Investigating the Chemical Evolution of the Universe via Numerical Simulations: Supernova Dust Destruction and Non-Equilibrium Ionization Chemistry

by

Devin W. Silvia

B.S., University of Washington, Astronomy, 2007

B.S., University of Washington, Physics, 2007

M.S., University of Colorado, Astrophysical and Planetary Sciences, 2010

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Department of Astrophysical and Planetary Sciences

2013

This thesis entitled:
Investigating the Chemical Evolution of the Universe via Numerical Simulations: Supernova Dust Destruction and Non-Equilibrium Ionization Chemistry
written by Devin W. Silvia
has been approved for the Department of Astrophysical and Planetary Sciences

J. Michael Shull

Prof. John Stocke

Prof. Nils Halverson

Prof. Jack Burns

Prof. G. Barney Ellison

Date

The final copy of this thesis has been examined by the signatories, and we find that both the content and the form meet acceptable presentation standards of scholarly work in the above mentioned discipline.

Silvia, Devin W. (Ph.D., Astrophysical and Planetary Sciences)

Investigating the Chemical Evolution of the Universe via Numerical Simulations: Supernova Dust Destruction and Non-Equilibrium Ionization Chemistry

Thesis directed by Prof. J. Michael Shull

The chemical evolution of the Universe is a complicated process with countless facets that define its properties over the course of time. In the early Universe, the metal-free first stars were responsible for originally introducing metals into the pristine gas left over from the Big Bang. Once these metals became prevalent, they forever altered the thermodynamics of the Universe. Understanding precisely where these metals originated, where they end up, and the conditions they experience along the way is of great interest in the astrophysical community. In this work, I have used numerical simulations as a means of understanding two separate phenomena related to the chemical evolution the Universe.

The first topic focuses on the question as to whether or not core-collapse supernovae in the high-redshift universe are capable of being "dust factories" for the production of galactic dust. To achieve this, I carried out idealized simulations of supernova ejecta clouds being impacted by reverse-shock blast waves. By post-processing the results of these simulations, I was able to estimate the amount of dust destruction that would occur due to thermal sputtering. In the most extreme scenarios, simulated with high relative velocities between the shock and the ejecta cloud and high gas metallicities, I find complete destruction for some grains species and only 44% dust mass survival for even the most robust species. This raises the question as to whether or not high-redshift supernova can produce dust masses in sufficient excess of the $\sim$1 $M_{\odot}$ per event required to match observations of high-z galaxies.

The second investigation was driven by the desire to find an answer to the missing baryon problem and a curiosity as to the impact that including a full non-equilibrium treatment of ionization chemistry has on simulations of the intergalactic medium. To address these questions, I have

helped to develop `Dengo`, a new software package for solving complex chemical networks. Once this new package was integrated into `Enzo`, I carried out a set of cosmological simulations that served as both a test of the new solver and a confirmation that non-equilibrium ionization chemistry produces results that are drastically different from those that assume collisional ionization equilibrium. Although my analysis of these simulations is in its early stages, I find that the observable properties of the intergalactic medium change considerably. Continued efforts to run state-of-the-art simulations of the intergalactic medium using `Dengo` are warranted.

Dedication

To those who fight for truth, justice, and the American way.

Acknowledgements

If one were to define getting a PhD as simply getting from Point A to Point B, my route was anything but direct and there were plenty of bumps along the way. In fact, without the aid of mentors, friends, and family, I might not have ever stumbled my way to Point B, which seemed, at times, to be tremendously elusive.

Given that, I would like to thank my advisor, Professor Mike Shull, for overseeing my research efforts and having the patience to deal with the sometimes frustrating nature of doing computational astrophysics. I thank Britton Smith for lending his expertise at those times when I felt a bit like I was in over my head. I also need to thank the people I've met through my collaborations and my participation in the use and development of multiple forms of open source software. In particular, I thank the `Enzo`, `yt`, and small, yet hopefully growing, `Dengo` communities.

I would also like to thank my housemates, both past and present, official and honorary, as you were all there through some of the highest and lowest points of this journey and I deeply appreciate your camaraderie over these past few years. Beyond my house, there are too many additional friends that I've made along the way to mention here (apparently I'm only allowed the one page), but know that you, too, were instrumental in my arrival at this transition point in my life.

My family was also unwavering in their support through this lengthy process, not only in their willingness to put up with another six years of my life as a "student", but also in their words of encouragement, advice, and praise. As I take this next step in my life, know that you have been, and will continue to be, the reason I have made it this far.

Finally, I thank Amy for putting up with me over the last two years while I have, at times perhaps, appeared to be teetering on the brink of insanity. While we met out of what was a fair bit of happenstance, my life as been immeasurably better with you in it.

Contents

Tables

Table

Figures

Figure

PREFACE

In an effort to help guide the reader through this thesis, I've include this brief preface, which highlights the nature and content of each chapter.

Chapter 1 serves as an introduction to the astrophysical topics relevant to this work and my motivations for studying them. Chapter 2 provides a brief description of the computational methods that enable this research. Specifically, I present the basic details of `Enzo`, the hydrodynamics + N-body code used to run the simulations included in this thesis, and `Dengo`, a new software package for solving chemical networks.

Chapters 3 and 4 are comprised of two previously published works, Silvia et al. (2010) and Silvia et al. (2012), respectively. Both of these works focus on dust destruction in supernova remnants. Since these are both refereed, self-contained articles, they are included in their entirety with permission from the American Astronomical Society. As such, portions of Chapters 3 and 4 will be redundant with the content in Chapters 1 and 2. Furthermore, when Chapter 4 refers to "Paper 1", it is actually referring to Chapter 3, as Silvia et al. (2010) and Silvia et al. (2012) were sequential papers. In addition to the publications contained in these chapters, the presented work was also used in Fesen et al. (2011) to compare the structures found in those simulations with the morphologies of Cas A observations.

Chapter 5 consists of entirely new, unpublished content. Since the content of this chapter is intended for a future publication it uses the first person plural whereas Chapters 1, 2, and 6, use the singular first person. Chapter 5 focuses on testing the new chemistry solver and exploring the early results of IGM simulations that use it.

Finally, I conclude with Chapter 6, which contains a brief summary of the results from Chapters 3, 4, and 5 and an outline of my plans for future work.

Chapter 1

Chemical Evolution in the Universe

1.1 Introduction

From the moment the first star ended its life in a catastrophic supernova explosion, metals have been "polluting" the Universe. Each successive star and collective generations of stars have continued to inject metals into the once pristine gas left over from the Big Bang. Along the way, those metals may have found themselves condensed into dust grains within cold, dense gas clouds or, at times, in a highly ionized state driven by intense shock heating or strong radiation fields. Understanding the nature of this metal enrichment and subsequent chemical evolution of the Universe is of great scientific interest.

Given the complexity of the chemical evolution of the Universe and the difficulties associated with the observations that shine light on this topic, computational astrophysics fills an important niche in advancing our understanding of the subject. Numerical simulations have the power not only to confirm current observations of astrophysical phenomena, but also to predict the results that can be expected from future observations.

Driven by curiosities about the chemical evolution of the Universe and aided by the investigative power of computational simulations, I have sought to understand two separate phenomena related to metal enrichment and subsequent processing of those metals. My earlier work focused on investigating a potential dust destruction mechanism in the early universe while my recent research efforts have revolved around accurately calculating the equations that govern ionization chemistry in the intergalactic medium. Sections 1.2 and 1.3 provide background information and research

goals for these two topics, respectively. The final section of this chapter outlines the remaining contents of this thesis.

1.2 Dust in the Universe and How to Destroy It

In a simple world, the universe would be free of dust grains and all astrophysical gas would happily obey the standard laws of gas physics. Of course, the reality is that dust is observed in nearly every location that astronomers have decided to look and its presence plays a significant role in the evolution of the Universe. Any gas that contains a non-negligible dust mass will have considerably different thermodynamic properties than gas that does not. As refractory elements condense out of the gas phase and become locked into dust grains, those grains become not only a large reservoir of metal coolants but serve as coolants themselves. Furthermore, once dust grains are formed, their surfaces can act as molecular hydrogen formation sites, which is not only an additional gas coolant, but also one the first ingredients for a rich molecular cloud chemistry. These sort of effects can drive the transition from the first population of stars in the Universe (Pop III) to the second (Pop II) by altering the star formation process and initial mass function of that population (Omukai et al., 2005; Clark et al., 2008). Dust also modifies the radiation field of the interstellar medium by reprocessing ultraviolet (UV) and optical star light into far-infrared (IR) and millimeter emission.

Given the significant impact of dust's presence in the Universe, understanding its creation and destruction mechanisms is of great importance in the astrophysics community. In the present day universe and our local stellar neighborhood, it is assumed that the majority of dust formation takes place in the stellar winds of metal-enriched red giant (RG) and asymptotic giant branch (AGB) stars (Dwek & Scalo, 1980). When we shift our thoughts to the early universe, the dominant source of dust grains becomes less certain.

At high redshift, it is important to consider the timescale limitations that observations impose on dust formation. Observations of quasars and galaxies at z $\sim$ 6 have led to estimates of total dust masses of order 10^8 $M_\odot$. Given that that the Universe is only $\sim$1 billion years old at this

epoch and many of the first AGB star progenitors have yet to evolve off of the main sequence, this is a perplexing amount of dust. To explain these unexpected observations, it has been proposed that the core-collapse supernova (SNe) of massive stars can act as the "dust factories" necessary for producing the high-redshift dust (Sugerman et al., 2006).

These cataclysmic events and their remnants are conducive to the formation of large amounts of dust because they 1) can eject significant quantities (30-100 $M_{\odot}$) of stellar-processed material and 2) the abundances of refractory elements (C, O, Si, Fe) in this material is high. Following the high energy ($\sim 10^{52}$ erg) explosion, as the ejected material expands and cools, these elements can condense and nucleate into dust grains. There have been several theoretical efforts (Kozasa et al., 1989, 1991; Todini & Ferrara, 2001; Nozawa et al., 2003; Bianchi & Schneider, 2007; Nozawa et al., 2010) to predict the amount of dust mass that could be expected from those SNe events, which have produced estimates of $\sim$0.1 - 0.3 $M_{\odot}$ per event. Encouragingly, these estimates are in rough agreement with the amount of dust necessary to match the high-redshift observations. However, when observers have sought to confirm such dust masses by observing local supernova remnants (SNRs), they often fall orders of magnitude short of the required values with estimates in the range of 10^{-4} - 10^{-2} $M_{\odot}$ (Stanimirović et al., 2005; Williams et al., 2006; Meikle et al., 2007; Rho et al., 2008, 2009; Kotak et al., 2009). However, more recent observations aimed at finding colder dust ($T < 40$ K; Sibthorpe et al., 2010; Barlow et al., 2010; Matsuura et al., 2011) find larger dust masses, possibly as much as $\sim$0.6 $M_{\odot}$ of dust in SN1987A (Matsuura et al., 2011).

This disagreement between theoretical dust mass predictions and observational estimates, presents another paradox in the quest to understand high-redshift dust. When considering explanations for this discrepancy, there are a number of avenues that one could investigate. First, it is possible that the theoretical calculations are in error, perhaps missing some fundamental component that would decrease the total produced dust. Second, the observations may simply be underestimating the amount of dust in local SNRs or the dust may reside in regimes that are difficult to probe observationally (as suggested by the more recent observations of SN 1987A). These observations must disentangle the thermal emission of dust grains from far-IR backgrounds and

separate newly-formed ejecta dust from that of the circumstellar and interstellar media. Theoretical and observational difficulties aside, it is also possible that there is a mechanism in which the freshly-formed dust grains produced in SNe are rapidly destroyed.

As the fast-moving ($v_{ej} \geq 1000$ km s^{-1}), dust-rich ejecta travels outward from the center of the supernova remnant, it will encounter the reverse shock of the remnant (see Figure 1.1 for a representation of the SNR structure) at relative velocities as high as 10^4 km s^{-1}. During this interaction, the ejecta gas will be compressed and shock heated. In this phase, the nascent embedded dust grains will be subject to ion sputtering and grain-grain collisions in this hot, dense medium. Both of these processes serve to decrease the size of the dust grains and, in the extreme cases, can lead to complete grain destruction.

