could drastically improve performance is to allow cells to change their timestep level in between full sweeps, in order to quickly react to sudden a sudden increase or decrease in the required timestep, something we chose not to do because of the additional complexity that comes along with the implementation. Doing so could help with increasing the accuracy of the integration (in the case of a suddenly decreasing timestep) and improve performance (in the case of a suddenly increasing timestep).

We believe that this new improvement to the Sweep method is a significant step in order to incorporate it into a full cosmological simulation with hydrodynamics, gravity and a proper treatment of star formation. Previously, the requirement of a global Sweep timestep that applies to all cells prevented Sweep from accurately integrating the cells that required it without incurring a large cost in computational time. One challenge with such an implementation is that most state-of-the-art cosmological codes already contain their own internal sub-timestep hierarchy, usually for both gravity and hydrodynamics. The inclusion of the Subsweep method would require properly integrating with those timestepping schemes, especially the hydrodynamical one, since

they interact - radiative transfer can lead to increases in temperature which could lead to a reduced hydrodynamical timestep. Vice versa, hydrodynamical interaction can also a sudden increase in the required accuracy of the integration of radiative transfer. However, despite all of this additional complexity, we believe that integration of the Subsweep method into a full hydrodynamical code could potentially yield large benefits by bringing a method that efficiently computes very accurate solutions to the radiative transfer equations to cosmological simulations.

Chapter 5

Studying Reionization via Postprocessing of the TNG simulations

In this chapter, we employ the radiative transfer code Subsweep which we introduced in Chapter 4 in order to study the era of reionization on the outputs of the TNG simulations. We introduce our remapping approach to postprocessing, discuss the source model and the setup of the TNG simulations that we use. We use the resulting simulations to find a reasonable value for the effective escape fraction and verify the physical accuracy of our results by studying the spatial behavior of reionization and the state of the ISM and the IGM in our simulations.

5.1 Introduction

5.1.1 Motivation

The cosmic microwave background (CMB) radiation originated during the recombination epoch of the Universe, approximately at a redshift of $z \approx 1100$ resulting in a minimal residual free electron fraction. However, observations of the present-day Universe indicate a highly ionized intergalactic medium (IGM), as evidenced by the absence of Gunn-Peterson troughs [Gunn and Peterson, 1965] in nearby quasar absorption spectra. Thus, we are led to believe that there is an 'epoch of reionization' (EoR) during which the hydrogen in the IGM was reionized, likely by photons emitted by stars and quasars [Barkana and Loeb, 2001, Fan et al., 2006a]. This process likely began around $z \approx 7 - 10$ for hydrogen and later for helium around $z \approx 3$, influenced by the harder radiation emitted by quasars [Schaye et al., 2000]. Observations of

distant quasars seem to confirm that reionization was largely completed by $z = 6$ [Fan et al., 2006b]. Combined with the observed high value of the optical depth to electron scattering inferred from the CMB [Bennett et al., 2013, Collaboration, 2020], this implies that reionization likely began before $z \approx 7$. However, the duration and spatial properties of this process, as well as the source population causing reionization remain areas of ongoing observational and theoretical investigation.

In the coming decades, a new generation of advanced instruments will revolutionize our ability to observe Reionization phenomena. The James Webb Space Telescope (JWST) began observations in 2022 and the Extremely Large Telescope (ELT) will soon be operational, allowing us to target galaxies involved in the reionization process. Additionally, forthcoming radio experiments like the Square Kilometre Array (SKA) will offer opportunities for direct examination of the density and spatial distribution of neutral Hydrogen gas. Understanding and interpreting these new datasets will require extensive modeling and simulation efforts to uncover the underlying physical constraints of Reionization.

Various theoretical models have been developed to understand cosmic reionization, often employing semi-analytic methods or radiative transfer post-processing of dark matter simulation outputs. Notably, the excursion set approach to reionization developed in Furlanetto et al. [2004] has been widely utilized in analytical and semi-numerical models [Alvarez and Abel, 2007, Mesinger and Furlanetto, 2007].

Recent advancements have enabled the development of full radiation hydrodynamics simulations that simultaneously track cosmic reionization and structure formation in a self-consistent manner. While pioneering work such as Gnedin and Abel [2001a] laid the foundation for these kinds of simulations, it has only recently become feasible to study cosmological volumes in detail, allowing for the exploration of radiative feedback processes on galaxy formation, such as inhomogeneous photoionization heating [Paardekooper et al., 2013, Petkova and Springel, 2011].

Theoretical investigations suggest that the primary contributors to reionization are likely small, star-forming galaxies at high redshifts. While Pop-III stars may contribute to the production of high redshift photons, their overall impact is considered small compared to that of small galaxies [Paardekooper et al., 2013, Wise and Abel, 2008, Wise et al., 2014]. Similarly, active galactic nuclei (AGN) are not expected to play a critical role at high redshift due to their limited total luminosity [Faucher-Giguère et al., 2009, Hopkins et al., 2007]. Instead, it is proposed that small protogalaxies, with stellar masses around $10^4 M_\odot$ or lower, dominate the ionizing budget, potentially initiating reionization earlier [Ahn et al., 2012, Paardekooper et al., 2013].

Recent advancements in cosmological hydrodynamic simulations, such as the Il-

lustris, Eagle and TNG projects [Marinacci et al., 2018, Naiman et al., 2018, Nelson et al., 2018, 2021, Pillepich et al., 2018, Schaye et al., 2015, Springel et al., 2018, Vogelsberger et al., 2014a,b], have produced realistic galaxy populations both at $z = 0$ and high redshifts, incorporating sub-grid models of feedback processes that regulate star formation. Ideally, we would like to study the era of reionization self consistently within the framework of such hydrodynamical simulations. However, due to computational constraints we will focus on postprocessing studies in which radiative transfer is performed on the simulation outputs. This has the primary drawback that feedback from the radiation is not taken into account.

Previous studies such as Bauer et al. [2015] have performed such postprocessing analysis of the Illustris simulations. In this chapter, we will focus on postprocessing of the TNG simulations using our novel Subsweep method, which we introduced in Chapters 3 and 4.

5.2 Methods

5.2.1 Remapping of temperatures and ionization fractions

Ideally, we would like to model reionization of the model universe self-consistently, by incorporating on-the-fly radiative transfer using the Sweep algorithm alongside the simulation. In practice, large cosmological simulations are already extremely computationally intensive and already stretch available resources to the maximum. As we have already discussed in 3.2.2, radiative transfer is an extremely expensive process by itself, both due to the intractable nature of the underlying equation and the large dimensionality of the quantity of interest, specific intensity. Another complication for incorporating radiative transfer on-the-fly into hydrodynamic simulations is that the thermal feedback of the radiation can rapidly heat the gas surrounding sources of ionizing photons. This feedback can drastically reduce the timestep required for the accurate numerical integration of the hydrodynamical equations in a large number of cells, which can, in turn, increase the amount of work that has to be performed as well as affect processor load balance negatively. Moreover, depending on the hydrodynamic solver, one must be very careful in order to guarantee numerical stability of the solver in the presence of such sudden changes in the temperature, often requiring some sort of anticipatory mechanisms that can predict approximately where feedback will have an effect shortly after. While this is a common challenge for feedback mechanisms in general, very local feedback processes such as supernova feedback tend to be more predictable, both in time and location than radiative transfer, which can, by its nature,

have a strong, sudden impact on gas far away from the original photon sources.

The above arguments do not make applications of Sweep to large hydrodynamical cosmological simulations impossible, but they do make it technically challenging. For this reason, we decide to first pursue a simpler approach in which we only use Sweep as a form of postprocessing of the TNG simulations. The basic idea is that if every quantity except the chemical composition of the gas and the temperature are kept fixed and radiative transfer along with chemical evolution of the local cells is performed for a long enough time, the system will eventually reach an equilibrium state on a timescale T_{eq}. One might initially expect that T_{eq} is quite small, such that given any snapshot of the TNG simulations, we could quickly reach equilibrium and could use this equilibrium state as a good guess for the state in which the same TNG snapshot would have been if radiative transfer had been performed along with the simulation. However, in practice, we find that T_{eq} is large compared to the times between two subsequent snapshots of TNG. This means that, in running our Sweep simulation all the way to equilibrium, we are computing a state that is already inconsistent with TNG. Here, we will describe a method to deal with this problem in a more dynamic fashion.

In the beginning, we take the first TNG snapshot at the first redshift z_1 which we are interested and evolve the system using radiative transfer and chemistry until we reach the redshift of the next TNG snapshot at z_2. Now, we have two different sources of information about our system at z_2. The TNG snapshot gives the structure on which we would like to continue our computation. This includes the positions of the particles which may have moved, been refined or de-refined and also includes updated densities, heating and changes in the chemistry due to feedback processes (other than radiative transfer) and newly formed stars. The output of the Subsweep run has data on the chemical composition of the gas and heating due to the radiation. In order to combine the individual bits of information that both of these snapshots contain, we perform a procedure that we call remapping. Since we'd like to continue with the grid of the TNG snapshot, we need to map the information of our previous Sweep result onto the new TNG snapshot. In order to do so, we loop over all of the particles in the new TNG snapshot and perform a nearest-neighbor lookup on all of the particles in the Subsweep snapshot to find the closest particle. Given the values of temperature and ionization fraction in the TNG snapshot $(T_{\mathrm{tng}}, x_{i,\mathrm{tng}})$ and the temperature and ionization fraction of the Subsweep snapshot $(T_{\mathrm{sweep}}, x_{i,\mathrm{sweep}})$ we then set the real value to the maximum of each of the quantities: $T = \max(T_{\mathrm{tng}}, T_{\mathrm{sweep}})$, $x_i = \max(x_{i,\mathrm{tng}}, x_{i,\mathrm{sweep}})$. Using this remapping procedure we retain the ionization and heating of the Sweep result, which has the largest effect in the IGM without overwriting these effects where they

take place in TNG itself, which can become ionized for example due to collisional reionization in the high density regions.

In order to perform the nearest-neighbor lookup above efficiently, we first perform the domain decomposition according to the description in Section 4.2.2 using the positional data of the particles in the TNG snapshot. Then, we load the particle data of the corresponding Subsweep snapshot. The data is first distributed among all processors to allow efficient parallel read-in of the file and then re-distributed according to the same domain decomposition that we used for the TNG particles. We then construct a KD-tree over the TNG particles and first use a local nearest-neighbor lookup in which every processor determines the closest particle from the Subsweep snapshot that it contains locally. Since both data sets have been distributed due to the same domain decomposition, this local nearest neighbor will also be the global nearest neighbor most of the time. However, there are cases where a particle on some other processor might be closer. To obtain the right data in this case, each processor checks which other processors have a domain that is close enough (that is, closer than the local nearest neighbor) that a tree search should be performed on them. All particles with this property are then sent to the respective processors and a KD-tree lookup is performed on them. Anytime we find a particle that is even closer than the local nearest neighbor, it is reported back to the origin processor alongside the temperature and chemical abundances needed for the remapping.

5.2.2 Preventing cooling

During the Subsweep simulation, any heating due to hydrodynamic effects or feedback mechanisms such as supernovae that are present in the original TNG simulations cannot be accounted for. This means that, if left untouched, some of the dense gas in galaxies would immediately start cooling rapidly due to - for example - recombination. However, this cooling is not only spurious, but could also strongly affect the result of radiative transfer. In order to prevent this undesirable effect from occurring, we prevent any net cooling and net recombination in all cells of the simulation. Effectively, this is done by constraining the change in hydrogen ionization fraction and temperature to always be positive. Care must be taken to also apply this constraint during substeps of the chemistry, in order to prevent the solver from internally cooling during its own substeps, resulting in either invalid physical states or a large drain on performance due to the requirement of very small timesteps.

This fairly drastic operation of preventing cooling and recombination is based on the reasonable assumption that the majority of gas will only heat and ionize during recombination. While it is of course possible that small local pockets of gas would

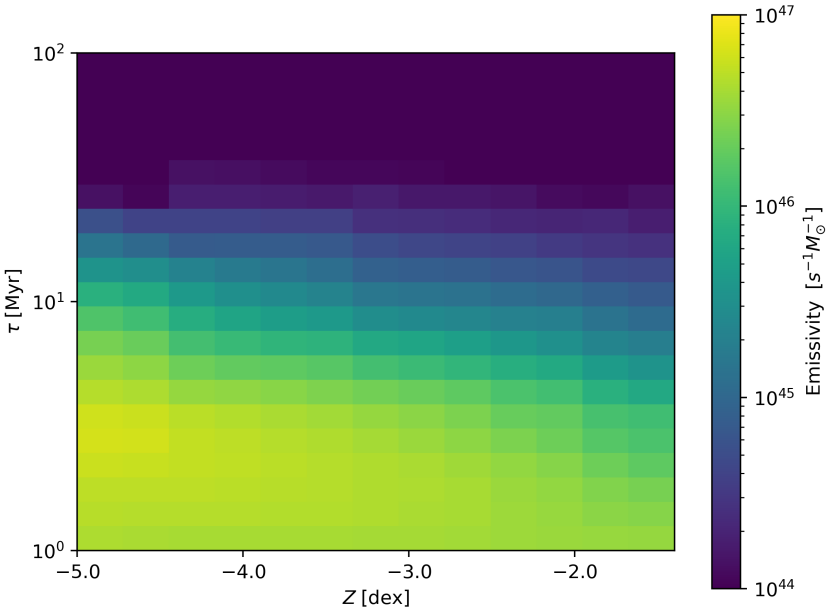

Figure 5.1. The interpolation table for the emissivity in $s^{-1}M_\odot^{-1}$ as a function of stellar metallicity Z and stellar age τ.

cool and recombine, for example if they are temporarily shielded from radiation by a dense cloud of gas, we believe that this is a reasonable assumption to more accurately model the large scale structure of reionization.

5.2.3 Source model

In order to model radiative transfer, we require an expression for the luminosity of the particles corresponding to stars in our simulations. Here, we are only interested in hydrogen reionization and therefore consider only photons with energies above $E = 13.65\,\mathrm{eV}$. We employ the Binary Population and Spectral Synthesis model (BPASS) [Eldridge et al., 2017], specifically v2.2.1 with binary populations, a top-heavy Kroupa IMF which is cut off at stellar masses above 100 $M_\odot$.

We compute the luminosity L via

$$L = \frac{M}{M_\odot}\bar{L}(Z,\tau), \tag{5.1}$$

where M is the mass of the star particle and $\bar{L}(Z,\tau)$ is an interpolation function based on the metallicity Z and age τ of the star particle. The interpolation is performed via linear interpolation between the two closest values in a pre-computed table of luminosities (see Fig. 5.1). The values show a clear dependence on both metallicity and age, with the emissivity decreasing slightly with increasing metallicity and decreasing strongly with increasing age. For each star particle, we use the value of the

metallicity of the TNG output and compute the age from the stellar formation time. Using the interpolated value of the emissivity, we multiply by the mass of the stellar particle to get the total luminosity L in photons per second. In order to incorporate the source terms into Subsweep, we map each star particle to its closest Voronoi cell by performing a parallel KD-tree lookup (for performance reasons, the lookup is implemented analogously to the lookup described in 5.2.1).

5.2.4 Escape fraction

The escape fraction, defined as the ratio of photons escaping a galaxy to the number of photons originally emitted is a crucial quantity in the study of reionization. At the heart of the problem is the difficult challenge of simultaneously simulating a large enough volume of space in order to obtain a cosmologically representative result and at the same time obtain high enough resolution in each individual galaxy of the simulation in order to accurately model the inner-galactic processes. Since even state-of-the-art simulations such as TNG cannot currently span all of these orders of magnitude, the result is that some of the interstellar medium will not be accurately modeled in the simulation. For our purposes, that means that the absorption of photons due to the dense ISM is usually underestimated, resulting in an overall larger number of photons available for reionization of the IGM than otherwise. In order to account for this fact, we introduce the artificial escape fraction f, a free parameter that accounts for this effect by reducing the amount of emitted photons. It should be noted that f may very well evolve with time and models such as the one proposed in [Kuhlen and Faucher-Giguère, 2012] have proposed a model for the time evolution of the escape fraction. For simplicity, we will assume an escape fraction that is constant in time in this work.