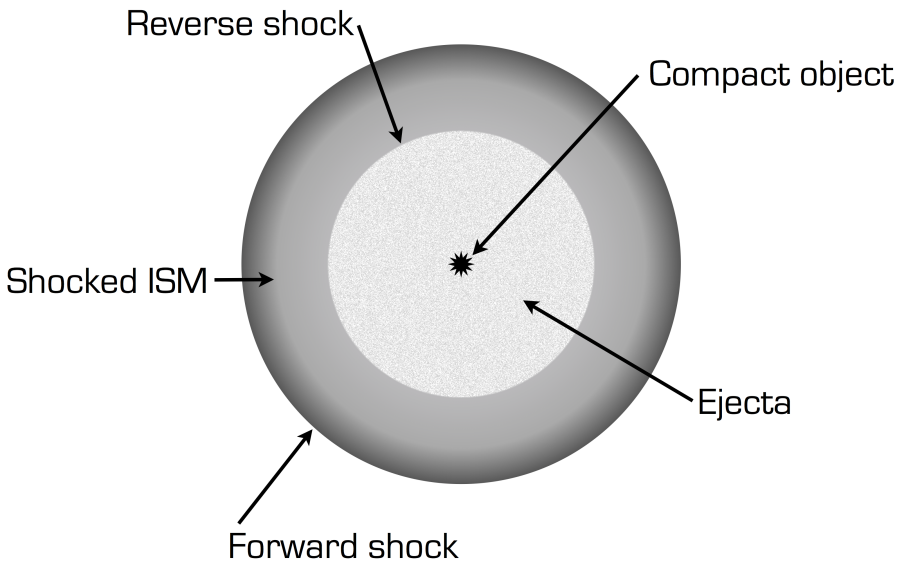


Figure 1.1: A schematic representation of a supernova remnant. The forward shock is created as the supernova shock expands into the interstellar medium (ISM). As the forward shock heats and compresses the ISM, a reverse shock is driven back into the supernova ejecta.

Ion sputtering is a process that occurs any time a solid state material is bathed in a plasma of sufficiently high-energy particles. In the context of a supernova remnants, dust grains play the role of the solid matter and the high-energy particles are the ions of the metal-enriched supernova ejecta. When one of these ions impacts the surface of a dust grain, it will penetrate the surface of

the grain until it eventually collides with an atom trapped in the lattice of the grain. If the energy of the ion is high enough, it will cause a collision cascade that will propagate back toward the dust grain surface. When the cascade recoil reaches the surface, if the energy of the recoil is greater than the surface binding energy of the lattice, an atom will be ejected from the dust grain. Refer to Figure 1.2 for a diagram from Behrisch (1981) of the interaction between the high-energy ion and the solid dust grain.

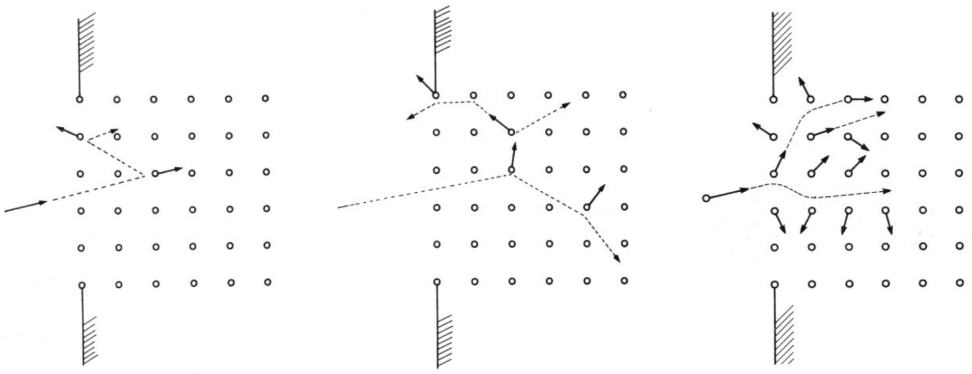


Figure 1.2: Reproduced from Behrisch (1981): Three regimes of sputtering by elastic collisions. (left) The single-knockon regime. Recoil atoms from ion-target collisions receive sufficiently high energy to get sputtered, but not enough to generate recoil cascades. (center) The linear cascade regime. Recoil atoms from ion-target collisions receive sufficiently high energy to generate recoil cascades. The density of recoil atoms is sufficiently low so that knock-on collisions dominate and collisions between moving atoms are infrequent. (right) The spike regime. The density of recoil atoms is so high that the majority of atoms within a certain volume (the spike volume) are in motion.

The exact sputtering yield, the number atoms ejected per incident ion, depends on the mass and energy of the incident ion, the angle of incidence, and the surface binding energy of the dust grain. In the case of metal-enriched supernova ejecta, the dependence on ion mass is of considerable importance. The yields for various combinations of dust grain species and incident high-mass ions have been studied in laboratory settings and, for those combinations that have not been experimentally tested, theoretical models are capable of extrapolating to higher ion masses. Figure 1.3 from Nozawa et al. (2006) shows experimental and model data for various ions impacting silicates and theoretical curves for those same combinations. The dependence of the sputtering yield on ion mass is clearly evident in the large jump from hydrogen and helium ions to argon ions.

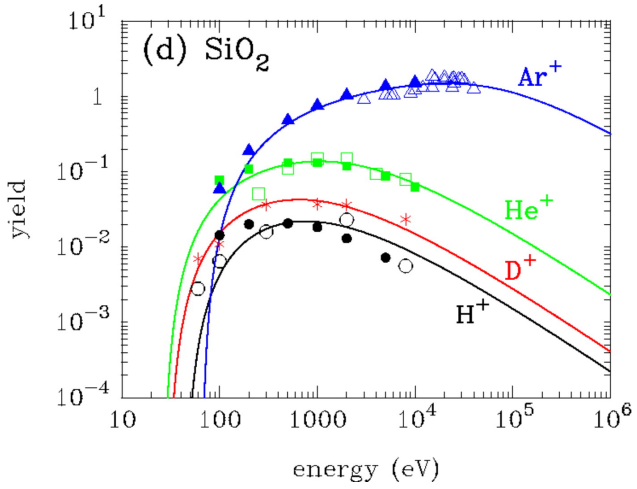


Figure 1.3: Reproduced from Nozawa et al. (2006): Sputtering yields of SiO_2 grains vs. incident energy of projectiles. The incident ion species are H+, D+, He+, and Ar+. The experimental data on sputtering yield are represented by open circles (H+), asterisks (D+), open squares (He+), and open triangles (Ar+), and the results of the sputtering yield calculated by the EDDY code are denoted by filled circles (H+), filled squares (He+), and filled triangles (Ar+). The solid curves show the best-fitting theoretical yields calculated by the universal relation.

Using such data for sputtering yields, it is possible to compute the erosion rates of various dust grains as a function of the density and temperate of the surrounding gas. Those rates can then be coupled to numerical simulations of shock waves interacting with ejecta gas to compute the total amount of dust destruction caused by ion sputtering. When considering running such simulations, it is important that the simulation set up and input parameters be motivated by observational data. To that effect, for the simulations contained in this work, I have referred to the observations of Hammell & Fesen (2008) and Fesen et al. (2011) to guide my selection of simulation parameters and design my simulation mechanics.

In particular, observations like those carried out by Hammell & Fesen (2008, see Figure 1.4) have shown clearly that the ejecta of SNRs are broken up into rapidly expanding "knots" of metal-enriched material. Although the knots catalogued in that work have already expanded beyond the forward shock of the remnant, it is reasonable to assume that these knots are the descendants of likely larger knots that were originally interior to the forward and reverse shocks of the remnant. This motivated my decision to set up "cloud-crushing" simulations akin to those used to study

the interaction between supernova shocks and interstellar gas clouds. The basic premise of such simulations is to drive a shock wave into an over dense, spherical cloud of gas. A detailed description of the simulation set up and the parameters explored is included in Chapters 3 and 4 of this work.

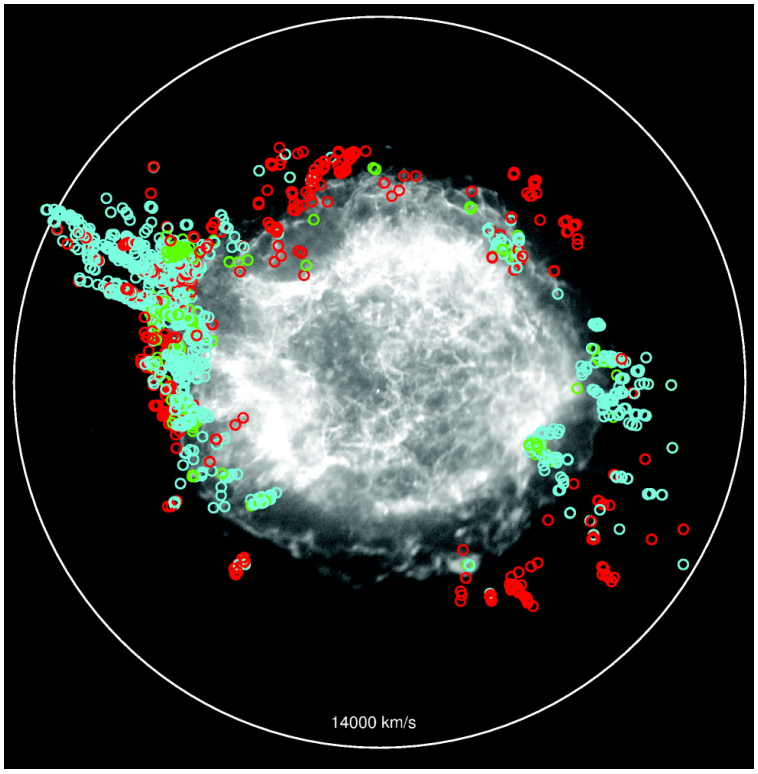


Figure 1.4: Reproduced from Hammell & Fesen (2008): The 1 Ms exposure Chandra image of Cas A, with the locations of all 1825 identified outer ejecta knots marked, color coded by their emission properties. Red open circles indicate knots with strong N II line emission, green open circles knots with strong O II emission, and cyan open circles FMK-like outlying knots. FMK is short for "fast-moving knot", a population of knots observed in Cas A with velocities of order a few $\times$ 10^3 km s^{-1}.

By coupling calculated erosions rates to observationally-motivated numerical simulations, I have worked to answer the question as to how much of newly-formed ejecta dust is capable of surviving the harsh environment of the supernova remnant. From there, I put my results into the context of the larger question about whether or not high-redshift SNe are the dust factories necessary to match the observationally estimated dust masses in z $\sim$ 6 galaxies. Although these questions are far from fully answered, I have brought into debate the idea that SNe alone may not produce sufficient dust to match observations and built a foundation for future work to pursue this

facet of chemical evolution in the Universe.

1.3 The Ionization Structure of the Intergalactic Medium

The studies that revolve around the bulk chemical evolution of the Universe cover an expansive range in redshift, from the formation of the first stars ($z \sim 20 - 30$) to the present-day ($z = 0$) properties of our own Milky Way halo. At very high redshift ($z \gtrsim 8$), the majority of the work in the field has been based in theory and numerical simulations, as the farthest observed objects lie at redshifts of $z \simeq 6 - 9$ (the furthest spectroscopically confirmed galaxy being at $z = 8.55$; Lehnert et al., 2010). In this regime, numerical efforts have attempted to understand the nature of the first stars and the effects of their feedback on their surroundings (Abel et al., 2002; Bromm et al., 2002; O'Shea & Norman, 2007, 2008; Smith et al., 2009; Turk et al., 2009). Eventually, as new observatories come online (e.g. The James Webb Space Telescope), the observational limit will be pushed progressively deeper and observations and theory of this epoch will begin to meet.

If we shift our attention to lower redshifts ($z \approx 4 - 7$), we enter a period in the chemical history of the universe that can be studied by both theory and observations. Theoretical predictions from Big Bang nucleosynthesis, measurements of the cosmic microwave background (CMB) (Komatsu et al., 2011), and observations of the Lyα forest are all in agreement on the cosmic baryon budget. However, moving to $z < 4$, observations run into what is classically known as the "missing baryon" problem (Shull, 2003; Bregman, 2007). Early numerical work (Cen & Ostriker, 1999; Davé et al., 1999, 2001) suggested that much of this missing matter resides in a gaseous phase with temperatures of order $10^5 - 10^7$ K and moderate to low densities, commonly referred to as the warm-hot intergalactic medium (WHIM). Unfortunately, the tenuous nature of the WHIM makes it very difficult to observe. While there has been some success in probing this regime via ultraviolet (UV) spectra (Tripp et al., 2000; Danforth & Shull, 2005, 2008; Danforth et al., 2006; Tripp et al., 2008; Tilton et al., 2012), much of the baryonic matter in this epoch remains missing. The recent results by Shull et al. (2012), which focused on updating the baryon census aided by information from numerical simulations, still find that ∼30% of this matter is unaccounted for (see Figure 1.5).

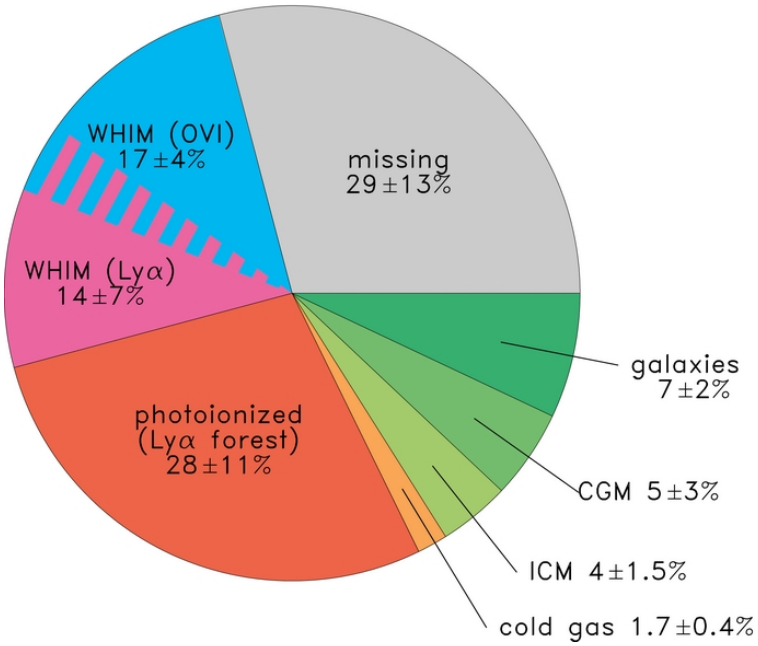


Figure 1.5: Reproduced from Shull et al. (2012): Compilation of current observational measurements of the low-redshift baryon census. Slices of the pie chart show baryons in collapsed form, in the circumgalactic medium (CGM) and intercluster medium (ICM), and in cold gas (H I and He I). Primary baryon reservoirs include diffuse photoionized Lyα forest and WHIM traced by O VI and broad Lyα absorbers. Blended colors (BLAs and O VI) have a combined total of 25% ± 8%, smaller than their direct sum (17% ± 14%) owing to corrections for double-counting of WHIM at 10^5-10^6 K with detectable metal ions. Collapsed phases (galaxies, CGM, ICM, cold neutral gas) total 18% ± 4%. Formally, 29% ± 13% of the baryons remain unaccounted for. Our simulations suggest that an additional 15% reside in X-ray-absorbing gas at T $\geq 10^6$ K. Additional baryons may be found in weaker lines of low-column-density O VI and Lyα absorbers. Deeper spectroscopic UV and X-ray surveys are desirable.

Using prescriptions for star formation rates and stellar feedback, cosmological simulations of large-scale structure can make estimates of the expected metal-enrichment of the multi-phase intergalactic medium (IGM). Additionally, unlike observations, simulations have no obstacles in tracking matter that might exist in a difficult-to-observe gaseous phase. The WHIM is a prime example of such gas. For this reason, there have been several theoretical efforts that have sought to understand the location and properties of the WHIM (Cen & Ostriker, 1999; Davé et al., 1999, 2001; Cen et al., 2001; Chen et al., 2003; Fang et al., 2005; Cen & Fang, 2006; Oppenheimer & Davé, 2009; Smith et al., 2011). When these authors have tried to compare their simulation results to observations, some have found that common absorbers (e.g. O VI; $\lambda\lambda 1032, 1038$) thought to

trace the WHIM may, in fact, trace much cooler gas (∼15,000 K; Oppenheimer & Davé, 2009).

Recently, Smith et al. (2011) carried out high-resolution cosmological simulations with careful attention paid to the effects that the method for stellar feedback has on the global star formation rate and mass fraction in various baryonic phases. Figure 1.6 shows volume renderings of one of the simulations from this work. The important feature to note is how the moderate to low density, WHIM-temperature regions of the domain trace the presence of O VI. From these simulations, the authors created synthetic quasar sight lines to estimate the number of O VI absorbers and tally the number of baryons traced by such absorbers. Although they provide reasonable fits to the number density of observed absorbers (with 57% in the WHIM, contrary to Oppenheimer & Davé (2009)), one caveat to the results presented in this work is the assumption that the metal species are in collisional ionization equilibrium (CIE) with a uniform photoionizing background.

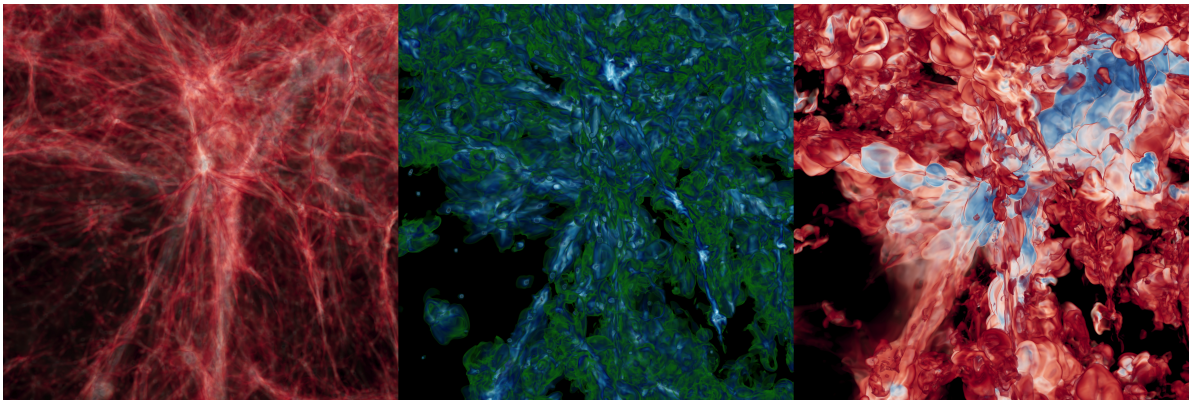


Figure 1.6: Volume renderings of a simulation from Smith et al. (2011). The physical distance spanned by each image is roughly 40 Mpc. The rendered quantities are: (left) gas density, which highlights the filamentary nature of cosmological large-scale structure; (center) post-processed density of O VI, a common WHIM tracer; and (right) temperature, ranging from 10^5 K [red] to 10^7 K [blue]. Image created by Britton Smith and Sam Skillman.

To test the validity of such an assumption, it is worth reviewing the reactions and corresponding rates that determine the abundance of a given ion as a function of time. As a means of simplifying the problem, I present this review in the context of O VI. The reactions that control the creation and destruction of O VI (in the absence of a radiation field) are:

$$\mathrm{O\ V} + e^- \quad \Rightarrow \quad \mathrm{O\ VI} + 2e^-, \tag{1.1}$$

$$\text{O VI} + e^- \Rightarrow \text{O V} + \gamma \textbf{ or } \text{O V}^{**}, \tag{1.2}$$

$$\text{O VI} + e^- \Rightarrow \text{O VII} + 2e^-, \tag{1.3}$$

$$\text{O VII} + e^- \Rightarrow \text{O VI} + \gamma \textbf{ or } \text{O VI}^{**}, \tag{1.4}$$

where Equations 1.1 and 1.3 are the reactions for collisional ionization of O V and O VI and Equations 1.2 and 1.4 are the reactions for recombination of O VI and O VII. Note that these equations show that there are two forms of recombination, radiative recombination and dielectronic recombination. Radiative recombination occurs when a ion captures an electron and simultaneously emits a photon (γ) while dielectronic recombination occurs when the energy of the captured proton goes into the promotion of another bound electron to an excited bound state ('**' denotes this doubly excited state). From these reactions, the rate of change of O VI is given by:

$$\frac{dn_{OVI}}{dt} = \beta_{OV}\ n_e\ n_{OV} - (\beta_{OVI} + \alpha_{OVI})\ n_e\ n_{OVI} + \alpha_{OVII}\ n_e\ n_{OVII}, \tag{1.5}$$

where β_{OV} is the collisional ionization rate of O V into O VI, β_{OVI} is the collisional ionization rate of O VI into O VII, α_{OVI} is the recombination rate of O VI into O V, α_{OVII} is the recombination rate of O VII into O VI, and n_e and n_{OVI} are the number densities of electrons and O VI, respectively. In the presence of a radiation field, Equation 1.6 becomes:

$$\frac{dn_{OVI}}{dt} = [\beta_{OV}\ n_e + \Gamma_{OV}]\ n_{OV} - [(\beta_{OVI} + \alpha_{OVI})\ n_e + \Gamma_{OVI}]\ n_{OVI} + \alpha_{OVII}\ n_e\ n_{OVII}, \tag{1.6}$$

where Γ_{OV} is the photoionization term for O V and Γ_{OVI} is the photoionization term for O VI. I should note that here I have simply written the recombination rate as α_{OVI} where in reality this factor is the sum of radiative, α_{rad}, and dielectronic, α_{di}, recombination rates.

If we consider that, over time, some fraction of the total oxygen will end up in O VI, a highly sought after observable in the UV, then it is important to understand how long it might stay in that state. From Equation 1.6, the recombination time for O VI can be estimated as:

$$\tau_{\alpha_{OVI}} = \frac{1}{n_e\ \alpha_{OVI}}, \tag{1.7}$$

At the densities and temperatures of the WHIM, this becomes:

$$\tau_{\alpha_{OVI}} \approx \frac{1}{1.16 n_H \times (7 \times 10^{-12}\ cm^3\ s^{-1})} \approx \frac{1}{1.16 \times 10^{-5}\ cm^{-3} \times (7 \times 10^{-12}\ cm^3\ s^{-1})}, \tag{1.8}$$

$$\tau_{\alpha_{OVI}} \approx 1.2 \times 10^{16}\ s \approx 3.8 \times 10^{8}\ yr, \tag{1.9}$$

where we have assumed a number density of $n_H = 10^{-5}$ cm^{-3}, a temperature of T = 10^6 K for the recombination rate of O VI (pulled from the CHIANTI atomic database; Dere et al., 2009), and an electron density of $n_e \approx 1.16\ n_H$ for gas comprised of fully ionized H and He. While this number is already only a couple orders of magnitude from a Hubble time, for lower densities or higher temperatures, it becomes even more comparable. Given this, it becomes important to carry out non-equilibrium (NEQ) ionization calculations to accurately map the ionization structure of WHIM gas.

Furthermore, since certain ion species can be powerful coolants in intergalactic gas, divergence from CIE values could have drastic feedback effects on the thermodynamic properties of such gas. Accurately calculating the net cooling rate of gas that is out of ionization equilibrium requires including the ion-by-ion cooling efficiencies, like those produced by Gnat & Ferland (2012), in numerical calculations. Figure 1.7 shows how the overall cooling function can differ from the CIE values when a gas, which is initialized at 5 $\times$ 10^6 K, is allowed to cool and the ion fractions shift out of equilibrium.