This artificial escape fraction f is also clearly dependent on the resolution of the underlying simulation. High resolution simulations will naturally resolve the ISM more accurately, leading to higher densities and thus higher absorption, which in turn means that the artificial escape fraction which we introduce does not need to be as low.

For our simulation results, we will briefly look at a number of simulations with different values of the escape fraction in order to roughly reproduce the ionization behavior we expect. However, it should be noted that this method can be improved by deriving the escape fraction, either from first principles, by performing a subset of high resolution simulations or by estimating a functional dependence of the artificial escape fraction on the resolution of the simulation, for example by computing clumping factors. For simplicity, in this work, we resort to simply leaving f as a free parameter,

but emphasize that this can be improved upon.

5.3 Results

5.3.1 Setting up the TNG simulation for postprocessing

In these results, we use Subsweep to perform radiative transfer postprocessing of the TNG100-3 box [Marinacci et al., 2018, Naiman et al., 2018, Nelson et al., 2018, 2021, Pillepich et al., 2018, Springel et al., 2018]. This is a lower resolution version of the original TNG100 (TNG100-1) simulation. Compared to the full simulation which consists of 1820^3 gas particles, the TNG100-3 box has a quarter of the resolution per axis, for a total of 455^3 gas particles. Other than the resolution, most of the parameters of the simulation are the same between the TNG100-1 and TNG100-3. The simulation box represents a volume with side length of $75\,\mathrm{cMpc\,h^{-1}}$. The cosmological parameters are based on the data from the 2015 release of the Planck collaboration [Planck Collaboration et al., 2016]. In particular, $\Omega_\mathrm{m} = 0.3089, \Omega_\Lambda = 0.6911, \Omega_\mathrm{b} = 0.0486$ and $h = 0.6774$. The original TNG100 simulation started at redshift $z = 127$ and ran until $z = 0$. Since we are mostly interested in reionization, we will only perform radiative transfer postprocessing for the time of interest. In this box, the first star forms at $z \approx 11.9$, but we find that luminosities do not reach significant levels until about $z = 8.5$, which is where we begin the postprocessing.

Before we begin postprocessing, we perform a rerun of the TNG100-3 box from $z = 127$ until $z = 5$ in order to achieve two things.

First of all, we disable the ultraviolet background (UVBG) of the original simulation. The UVBG was introduced in the original simulations in order to introduce effective reionization without requiring radiative transfer. Since this is exactly the process we want to study in more detail, we disable the UVBG, so that ionization and heating due to ultraviolet radiation can be computed by Subsweep. It should be noted that disabling the UVBG does mean that any feedback due to heating that had been introduced is not present in our rerun of the simulation.

The second reason for the rerun is that we want to reduce the time differential between subsequent output snapshots of the simulation. This is done in order to, at any given time, minimize the discrepancy between the original TNG result and the current snapshot being postprocessed in Subsweep.

5.3.2 Postprocessing

Using the outputs of the simulation with higher output frequency and no ultraviolet background radiation, we now use Subsweep to perform radiative transfer postprocessing of the simulation outputs. In this section we will detail our choice of Subsweep parameters for these simulations.

For all of the following simulations, we use the following parameters: For the number of directional bins, we choose $n_{\text{dir}} = 84$. The safety parameter x (see Eq. 4.6) that controls the accuracy of the integration is chosen as $x = 10\ \%$.

All simulations are performed with periodic boundaries enabled. As discussed in Section 4.2.7, we do not perform any additional periodic boundary iterations, since we deem the combination of Warmstarting and substepping to result in acceptable accuracy.

Simulation output of the relevant fields in every cell is performed every 50 Myr, but some quantities are computed within the simulation and output at every timestep. These include the volume- and mass- averaged ionization fraction, temperature, photoionization rate and information about the performance of the code as well as the placement of the particles on the timestepping hierarchy.

For the initial conditions, we take the first snapshot of the TNG simulation and use it to initialize the fractions of ionized hydrogen and temperatures for all cells. Densities are taken as-is from the TNG values and not modified during the Subsweep run, except for the (relatively small) effect due to expansion during the Subsweep run, which are changed according to $\rho(t) = \left(\frac{a(t)}{a(0)}\right)^{-3} \rho(0)$ where $\rho(t)$ and $a(t)$ are the density and scale factor at the time t since the beginning of the simulation (so that $\rho(0)$ and $a(0)$ are the initial density and scale factor).

5.3.3 Escape fraction

As a first step, our goal is to find an appropriate value for the artificial escape fraction f. To this end, we perform 4 different simulations with $f = 0.1, 0.2, 0.3$ and 0.5. For each simulation, we compute the (mass- and volume-) averaged neutral hydrogen fraction as a function of redshift.

Figure 5.2 shows the neutral fraction as a function of redshift compared to a number of constraints on the volume-weighted average obtained from a variety of observational and computational methods [Becker et al., 2015, Bosman et al., 2022, Ďurovčíková et al., 2020, Fan et al., 2006b, McGreer et al., 2015, Ouchi et al., 2010, Tilvi et al., 2014, Totani et al., 2016]. First of all, we find that the overall shape of the reionization history is reproduced by our results well within the statistical error

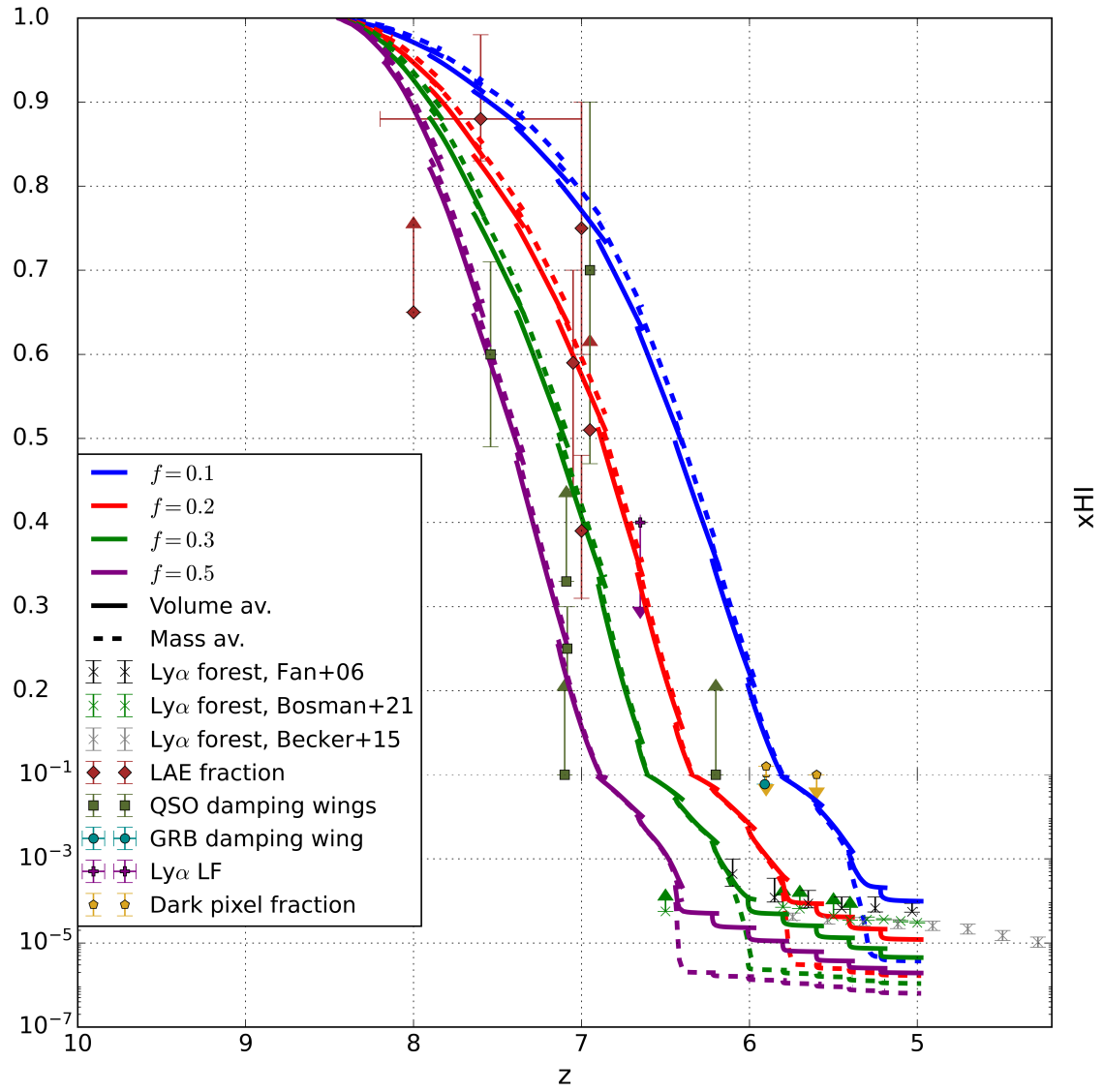

Figure 5.2. The hydrogen neutral fraction x_{HI} as a function of redshift z. Blue: Subsweep result with $f = 0.1$. Red: Subsweep result with $f = 0.2$. Green: Subsweep result with $f = 0.3$. Purple: Subsweep result with $f = 0.5$. Solid lines: volume average. Dashed lines: mass average. Axes in linear scale for $x_{\mathrm{HI}} > 0.1$ and in log-scale below $x_{\mathrm{HI}} < 0.1$. The constraints and their corresponding uncertainties apply to the volume-averaged fraction and are obtained from Lyman-α forests [Becker et al., 2015, Bosman et al., 2022, Fan et al., 2006b, Tilvi et al., 2014], quasar damping wings [Ďurovčíková et al., 2020], gamma ray burst damping wings [Totani et al., 2016], measurements of the Lyman alpha luminosity function (Ly_α LF) [Ouchi et al., 2010] as well as dark pixel analysis [McGreer et al., 2015].

Δt_{max}	n
1.0 Myr	2
1.0 Myr	6
4.0 Myr	2
4.0 Myr	4
4.0 Myr	6

Table 5.1. The different simulations we perform in order to assess the timestep parameters. Shown are the number of timestep levels n and the maximum timestep Δt_{max}.

bars given by the various observational constraints. As expected, the neutral fraction starts out at 1 at $z = 8.5$ before slowly decreasing as the first ionized bubbles form around the first stars. Reionization begins to accelerate at $z = 7.5$ with x_{HI} reaching 0.1 by $z = 7$ and $z = 5.5$, depending on the escape fraction. At $x_{\mathrm{HI}} \approx 1 \times 10^{-4}$, the mass and volume average begin drifting apart significantly, with the volume average tapering off before the mass average. Even at $z = 5$, neither average has converged fully. We also reproduce the tail, in which reionization slows down drastically as the amount of non-ionized gas in the box is depleted more and more. While we do not have very strong constraints on the neutral fraction, we find that the simulation with $f = 0.2$ fits the observational constraints best, so that we will proceed all further tests by performing simulations with this value of f.

All trajectories show a number of small discontinuities, which occur due to switching the underlying TNG snapshot during postprocessing. Switching snapshots requires a remapping of the quantities, which can lead to a sudden change in the averages over the entire volume. Moreover, using a new TNG snapshot means a change in the spatial distribution of sources and their luminosities, which can lead to non-continuous behavior as well.

5.3.4 Timestep parameters

In order to check whether we have chosen reasonable numerical parameters, we perform a set of simulations on the 455^3 TNG-100 box in which we vary the used timestep parameters, the maximum allowed timestep Δt_{max} and the number of sub-timestep levels n. We perform five simulations with $f = 0.4$ and vary the parameters as shown in 5.1.

In order to check whether our timestep parameters are reasonable, we compute the neutral fraction of hydrogen as a function of redshift for each of these simulations.

Figure 5.3 shows the obtained results. We find that each time we decrease the effective timestep (by lowering $\Delta t_{\max}$ or increasing n, ionization takes place slightly earlier. This is consistent with our expectations, and the tests we performed in 4.3, which show that ionization fronts move slower if the effective timesteps are too large.

We find that, initially, decreasing the timestep has a drastic effect on the result. For example, the run with $\Delta t_{\max} = 4\,\mathrm{Myr}, n = 2$ clearly does not resolve ionization fronts properly, leading to a much delayed timing of reionization to a degree that is unacceptable in terms of its accuracy. On the other hand, we find that, for example, the two runs with $\Delta t_{\max} = 4\,\mathrm{Myr}, n = 6$ and $\Delta t_{\max} = 1\,\mathrm{Myr}, n = 6$, which differ in their effective timestep by a factor of 4, both give very similar results for the history of reionization and differ by an amount that is well within what we consider acceptable.

We note that this result holds only for this particular box. Should the physical setup, in particular the resolution, of the simulation change, another set of such convergence tests will be required in order to ensure that the results are physically accurate. Given that we have confirmed that the results convergence reasonably for the run with $\Delta t_{\max} = 1\,\mathrm{Myr}, n = 6$, we will now proceed our analysis using the results from this run only.

5.3.5 Spatial Distribution

In Fig. 5.4, we show a slice through the center of the box in the $x - y$ plane at three different redshifts $z = 7.9,\ 6.9$ and 6.4. Shown are the temperature T, the ionization fraction x_{HII} and the density ρ. The picture shows the stages of reionization, beginning with a number of small bubbles forming around the most luminous galaxies. This is followed up by expansion of those bubbles until they eventually overlap and begin to increase in size before they eventually cover the entire box. The overall picture is consistent with the idea that overdense regions are ionized before underdense regions [Bauer et al., 2015, Razoumov et al., 2002]. Due to the ionizing radiation, the gas in the IGM is heated up until it eventually reaches temperatures of about $2 \times 10^4\,\mathrm{K}$ almost everywhere.