While some work has been done to include NEQ ionization effects for oxygen in simulations (Cen & Fang, 2006), these authors did not explicitly check whether resulting absorptions systems traced the WHIM. Motivated by the importance of understanding the mass content of this hard-to-detect phase of the IGM, I have worked to develop a new NEQ ionization chemistry solver to calculate not only the ionization structure of oxygen, but of all interesting metal species. For the purposes of this thesis, however, I have focused on the results of applying this new solver to oxygen ion species and have attempted to compare these results to those of Smith et al. (2011). The work completed to date is discussed in Chapter 5

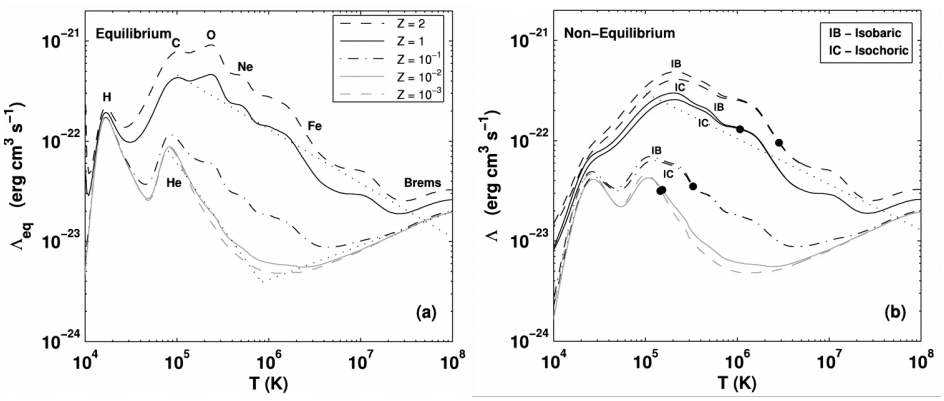


Figure 1.7: Reproduced from Gnat & Sternberg (2007): Cooling efficiency (ergs cm^3 s^{-1}) vs. temperature for Z from 10^{-3} to 2 times solar. (a) Equilibrium cooling. The dominant cooling elements at various temperatures are indicated near the curves. The dotted lines are our power-law fits for $Z/Z_\odot = 1$ and 10^{-3}. (b) Non-equilibrium cooling. For each Z, two curves are displayed: the lower curve is for isochoric (IC) cooling, and the upper curve is isobaric (IB) cooling. The dotted line is our power-law fit for $Z/Z_\odot = 1$. The filled circles indicate the temperatures at which time-dependent cooling first deviate from the equilibrium cooling by 5%.

In addition to UV studies, observations at X-ray wavelengths have sought to probe WHIM-phase gas via more highly ionized metals (e.g. O VII and O VIII) that should trace the hotter end of the temperature range. Very recently, Gupta et al. (2012) have used such observations to make claims that $\sim 7 \times 10^{10} M_\odot$ of matter could exist in the circumgalactic medium (CGM) surrounding the Milky Way. This is over an order of magnitude more mass than estimated to exist in the CGM of other star-forming galaxies by Tumlinson et al. (2011) and Stocke et al. (2013) using O VI. This X-ray work does come with the caveat that they used an ionization model that assumed a single temperature to fit O VI, O VII, and O VIII, which could lead to unrealistic mass estimates. However, if this is true and not unique to the Milky Way, this could explain the majority of the missing baryons. By coupling this new NEQ chemistry solver to simulations of CGM, it will be possible to determine if simulations can support such claims.

1.4 Thesis Outline

Beyond the present chapter, this thesis consists of five additional chapters. Chapter 2 focuses on the computational methods that enabled this work. It includes a description of the primary

simulation code I used for my investigations of chemical evolution, `Enzo`, and the new software package, `Dengo` , that has been coupled to `Enzo` to allow for NEQ ionization chemistry calculations. Chapters 3 and 4 present the entirety of my work on using hydrodynamic simulations to understand dust destruction in SNRs. Chapter 5 showcases my most recent work, which has been focused on testing the new `Dengo` package and applying it to simulations of the IGM. Finally, I provide a brief summary of my research efforts and a discussion of my ideas for future work in Chapter 6.

Chapter 2

Computational Methods

Given that the bulk of my thesis work resides firmly in the world of computational astrophysics, this chapter serves to explain the nature of 1) the simulation code, `Enzo`, used to investigate the astrophysical phenomena discussed in Chapter 1 and 2) the details of the newly developed software package for solving chemical networks, `Dengo`. For any details beyond those discussed in this chapter, I refer the reader to the cited articles that explain each code extensively.

2.1 `Enzo`

The simulation work outlined in the following chapters employs the publicly available `Enzo`[1] code (Bryan & Norman, 1997; Norman & Bryan, 1999; O'Shea et al., 2005; The Enzo Collaboration et al., 2013). `Enzo` is an adaptive mesh refinement (AMR), grid-based hybrid (hydrodynamics + N-body) code that was originally written by Dr. Greg Bryan to perform cosmological simulations. The code has since been developed by a widespread community of individuals at a variety of institutions, including the University of Colorado.

`Enzo` uses an N-body particle mesh method to solve the gravitational dynamics of collisionless dark matter particles and an Eulerian block-structured AMR method (Berger & Colella, 1989) combined with a number of available numerical methods for solving the ideal gas dynamics. For this work, some of my simulations use the Piecewise Parabolic Method (PPM; Colella & Woodward, 1984), while others use the ZEUS solver of Stone & Norman (1992b) for solving the hydrodynamic

[1] `http://enzo-project.org/`

equations. The specification of which simulations used which solver is included in Sections 2.1.1 and 2.1.2.

`Enzo` also comes pre-equipped with a variety of physics capabilities, including: radiative cooling of both primordial and metal-enriched gas, a semi-implicit backward-differencing non-equilibrium (NEQ) primordial (H, He, H_2) chemistry solver (Abel et al., 1997; Anninos et al., 1997), a number of photoionizing/heating UV background models (e.g. Haardt & Madau, 1996), algorithms for star formation and feedback (e.g. Cen & Ostriker, 1992), two ideal magneto-hydrodynamics solvers (Wang & Abel, 2009; Collins et al., 2010), and two methods for radiation transport (Reynolds et al., 2009; Wise & Abel, 2011). The following two sections describe, in detail, which of these capabilities, or any additional capabilities, are used in the simulations presented in this work.

2.1.1 Cloud-crushing Simulations

For the cloud-crushing simulations mentioned in Chapter 1 and discussed in detail in Chapters 3 and 4, none of the cosmological or gravity-solving components of `Enzo` were used. Given the idealized nature of those SNR simulations and the fact that the gas hydrodynamics alone should dominate the evolution of the ejecta-shock interaction, there was neither a need nor a benefit to using those capabilities. To solve the equations that govern the hydrodynamics, I used `Enzo`'s PPM-based solver, which is particularly well suited for capturing the detailed shock structure expected when a blast wave is driven through and around an overdense cloud of gas.

In addition to solving the hydrodynamics, I also employed Enzo's method of computing radiative cooling rates via static look-up tables. The early work (Chapter 3) made use of `Enzo`'s pre-packaged, tabulated cooling rates defined by the analytic formula of Sarazin & White (1987) for $Z/Z_\odot = 0.5$ gas assumed to be in collisional ionization equilibrium. However, in Chapter 4, I used new cooling rate look-up tables for metallicities of $Z/Z_\odot = 1$, 10, and 100 created using `Cloudy`[2] (Ferland et al., 1998). Again, for these tables, the elements were assumed to be in ionization equilibrium. These new rate tables allowed me to approximate the cooling efficiencies

[2] http://nublado.org

for the metal-enriched gas expected in supernova ejecta.

My cloud-crushing simulations also took advantage of the AMR capabilities of `Enzo` through the addition of an extra data field and a new refinement criterion. Although these additions are discussed in detail in Chapter 3, for completeness, I have included an explanation here as well. In order to adequately track the evolution of the disrupted ejecta cloud, I defined a new scalar field, ξ, which was used to represent "cloud material". During simulation initialization, all cells enclosed within the cloud radius are defined to have a ξ value scaled to the gas density of the cloud such that the cells with the highest gas densities are set to $\xi = 1$ and cells with lower densities have proportionately lower ξ values. This field allows for a rather simple refinement criterion: any time a cell contains more than a specified amount of cloud material "mass" ($m_\xi = \xi \times$ cell volume), the cell is refined. All cells that get flagged by the refinement are re-gridded, with every spatial dimension being cut in half. This refinement method assures that the simulation maintains the resolution necessary to capture the full evolution of the cloud as it is disrupted by the inflowing shock without wasting computational resources on the comparatively uninteresting ambient medium.

Finally, as a means of estimating the dust destruction that might occur during the ejecta-shock interaction, I took advantage of `Enzo`'s ability to place Lagrangian tracer particles within the simulation domain. These particles are massless and do not alter the thermodynamic properties of the gas. When placed within a simulation, these particles are simply advected along with hydrodynamic flows present in the gas. These particles can be of tremendous utility because they allow one to track the thermodynamic history of individual gas parcels, a capability normally only afforded to smoothed-particle hydrodynamic (SPH) codes. For my simulations, I implanted tracer particles only within the ejecta cloud and used them represent discrete populations of ejecta dust. Once a simulation ran to completion, I took the density and temperature histories of my "dust particles" to calculate the amount of dust destruction that would have occurred due to the thermal sputtering of the dust grains. As mentioned before, a more detailed description of these sputtering calculations is contained in Chapters 3 and 4.

2.1.2 IGM Simulations

In Chapter 5, I present simulations that use the standard cosmological faculties of `Enzo`. While the specific details of the simulation parameters are included in that chapter, this section serves to highlight the various physics modules that are utilized. First, we initialize the simulations under the assumption of a ΛCDM universe with a power spectrum of density fluctuations given by Eisenstein & Hu (1999). During run time, the simulations include a prescription of star formation and distributed feedback analogous to that of Smith et al. (2011), who used a modified version of Cen & Ostriker (1992)'s star formation prescription and a new distributed-feedback method of injecting gas, metals, and thermal energy into the twenty-seven cells immediately surrounding a star particle. To account for the effects of radiative heating and cooling, a uniform UV metagalactic background that evolves with redshift following Haardt & Madau (2001) serves as the heating term, while cooling rates are calculated using the previously mentioned Sarazin & White (1987) analytic formula look-up table for $Z/Z_{\odot} = 0.5$ gas.

Unlike the cloud-crushing simulations, in these cosmological simulations, the equations of hydrodynamics are solved using the Stone & Norman (1992b) ZEUS solver. The choice to use this solver over the PPM-based one was motivated by the long term goal of doing an "apple-to-apples" comparison of my NEQ results with the CIE-based results of Smith et al. (2011). In this same vein, the simulations of the intergalactic medium (IGM) do not employ any refinement criteria to trigger the AMR functionally in `Enzo` and are instead comprised only of large static grids, referred to in Chapter 5 as "unigrid" simulations. Since the WHIM is expected to exist over a large volume of physical space, using AMR to resolve the regions of interest would likely result in the refinement of a significant fraction of the computational domain. In this scenario, it can be computationally favorable to increase the resolution of the entire domain and avoid the need for processors to communicate information between numerous AMR-created grids. Additionally, high-resolution unigrid simulations ensure that the physical structures that will eventually comprise the WHIM are adequately resolved at all epochs, especially at high redshifts when AMR criteria

might not yet be met. Ideas for choosing efficient refinement criteria for future work are discussed in Chapter 6.

Finally, the most important component included in these IGM simulations is the "on-the-fly" calculation of the ionization state of metal species, specifically that of oxygen. While `Enzo` originally included machinery for solving non-equilibrium primordial chemistry, my additions to the code allow for hydrogen, helium, and heavy elements ("metals") to be solved simultaneously. This capability is enabled by `Dengo`, a newly developed software package for solving complex chemical networks. The following section provides a a detailed explanation of the methods used and the functionality provided by `Dengo`.

2.2 `Dengo`

In the world of computational astrophysics, there are several scenarios that require accurate solutions to complex chemical networks to produce physically realistic results. These chemical networks need not only to calculate the changes in the chemical composition of an astrophysical fluid but also the corresponding heating and/or cooling rates of that fluid. To achieve this, these chemical networks must be comprised of many interconnected components, including the involved species, the reaction equations, and the reaction rate coefficients. Depending on the nature of the specific astrophysical system of interest, such networks can rapidly become large and, at times, unwieldy. Furthermore, as computational resources grow and their capabilities improve, the chemical networks that can be feasibly solved will only increase in size and complexity. To address this impending computational obstacle, I have worked with Dr. Matthew Turk and Professor Dan Reynolds to develop a new software package for building and solving large chemical networks, which we refer to as `Dengo`[3] .

`Dengo` serves as a means of defining chemical networks that are both extendable and contractable with components that are easily connected, yet simultaneously separable. As input, `Dengo` requires a list of the species of interest, the reactions that define their creation and destruc-

[3] https://bitbucket.org/MatthewTurk/dengo

tion, and the rate coefficients that correspond those the governing reactions. With this information in hand, `Dengo` proceeds to match right-hand and left-hand sides (R/LHS) of the equations for the rate of change of every chemical species by aggregating all of the source and sink terms from the full set of reactions. `Dengo` also calculates the partial pressures contributed from each species, which can be then used to compute the temperature of the gas self-consistently with the chemical composition. Once the temperature is calculated, the rate coefficients are interpolated from a set of lookup tables and fed into the complete set of RHS's for the chemical network. If the effects of radiative cooling from the chemical species are also desired, the cooling term, which will alter the gas energy, is also computed. Once all of these equations are defined, `Dengo` calculates the corresponding Jacobian matrix for use in solving the system of equations. `Dengo` then passes all of these components off to a generic solver that is well-suited to solve the system of ordinary differential equations (ODEs). Finally, `Dengo` takes all of the various pieces, combines them, and outputs a complete chemical network solver that can be "plugged in" to virtually any simulation code. Figure 2.1 serves to visualize `Dengo`'s overall conceptual framework.

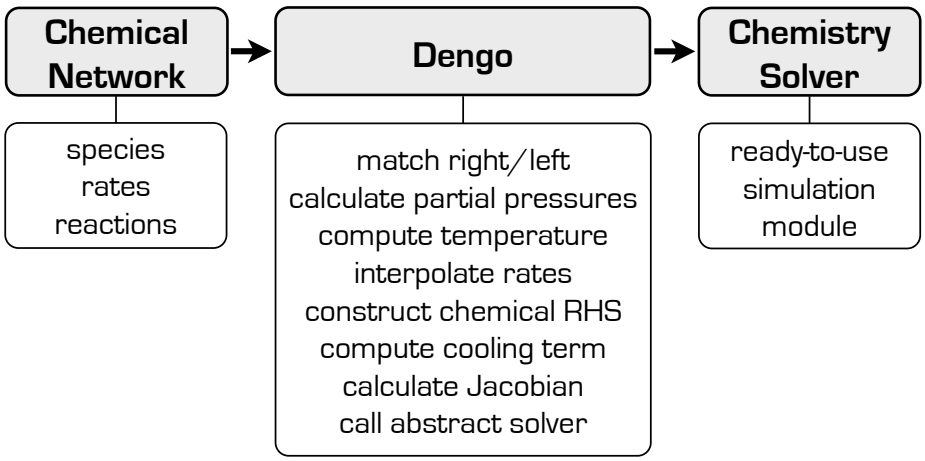


Figure 2.1: A visual representation of the concept behind the design of the `Dengo` software package. First, a chemical network is defined, which consists of the species, the rates that control their formation and destruction, and the reactions that use those rates. Second, `Dengo` 1) matches up all the species on the right and left sides of the reaction equations 2) computes the pressure, temperature, and cooling 3) constructs the right-hand-sides of all the rate-of-change equations and their corresponding Jacobian matrix and 4) calls an abstract solver. Last, `Dengo` outputs all of this information into a self-contained module that can be plugged into a simulation code.

On the technical side, the `Dengo` package is primarily written in `Python`[4] . This affords the use of `SymPy`[5] , a python library that enables `Dengo` to do symbolic mathematics for the construction of the RHS's and Jacobian of the system of equations. To output the final chemistry solver, the python-based template engine `Jinja2`[6] is used to write the C/C++ code required by `Enzo`. However, this could be easily modified to produce files in a variety of programming languages so that `Dengo` could be used for any simulation code.

For the work presented in this thesis, all solutions to the chemical networks are calculated using the backward Euler method, an implicit method ideal for solving a systems of stiff equations, where stiff systems are those with inherent timescales that vary across the system. In the case of the IGM, these timescales include the ionization time, the recombination time and the cooling time. In the backward Euler scheme, a system of equations defined by:

$$\frac{d\mathbf{u}}{dt} = \mathbf{f}(\mathbf{u}), \tag{2.1}$$

is integrated by evaluating:

$$\mathbf{u}_{n+1} = \mathbf{u}_n + h\mathbf{f}(\mathbf{u}_{n+1}), \tag{2.2}$$

where h is the step size and the variables in bold represent vectors. This can then be linearized using Newton's method, such that:

$$\mathbf{u}_{n+1} = \mathbf{u}_n + h\left[\mathbf{f}(\mathbf{u}_n) + \left.\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right|_{\mathbf{u}_n} \cdot (\mathbf{u}_{n+1} - \mathbf{u}_n)\right], \tag{2.3}$$

where $\partial\mathbf{f}/\partial\mathbf{u}$ is the Jacobian matrix. To simplify things further, Equation 2.4 can be rearranged to isolate $\mathbf{u}_{n+1}$:

$$\mathbf{u}_{n+1} = \mathbf{u}_n + h\left[\mathbf{1} - \frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right]^{-1} \cdot \mathbf{u}_n, \tag{2.4}$$

which provides a straightforward solution to Equation 2.1. During simulation run time, for every cell, this backward Euler solver is sub-cycled within a given hydrodynamic time step to evolve the

[4] http://www.python.org/
[5] http://sympy.org/en/index.html
[6] http://jinja.pocoo.org/docs/

chemical species. If the effects of radiative cooling are desired, the expected change in the gas energy due to the chemical species is computed simultaneously and applied to the cell. The specific species, reactions, and rate coefficients used for this work are discussed in Chapter 5.

For extensive details about the inner workings of `Dengo` and a comprehensive list of its capabilities, I refer the reader to the papers in preparation of Turk et al. (2013), Silvia et al. (2013), and Reynolds et al. (2013), which serve to introduce `Dengo` and the primordial network, discuss large chemical networks that include metal ions, and investigate the optimal numerical solvers for such networks, respectively. My primary contributions to the development of `Dengo` include building components of its basic framework for parsing a chemical network and outputting a complete solver, incorporating the reactions and rates for metal species, testing the validity of the solver, and integrating it into `Enzo`.

Chapter 3

Numerical Simulations of Supernova Dust Destruction. Cloud-crushing and Post-processed Grain Sputtering

ABSTRACT

We investigate through hydrodynamic simulations the destruction of newly-formed dust grains by sputtering in the reverse shocks of supernova remnants. Using an idealized setup of a planar shock impacting a dense, spherical clump, we implant a population of Lagrangian particles into the clump to represent a distribution of dust grains in size and composition. We then post-process the simulation output to calculate the grain sputtering for a variety of species and size distributions. We explore the parameter space appropriate for this problem by altering the over-density of the ejecta clumps and the speed of the reverse shocks. Since radiative cooling could lower the temperature of the medium in which the dust is embedded and potentially protect the dust by slowing or halting grain sputtering, we study the effects of different cooling methods over the time scale of the simulations. In general, our results indicate that grains with radii less than 0.1 μm are sputtered to much smaller radii and often destroyed completely, while larger grains survive their interaction with the reverse shock. We also find that, for high ejecta densities, the percentage of dust that survives is strongly dependent on the relative velocity between the clump and the reverse shock, causing up to 50% more destruction for the highest velocity shocks. The fraction of dust destroyed varies widely across grain species, ranging from total destruction of Al_2O_3 grains to minimal destruction of Fe grains (only 20% destruction in the most extreme cases). C and SiO_2 grains show moderate to strong sputtering as well, with 38% and 80% mass loss. The survival rate of grains formed by early supernovae is crucial in determining whether or not they can act as the "dust factories" needed to explain high-redshift dust.

3.1 Introduction

The presence of dust grains in the interstellar medium (ISM) of galaxies can have a wide range of impacts on the evolution of the universe, both at present and at high redshift. As refractory elements become depleted from the gas phase and locked into grains, they form a considerable reservoir of coolants in the solid-state phase. The surfaces of these grains are also believed to act as formation sites for molecular hydrogen (H_2), which not only serves as an additional coolant but can provide the initial conditions for a rich cloud chemistry. Grains are capable of reprocessing the radiation field of the ISM by providing a critical transfer mechanism for turning ultraviolet/optical starlight into far-infrared and submillimeter emission seen in interstellar molecular clouds and high-redshift galaxies. The earliest dust grains may also have influenced the low-metallicity gas by modifying the thermodynamic conditions in the high-redshift universe and altering the star formation process and the initial mass function (IMF) for the earliest second-generation stars (Omukai et al., 2005; Clark et al., 2008).

One paradox arises from observations of dust at high redshift. Several observations of damped Lyα systems indicate the presence of dust grains (Pettini et al., 1994; Ledoux et al., 2002), and high-redshift ($z > 6$) quasars seen in the Sloan Digital Sky Survey (SDSS) along with galaxies in the Hubble Deep Field show 1.2 mm thermal emission from dust (Smail et al., 1997; Hughes et al., 1998; Bertoldi et al., 2003). These observations imply a total dust mass of $\sim 10^8\ M_\odot$ formed within the first Gyr after the Big Bang, which is difficult to explain using traditional models of grain formation in evolved low-mass stars (Dunne et al., 2003; Dwek et al., 2007) and transportation of dust into the ISM through stellar winds (Whittet, 1992).

In an effort to find other sources for high-redshift dust, recent models have looked at the ejecta of core-collapse supernovae (SNe) as possible formation sites, which can occur on timescales shorter than the evolution of low-mass stars (Dwek & Scalo, 1980). Since the inner ejecta of core-collapse SNe are generally cold, dense, and metal-enriched, they are conducive to producing significant amounts of new dust grains. By utilizing theories of nucleation and grain growth made

by Kozasa & Hasegawa (1987), two groups (Kozasa et al., 1989, 1991; Todini & Ferrara, 2001) have attempted to estimate the dust mass formed in expanding SNe ejecta of varying metallicities, with results ranging from ∼0.1–0.3 $M_{\odot}$. These initial results were extended by Nozawa et al. (2003) and Schneider et al. (2004) to the case of zero-metallicity SNe with significant production of O, Si, and other heavy elements, as would be created by Population III stars.

Although SNe are appealing as potential "dust factories", current observations of nearby SNe appear to fall short of the required dust masses (∼0.2 $M_{\odot}$, or possibly as high as 1 $M_{\odot}$ per event; Dwek et al., 2007) by factors of 10-100. Sugerman et al. (2006) produced a review of Type II SN 2003gd in which they used radiative transfer models to conclude that up to 0.02 $M_{\odot}$ of dust is present in the ejecta, but this is contradicted by Meikle et al. (2007) who found only $4 \times 10^{-5} M_{\odot}$ of newly formed dust using mid-IR observations. Similar values were found in a young supernova remnant (SNR) located in the Small Magellanic Cloud (SMC) studied by Stanimirović et al. (2005), who claimed to observe $10^{-3} M_{\odot}$ of hot dust. Kotak et al. (2009) assert a direct detection of ejecta dust in Type II-plateau SN 2004et, but again only estimate a few times 10^{-4} $M_{\odot}$. One promising observation for SNe as dust factories carried out by Dunne et al. (2003) argued for the existence of ∼3 $M_{\odot}$ of dust in the direction of Cassiopeia A based on a submillimeter detection, but it was quickly contradicted by Krause et al. (2004) who attributed this emission to interstellar dust contained in an adjacent molecular cloud. In the Large Magellanic Cloud (LMC), Williams et al. (2006) observed four SNRs using the 24 and 70 μm bands on *Spitzer Space Telescope* and inferred dust masses of 0.01-0.1 $M_{\odot}$. More recently, Rho et al. (2008, 2009) used *Spitzer* to study Cassiopeia A and young supernova remnant 1E0102-7219 to find 0.020-0.054 $M_{\odot}$ and 0.014 $M_{\odot}$ of dust respectively. To continue the hunt for dust in Cassiopeia A, data from AKARI and the Balloon-borne Large Aperture Submillimeter Telescope (BLAST) were used by Sibthorpe et al. (2009) to find 0.06 $M_{\odot}$ of new "cold" ($T \sim 35$ K) dust, which when combined with previous "warm" ($T \sim 100$K) dust estimates, could make the dust-factory argument more plausible.

Observational tests that seek to confirm the predictions made by models of SNe are difficult. It is necessary to disentangle far-IR backgrounds from the thermal dust emission and separate the

newly formed ejecta dust from grains contained in circumstellar and interstellar material. One must also determine what fraction of the dust can be destroyed by the interaction between SNR reverse shocks and the ejecta. As a result of these difficulties, some dust may be located in undetected, unshocked ejecta. SN 1987A presents a particularly good case of the challenge of producing accurate dust mass estimates, since it has been inferred that dust formation is almost certainly occurring, but only $10^{-3} M_{\odot}$ was detected (Dwek, 2006; Dwek & Arendt, 2007). Observed changes in the optical and bolometric fluxes and asymmetric red and blue emission-line wings indicate that more dust is most likely present (McCray, 1993, 2007). Furthermore, since SN 1987A is a young remnant, the reverse shock has not yet impacted the inner ejecta material — a location that is very likely to harbor additional dust.