We define the ionization redshift z_{ion} and the ionization time t_{ion} as the first redshift/time at which

$$x_{\mathrm{HII}} = 0.5. \tag{5.2}$$

Using the ionization time, we also define the ionization velocity v_{ion} as

$$v_{\mathrm{ion}} = \boldsymbol{\nabla} t_{\mathrm{ion}}, \tag{5.3}$$

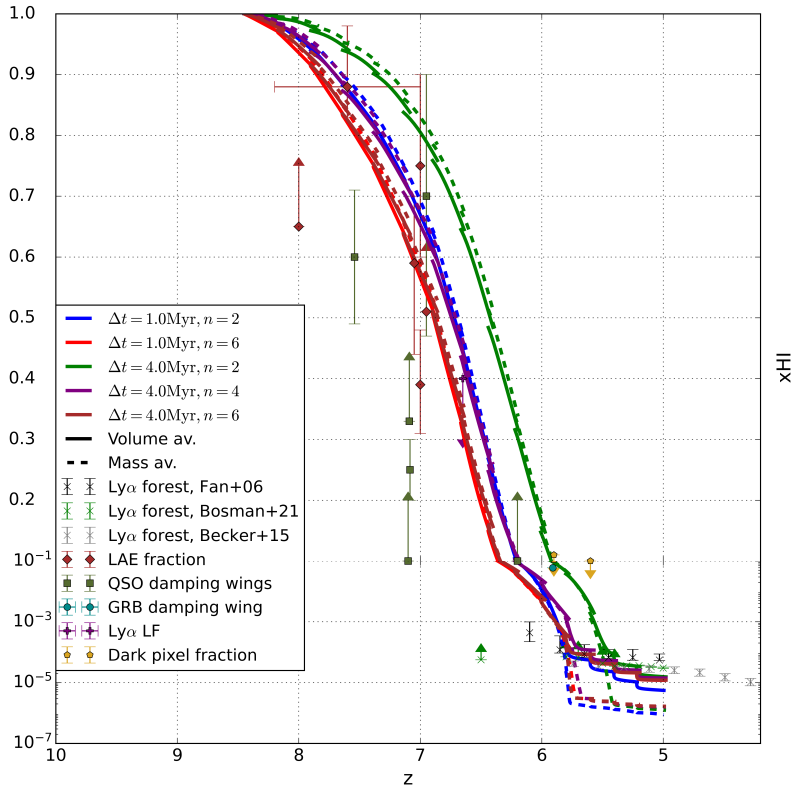

Figure 5.3. The hydrogen neutral fraction x_{HI} as a function of redshift z. Blue: $\Delta t_{\mathrm{max}} = 1\,\mathrm{Myr}, n = 2$, Red: $\Delta t_{\mathrm{max}} = 1\,\mathrm{Myr}, n = 6$, Green: $\Delta t_{\mathrm{max}} = 4\,\mathrm{Myr}, n = 2$, Purple: $\Delta t_{\mathrm{max}} = 4\,\mathrm{Myr}, n = 4$, Brown: $\Delta t_{\mathrm{max}} = 4\,\mathrm{Myr}, n = 6$, Solid lines: volume average. Dashed lines: mass average. Axes in linear scale for $x_{\mathrm{HI}} > 0.1$ and in log-scale below $x_{\mathrm{HI}} < 0.1$. The constraints and their corresponding uncertainties are obtained from Lyman-α forests [Becker et al., 2015, Bosman et al., 2022, Fan et al., 2006b, Tilvi et al., 2014], quasar damping wings [Ďurovčíková et al., 2020], gamma ray burst damping wings [Totani et al., 2016], measurements of the Lyman alpha luminosity function (Ly$_\alpha$ LF) [Ouchi et al., 2010] as well as dark pixel analysis [McGreer et al., 2015].

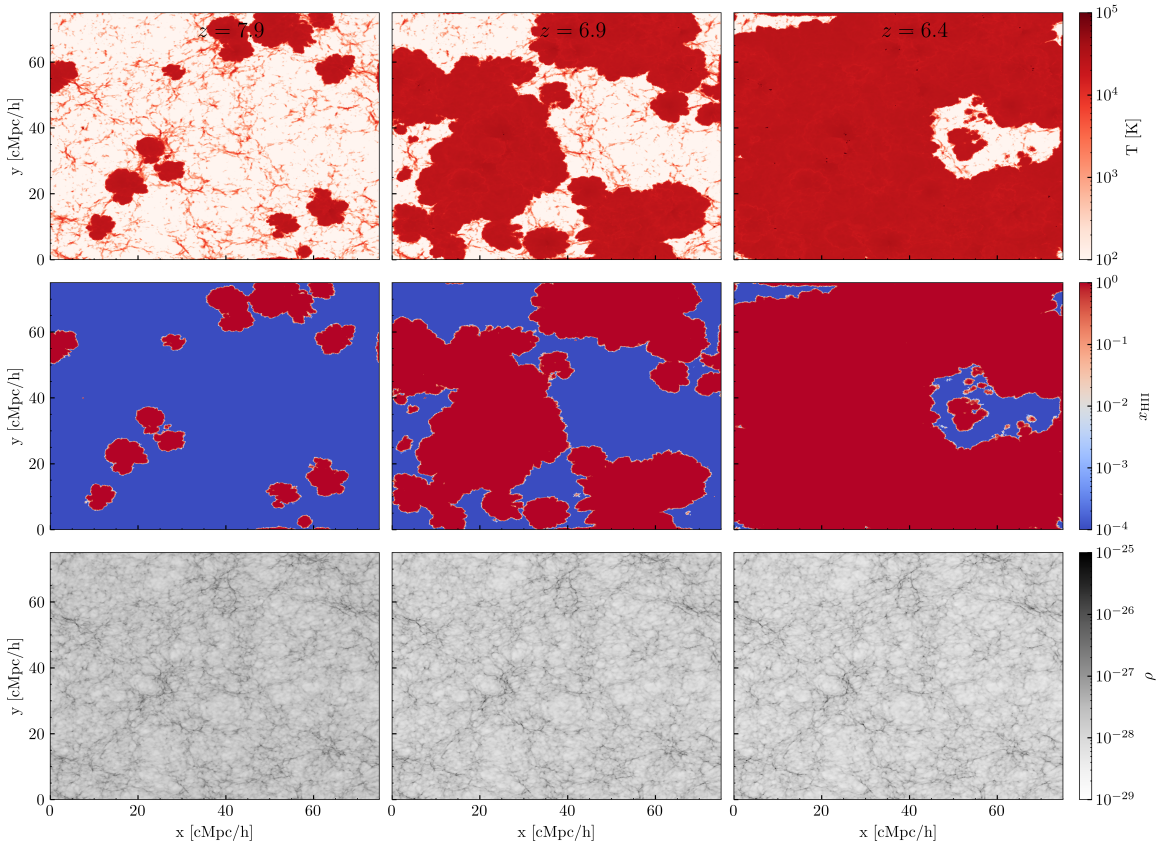

Figure 5.4. A slice through the simulated volume at $z = 37.5\,\mathrm{Mpc}$. Rows represent different quantities: Top row: Temperature T in K. Middle row: Ionized hydrogen fraction x_{HII}. Bottom row: Density ρ in $\mathrm{g}^{-1}\,\mathrm{cm}^{-3}$. Columns represent different redshifts. Left column: $z = 7.9$ Middle column: $z = 6.9$ Right column: $z = 6.4$

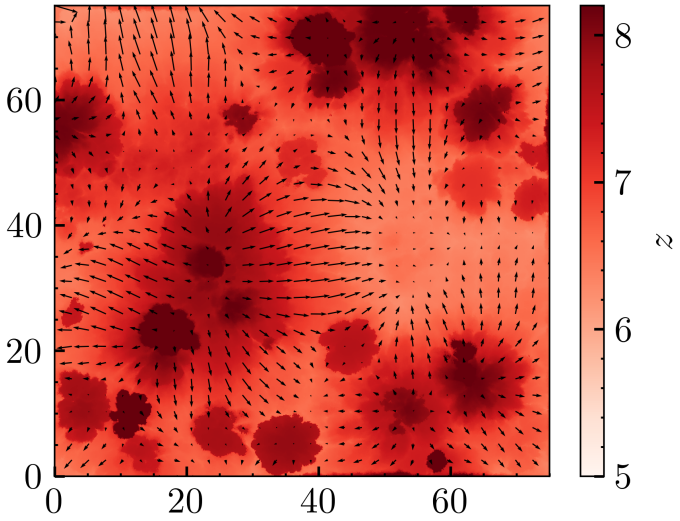

Figure 5.5. The ionization redshift z_{ion} (see Eq. 5.2) in a slice along the $x - y$ axis through the box at $z = 37.5\,\text{Mpc}$. The arrows denote the velocity v_{ion} of the ionization front at the given point in space, computed as per Eq. 5.3 (Gaussian smoothed with a characteristic length of $L = 5\,\text{kpc}$.

which we will use as a proxy for the direction and speed with which the ionization front moves. These definitions allows us to capture both spatial and temporal information in a single slice through the box, such as in Fig. 5.5, which shows the ionization redshift as well as the ionization velocity.

The results show the initial bubbles as fairly well-defined. We note that some of the bubbles have a particularly sharp edge, at which the expansion of the ionization front seems to stop. This can partially be explained by the fact that, as discussed in 5.2.3, the dependence of the source term on stellar age is quite strong, such that a small number of young stars can dominate reionization in the entire bubble. As those stars age and lose a sizable fraction of their luminosity, the size of the bubble is now large compared to the expected size for an R-type expansion with the given source term and the given density of the medium, leading to stagnation of the expansion.

5.3.6 State of the Gas

Figure 5.6 shows temperature-density histograms of the simulation box at two different redshifts $z = 8.16$ and $z = 6.66$. At $z = 8.16$, reionization has barely begun. At this point, the low-density gas of the IGM with about $1 \times 10^{-28}\,\text{g}\,\text{cm}^{-3}$ to $1 \times 10^{-26}\,\text{g}\,\text{cm}^{-3}$ is determined by the TNG model and spans a wide range of temperatures from $T = 1 \times 10^{1}\,\text{K}$ to about $T = 1 \times 10^{4}\,\text{K}$. The high density gas above $1 \times 10^{-24}\,\text{g}\,\text{cm}^{-3}$ is constrained extremely strongly by the equation of state to temperatures between

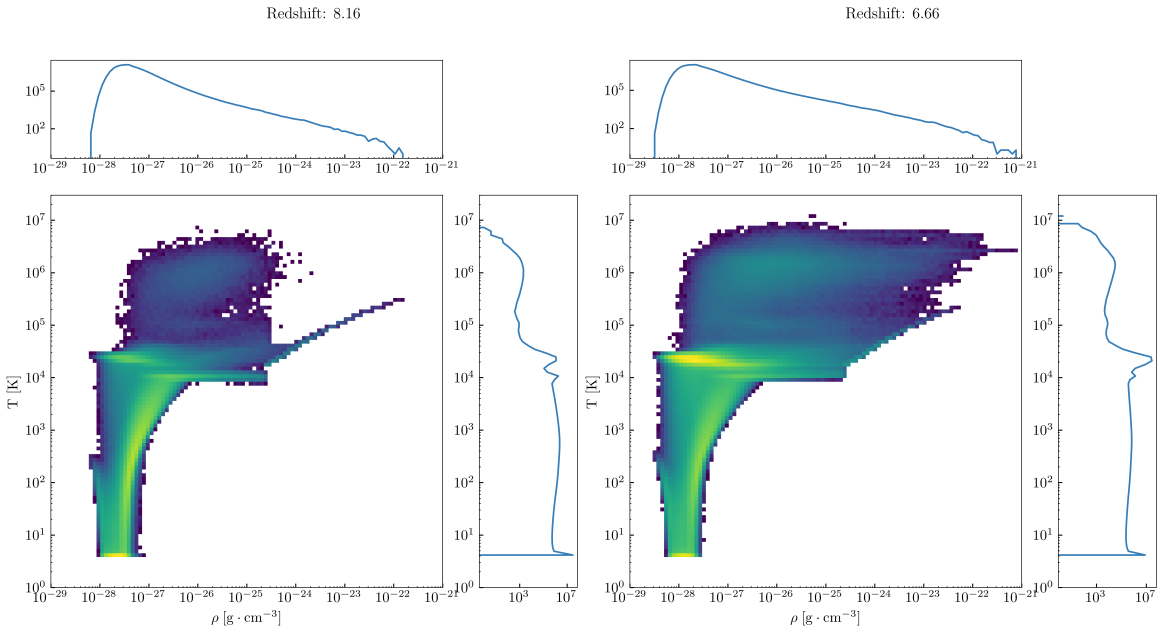

Figure 5.6. Central panel of each figure: Histograms of density ρ in $\mathrm{g\,cm^{-3}}$ vs. temperature T in K for all cells in the simulation. Logarithmic color scale. Right and top panel: One-dimensional histograms of densities (top panel) and temperatures (right panel). Left figure: $z = 8.16$. Right figure: $z = 6.66$.

$5 \times 10^4\,\mathrm{K}$ and $5 \times 10^5\,\mathrm{K}$. Local feedback processes have caused a small fraction of the medium-density gas to reach temperatures of about $1 \times 10^6\,\mathrm{K}$.

At $z = 6.66$ reionization has progressed to a point where approximately 60 % of the volume is ionized. This can be seen in the histogram primarily due to the strong peak at approximately $T = 3 \times 10^4\,\mathrm{K}, 5 \times 10^{-27}\,\mathrm{g\,cm^{-3}}$ which corresponds to the heated, ionized IGM. At the same time, the extremely hot gas with temperatures of about $T = 1 \times 10^6\,\mathrm{K}$ is still present. This ability to capture both the feedback due to photoionization as it is modeled by Subsweep while also incorporating local feedback from the TNG model is the primary advantage of the remapping approach described in 5.2.1.

Figure 5.7 shows temperature-ionization fraction histograms of the simulation box at two different redshifts $z = 8.16$ and $z = 6.66$. Similar to the temperature density histograms, the gas is initially (at $z = 8.16$) split in a bimodal distribution, with one peak at completely neutral gas which spans temperatures from $T = 7\,\mathrm{K}$ to $T = 2 \times 10^4\,\mathrm{K}$ and a second peak at fully ionized gas spanning temperatures from $T = 1 \times 10^4\,\mathrm{K}$ to $T = 5 \times 10^6\,\mathrm{K}$. The former peak is caused by the IGM which is completely neutral at this redshift and the latter peak corresponds to the gas heated from feedback processes in TNG. As reionization progresses (at $z = 6.66$), we see that some of the neutral gas has been ionized and heated, occupying the previously seen peak at $T = 3 \times 10^4\,\mathrm{K}$.

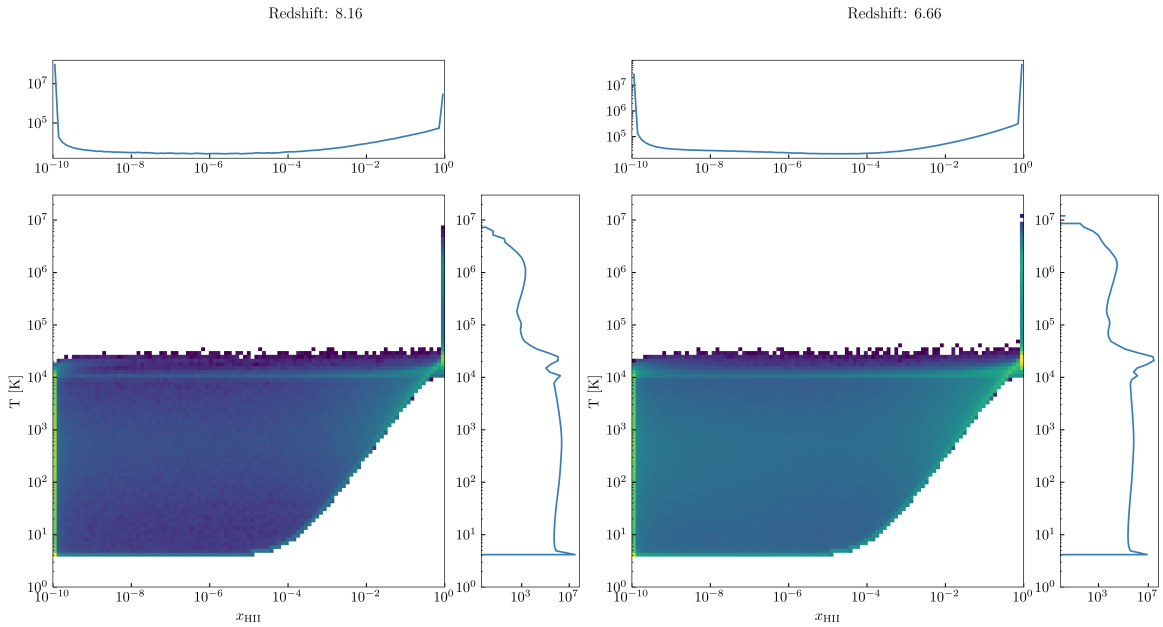

Figure 5.7. Central panel of each figure: Histograms of ionization fraction x_{HII} vs. temperature T in K for all cells in the simulation. Logarithmic color scale. Right and top panel: One-dimensional histograms of ionization fractions (top panel) and temperatures (right panel). Left figure: $z = 8.16$. Right figure: $z = 6.66$.