The question arises as to whether or not the fast-moving dust ($V_{\rm ej} \geq 1000$ km s^{-1}) can survive its impact with the reverse shock, at which point it will be subject to thermal sputtering and grain-grain collisions. Since the relative speed between the shock and the ejecta clumps can be large, the temperature of the ejecta can be raised to a level that might completely destroy these newly formed grains. Recent work by Nozawa et al. (2007) investigates the interaction between reverse shocks and SN ejecta through one-dimensional hydrodynamic simulations. They conclude that 20-100% of the dust mass is destroyed, depending on the ambient gas density and the initial energy of the SN explosion. Bianchi & Schneider (2007) also used a combination of numerical and semi-analytic methods to conclude that only a small fraction of the dust surives — as low as 7% in some of their models. Further efforts were made by Nath et al. (2008) who performed one-dimensional analytic calculations of the reverse shock impacting ejecta material that followed a power-law density profile. They found that only 1-20% of the dust mass in the form of silicates and graphites was sputtered, depending on the steepness of the profile and the grain size distribution.

While the above studies have treated the ejecta material primarily as a continuous medium, observations of SNR ejecta indicate that in reality the material is knotted and clumpy (Fesen et al., 2006; Hammell & Fesen, 2008). This lends itself well to the classic "cloud-crushing" problem (Woodward, 1976) in which an over-dense clump/cloud of material is impacted by a shock wave.

Such a problem has been studied using numerical hydrodynamic simulations in the context of an ISM cloud getting hit by the forward shock of a supernova explosion (Bedogni & Woodward, 1990; Stone & Norman, 1992a; Klein et al., 1994; Mac Low et al., 1994; Orlando et al., 2005; Patnaude & Fesen, 2005; Nakamura et al., 2006).

In this paper, we investigate the effects of reverse shocks on SN ejecta by performing three-dimensional cloud-crushing simulations and focusing on the case of an individual ejecta clump encountering a reverse shock. In Section 3.2, we describe the code that we have used to carry out these simulations and our particular simulation set up. In Section 3.3, we discuss the simulations examined in this paper, which include comparisons to previous work and runs that serve as our means of computing reverse-shock dust destruction. In Section 3.4, we highlight our methods for calculating dust destruction. In Section 3.5, we present the results of calculations, and we conclude in Section 3.6 with a summary and discussion of future work.

3.2 Methodology

3.2.1 Enzo

For this study, we use the cosmological, Eulerian adaptive mesh refinement (AMR), hydrodynamics + N-body code, Enzo (Bryan & Norman, 1997; Norman & Bryan, 1999; O'Shea et al., 2005). Owing to the idealized nature of our problem, we do not use any of the cosmological or gravity-solving components of the code. To handle the fluid dynamics in our work, we use Enzo's piecewise parabolic method (PPM; Colella & Woodward, 1984) for solving the hydrodynamic equations. For more details on the inner-workings of Enzo, refer to the cited articles.

The AMR capabilities of Enzo are particularly useful to resolve a range of physical scales or to ensure that only the regions of particular concern remain well resolved, so that we do not waste computation time on unimportant or uninteresting portions of the computational domain. The refinement algorithm works by taking any cells that meet a user-specified refinement criterion and spawning a new grid of cells with twice the resolution. For example, a three-dimensional cell that

satisfies any of the desired refinement criteria will be split into eight cells with cell edges that are now half as long as the parent cell from which they were spawned. Once this cell is refined to a higher level, all of the new child cells are then checked to see if any still meet the refinement parameters. If so, they will be refined to a higher level in the same way, becoming parents to new children. Higher and higher levels of refinement can be created to theoretically infinite resolution. However, since large amounts of refinement can quickly become computationally expensive, a maximum level is often specified when running to the code to halt refinement at the highest resolution that can be afforded by the simulator. Enzo comes preloaded with the ability to refine based on a variety of parameters: matter overdensity; gradients in pressure, energy, or density; Jeans length; cooling time; and shocks. It is also possible to define new refinement criteria that are simulation specific. We have created such a criterion for our cloud-crushing simulations that refines on gas that is initially contained within the cloud. The implementation of that refinement method is discussed in Section 3.3.2.

In addition to the standard hydrodynamic capabilities of Enzo, we utilize "tracer particles" in all of our simulations. The particles are Lagrangian in nature and simply advect with the flow present in the grids. In the current format, the tracer particles do not alter the state of their surrounding medium and serve only as a means of tracking the conditions (i.e., density and temperature) of the fluid they are embedded in. However, the machinery for allowing the tracer particles to produce feedback, such as metal injection or dust cooling, exists and will be utilized in our future papers. Currently, the tracer particles are particularly useful in representing the presence of dust in our simulations, and their use as such will be addressed in Section 3.4.1.

For a subset of our simulations, we also include the effects of radiative cooling. Currently, we use the analytic formula defined by Sarazin & White (1987), which approximates the cooling rate as a function of temperature for $Z = 0.5Z_{\odot}$ gas in ionization equilibrium. While we expect the ejecta material to be metal enriched, with relative abundances that likely deviate from solar values, for the purposes of this paper we sought only to investigate the effect of the presence of metals (cooling on) versus a complete lack of metals (cooling off). Future work will include more variable

abundances to allow for the enrichment of the ejecta by metals such as C, O, Si, Mg, Mn, and Fe which have been shown to have significant impacts on the cooling rates (Santoro & Shull, 2006) and are key ingredients in most dust grains. When implementing these adjustable abundances and altered cooling rates, we plan to use a Cloudy (Ferland et al., 1998) cooling method similar to Smith et al. (2008).

3.2.2 Simulation Setup

To address the proposed problem of a supernova remnant reverse shock impacting a clump of newly formed ejecta, we added to the suite of pre-packaged Enzo simulations. In this particular problem type, we seek to simulate the previously studied "cloud-crushing" problem in which a shock is driven into a cloud of over-dense material. Although previous work focused on the forward shock of a supernova propagating into the ISM, the process is analogous to that of a reverse shock penetrating the clumpy supernova ejecta.

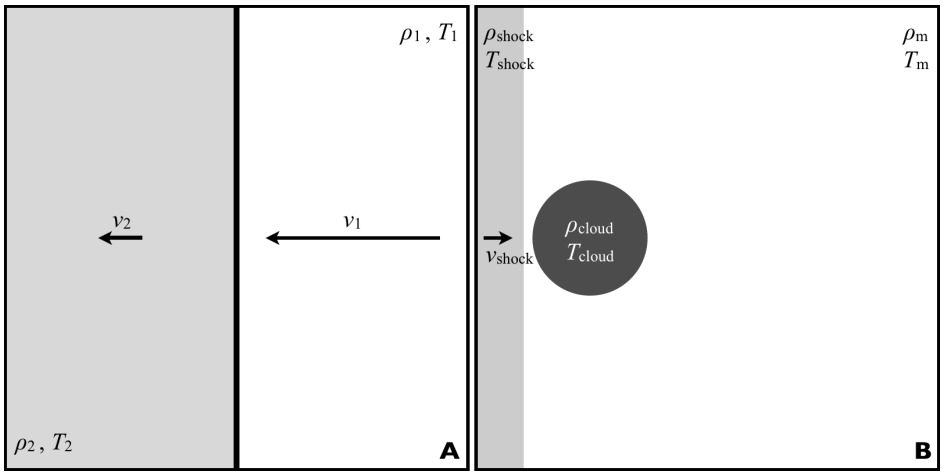


Figure 3.1: (a) Jump condition quantities in the frame of the shock front, represented by the vertical black line. (b) Depiction of cloud-crushing setup in the frame of the ambient medium. The medium and the cloud are both at rest.

The basic nature of our new problem type is to initialize a spherical cloud with constant overdensity, χ, embedded in an ambient medium of density ρ_m such that $\rho_{cloud} = \chi\rho_m$. We then send a shock wave with Mach number, $\mathcal{M} = v/c_s$ where c_s is the adiabatic sound speed with

$\gamma = 5/3$, in the cartesian x-direction toward the cloud. Additional user-defined parameters required to initialize the simulations include the radius of the cloud, $r_{\rm cloud}$, and the temperature of the cloud, $T_{\rm cloud}$. All of the remaining quantities needed to start the simulation are derived from the above mentioned user-specified parameters. These derived values include temperature of the ambient medium, $T_{\rm m}$, velocity of the shock, $v_{\rm shock}$, post-shock density, $\rho_{\rm shock}$, and post-shock temperature, $T_{\rm shock}$. We set $T_{\rm m}$ so that the cloud remains in pressure equilibrium, while the shock-related values are calculated using the Rankine-Hugoniot jump conditions, which when defined in the frame of the shock front, are given by:

$$\frac{\rho_2}{\rho_1} = \left[\frac{(\gamma+1)\,\mathcal{M}^2}{(\gamma-1)\,\mathcal{M}^2+2}\right], \tag{3.1}$$

$$\frac{v_1}{v_2} = \left[\frac{(\gamma+1)\,\mathcal{M}^2}{(\gamma-1)\,\mathcal{M}^2+2}\right], \tag{3.2}$$

$$\frac{T_2}{T_1} = \left[\frac{\rho_1}{\rho_2}\right]\left[\frac{2\gamma\mathcal{M}^2-(\gamma-1)}{(\gamma+1)}\right]. \tag{3.3}$$

Here, the subscripts 1 and 2 refer to the upstream and downstream quantities of the flow with respect to the shock front and $\mathcal{M} = v_1/c_{\rm s}$. See Figure 3.1a for a schematic of the flow. From these jump conditions, we shift from the rest frame of the shock front to the rest frame of the ambient medium to define the quantities used in our simulation. Under this transformation, the upstream and downstream values for density and temperature become the medium and shock quantities, respectively. However, the velocity of the inflowing material is defined as the difference between upstream and downstream values, $v_{\rm shock} \equiv v_1 - v_2$. This results in the following set of equations to define the initial conditions for the simulation:

$$\rho_{\rm shock} = \rho_{\rm m}\left[\frac{(\gamma+1)\,\mathcal{M}^2}{(\gamma-1)\,\mathcal{M}^2+2}\right], \tag{3.4}$$

$$T_{\rm shock} = T_{\rm m}\left[\frac{\rho_{\rm m}}{\rho_{\rm shock}}\right]\left[\frac{2\gamma\mathcal{M}^2-(\gamma-1)}{(\gamma+1)}\right], \tag{3.5}$$

$$v_{\rm shock} \;=\; c_{\rm s}\mathcal{M}\left(1-\frac{\rho_{\rm m}}{\rho_{\rm shock}}\right), \tag{3.6}$$

where the meanings of $\mathcal{M}$ and $c_{\rm s}$ remain unchanged and the flow is now represented by Figure 3.1b.

While this formulation of the cloud-crushing problem is sufficient to study the general properties of the shock-cloud interaction and compare to previous works, we made provisions for additional complexities worthy of investigation. In one avenue of investigation, we allowed the cloud to take on a variety of density profiles. While the simplest implementation is to define the cloud as a sphere of uniform density, which we explore in some of our simulations, we expect that the boundary between the ejecta clump and the ambient medium should be smoothly varying, analogous to the ISM clouds of Patnaude & Fesen (2005) and Nakamura et al. (2006). In addition, a "soft" cloud boundary reduces undesirable numerical effects commonly produced by a hard edge. Therefore, we define the cloud to consist of a smoothly varying envelope surrounding a uniform core for a fraction of our simulations. Specifically, we allow for two types of cloud envelopes — one given by a $1/r$ power-law fall-off,

$$\rho(r) \;=\; \begin{cases} \rho_{\rm core} & 0 \le r \le r_{\rm core} \\ \rho_{\rm core} + \left[\rho_{\rm m}-\rho_{\rm core}\right]\left[\frac{1/r-1/r_{\rm core}}{1/r_{\rm cloud}-1/r_{\rm core}}\right] & r_{\rm core} < r \le r_{\rm cloud} \end{cases}, \tag{3.7}$$

and another by a gaussian fall-off,

$$\rho(r) \;=\; \begin{cases} \rho_{\rm core} & 0 \le r \le r_{\rm core} \\ \rho_{\rm core}\left[\frac{\rho_{\rm m}}{\rho_{\rm core}}\right]^{\left(\frac{r-r_{\rm core}}{r_{\rm cloud}-r_{\rm core}}\right)^2} & r_{\rm core} < r \le r_{\rm cloud} \end{cases}, \tag{3.8}$$

where $r_{\rm core}$ is the inner radius at which the cloud is assumed to have a uniform density and $\rho_{\rm core} = \chi\rho_{\rm m}$ is the density of that core. In the case of a uniform cloud $\rho_{\rm core}$ and $\rho_{\rm cloud}$ are interchangeable as indicated in Figure 3.1b.

For all of the simulations presented in this paper, we have treated the shock as a constant inflow of material. This is representative of the scenario in which the width of the shock is much large than the radius of the cloud (effectively infinite). For the situation in which the shock is of

finite width and comparable in size to that of the cloud, it would be necessary to send a time-limited pulse of post-shock material. Such methods are being investigated for future work.

3.2.3 Cloud Tracking

Finally, we implement two additional features that allow us to track the evolution of the ejecta clump. First, we populate all of our clouds with a uniform density of tracer particles to represent dust contained in the ejecta material. This allows us to track the movement of the dust while the cloud is crushed, as well as the nature of the medium in which the dust is embedded. We can then post-process the history of each tracer particle to calculate dust destruction by assigning each particle some distribution of dust-grain properties. Second, we add a simple scalar field that is advected with the flow in the same manner as density, but does not alter the state of the system. We use this field to represent "cloud material", ξ, by initializing the values to be non-negligible only in cells that fall within the radius of the cloud. This field is motivated purely by computational necessity, and all cloud material values are dimensionless and devoid of physical meaning. To avoid potential round-off errors in the numerics, we set cells that are outside the cloud radius to be small but non-zero (10^{-20}). The non-negligible values are set to be dependent on the density of the cloud such that a cell with initial density $\rho_{\rm core}$ is assigned a cloud material value of $\xi = 1$. This means that a uniform density cloud would be filled with 1's while a cloud with a power-law or gaussian envelope would have 1's in the core and then a fall-off in cloud material that scales directly with the density profile. This field proves to be particularly useful as a refinement criterion for the AMR component of Enzo to ensure that the hydrodynamic evolution of the cloud is tracked with sufficient resolution. Section 3.3.2 discusses its uses as a refinement criterion.

3.3 The Simulations

3.3.1 Dimensionality and Comparison to Previous Works

Before addressing the ejecta-reverse shock interaction, we present simulations in which we demonstrate Enzo's ability to successfully handle the cloud-crushing problem. We also compare it to previous studies that investigated the interactions between the forward shock and the ISM. In this simulation, we construct a computational volume that consists of 512^3 cells and a physical size of 2.47×10^{18} cm (0.8 pc) on each side. We define the cloud of over-dense material to have a radius $r_{\rm cloud} = 3.09 \times 10^{17}$ cm (0.1 pc) and an over-density factor $\chi = 10$ in a higher resolution nested grid such that there are 128 cells per cloud radius, a physical resolution of 2.41×10^{15} cm ($\sim$161 AU) per cell edge. This choice is motivated by Klein et al. (1994) and Orlando et al. (2005) who found that this provides sufficient resolution to capture the physical properties of the problem. We then use inflow boundary conditions to send material with the appropriate shock values to produce a $\mathcal{M} = 10$ shock wave in the direction of the cloud. We allow the simulation to run for a time $t = 3t_{\rm cc}$, where $t_{\rm cc} = \chi^{1/2} r_{\rm cloud}/v_{\rm shock}$ is the cloud-crushing time as defined by Klein et al. (1994), at which point the cloud flows out of the domain. We find that this set of input parameters compares closely to simulations carried out by Bedogni & Woodward (1990), Stone & Norman (1992a), Klein et al. (1994), and Orlando et al. (2005) and therefore creates a simulation that can be compared to those previous works. We maintain the 2.41×10^{15} cm per cell edge resolution over the computational domain that encases the cloud for the full duration of the simulation.

In order to explore the effects of dimensionality on this problem, we investigate the evolution of this particular set up by starting first with a one-dimensional (1D) simulation and then progressing to higher dimensions. The 1D case corresponds to a scenario in which the inflowing shock impacts an effective "wall" of cloud material. This situation lends itself well to tracking the various transmitted and reflected shocks present in the problem as well as for identifying these features in higher dimension simulations. The first example of such shock propagation occurs when the inflowing shock impacts the cloud wall and transmits a slower shock through the cloud material and a

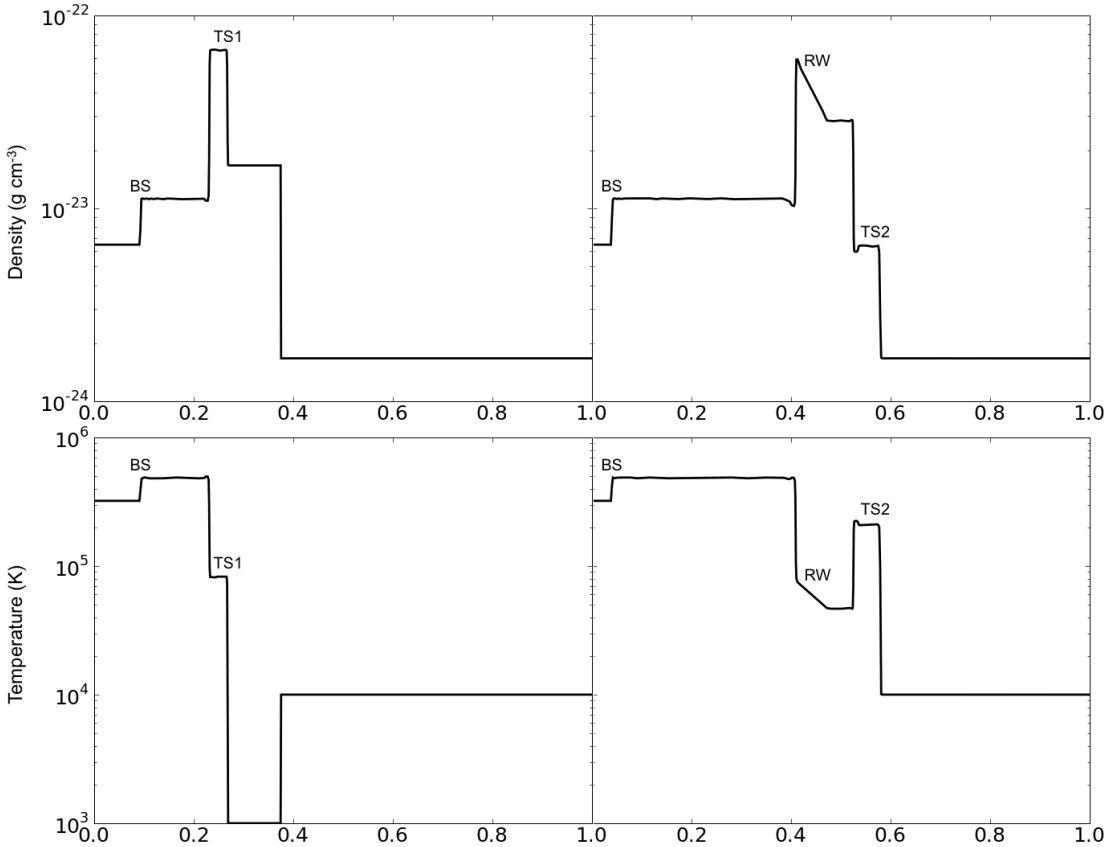


Figure 3.2: Density (top) and temperature (bottom) profiles for the 1D comparison simulation. The left column corresponds to $t = 0.52t_{\rm cc}$, and the right column corresponds to $t = 1.40t_{\rm cc}$, where $t = 0$ is when the shock first impinges upon the cloud edge. The labels BS, TS1, TS2, and RW mark the locations of the bow shock, first transmitted shock, second transmitted shock, and rarefaction wave respectively.

bow shock is reflected back towards the inflow. When the first transmitted shock crosses the back end of the cloud wall, another shock is transmitted into the ambient medium, while a rarefaction wave travels in the direction opposite the inflow through the newly-shocked cloud material. At this point, additional ringing occurs as waves bounce back and forth inside the cloud wall. See Figure 3.2 for density and temperature profiles of the 1D simulation at various points in time which display the above mentioned features.

For the two-dimensional (2D) case, we find many of the same features as in the 1D case, with the exception that the inflowing shock is capable of wrapping around and re-converging behind the cloud before the shock transmitted into the cloud traverses the full extent of the cloud. This is

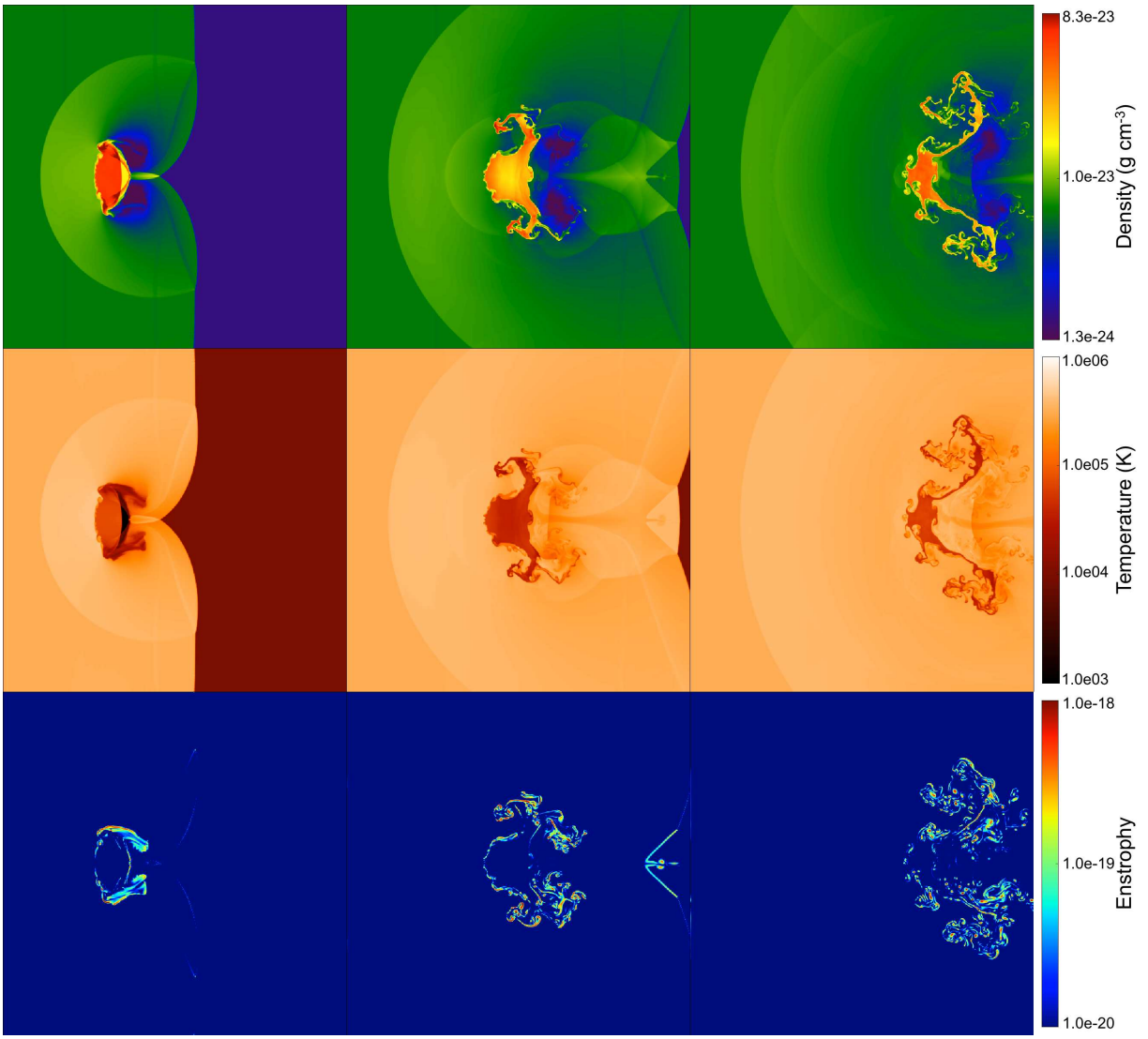


Figure 3.3: Density (top), temperature (middle), and enstrophy (bottom) for the 2D comparison simulation. The left column shows the state of the system at $t = 0.81t_{\rm cc}$, the middle at $t = 1.69t_{\rm cc}$, and the right at $t = 2.56t_{\rm cc}$

because the cloud no longer has the wall-like presence it had in the 1D case, and is now a circular object surrounded by ambient medium. As a result, an additional reverse shock is driven through the cloud opposite the flow as the inflowing shock envelops the cloud and exerts added pressure on all sides. The second dimension also allows for vortex creation, as material is stripped off the outer edges of the cloud and pushed into the evacuated region directly behind the cloud by the high pressure inflow material that surrounds it. In our 2D realization of this problem, we find qualitatively similar results to the work of Bedogni & Woodward (1990) — further confirmation that Enzo provides adequate machinery to study this problem. Figure 3.3 shows the results of this 2D run in density, temperature, and enstrophy, the square of the magnitude of the vorticity vector, $|\mathbf{w}|^2 = |\nabla \times \mathbf{v}|^2$.