The overall picture is corroborated by Fig. 5.8, which shows the mass-averaged temperature in K as a function of redshift z in three different density bins. The lowest density bins follows a comparably smooth distribution, increasing from an average temperature of about $2 \times 10^3\,\mathrm{K}$ at $z = 8$ to about $3 \times 10^4\,\mathrm{K}$ at $z = 5$.

In comparison, the medium and high density bins show a very different evolution. While the gas is also heated over time, with average temperature increasing from approximately $1 \times 10^4\,\mathrm{K}$ to $2 \times 10^5\,\mathrm{K}$ for the medium density gas and from $6 \times 10^4\,\mathrm{K}$ to $1 \times 10^6\,\mathrm{K}$ for the high density gas, this increase does not occur smoothly but in a series of jumps. These jumps are due to the remapping of the values from the TNG simulations, in which the local feedback in galaxies can cause the gas to heat to very high temperatures. The effect is especially apparent for lower redshifts, where radiation feedback does not increase the temperature of the dense gas anymore so that the temperature stays almost constant during each individual Subsweep run but is then increased as the values are updated from the next TNG snapshot.

5.3.7 Photoionization Rate

In Fig. 5.9 we show the volume-averaged photoionization rate Γ as a function of redshift z, in comparison to observational constraints from [Calverley et al., 2011, D'Aloisio et al., 2018]. Each individual simulation initially starts out with a lower

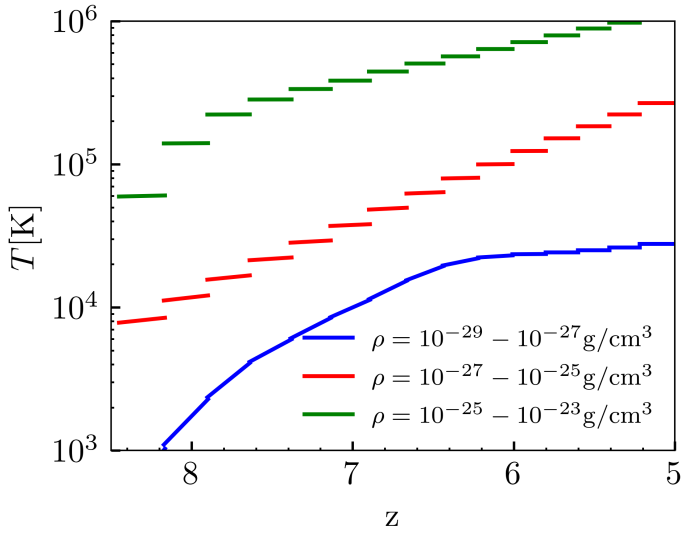

Figure 5.8. The mass-averaged temperature T in K as a function of redshift z for three different bins of the density ρ: Blue: $1 \times 10^{-29}\,\mathrm{g\,cm^{-3}} < \rho < 1 \times 10^{-27}\,\mathrm{g\,cm^{-3}}$. Red: $1 \times 10^{-27}\,\mathrm{g\,cm^{-3}} < \rho < 1 \times 10^{-25}\,\mathrm{g\,cm^{-3}}$. Green: $1 \times 10^{-25}\,\mathrm{g\,cm^{-3}} < \rho < 1 \times 10^{-23}\,\mathrm{g\,cm^{-3}}$.

photoionization rate before it rapidly increases and begins plateauing. We believe that this effect is mostly of numerical nature and caused by the solver not having converged, so that the eventual plateau is a more reasonable value for the photoionization rate. However, we still find that our simulations have a stronger rate of ionization than observations in the $z = 5$ to $z = 6$ range, up to a factor of 10.

5.3.8 Performance

In Table 5.2, we show a number of the simulations we have performed along their computational cost. Each run was performed on 512 cores. For each run, we show the number of timestep levels n, the maximum timestep Δt_{max}, as well as the total runtime cost, which is the sum of the time taken by each individual simulation times the number of cores used (512). The overall cost is only a measure of how expensive the simulation is to run, but does not take into account the different levels of accuracy achieved by each simulation, which is why we also show the total runtime cost divided by the number of lowest-timestep sweeps, which takes into account the additional cost that simulations at lower timesteps and simulations with different numbers of timestep levels require. We find that all of our simulations are reasonably cheap to run. These results also show that sub-timestep sweeps help increasing accuracy without decreasing the overall runtime cost, as can be seen in the comparison of the run with $n = 4, \Delta t_{\mathrm{max}} = 4\,\mathrm{Myr}$ and that with $n = 2, \Delta t_{\mathrm{max}} = 1\,\mathrm{Myr}$, which have the

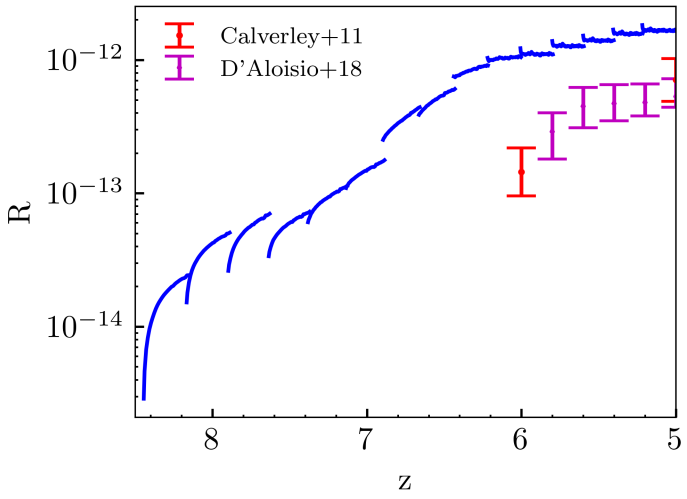

Figure 5.9. The ionization rate Γ in photons/s as a function of redshift z. Blue: Simulation result. Red: Constraints from [Calverley et al., 2011]. Purple: Constraints from [D'Aloisio et al., 2018].

$\Delta t_{\max}$	n	cost [core h]	cost / sweep [core h]
1.0 Myr	2	7.378×10^3	8.30
1.0 Myr	6	1.91×10^4	1.98
4.0 Myr	2	2.27×10^3	7.51
4.0 Myr	4	3.92×10^3	4.27
4.0 Myr	6	7.66×10^3	2.42

Table 5.2. Performance data on different simulation runs. For each run, we show the number of timestep levels n, the maximum timestep $\Delta t_{\max}$, the total runtime cost in core hours, as well as the total runtime cost divided by the number of sweeps at the lowest timestep level.

same effective minimum timestep but differ in the total cost by about a factor of 2.

5.4 Conclusion

In this chapter, we have applied our numerical radiative transfer postprocessing code Subsweep to the TNG simulations. In particular, we performed a set of simulations in which we studied reionization of a low-resolution version of the TNG100 simulation. To this end we performed a rerun of the simulation itself without the ultraviolet background in order to give full responsibility over reionization to Subsweep. The simulations themselves were performed by a sequence of sub-simulations beginning from redshift $z = 8.5$ to $z = 5$. The result of each sub-simulation is then combined

with the next TNG snapshot by remapping temperatures and ionization fraction to the respective maximum of each result. This method helps incorporate both the heating and ionization from the ionizing radiation within Subsweep, as well as from local feedback processes, such as supernovae into the final result.

As a source model, we use an interpolated version of the BPASS tables according to stellar age and metallicity, under the assumption of a Kroupa IMF and a high mass cutoff. We only perform hydrogen reionization and therefore only regard a single frequency bin of photons above $E = 13.65\,\mathrm{eV}$.

We performed a series of simulations which differ in their effective escape fraction in order to find that, while all of the simulations we performed reproduce the overall shape of the time evolution of reionization, the simulation with $f = 0.2$ seems to give the best fit to the observational constraints. We then studied the simulation with $f = 0.2$ in more detail. We begin by studying the spatial distribution and the formation of ionized bubbles and find that our results are compatible with the idea that reionization happens in the high density medium around galaxies first before forming bubbles that eventually cover the medium density gas and low density IGM.

We also find that the state of the gas is consistent with our expectations. Initially, the box starts out at relatively low temperatures everywhere except in the high density regions surrounding the galaxies. As reionization progresses, more and more of the gas is ionized and heated up, until a large fraction of the gas has temperatures of $T = 3 \times 10^4\,\mathrm{K}$ and above. Our comparison of the ultraviolet background produced by radiative transfer in our simulations to the constraints given by observations finds that we overestimate the photoionization rate by a factor of up to 10.

In future work, we aim to perform this set of simulations for higher resolution versions of the TNG box. We have already confirmed in a number of test runs that Subsweep is capable of handling those larger boxes with reasonable amounts of computational effort, so that running the simulations and performing the data analysis is the only work left to do.

Another interesting approach to making these results more accurate is to incorporate Helium, both for its effects on Hydrogen reionization and for the separate problem of understanding HeII reionization in more detail. Incorporating Helium would increase the amount of memory required for these simulations drastically, due to the additional required bin of photon frequencies. However, we believe that the increase in computational cost would be much less than the expected factor of 2, due to the fact that the most costly operation during a Sweep is the dependency counting and iteration through the grid, which would still only need to be performed once. Moreover, the simulation would have to be performed for a longer physical time, since

Helium reionization is expected to occur later, increasing the amount of computational time required. Another improvement that should be considered in order to accurately study Helium reionization is to extend the source model to incorporate AGNs, which are believed to be crucial for Helium reionization.

Chapter 6

Conclusion

In this work, we developed the novel Sweep method for radiative transfer in astrophysical applications. We discuss both the original implementation of Sweep in the cosmological simulation code Arepo [Springel, 2010b], as well as the standalone radiative transfer postprocessing code Subsweep. Ultimately, we use Subsweep to perform postprocessing of the TNG simulations with the intent to study the era of reionization in the TNG universe.

6.1 Summary

In the methodological part of this work, we describe the Sweep algorithm and its implementation details in Chapters 3 and 4. The Sweep algorithm enables the efficient and highly parallel computation of accurate solutions to the radiative transfer equation. It falls in the category of discrete ordinate methods in which the radiative transfer equation is solved via discretization in every variable, namely time, space, frequency and direction. In this work we focus primarily on Voronoi grids, but the method extends to a variety of other spatial discretizations, such as adaptive mesh refinement. In order to solve the resulting discrete equation, Sweep performs an ad-hoc topological sorting of the dependency graph induced by the neighborhood property of the grid (Section 3.2.5, Section 3.2.4 and Section 3.2.6). To ensure that the induced dependency graph is acyclic, the Sweep procedure performs source iteration in order to account for both scattering and periodic boundary conditions, requiring solutions only for the scattering-less radiative transfer equation (Section 3.2.3, Section 3.2.7).

Sweep is then integrated closely with a radiation chemistry solver, such as the SPRAI/SGChem modules in Arepo [Glover and Jappsen, 2007, Jaura et al., 2018] or a standalone radiation chemistry solver in the case of Subsweep. One of the major advantages of Subsweep over its predecessor is the ability to perform adaptive timestep-

ping in order to avoid the need for a global, low timestep. This is not necessary for the solution to the radiative transfer equation itself (which relaxes instantaneously under the infinite-speed-of-light assumption), but is absolutely crucial in order to allow the combined system of radiation chemistry and radiative transfer to follow the physical trajectory accurately.

We tested both Sweep and Subsweep extensively in this work with a variety of physical tests, such as the R-type expansion of a spherical ionized bubble around a single source in a medium of uniform density (Section 3.3.1, Section 4.3.1), the D-type expansion (Section 3.3.1) and the formation of shadows behind overdense regions (Section 4.3.3 and Section 3.3.2). We also performed a variety of tests in order to verify the behavior of both codes in the presence of periodic boundary conditions (Section 3.3.4 and Section 4.3.2) and its ability to include scattering (Section 3.3.3). Moreover, we studied the computational performance and ability to parallelize of the algorithm (Section 3.3.5, Section 3.3.6) as well as the performance of the substepping method and its ability to cut down on the overall runtime of the solver without sacrificing accuracy in addition to its ability to alleviate the need for additional iterations of the solver in order to account for periodic boundary conditions (Section 4.3.4).

Overall, we find that the Sweep method performs excellently on our test problems, and scales extremely efficiently on a larger number of processors, enabling the incorporation of radiative transfer into large simulations.

In the scientific part of this work (Chapter 5), we studied the era of reionization in the context of the TNG simulation suite, in particular on a lower resolution version of the TNG100 simulations [Marinacci et al., 2018, Naiman et al., 2018, Nelson et al., 2018, 2021, Pillepich et al., 2018, Springel et al., 2018]. To this end, we employ radiative transfer postprocessing using the Subsweep code. In order to incorporate the effects of local feedback, such as supernovae in our postprocessed results, we perform a hybrid approach in which we postprocess each individual output of the TNG simulation for the time between it and the next output and then perform a remapping procedure (Section 5.2.1) in which temperature and ionization fractions are mapped onto the grid given by the next snapshot. In this manner, we incorporate radiative transfer while still following the overall evolution of the TNG box closely. In order to consistently treat radiative transfer, we perform our simulations on a result that is obtained by rerunning the TNG100 model without an ultraviolet background (Section 5.3.1).

Using our method, we find that we are able to reproduce the history of reionization closely, as long as the value of the effective escape fraction is chosen to account for the absorption of the unresolved part of the dense ISM (Discussion in Section 5.2.4, results

in Section 5.3.3). After ensuring that we choose reasonable parameters for our solver (Section 5.3.4), we study the history of reionization and its spatial distribution (Section 5.3.5), the state of the gas during reionization (Section 5.3.6) and compare values of the photoionization rate to constraints (Section 5.3.7) before we briefly discuss the performance of the code in these simulations (Section 5.3.8).

6.2 Outlook

Subsweep was developed with the goal of postprocessing the TNG simulations. While we have shown that our method works and can be applied successfully, we have only discussed results for a low resolution box. In the future, we aim to perform postprocessing of either the second highest resolution of TNG100 (with 910^3 particles) or TNG50 (with 1080^3 particles) or the full resolution of TNG100 (with 1820^3 particles). Higher resolution simulations will enable more precise analysis of reionization, especially with a focus on the properties of the ISM. We have already established that postprocessing of the second highest resolution is feasible and not overly expensive by performing a series of proof-of-concept runs. However, we still need to perform a rerun of the hydrodynamic simulation itself without the ultraviolet background, investigate the required parameters, namely the effective escape fraction and the timestep parameters and perform the full set of simulations.

Eventually, an obvious next step in this work is to perform simulations akin to the TNG simulations but with on-the-fly radiative transfer feedback, focusing on the era of reionization in particular. We believe that the Sweep method is a great candidate as a method for on-the-fly simulations, owing to the fact that the method produces accurate results while requiring relatively few computational resources, which is essential in such extremely costly simulations.

Despite the good performance and parallel scaling of Sweep, radiative transfer is still an extremely expensive computation, such that incorporating it into hydrodynamic simulations is a challenge. In our original implementation of Sweep within Arepo, we found that radiative transfer can often require a large fraction of the overall runtime of the simulation, so that any such simulations could potentially become very costly to run.

We believe that one very important improvement on the path to fully integrated hydrodynamical simulations with Sweep is the sub-timestepping procedure which we discussed in this work. From a theoretical standpoint, this method should integrate perfectly well into a hydrodynamical setup. However, from an implementation standpoint, doing so is a challenge. In most cases, the hydrodynamics solver itself already

has a system for sub-timesteps in place and the introduction of another, independent timestep hierarchy can significantly increase code complexity. Moreover, while the two hierarchies need to be independent to provide their full benefits, they still affect each other, for example through heating caused by radiative feedback lowering the required hydrodynamical timestep in a cell. Any such implementation is not only complex by itself but also needs to be tested rigorously for accuracy and stability.

Despite all of these complications, we believe that fully integrated radiative transfer in hydrodynamical cosmological simulations is a necessary step in order to fully understand the complex processes involved in structure formation and the era of reionization in particular.

Appendix A

Appendices

A.1 Proof that sweep dependency graphs induced by Voronoi grids are acyclic

The solution of a cell c' in a sweep in the direction $\boldsymbol{\Omega}$ depends on fluxes of a neighboring cell c if the normal n of the face connecting c and c' (defined such that it points towards c') fulfills

$$\boldsymbol{n} \cdot \boldsymbol{\Omega} > 0. \tag{A.1}$$

In a Voronoi grid, the face normal $\boldsymbol{n}$ is given by

$$\boldsymbol{n} = \frac{\boldsymbol{p}' - \boldsymbol{p}}{|\boldsymbol{p}' - \boldsymbol{p}|}, \tag{A.2}$$

where p and p' are the Delaunay points corresponding to the respective Voronoi cells.

Now assume that there is a cycle $c_1, c_2, \ldots, c_n, c_{n+1}$ in the dependencies, such that each cell in the cycle depends on the next and c_{n+1} is the same cell as c_1. Now, clearly

$$\boldsymbol{\Omega} \cdot \left(\sum_{i=1}^{n} (\boldsymbol{p}_{i+1} - \boldsymbol{p}_i) \right) = \boldsymbol{\Omega} \cdot \boldsymbol{0}, \tag{A.3}$$

but combining Eq. A.1 with Eq. A.2 yields that each term in the sum in Eq. A.3 is larger than zero, which is a contradiction. Therefore, there are no cycles in the sweep dependencies induced by a Voronoi grid.

A.2 Details of the radiation chemistry

Here, we specify the exact equations used in our radiation chemistry solver described in Section 4.2.9. The photo-heating term $H_{\rm photo}$ is given by

$$H_{\rm photo} = \Gamma \left(1 - e^{-n_{\rm H\,I}\sigma l}\right) (E_{\rm avg} - E_{\rm Ryd}), \tag{A.4}$$

where Γ is the photon rate in $\mathrm{s}^{-1}\ \mathrm{cm}^{-3}$, $E_{\rm avg}$ is the number-averaged photon energy, which for consistency with the BPASS values is chosen as 18.028 eV and $E_{\rm Ryd} = 13.65\,\mathrm{eV}$ is the Rydberg energy. Collisional ionization rates $\beta(T)$ and collisional ionization cooling rates $\zeta(T)$ are given by [Cen, 1992]

$$\beta(T) = 5.85 \times 10^{-11}\,\mathrm{cm}^3\,\mathrm{s}^{-1}\sqrt{T/(1K)}\left(1 + \sqrt{T/\left(1 \times 10^5\,\mathrm{K}\right)}\right)^{-1} e^{-157\,809\,\mathrm{K}/T} \tag{A.5}$$

$$\zeta(T) = 1.27 \times 10^{-21}\,\mathrm{erg}\,\mathrm{cm}^3\,\mathrm{s}^{-1}\sqrt{T/(1K)}\left(1 + \sqrt{T/\left(1 \times 10^5\,\mathrm{K}\right)}\right)^{-1} e^{-157\,809\,\mathrm{K}/T}. \tag{A.6}$$

Collisional excitation rates $\psi(T)$ are given by [Cen, 1992]

$$\psi(T) = 7.5 \times 10^{-19}\,\mathrm{erg}\,\mathrm{cm}^3\,\mathrm{s}^{-1}\sqrt{T/(1K)}\left(1 + \sqrt{T/\left(1 \times 10^5\,\mathrm{K}\right)}\right)^{-1} e^{-118\,348\,\mathrm{K}/T}. \tag{A.7}$$

As discussed in 4.2.9, we use the on-the-spot approximation and therefore only consider case-B recombination. The case-B recombination rates $\alpha(T)$ and recombination cooling rates $\eta(T)$ are given by [Hui and Gnedin, 1997]

$$\alpha(T) = 1.269 \times 10^{-13}\,\mathrm{cm}^3\,\mathrm{s}^{-1} T \frac{\lambda^{1.503}}{\left(1 + (\lambda/0.522)^{0.47}\right)^{1.923}} \tag{A.8}$$

$$\eta(T) = 3.435 \times 10^{-30}\,\mathrm{erg}\,\mathrm{cm}^3\,\mathrm{s}^{-1}\,\mathrm{K}^{-1} T \frac{\lambda^{1.97}}{\left(1 + (\lambda/2.25)^{0.376}\right)^{3.72}}, \tag{A.9}$$

where $\lambda = 315\,614\,\mathrm{K}/T$. The Bremsstrahlung cooling rate coefficient $\Theta(T)$ is given by [Osterbrock and Ferland, 2006]

$$\Theta(T) = 1.42 \times 10^{-27}\,\mathrm{erg}\,\mathrm{cm}^3\,\mathrm{s}^{-1}\sqrt{T/(1\,\mathrm{K})} \tag{A.10}$$

Compton cooling $\bar{\omega}(T)$ is defined as [Haiman et al., 1996]

$$\bar{\omega}(T) = 1.017 \times 10^{-37}\,\mathrm{erg}\,\mathrm{s}^{-1}\left(\frac{2.727}{a}\right)^4\left(T - \frac{2.727}{a}\right), \tag{A.11}$$

where $2.727\,\mathrm{K}/a$ is the CMB temperature and a is the cosmological scale factor. Our chemistry solver also uses the temperature derivatives of the rate coefficients, which are obtained by symbolic differentiation.

The value of the heating and cooling rates are shown in Fig. A.1.

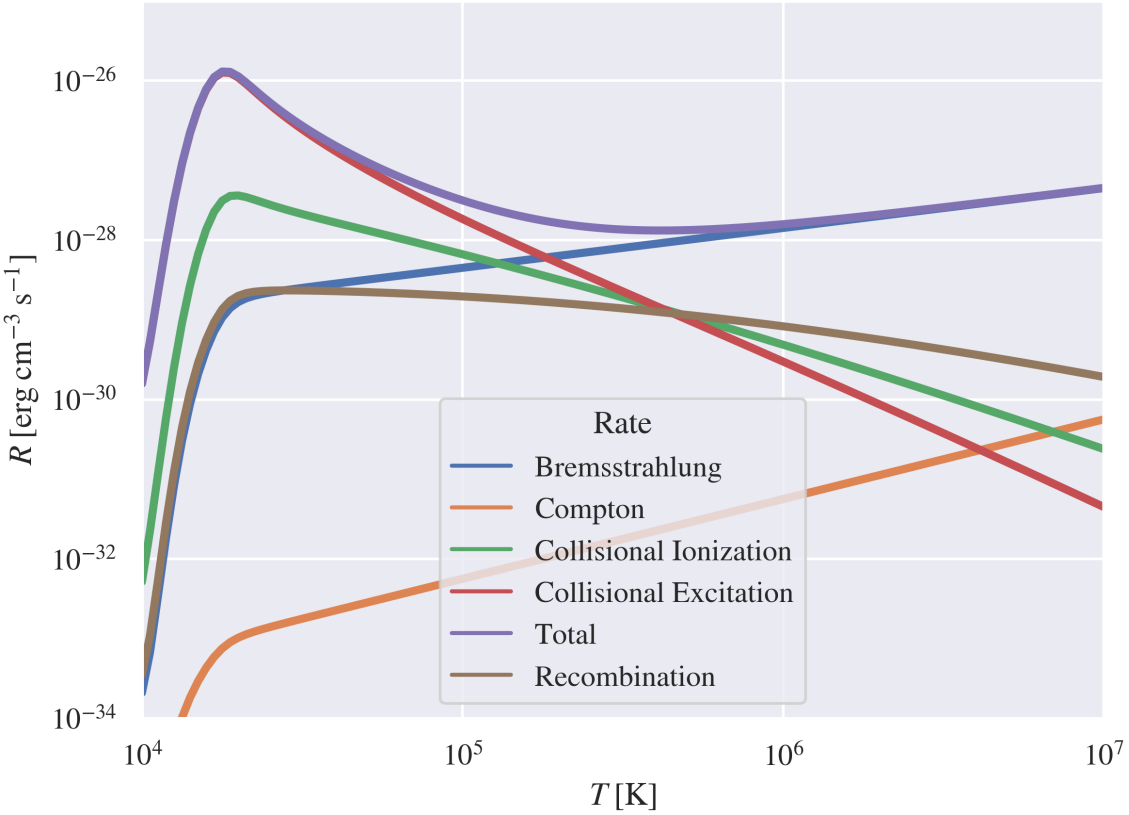

Figure A.1. The value of the different cooling rates as a function of temperature.

Own publications

[Own1] Toni Peter, Ralf S Klessen, Guido Kanschat, Simon C O Glover, and Peter Bastian. The sweep method for radiative transfer in AREPO. *Monthly Notices of the Royal Astronomical Society*, 519(3):4263–4278, January 2023. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/stac3034.

Bibliography

Tom Abel and Benjamin D. Wandelt. Adaptive ray tracing for radiative transfer around point sources. *Monthly Notices of the Royal Astronomical Society*, 330(3): L53–L56, March 2002. ISSN 00358711, 13652966. doi: 10.1046/j.1365-8711.2002.05206.x.

Tom Abel, Michael L. Norman, and Piero Madau. Photon-conserving Radiative Transfer around Point Sources in Multidimensional Numerical Cosmology. *The Astrophysical Journal*, 523(1):66–71, September 1999a. ISSN 0004-637X, 1538-4357. doi: 10.1086/307739.

Tom Abel, Michael L. Norman, and Piero Madau. Photon-conserving Radiative Transfer around Point Sources in Multidimensional Numerical Cosmology. *The Astrophysical Journal*, 523:66–71, September 1999b. ISSN 0004-637X. doi: 10.1086/307739.

Tom Abel, Greg L. Bryan, and Michael L. Norman. The Formation of the First Star in the Universe. *Science*, 295(5552):93–98, January 2002. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.1063991.

Michael P. Adams, Marvin L. Adams, W. Daryl Hawkins, Timmie Smith, Lawrence Rauchwerger, Nancy M. Amato, Teresa S. Bailey, Robert D. Falgout, Adam Kunen, and Peter Brown. Provably Optimal Parallel Transport Sweeps on Semi-Structured Grids. *arXiv:1906.02950 [physics]*, June 2019.

Oscar Agertz, Ben Moore, Joachim Stadel, Doug Potter, Francesco Miniati, Justin Read, Lucio Mayer, Artur Gawryszczak, Andrey Kravtsov, Åke Nordlund, Frazer Pearce, Vicent Quilis, Douglas Rudd, Volker Springel, James Stone, Elizabeth Tasker, Romain Teyssier, James Wadsley, and Rolf Walder. Fundamental differences between SPH and grid methods: Simulating fluids using SPH and grid techniques. *Monthly Notices of the Royal Astronomical Society*, 380(3):963–978, August 2007. ISSN 00358711. doi: 10.1111/j.1365-2966.2007.12183.x.

Kyungjin Ahn, Ilian T. Iliev, Paul R. Shapiro, Garrelt Mellema, Jun Koda, and Yi Mao. Detecting the Rise and Fall of the First Stars by Their Impact on Cosmic

Reionization. *The Astrophysical Journal*, 756:L16, September 2012. ISSN 0004-637X. doi: 10.1088/2041-8205/756/1/L16.

Marcelo A. Alvarez and Tom Abel. Quasar HII regions during cosmic reionization. *Monthly Notices of the Royal Astronomical Society*, 380:L30–L34, September 2007. ISSN 0035-8711. doi: 10.1111/j.1745-3933.2007.00342.x.

C. Baczynski, S. C. O. Glover, and R. S. Klessen. Fervent : Chemistry-coupled, ionizing and non-ionizing radiative feedback in hydrodynamical simulations. *Monthly Notices of the Royal Astronomical Society*, 454(1):380–411, November 2015. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/stv1906.

J. S. Bagla. TreePM: A Code for Cosmological N-Body Simulations. *Journal of Astrophysics and Astronomy*, 23:185–196, December 2002. ISSN 0250-6335. doi: 10.1007/BF02702282.

Randal S. Baker and Kenneth R. Koch. An S_n Algorithm for the Massively Parallel CM-200 Computer. *Nuclear Science and Engineering*, 128(3):312–320, March 1998. ISSN 0029-5639, 1943-748X. doi: 10.13182/NSE98-1.

Randal Scott Baker. An S_N Algorithm for Modern Architectures. Technical Report LA-UR–16-26593, 1312619, August 2016.

R. Barkana and A. Loeb. In the beginning: The first sources of light and the reionization of the universe. *Physics Reports*, 349:125–238, July 2001. ISSN 0370-1573. doi: 10.1016/S0370-1573(01)00019-9.

Josh Barnes and Piet Hut. A hierarchical O(N log N) force-calculation algorithm. *Nature*, 324:446–449, December 1986. ISSN 0028-0836. doi: 10.1038/324446a0.

Andreas Bauer, Volker Springel, Mark Vogelsberger, Shy Genel, Paul Torrey, Debora Sijacki, Dylan Nelson, and Lars Hernquist. Hydrogen Reionization in the Illustris universe. *Monthly Notices of the Royal Astronomical Society*, 453(4):3594–3611, November 2015. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/stv1893.

George D. Becker, James S. Bolton, Piero Madau, Max Pettini, Emma V. Ryan-Weber, and Bram P. Venemans. Evidence of patchy hydrogen reionization from an extreme Lyα trough below redshift six. *Monthly Notices of the Royal Astronomical Society*, 447:3402–3419, March 2015. ISSN 0035-8711. doi: 10.1093/mnras/stu2646.

Robert H. Becker, Xiaohui Fan, Richard L. White, Michael A. Strauss, Vijay K. Narayanan, Robert H. Lupton, James E. Gunn, James Annis, Neta A. Bahcall, J. Brinkmann, A. J. Connolly, István Csabai, Paul C. Czarapata, Mamoru

Doi, Timothy M. Heckman, G. S. Hennessy, Željko Ivezić, G. R. Knapp, Don Q. Lamb, Timothy A. McKay, Jeffrey A. Munn, Thomas Nash, Robert Nichol, Jeffrey R. Pier, Gordon T. Richards, Donald P. Schneider, Chris Stoughton, Alexander S. Szalay, Aniruddha R. Thakar, and D. G. York. Evidence for Reionization at [ITAL][CLC]z[/CLC][/ITAL] $\sim$ 6: Detection of a Gunn-Peterson Trough in a [ITAL][CLC]z[/CLC][/ITAL] = 6.28 Quasar. *The Astronomical Journal*, 122(6): 2850–2857, December 2001. ISSN 00046256. doi: 10.1086/324231.

C. L. Bennett, D. Larson, J. L. Weiland, N. Jarosik, G. Hinshaw, N. Odegard, K. M. Smith, R. S. Hill, B. Gold, M. Halpern, E. Komatsu, M. R. Nolta, L. Page, D. N. Spergel, E. Wollack, J. Dunkley, A. Kogut, M. Limon, S. S. Meyer, G. S. Tucker, and E. L. Wright. Nine-year Wilkinson Microwave Anisotropy Probe (WMAP) Observations: Final Maps and Results. *The Astrophysical Journal Supplement Series*, 208:20, October 2013. ISSN 0067-0049. doi: 10.1088/0067-0049/208/2/20.

Rebekka Bieri, Yohan Dubois, Joakim Rosdahl, Alexander Wagner, Joseph Silk, and Gary A. Mamon. Outflows driven by quasars in high-redshift galaxies with radiation hydrodynamics. *Monthly Notices of the Royal Astronomical Society*, 464(2):1854–1873, January 2017. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/stw2380.

Sarah E. I. Bosman, Frederick B. Davies, George D. Becker, Laura C. Keating, Rebecca L. Davies, Yongda Zhu, Anna-Christina Eilers, Valentina D'Odorico, Fuyan Bian, Manuela Bischetti, Stefano V. Cristiani, Xiaohui Fan, Emanuele P. Farina, Martin G. Haehnelt, Joseph F. Hennawi, Girish Kulkarni, Andrei Mesinger, Romain A. Meyer, Masafusa Onoue, Andrea Pallottini, Yuxiang Qin, Emma Ryan-Weber, Jan-Torge Schindler, Fabian Walter, Feige Wang, and Jinyi Yang. Hydrogen reionization ends by z = 5.3: Lyman-α optical depth measured by the XQR-30 sample. *Monthly Notices of the Royal Astronomical Society*, 514:55–76, July 2022. ISSN 0035-8711. doi: 10.1093/mnras/stac1046.

Alan P. Boss. Flux-limited Diffusion Approximation Models of Giant Planet Formation by Disk Instability. *The Astrophysical Journal*, 677:607–615, April 2008. ISSN 0004-637X. doi: 10.1086/533496.

A. Bowyer. Computing Dirichlet tessellations. *The Computer Journal*, 24(2):162–166, February 1981. ISSN 0010-4620, 1460-2067. doi: 10.1093/comjnl/24.2.162.

Achi Brandt. Multi-Level Adaptive Solutions to Boundary-Value Problems. *Mathematics of Computation*, 31(138):333–390, 1977. ISSN 0025-5718. doi: 10.2307/2006422.

Alexander P. Calverley, George D. Becker, Martin G. Haehnelt, and James S. Bolton. Measurements of the ultraviolet background at $4.6 < z < 6.4$ using the quasar proximity effect. *Monthly Notices of the Royal Astronomical Society*, 412:2543–2562, April 2011. ISSN 0035-8711. doi: 10.1111/j.1365-2966.2010.18072.x.

Renyue Cen. A hydrodynamic approach to cosmology - Methodology. *The Astrophysical Journal Supplement Series*, 78:341, February 1992. ISSN 0067-0049, 1538-4365. doi: 10.1086/191630.

Philip Chang, Shane W Davis, and Yan-Fei Jiang. Time-dependent radiation hydrodynamics on a moving mesh. *Monthly Notices of the Royal Astronomical Society*, 493(4):5397–5407, April 2020. ISSN 0035-8711. doi: 10.1093/mnras/staa573.

Planck Collaboration. Planck 2018 results. VI. Cosmological parameters. *Astronomy & Astrophysics*, 641:A6, September 2020. ISSN 0004-6361, 1432-0746. doi: 10.1051/0004-6361/201833910.

James W. Cooley and John W. Tukey. An algorithm for the machine calculation of complex Fourier series. *Mathematics of Computation*, 19(90):297–301, 1965. ISSN 0025-5718, 1088-6842. doi: 10.1090/S0025-5718-1965-0178586-1.

Tiago Costa, Joakim Rosdahl, Debora Sijacki, and Martin G. Haehnelt. Driving gas shells with radiation pressure on dust in radiation-hydrodynamic simulations. *Monthly Notices of the Royal Astronomical Society*, 473:4197–4219, January 2018. ISSN 0035-8711. doi: 10.1093/mnras/stx2598.

Richard H. Cyburt, Brian D. Fields, Keith A. Olive, and Tsung-Han Yeh. Big bang nucleosynthesis: Present status. *Reviews of Modern Physics*, 88:015004, January 2016. ISSN 0034-6861. doi: 10.1103/RevModPhys.88.015004.

Anson D'Aloisio, Matthew McQuinn, Frederick B. Davies, and Steven R. Furlanetto. Large fluctuations in the high-redshift metagalactic ionizing background. *Monthly Notices of the Royal Astronomical Society*, 473:560–575, January 2018. ISSN 0035-8711. doi: 10.1093/mnras/stx2341.

Olivier Doré, Gil Holder, Marcelo Alvarez, Ilian T. Iliev, Garrelt Mellema, Ue-Li Pen, and Paul R. Shapiro. Signature of patchy reionization in the polarization anisotropy of the CMB. *Physical Review D*, 76(4):043002, August 2007. ISSN 1550-7998, 1550-2368. doi: 10.1103/PhysRevD.76.043002.

B. Dubroca and J. Feugeas. Etude théorique et numérique d'une hiérarchie de modèles aux moments pour le transfert radiatif. *Academie des Sciences Paris Comptes*

Rendus Serie Sciences Mathematiques, 329:915–920, November 1999. ISSN 0764-4450. doi: 10.1016/S0764-4442(00)87499-6.

C. P. Dullemond, A. Juhasz, A. Pohl, F. Sereshti, R. Shetty, T. Peters, B. Commercon, and M. Flock. RADMC-3D: A multi-purpose radiative transfer tool. *Astrophysics Source Code Library*, page ascl:1202.015, February 2012.

Dominika Ďurovčíková, Harley Katz, Sarah E I Bosman, Frederick B Davies, Julien Devriendt, and Adrianne Slyz. Reionization history constraints from neural network based predictions of high-redshift quasar continua. *Monthly Notices of the Royal Astronomical Society*, 493(3):4256–4275, April 2020. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/staa505.

H. Edelsbrunner and N. R. Shah. Incremental topological flipping works for regular triangulations. *Algorithmica*, 15(3):223–241, March 1996. ISSN 0178-4617, 1432-0541. doi: 10.1007/BF01975867.

G. Efstathiou, M. Davis, S. D. M. White, and C. S. Frenk. Numerical techniques for large cosmological N-body simulations. *The Astrophysical Journal Supplement Series*, 57:241, February 1985. ISSN 0067-0049, 1538-4365. doi: 10.1086/191003.

Daniel J. Eisenstein and Wayne Hu. Baryonic Features in the Matter Transfer Function. *The Astrophysical Journal*, 496(2):605–614, April 1998. ISSN 0004-637X, 1538-4357. doi: 10.1086/305424.

J. J. Eldridge, E. R. Stanway, L. Xiao, L. A. S. McClelland, G. Taylor, M. Ng, S. M. L. Greis, and J. C. Bray. Binary Population and Spectral Synthesis Version 2.1: Construction, Observational Verification, and New Results. *Publications of the Astronomical Society of Australia*, 34:e058, November 2017. ISSN 1323-3580. doi: 10.1017/pasa.2017.51.

Xiaohui Fan, Vijay K. Narayanan, Michael A. Strauss, Richard L. White, Robert H. Becker, Laura Pentericci, and Hans-Walter Rix. Evolution of the Ionizing Background and the Epoch of Reionization from the Spectra of [ITAL][CLC]z[/CLC][/ITAL] $\sim$ 6 Quasars. *The Astronomical Journal*, 123(3):1247–1257, March 2002. ISSN 00046256, 15383881. doi: 10.1086/339030.

Xiaohui Fan, C.L. Carilli, and B. Keating. Observational Constraints on Cosmic Reionization. *Annual Review of Astronomy and Astrophysics*, 44(1):415–462, September 2006a. ISSN 0066-4146, 1545-4282. doi: 10.1146/annurev.astro.44.051905.092514.

Xiaohui Fan, Michael A. Strauss, Robert H. Becker, Richard L. White, James E. Gunn, Gillian R. Knapp, Gordon T. Richards, Donald P. Schneider, J. Brinkmann, and Masataka Fukugita. Constraining the Evolution of the Ionizing Background and the Epoch of Reionization with z ˜ 6 Quasars. II. A Sample of 19 Quasars. *The Astronomical Journal*, 132(1):117–136, July 2006b. ISSN 0004-6256, 1538-3881. doi: 10.1086/504836.

Claude-André Faucher-Giguère, Adam Lidz, Matias Zaldarriaga, and Lars Hernquist. A New Calculation of the Ionizing Background Spectrum and the Effects of He II Reionization. *The Astrophysical Journal*, 703(2):1416–1443, October 2009. ISSN 0004-637X, 1538-4357. doi: 10.1088/0004-637X/703/2/1416.

Steven L. Finkelstein, Anson D'Aloisio, Jan-Pieter Paardekooper, Russell Ryan Jr, Peter Behroozi, Kristian Finlator, Rachael Livermore, Phoebe R. Upton Sanderbeck, Claudio Dalla Vecchia, and Sadegh Khochfar. Conditions for Reionizing the Universe with a Low Galaxy Ionizing Photon Escape Fraction. *The Astrophysical Journal*, 879(1):36, July 2019. ISSN 0004-637X. doi: 10.3847/1538-4357/ab1ea8.

Steven R. Furlanetto, Matias Zaldarriaga, and Lars Hernquist. The Growth of H II Regions During Reionization. *The Astrophysical Journal*, 613:1–15, September 2004. ISSN 0004-637X. doi: 10.1086/423025.

S. C. O. Glover and A.-K. Jappsen. Star Formation at Very Low Metallicity. I. Chemistry and Cooling at Low Densities. *The Astrophysical Journal*, 666(1):1–19, September 2007. ISSN 0004-637X, 1538-4357. doi: 10.1086/519445.

Nickolay Y. Gnedin and Tom Abel. Multi–dimensional Cosmological Radiative Transfer with a Variable Eddington Tensor Formalism. *New Astronomy*, 6(7):437–455, October 2001a. ISSN 13841076. doi: 10.1016/S1384-1076(01)00068-9.

Nickolay Y. Gnedin and Tom Abel. Multi-dimensional cosmological radiative transfer with a Variable Eddington Tensor formalism. *New Astronomy*, 6(7):437–455, October 2001b. ISSN 13841076. doi: 10.1016/S1384-1076(01)00068-9.

James E. Gunn and Bruce A. Peterson. On the Density of Neutral Hydrogen in Intergalactic Space. *The Astrophysical Journal*, 142:1633, November 1965. ISSN 0004-637X, 1538-4357. doi: 10.1086/148444.

Zoltan Haiman, Anne A. Thoul, and Abraham Loeb. Cosmological Formation of Low-Mass Objects. *The Astrophysical Journal*, 464:523, June 1996. ISSN 0004-637X. doi: 10.1086/177343.

Tilman Hartwig, Simon C. O. Glover, Ralf S. Klessen, Muhammad A. Latif, and Marta Volonteri. How an improved implementation of H $_2$ self-shielding influences the formation of massive stars and black holes. *Monthly Notices of the Royal Astronomical Society*, 452(2):1233–1244, September 2015. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/stv1368.

John C. Hayes and Michael L. Norman. Beyond Flux-limited Diffusion: Parallel Algorithms for Multidimensional Radiation Hydrodynamics. *The Astrophysical Journal Supplement Series*, 147:197–220, July 2003. ISSN 0067-0049. doi: 10.1086/374658.

Matthew Hayes, Daniel Schaerer, Göran Östlin, J. Miguel Mas-Hesse, Hakim Atek, and Daniel Kunth. ON THE REDSHIFT EVOLUTION OF THE Lyα ESCAPE FRACTION AND THE DUST CONTENT OF GALAXIES. *The Astrophysical Journal*, 730(1):8, February 2011. ISSN 0004-637X. doi: 10.1088/0004-637X/730/1/8.

Lars Hernquist, Piet Hut, and Jun Makino. Discreteness Noise versus Force Errors in N-Body Simulations. *The Astrophysical Journal*, 402:L85, January 1993. ISSN 0004-637X. doi: 10.1086/186706.

R.W Hockney and J.W Eastwood. *Computer Simulation Using Particles*. CRC Press, 0 edition, March 2021. ISBN 978-0-367-80693-4. doi: 10.1201/9780367806934.

Philip F. Hopkins, Gordon T. Richards, and Lars Hernquist. An Observational Determination of the Bolometric Quasar Luminosity Function. *The Astrophysical Journal*, 654:731–753, January 2007. ISSN 0004-637X. doi: 10.1086/509629.

Edwin Hubble. A Relation between Distance and Radial Velocity among Extra-Galactic Nebulae. *Proceedings of the National Academy of Science*, 15:168–173, March 1929. ISSN 0027-8424. doi: 10.1073/pnas.15.3.168.

Lam Hui and Nickolay Y. Gnedin. Equation of state of the photoionized intergalactic medium. *Monthly Notices of the Royal Astronomical Society*, 292:27–42, November 1997. ISSN 0035-8711. doi: 10.1093/mnras/292.1.27.

Ilian T. Iliev, Benedetta Ciardi, Marcelo A. Alvarez, Antonella Maselli, Andrea Ferrara, Nickolay Y. Gnedin, Garrelt Mellema, Taishi Nakamoto, Michael L. Norman, Alexei O. Razoumov, Erik-Jan Rijkhorst, Jelle Ritzerveld, Paul R. Shapiro, Hajime Susa, Masayuki Umemura, and Daniel J. Whalen. Cosmological Radiative Transfer Codes Comparison Project I: The Static Density Field Tests. *Monthly Notices of the Royal Astronomical Society*, 371(3):1057–1086, September 2006. ISSN 00358711, 13652966. doi: 10.1111/j.1365-2966.2006.10775.x.

Ilian T. Iliev, Paul R. Shapiro, Garrelt Mellema, Ue-Li Pen, Patrick McDonald, and J. Richard Bond. Simulating Reionization: Character and Observability. In *FIRST STARS III: First Stars II Conference*, pages 442–444, Santa Fe (New Mexico), 2008. doi: 10.1063/1.2905659.

Ilian T. Iliev, Daniel Whalen, Garrelt Mellema, Kyungjin Ahn, Sunghye Baek, Nickolay Y. Gnedin, Andrey V. Kravtsov, Michael Norman, Milan Raicevic, Daniel R. Reynolds, Daisuke Sato, Paul R. Shapiro, Benoit Semelin, Joseph Smidt, Hajime Susa, Tom Theuns, and Masayuki Umemura. Cosmological Radiative Transfer Comparison Project II: The Radiation-Hydrodynamic Tests. *Monthly Notices of the Royal Astronomical Society*, 400(3):1283–1316, December 2009. ISSN 00358711, 13652966. doi: 10.1111/j.1365-2966.2009.15558.x.

Ilian T. Iliev, Garrelt Mellema, Kyungjin Ahn, Paul R. Shapiro, Yi Mao, and Ue-Li Pen. Simulating cosmic reionization: How large a volume is large enough? *Monthly Notices of the Royal Astronomical Society*, 439(1):725–743, March 2014. ISSN 1365-2966, 0035-8711. doi: 10.1093/mnras/stt2497.

O Jaura, S C O Glover, R S Klessen, and J-P Paardekooper. SPRAI: Coupling of radiative feedback and primordial chemistry in moving mesh hydrodynamics. *Monthly Notices of the Royal Astronomical Society*, 475(2):2822–2834, April 2018. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/stx3356.

Ondrej Jaura, Mattis Magg, Simon C O Glover, and Ralf S Klessen. SPRAI-II: Multifrequency radiative transfer for variable gas densities. *Monthly Notices of the Royal Astronomical Society*, 499(3):3594–3609, October 2020. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/staa3054.

A. Jenkins, C. S. Frenk, S. D. M. White, J. M. Colberg, S. Cole, A. E. Evrard, H. M. P. Couchman, and N. Yoshida. The mass function of dark matter haloes. *Monthly Notices of the Royal Astronomical Society*, 321:372–384, February 2001. ISSN 0035-8711. doi: 10.1046/j.1365-8711.2001.04029.x.

Yan-Fei Jiang, James M. Stone, and Shane W. Davis. An Algorithm for Radiation Magnetohydrodynamics Based on Solving the Time-dependent Transfer Equation. *The Astrophysical Journal Supplement Series*, 213:7, July 2014. ISSN 0067-0049. doi: 10.1088/0067-0049/213/1/7.

Rahul Kannan, Mark Vogelsberger, Federico Marinacci, Ryan McKinnon, Rüdiger Pakmor, and Volker Springel. Arepo-RT: Radiation hydrodynamics on a moving

mesh. *Monthly Notices of the Royal Astronomical Society*, 485(1):117–149, May 2019. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/stz287.

Jeong-Gyu Kim, Woong-Tae Kim, and Eve C. Ostriker. Modeling UV Radiation Feedback from Massive Stars. II. Dispersal of Star-forming Giant Molecular Clouds by Photoionization and Radiation Pressure. *The Astrophysical Journal*, 859:68, May 2018. ISSN 0004-637X. doi: 10.3847/1538-4357/aabe27.

K R Koch, R S Baker, and R E Alcouffe. Solution of the first-order form of the 3-D discrete ordinates equations on a massively parallel machine. *Transactions of the American Nuclear Society*, 65:198, January 1991.

C. J. H. Kruip, J.-P. Paardekooper, B. J. F. Clauwens, and V. Icke. Mathematical properties of the *SimpleX* algorithm. *Astronomy and Astrophysics*, 515:A78, June 2010. ISSN 0004-6361, 1432-0746. doi: 10.1051/0004-6361/200913439.

Mark R. Krumholz, Richard I. Klein, and Christopher F. McKee. Radiation-Hydrodynamic Simulations of Collapse and Fragmentation in Massive Protostellar Cores. *The Astrophysical Journal*, 656:959–979, February 2007. ISSN 0004-637X. doi: 10.1086/510664.

Michael Kuhlen and Claude-André Faucher-Giguère. Concordance models of reionization: Implications for faint galaxies and escape fraction evolution. *Monthly Notices of the Royal Astronomical Society*, 423:862–876, June 2012. ISSN 0035-8711. doi: 10.1111/j.1365-2966.2012.20924.x.

G. Lemaître. Un Univers homogène de masse constante et de rayon croissant rendant compte de la vitesse radiale des nébuleuses extra-galactiques. *Annales de la Société Scientifique de Bruxelles*, 47:49–59, January 1927.

C. D. Levermore and G. C. Pomraning. A flux-limited diffusion theory. *The Astrophysical Journal*, 248:321, August 1981. ISSN 0004-637X, 1538-4357. doi: 10.1086/159157.

Adam Lidz, Matthew McQuinn, Matias Zaldarriaga, Lars Hernquist, and Suvendra Dutta. Quasar Proximity Zones and Patchy Reionization. *The Astrophysical Journal*, 670(1):39–59, November 2007. ISSN 0004-637X, 1538-4357. doi: 10.1086/521974.

Abraham Loeb and Rennan Barkana. The Reionization of the Universe by the First Stars and Quasars. *Annual Review of Astronomy and Astrophysics*, 39(1):19–66, September 2001. ISSN 0066-4146, 1545-4282. doi: 10.1146/annurev.astro.39.1.19.

Jose Pablo Lucero Lorca. *Multilevel Schwarz Methods for Multigroup Radiation Transport Problems*. PhD thesis, Heidelberg University, 2018.

Federico Marinacci, Mark Vogelsberger, Rüdiger Pakmor, Paul Torrey, Volker Springel, Lars Hernquist, Dylan Nelson, Rainer Weinberger, Annalisa Pillepich, Jill Naiman, and Shy Genel. First results from the IllustrisTNG simulations: Radio haloes and magnetic fields. *Monthly Notices of the Royal Astronomical Society*, August 2018. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/sty2206.

Ian D. McGreer, Andrei Mesinger, and Valentina D'Odorico. Model-independent evidence in favour of an end to reionization by z $\approx$ 6. *Monthly Notices of the Royal Astronomical Society*, 447(1):499–505, February 2015. ISSN 1365-2966, 0035-8711. doi: 10.1093/mnras/stu2449.

Andrei Mesinger and Steven Furlanetto. Efficient Simulations of Early Structure Formation and Reionization. *The Astrophysical Journal*, 669:663–675, November 2007. ISSN 0004-637X. doi: 10.1086/521806.

Dimitri Mihalas and Barbara Weibel-Mihalas. *Foundations of Radiation Hydrodynamics*. Dover, Mineola, N.Y, 1999. ISBN 978-0-486-40925-2.

Jill P. Naiman, Annalisa Pillepich, Volker Springel, Enrico Ramirez-Ruiz, Paul Torrey, Mark Vogelsberger, Rüdiger Pakmor, Dylan Nelson, Federico Marinacci, Lars Hernquist, Rainer Weinberger, and Shy Genel. First results from the IllustrisTNG simulations: A tale of two elements – chemical evolution of magnesium and europium. *Monthly Notices of the Royal Astronomical Society*, 477(1):1206–1224, June 2018. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/sty618.

Julio F. Navarro, Carlos S. Frenk, and Simon D. M. White. A Universal Density Profile from Hierarchical Clustering. *The Astrophysical Journal*, 490:493–508, December 1997. ISSN 0004-637X. doi: 10.1086/304888.

Dylan Nelson, Annalisa Pillepich, Volker Springel, Rainer Weinberger, Lars Hernquist, Ruediger Pakmor, Shy Genel, Paul Torrey, Mark Vogelsberger, Guinevere Kauffmann, Federico Marinacci, and Jill Naiman. First results from the IllustrisTNG simulations: The galaxy color bimodality. *Monthly Notices of the Royal Astronomical Society*, 475(1):624–647, March 2018. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/stx3040.

Dylan Nelson, Annalisa Pillepich, Volker Springel, Ruediger Pakmor, Rainer Weinberger, Shy Genel, Paul Torrey, Mark Vogelsberger, Federico Marinacci, and Lars

Hernquist. First Results from the TNG50 Simulation: Galactic outflows driven by supernovae and black hole feedback. *Monthly Notices of the Royal Astronomical Society*, 490(3):3234–3261, December 2019. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/stz2306.

Dylan Nelson, Volker Springel, Annalisa Pillepich, Vicente Rodriguez-Gomez, Paul Torrey, Shy Genel, Mark Vogelsberger, Ruediger Pakmor, Federico Marinacci, Rainer Weinberger, Luke Kelley, Mark Lovell, Benedikt Diemer, and Lars Hernquist. The IllustrisTNG Simulations: Public Data Release, January 2021.

Ulrich M. Noebauer and Stuart A. Sim. Monte Carlo Radiative Transfer. *Living Reviews in Computational Astrophysics*, 5(1):1, December 2019. ISSN 2367-3621, 2365-0524. doi: 10.1007/s41115-019-0004-9.

Donald E. Osterbrock and Gary J. Ferland. *Astrophysics of Gaseous Nebulae and Active Galactic Nuclei*. January 2006.

Masami Ouchi, Kazuhiro Shimasaku, Hisanori Furusawa, Tomoki Saito, Makiko Yoshida, Masayuki Akiyama, Yoshiaki Ono, Toru Yamada, Kazuaki Ota, Nobunari Kashikawa, Masanori Iye, Tadayuki Kodama, Sadanori Okamura, Chris Simpson, and Michitoshi Yoshida. STATISTICS OF 207 Lyα EMITTERS AT A REDSHIFT NEAR 7: CONSTRAINTS ON REIONIZATION AND GALAXY FORMATION MODELS. *The Astrophysical Journal*, 723(1):869–894, November 2010. ISSN 0004-637X, 1538-4357. doi: 10.1088/0004-637X/723/1/869.

S. Oxley and M. M. Woolfson. Smoothed particle hydrodynamics with radiation transfer. *Monthly Notices of the Royal Astronomical Society*, 343(3):900–912, August 2003. ISSN 0035-8711. doi: 10.1046/j.1365-8711.2003.06751.x.

J.-P. Paardekooper, C. J. H. Kruip, and V. Icke. SimpleX2: Radiative transfer on an unstructured, dynamic grid. *Astronomy and Astrophysics*, 515:A79, June 2010. ISSN 0004-6361, 1432-0746. doi: 10.1051/0004-6361/200913821.

J. P. Paardekooper, S. Khochfar, and C. V. Dalla. The first billion years project: Proto-galaxies reionizing the universe. *Monthly Notices of the Royal Astronomical Society*, 429:L94–L98, February 2013. ISSN 0035-8711. doi: 10.1093/mnrasl/sls032.

Shawn D Pautz. An Algorithm for Parallel S_n Sweeps on Unstructured Meshes. *Nuclear Science and Engineering*, 140(2):111–136, February 2002. ISSN 0029-5639, 1943-748X. doi: 10.13182/NSE02-1.

P. J. E. Peebles, R. A. Daly, and R. Juszkiewicz. The Masses of Rich Clusters of Galaxies as a Test of the Biased Cold Dark Matter Theory. *The Astrophysical Journal*, 347:563, December 1989. ISSN 0004-637X. doi: 10.1086/168149.

S. Perlmutter, G. Aldering, G. Goldhaber, R. A. Knop, P. Nugent, P. G. Castro, S. Deustua, S. Fabbro, A. Goobar, D. E. Groom, I. M. Hook, A. G. Kim, M. Y. Kim, J. C. Lee, N. J. Nunes, R. Pain, C. R. Pennypacker, R. Quimby, C. Lidman, R. S. Ellis, M. Irwin, R. G. McMahon, P. Ruiz-Lapuente, N. Walton, B. Schaefer, B. J. Boyle, A. V. Filippenko, T. Matheson, A. S. Fruchter, N. Panagia, H. J. M. Newberg, W. J. Couch, and The Supernova Cosmology Project. Measurements of Ω and Λ from 42 High-Redshift Supernovae. *The Astrophysical Journal*, 517(2): 565–586, June 1999. ISSN 0004-637X, 1538-4357. doi: 10.1086/307221.

Toni Peter, Ralf S Klessen, Guido Kanschat, Simon C O Glover, and Peter Bastian. The sweep method for radiative transfer in AREPO. *Monthly Notices of the Royal Astronomical Society*, 519(3):4263–4278, January 2023. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/stac3034.

Thomas Peters, Robi Banerjee, Ralf S. Klessen, Mordecai-Mark Mac Low, Roberto Galván-Madrid, and Eric R. Keto. H II Regions: Witnesses to Massive Star Formation. *The Astrophysical Journal*, 711:1017–1028, March 2010. ISSN 0004-637X. doi: 10.1088/0004-637X/711/2/1017.

Margarita Petkova and Volker Springel. A novel approach for accurate radiative transfer in cosmological hydrodynamic simulations: Radiative transfer in cosmological simulations. *Monthly Notices of the Royal Astronomical Society*, 415(4):3731–3749, August 2011. ISSN 00358711. doi: 10.1111/j.1365-2966.2011.18986.x.

Annalisa Pillepich, Dylan Nelson, Lars Hernquist, Volker Springel, Rüdiger Pakmor, Paul Torrey, Rainer Weinberger, Shy Genel, Jill Naiman, Federico Marinacci, and Mark Vogelsberger. First results from the IllustrisTNG simulations: The stellar mass content of groups and clusters of galaxies. *Monthly Notices of the Royal Astronomical Society*, 475(1):648–675, March 2018. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/stx3112.

Annalisa Pillepich, Dylan Nelson, Volker Springel, Ruediger Pakmor, Paul Torrey, Rainer Weinberger, Mark Vogelsberger, Federico Marinacci, Shy Genel, Arjen van der Wel, and Lars Hernquist. First Results from the TNG50 Simulation: The evolution of stellar and gaseous disks across cosmic time. *Monthly Notices of the*

Royal Astronomical Society, 490(3):3196–3233, December 2019. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/stz2338.

Planck Collaboration. *Planck* 2013 results. XVI. Cosmological parameters. *Astronomy & Astrophysics*, 571:A16, November 2014. ISSN 0004-6361, 1432-0746. doi: 10.1051/0004-6361/201321591.

Planck Collaboration, P. A. R. Ade, N. Aghanim, M. Arnaud, M. Ashdown, J. Aumont, C. Baccigalupi, A. J. Banday, R. B. Barreiro, J. G. Bartlett, N. Bartolo, E. Battaner, R. Battye, K. Benabed, A. Benoît, A. Benoit-Lévy, J. P. Bernard, M. Bersanelli, P. Bielewicz, J. J. Bock, A. Bonaldi, L. Bonavera, J. R. Bond, J. Borrill, F. R. Bouchet, F. Boulanger, M. Bucher, C. Burigana, R. C. Butler, E. Calabrese, J. F. Cardoso, A. Catalano, A. Challinor, A. Chamballu, R. R. Chary, H. C. Chiang, J. Chluba, P. R. Christensen, S. Church, D. L. Clements, S. Colombi, L. P. L. Colombo, C. Combet, A. Coulais, B. P. Crill, A. Curto, F. Cuttaia, L. Danese, R. D. Davies, R. J. Davis, P. de Bernardis, A. de Rosa, G. de Zotti, J. Delabrouille, F. X. Désert, E. Di Valentino, C. Dickinson, J. M. Diego, K. Dolag, H. Dole, S. Donzelli, O. Doré, M. Douspis, A. Ducout, J. Dunkley, X. Dupac, G. Efstathiou, F. Elsner, T. A. Enßlin, H. K. Eriksen, M. Farhang, J. Fergusson, F. Finelli, O. Forni, M. Frailis, A. A. Fraisse, E. Franceschi, A. Frejsel, S. Galeotta, S. Galli, K. Ganga, C. Gauthier, M. Gerbino, T. Ghosh, M. Giard, Y. Giraud-Héraud, E. Giusarma, E. Gjerløw, J. González-Nuevo, K. M. Górski, S. Gratton, A. Gregorio, A. Gruppuso, J. E. Gudmundsson, J. Hamann, F. K. Hansen, D. Hanson, D. L. Harrison, G. Helou, S. Henrot-Versillé, C. Hernández-Monteagudo, D. Herranz, S. R. Hildebrandt, E. Hivon, M. Hobson, W. A. Holmes, A. Hornstrup, W. Hovest, Z. Huang, K. M. Huffenberger, G. Hurier, A. H. Jaffe, T. R. Jaffe, W. C. Jones, M. Juvela, E. Keihänen, R. Keskitalo, T. S. Kisner, R. Kneissl, J. Knoche, L. Knox, M. Kunz, H. Kurki-Suonio, G. Lagache, A. Lähteenmäki, J. M. Lamarre, A. Lasenby, M. Lattanzi, C. R. Lawrence, J. P. Leahy, R. Leonardi, J. Lesgourgues, F. Levrier, A. Lewis, M. Liguori, P. B. Lilje, M. Linden-Vørnle, M. López-Caniego, P. M. Lubin, J. F. Macías-Pérez, G. Maggio, D. Maino, N. Mandolesi, A. Mangilli, A. Marchini, M. Maris, P. G. Martin, M. Martinelli, E. Martínez-González, S. Masi, S. Matarrese, P. McGehee, P. R. Meinhold, A. Melchiorri, J. B. Melin, L. Mendes, A. Mennella, M. Migliaccio, M. Millea, S. Mitra, M. A. Miville-Deschênes, A. Moneti, L. Montier, G. Morgante, D. Mortlock, A. Moss, D. Munshi, J. A. Murphy, P. Naselsky, F. Nati, P. Natoli, C. B. Netterfield, H. U. Nørgaard-Nielsen, F. Noviello, D. Novikov, I. Novikov, C. A. Oxborrow, F. Paci, L. Pagano, F. Pajot, R. Paladini, D. Paoletti, B. Partridge, F. Pasian, G. Patanchon, T. J. Pear-

son, O. Perdereau, L. Perotto, F. Perrotta, V. Pettorino, F. Piacentini, M. Piat, E. Pierpaoli, D. Pietrobon, S. Plaszczynski, E. Pointecouteau, G. Polenta, L. Popa, G. W. Pratt, G. Prézeau, S. Prunet, J. L. Puget, J. P. Rachen, W. T. Reach, R. Rebolo, M. Reinecke, M. Remazeilles, C. Renault, A. Renzi, I. Ristorcelli, G. Rocha, C. Rosset, M. Rossetti, G. Roudier, B. Rouillé d'Orfeuil, M. Rowan-Robinson, J. A. Rubiño-Martín, B. Rusholme, N. Said, V. Salvatelli, L. Salvati, M. Sandri, D. Santos, M. Savelainen, G. Savini, D. Scott, M. D. Seiffert, P. Serra, E. P. S. Shellard, L. D. Spencer, M. Spinelli, V. Stolyarov, R. Stompor, R. Sudiwala, R. Sunyaev, D. Sutton, A. S. Suur-Uski, J. F. Sygnet, J. A. Tauber, L. Terenzi, L. Toffolatti, M. Tomasi, M. Tristram, T. Trombetti, M. Tucci, J. Tuovinen, M. Türler, G. Umana, L. Valenziano, J. Valiviita, F. Van Tent, P. Vielva, F. Villa, L. A. Wade, B. D. Wandelt, I. K. Wehus, M. White, S. D. M. White, A. Wilkinson, D. Yvon, A. Zacchei, and A. Zonca. Planck 2015 results. XIII. Cosmological parameters. *Astronomy and Astrophysics*, 594:A13, September 2016. ISSN 0004-6361. doi: 10.1051/0004-6361/201525830.

Jonathan R. Pritchard and Abraham Loeb. 21 cm cosmology in the 21st century. *Reports on Progress in Physics*, 75:086901, August 2012. ISSN 0034-4885. doi: 10.1088/0034-4885/75/8/086901.

Alexei O. Razoumov, Michael L. Norman, Tom Abel, and Douglas Scott. Cosmological Hydrogen Reionization with Three-dimensional Radiative Transfer. *The Astrophysical Journal*, 572(2):695–704, June 2002. ISSN 0004-637X, 1538-4357. doi: 10.1086/340451.

J-F Ripoll, B Dubroca, and G Duffa. Modelling radiative mean absorption coefficients. *Combustion Theory and Modelling*, 5(3):261–274, September 2001. ISSN 1364-7830, 1741-3559. doi: 10.1088/1364-7830/5/3/301.

Jelle Ritzerveld and Vincent Icke. Transport on adaptive random lattices. *Physical Review E*, 74(2):026704, August 2006. ISSN 1539-3755, 1550-2376. doi: 10.1103/PhysRevE.74.026704.

Brant E. Robertson, Richard S. Ellis, Steven R. Furlanetto, and James S. Dunlop. Cosmic Reionization and Early Star-Forming Galaxies: A Joint Analysis of new Constraints from Planck and the Hubble Space Telescope. *The Astrophysical Journal Letters*, 802(2):L19, April 2015. ISSN 2041-8205. doi: 10.1088/2041-8205/802/2/L19.

J. Rosdahl and R. Teyssier. A scheme for radiation pressure and photon diffusion with

the M1 closure in ramses-rt. *Monthly Notices of the Royal Astronomical Society*, 449 (4):4380–4403, June 2015. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/stv567.

J. Rosdahl, J. Blaizot, D. Aubert, T. Stranex, and R. Teyssier. Ramses-rt: Radiation hydrodynamics in the cosmological context. *Monthly Notices of the Royal Astronomical Society*, 436(3):2188–2231, December 2013. ISSN 1365-2966, 0035-8711. doi: 10.1093/mnras/stt1722.

George B. Rybicki and Alan P. Lightman. *Radiative Processes in Astrophysics*. Wiley-VCH Verlag GmbH & Co. KGaA, Weinheim, Germany, May 1985. ISBN 978-3-527-61817-0 978-0-471-82759-7. doi: 10.1002/9783527618170.

Youcef Saad and Martin H. Schultz. GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, July 1986. ISSN 0196-5204, 2168-3417. doi: 10.1137/0907058.

Mario G. Santos, Asantha Cooray, Zoltan Haiman, Lloyd Knox, and Chung-Pei Ma. Small-Scale Cosmic Microwave Background Temperature and Polarization Anisotropies Due to Patchy Reionization. *The Astrophysical Journal*, 598(2):756–766, December 2003. ISSN 0004-637X, 1538-4357. doi: 10.1086/378772.

Anna T P Schauer, Simon C O Glover, Ralf S Klessen, and Daniel Ceverino. The influence of streaming velocities on the formation of the first stars. *Monthly Notices of the Royal Astronomical Society*, 484(3):3510–3521, April 2019. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/stz013.

Joop Schaye, Tom Theuns, Michael Rauch, George Efstathiou, and Wallace L. W. Sargent. The thermal history of the intergalactic medium*. *Monthly Notices of the Royal Astronomical Society*, 318:817–826, November 2000. ISSN 0035-8711. doi: 10.1046/j.1365-8711.2000.03815.x.

Joop Schaye, Robert A. Crain, Richard G. Bower, Michelle Furlong, Matthieu Schaller, Tom Theuns, Claudio Dalla Vecchia, Carlos S. Frenk, I. G. McCarthy, John C. Helly, Adrian Jenkins, Y. M. Rosas-Guevara, Simon D. M. White, Maarten Baes, C. M. Booth, Peter Camps, Julio F. Navarro, Yan Qu, Alireza Rahmati, Till Sawala, Peter A. Thomas, and James Trayford. The EAGLE project: Simulating the evolution and assembly of galaxies and their environments. *Monthly Notices of the Royal Astronomical Society*, 446:521–554, January 2015. ISSN 0035-8711. doi: 10.1093/mnras/stu2058.

Min-Su Shin, Hy Trac, and Renyue Cen. Cosmological HII Bubble Growth During Reionization. *The Astrophysical Journal*, 681(2):756–770, July 2008. ISSN 0004-637X, 1538-4357. doi: 10.1086/588247.

Aaron Smith, Rahul Kannan, Benny T. H. Tsang, Mark Vogelsberger, and Rüdiger Pakmor. AREPO-MCRT: Monte Carlo Radiation Hydrodynamics on a Moving Mesh. *The Astrophysical Journal*, 905:27, December 2020. ISSN 0004-637X. doi: 10.3847/1538-4357/abc47e.

Lyman Spitzer. *Physical Processes in the Interstellar Medium*. Wiley, 1 edition, 1978. ISBN 978-0-471-29335-4 978-3-527-61772-2. doi: 10.1002/9783527617722.

Volker Springel. Moving-mesh hydrodynamics with the AREPO code. *Proceedings of the International Astronomical Union*, 6(S270):203–206, May 2010a. ISSN 1743-9213, 1743-9221. doi: 10.1017/S1743921311000378.

Volker Springel. E pur si muove: Galilean-invariant cosmological hydrodynamical simulations on a moving mesh. *Monthly Notices of the Royal Astronomical Society*, 401(2):791–851, January 2010b. ISSN 00358711, 13652966. doi: 10.1111/j.1365-2966.2009.15715.x.

Volker Springel and Lars Hernquist. Cosmological smoothed particle hydrodynamics simulations: A hybrid multiphase model for star formation. *Monthly Notices of the Royal Astronomical Society*, 339:289–311, February 2003. ISSN 0035-8711. doi: 10.1046/j.1365-8711.2003.06206.x.

Volker Springel, Rüdiger Pakmor, Annalisa Pillepich, Rainer Weinberger, Dylan Nelson, Lars Hernquist, Mark Vogelsberger, Shy Genel, Paul Torrey, Federico Marinacci, and Jill Naiman. First results from the IllustrisTNG simulations: Matter and galaxy clustering. *Monthly Notices of the Royal Astronomical Society*, 475(1): 676–698, March 2018. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/stx3304.

Joachim Gerhard Stadel. *Cosmological N-body Simulations and Their Analysis*. PhD thesis, University of Washington, United States, January 2001.

Bengt Strömgren. The Physical State of Interstellar Hydrogen. *The Astrophysical Journal*, 89:526, May 1939. ISSN 0004-637X, 1538-4357. doi: 10.1086/144074.

Elizabeth J. Tasker, Riccardo Brunino, Nigel L. Mitchell, Dolf Michielsen, Stephen Hopton, Frazer R. Pearce, Greg L. Bryan, and Tom Theuns. A test suite for quantitative comparison of hydrodynamic codes in astrophysics. *Monthly Notices of*

the Royal Astronomical Society, 390(3):1267–1281, November 2008. ISSN 00358711, 13652966. doi: 10.1111/j.1365-2966.2008.13836.x.

V. Tilvi, C. Papovich, S. L. Finkelstein, J. Long, M. Song, M. Dickinson, H. C. Ferguson, A. M. Koekemoer, M. Giavalisco, and B. Mobasher. Rapid Decline of Lyα Emission toward the Reionization Era. *The Astrophysical Journal*, 794:5, October 2014. ISSN 0004-637X. doi: 10.1088/0004-637X/794/1/5.

Tomonori Totani, Kentaro Aoki, Takashi Hattori, and Nobuyuki Kawai. High-precision analyses of Lyα damping wing of gamma-ray bursts in the reionization era: On the controversial results from GRB 130606A at $z = 5.91$. *Publications of the Astronomical Society of Japan*, 68(1):15, February 2016. ISSN 2053-051X, 0004-6264. doi: 10.1093/pasj/psv123.

H. Trac, A. Sills, and U.-L. Pen. A comparison of hydrodynamic techniques for modelling collisions between main-sequence stars. *Monthly Notices of the Royal Astronomical Society*, 377(3):997–1005, May 2007. ISSN 0035-8711, 1365-2966. doi: 10.1111/j.1365-2966.2007.11709.x.

Jan I. C. Vermaak, Jean C. Ragusa, and Jim E. Morel. Massively Parallel Transport Sweeps on Meshes with Cyclic Dependencies. *arXiv:2004.01824 [physics]*, April 2020.

M. Viel, S. Matarrese, H.J. Mo, M.G. Haehnelt, and Tom Theuns. Probing the intergalactic medium with the Lyα forest along multiple lines of sight to distant QSOs. *Monthly Notices of the Royal Astronomical Society*, 329(4):848–862, February 2002. ISSN 0035-8711, 1365-2966. doi: 10.1046/j.1365-8711.2002.05060.x.

Mark Vogelsberger, Shy Genel, Volker Springel, Paul Torrey, Debora Sijacki, Dandan Xu, Gregory F. Snyder, Simeon Bird, Dylan Nelson, and Lars Hernquist. Properties of galaxies reproduced by a hydrodynamic simulation. *Nature*, 509(7499):177–182, May 2014a. ISSN 0028-0836, 1476-4687. doi: 10.1038/nature13316.

Mark Vogelsberger, Shy Genel, Volker Springel, Paul Torrey, Debora Sijacki, Dandan Xu, Gregory F. Snyder, Dylan Nelson, and Lars Hernquist. Introducing the Illustris Project: Simulating the coevolution of dark and visible matter in the Universe. *Monthly Notices of the Royal Astronomical Society*, 444(2):1518–1547, October 2014b. ISSN 0035-8711, 1365-2966. doi: 10.1093/mnras/stu1536.

J. W. Wadsley, G. Veeravalli, and H. M. P. Couchman. On the treatment of entropy mixing in numerical cosmology. *Monthly Notices of the Royal Astronomical Society*,

387(1):427–438, June 2008. ISSN 0035-8711, 1365-2966. doi: 10.1111/j.1365-2966.2008.13260.x.

Terry P. Walker, Gary Steigman, David N. Schramm, Keith A. Olive, and Ho-Shik Kang. Primordial Nucleosynthesis Redux. *The Astrophysical Journal*, 376:51, July 1991. ISSN 0004-637X. doi: 10.1086/170255.

D. F. Watson. Computing the n-dimensional Delaunay tessellation with application to Voronoi polytopes. *The Computer Journal*, 24(2):167–172, February 1981. ISSN 0010-4620, 1460-2067. doi: 10.1093/comjnl/24.2.167.

Daniel Whalen and Michael L. Norman. A Multistep Algorithm for the Radiation Hydrodynamical Transport of Cosmological Ionization Fronts and Ionized Flows. *The Astrophysical Journal Supplement Series*, 162:281–303, February 2006. ISSN 0067-0049. doi: 10.1086/499072.

Stuart C. Whitehouse and Matthew R. Bate. Smoothed particle hydrodynamics with radiative transfer in the flux-limited diffusion approximation. *Monthly Notices of the Royal Astronomical Society*, 353(4):1078–1094, October 2004. ISSN 0035-8711. doi: 10.1111/j.1365-2966.2004.08131.x.

John H. Wise. Cosmic reionisation. *Contemporary Physics*, 60:145–163, April 2019. ISSN 0010-7514. doi: 10.1080/00107514.2019.1631548.

John H. Wise and Tom Abel. How Very Massive Metal-Free Stars Start Cosmological Reionization. *The Astrophysical Journal*, 684(1):1–17, September 2008. ISSN 0004-637X, 1538-4357. doi: 10.1086/590050.

John H. Wise, Vasiliy G. Demchenko, Martin T. Halicek, Michael L. Norman, Matthew J. Turk, Tom Abel, and Britton D. Smith. The birth of a galaxy - III. Propelling reionization with the faintest galaxies. *Monthly Notices of the Royal Astronomical Society*, 442:2560–2579, August 2014. ISSN 0035-8711. doi: 10.1093/mnras/stu979.

Guohong Xu. A New Parallel N-Body Gravity Solver: TPM. *The Astrophysical Journal Supplement Series*, 98:355, May 1995. ISSN 0067-0049. doi: 10.1086/192166.

Naoki Yoshida. Formation of the first generation of stars and blackholes in the Universe. *Proceedings of the Japan Academy, Series B*, 95(1):17–28, January 2019. ISSN 0386-2208, 1349-2896. doi: 10.2183/pjab.95.002.

Saleem Zaroubi. The Epoch of Reionization. In Tommy Wiklind, Bahram Mobasher, and Volker Bromm, editors, *The First Galaxies*, volume 396, pages 45–101. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. ISBN 978-3-642-32361-4 978-3-642-32362-1. doi: 10.1007/978-3-642-32362-1_2.

Mo Zeyao and Fu Lianxiang. Parallel Flux Sweep Algorithm for Neutron Transport on Unstructured Grid. *The Journal of Supercomputing*, 30(1):5–17, October 2004. ISSN 0920-8542. doi: 10.1023/B:SUPE.0000032778.36178.d8.

List of Figures

List of Tables

Acknowledgements

I want to thank Prof. Ralf Klessen and Prof. Simon Glover for giving me the opportunity to write this thesis at ITA and Prof. Guido Kanschat and Prof. Peter Bastian for their help and advice. Moreover, I'd like to thank Joe Lewis for all his help during the development of Subsweep and my colleagues, in particular Loke Ohlin and Mattis Magg for helpful discussions throughout the years. We thank Volker Springel for giving us access to the Arepo code, and the TNG team, in particular Dylan Nelson and Annalisa Pillepich for their helpful discussions as well as the access to the TNG simulation data and parameters.

We acknowledge computing resources and data storage facilities provided by the State of Baden-Württemberg through bwHPC and the German Research Foundation (DFG) through grant INST 35/1134-1 FUGG and INST 35/1503-1 FUGG. We also thank for computing time from the Leibniz Computing Center (LRZ) in project pr74nu. We thank for funding from the Heidelberg Cluster of Excellence EXC 2181 (Project-ID 390900948) 'STRUCTURES: A unifying approach to emergent phenomena in the physical world, mathematics, and complex data' supported by the German Excellence Strategy. RSK, SCOG, and TP acknowledge financial support from the European Research Council in the ERC synergy grant 'ECOGAL – Understanding our Galactic ecosystem: From the disk of the Milky Way to the formation sites of stars and planets' (project ID 855130), from DFG via the Collaborative Research Center (SFB 881, Project-ID 138713538) 'The Milky Way System' (subprojects A1, B1, B2, B8), and from the German Ministry for Economic Affairs and Climate Action for funding in project 'MAINN – Machine learning in Astronomy: understanding the physics of stellar birth with Invertible Neural Networks' (funding ID 50OO2206).