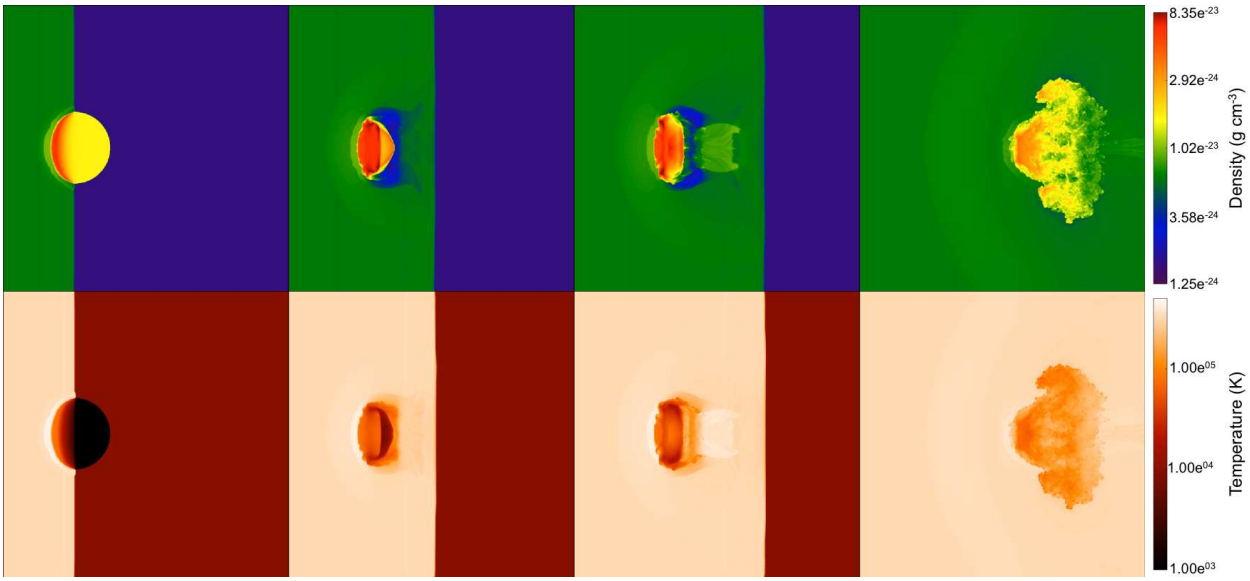


Figure 3.4: Density (top) and temperature (bottom) projected onto the x-y plane weighted by cloud material, ξ. Refer to Section 3.2.2 for a description of the cloud material. From left to right, the snapshots correspond to times $t = 0.23t_{\rm cc}$, $t = 0.72t_{\rm cc}$, $t = 1.02t_{\rm cc}$, and $t = 2.47t_{\rm cc}$, where $t = 0$ is when the shock first impinges upon the cloud edge.

In Figure 3.4, we see that the time evolution of the projected density for the three-dimensional (3D) simulation also compares favorably with previous works, particularly when we track the development of various features as a function of time. As in the 1D and 2D cases, when the inflowing shock first impacts the cloud, we see a slower shock transmitted into the cloud and a reflected bow

shock that propagates into the post-shock medium while the cloud becomes laterally flattened. By time $t = 0.75t_{\rm cc}$, the shock has flowed completely past the cloud and converged directly behind the cloud on the x-axis. At this point, the cloud is bathed in the high-pressure, post-shock medium and is compressed in all directions; an additional shock is driven back into the cloud opposite the direction of the inflow. Eventually, around $t = 1.2t_{\rm cc}$, the cloud begins to re-expand into the post shock medium owing to the compression caused when the shock first passed over it. Finally, for $t \gtrsim 2.0t_{\rm cc}$, the cloud is dominated by hydrodynamic instabilities and takes on a highly non-uniform, fragmented, and filamentary structure as it becomes comoving with the inflowing material. The evolution of vorticity in the 3D simulation becomes even more complex than the 2D case, as the third dimension allows for the stretching of vortex tubes and leads ultimately to a greater level of symmetry-breaking turbulence. All of these features were seen by Stone & Norman (1992a), Klein et al. (1994), and Orlando et al. (2005) at comparable moments in time. Figure 3.4 also shows the same evolution in projected temperature.

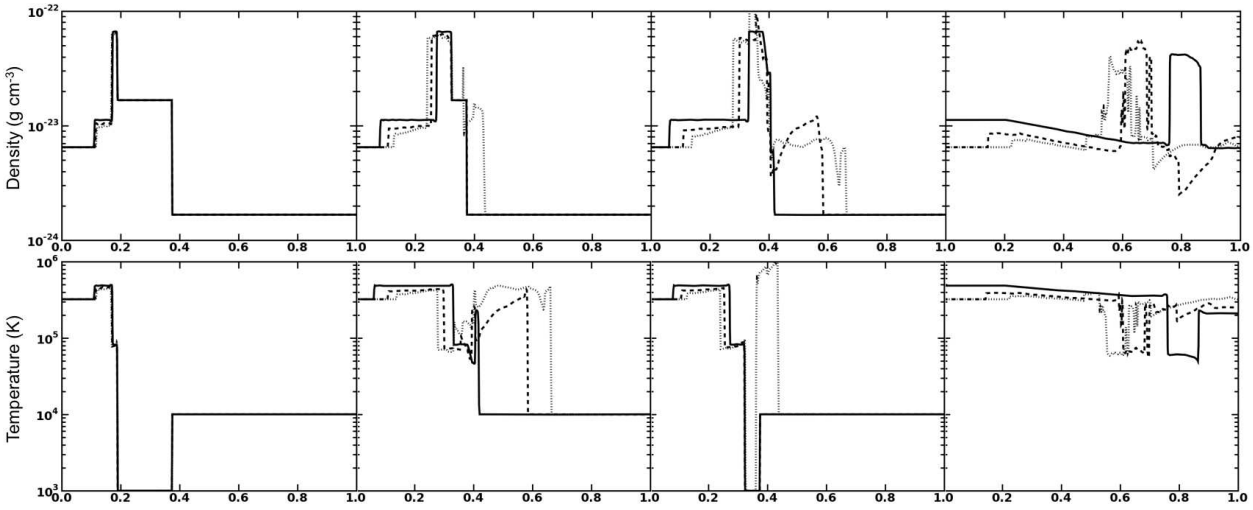


Figure 3.5: Density (top) and temperature (bottom) profiles for 1D (solid line), 2D (dashed line), and 3D (dotted line) comparison simulations. The x-axis is presented in code units. From left to right, the panels correspond to times $t = 0.23t_{\rm cc}$, $t = 0.72t_{\rm cc}$, $t = 1.02t_{\rm cc}$, and $t = 2.47t_{\rm cc}$.

To produce a more direct comparison of the three cases of varying dimensionality, we found it useful to look at 1D profiles of density and temperature as a function of time. For the 1D data,

these profiles are plots of the data as in Figure 3.2, while for the 2D and 3D simulations, they are found by tracing a path through the center of the computational domain that runs parallel to the x-axis. Figure 3.5 presents these profiles for select time steps. Upon inspection, the 2D and 3D cases are considerably more complex than the 1D case, but they still show similar evolution, especially in the time period before the inflowing shock wraps around the cloud in the 2D and 3D cases. We also point out that the cloud material moves across the computational domain more rapidly for lower dimension simulations. This can be explained by the fact that the momentum imparted by the planar shock has fewer degrees of freedom to transport into. For the 1D case, all of the material is forced to move in the x-direction by the inflow, but in the 2D and 3D cases it can be diffused into the y- and z-directions.

3.3.2 Reverse Shocks and Ejecta Clumps

While the simulations we use to study the interaction between a clump of ejecta and a supernova remnant reverse shock are qualitatively similar to the case of a forward shock penetrating the ISM, the values of the input parameters are considerably different. In the case of the reverse shock impacting newly formed ejecta, the relative velocity between the cloud and the shock wave is generally an order of magnitude higher ($\sim 10^3$ km s^{-1}) than the shock-ISM case ($\sim 10^2$ km s^{-1}), and the overdensity of the ejecta clump is on the order of $\chi \sim 1000$ versus the $\chi \sim 10$ values assumed for ISM clouds.

In addition to the modifications of the input parameters, we also made use of the variable cloud structure to explore what role, if any, the density profile of the cloud played in determining the amount of dust that would be sputtered as the ejecta are shocked. Using the full array of configurable options in our cloud-crushing problem type (shock width, cloud structure, radiative cooling, cloud overdensity, shock velocity), we ran a grid of simulations to determine the parameters that strongly influence the dust destruction induced by the shock-cloud interaction. Specifically, we investigate various combinations of power-law and gaussian envelopes; cloud over-densities of $\chi = 100$ and 1000; shock velocities of 1000, 3000, and 5000 km s^{-1}; and no cooling and the analytic

cooling of Sarazin & White (1987). For a list of the simulations and their corresponding parameters refer to Table 3.1.

All of the simulations have root-grid dimensions of 512×256×256, which provides a physical resolution of 1.25×10^{15} cm (∼84 AU) per cell edge. At this level, the radius of the cloud spans 8 cells for a physical radius of 10^{16} cm. In addition to the base-level resolution, we include a problem-specific refinement criterion that allows for two additional levels of resolution. Cell refinement is controlled using the previously mentioned cloud material field and flags cells for refinement based on a dimensionless cloud material "mass", $m_\xi = \xi V_{\rm cell}$, where $V_{\rm cell}$ is the volume of the cell. To determine whether or not a cell should be flagged it must have a cloud material mass greater than $m_{\rm flag} = m_{\rm min} r^{l\alpha}$, where $m_{\rm min}$ is the minimum mass for refinement, r is the refinement factor, l is the current refinement level of the cell, and α is a tunable exponent used to control whether the refinement acts in a Lagrangian ($\alpha = 0$), super-Lagrangian ($\alpha < 0$), or sub-Lagrangian ($\alpha > 0$) manner. For all simulations presented in this paper, we use $m_{\rm min} = 10^{-8}$, $r = 2$, and $\alpha = 0$, where $m_{\rm min}$ is a dimensionless number. By allowing the grid to refine to two levels beyond the root grid, we achieve a physical resolution of 3.125×10^{14} cm (∼21 AU) on the highest level. Although these reverse-shock simulations do not have the same number of cells per cloud radius as presented in the comparison case, we find that this particular combination of initial root-grid resolution, cloud material refinement criterion, and two-level refinement is sufficient to track and study the properties of the interactions between the reverse shock and the ejecta cloud. Additionally, the final dust masses do not change significantly with increased resolution.

We set the duration of our reverse-shock simulations based on the longest computationally manageable timescale for the parameters that produce the longest cloud-crushing time. For the above-mentioned parameter space, this corresponds to simulations with $r_{\rm cloud} = 10^{16}$ cm, $\chi = 1000$, and $v_{\rm shock} = 1000$ km s^{-1}, giving $t_{\rm cc} \approx 100$ yrs. With this constraint, we are able to run these simulations to a physical time of $t \approx 10^3$ yrs $\approx 10 t_{\rm cc}$. Therefore, we run all simulations to a minimum of 10 cloud-crushing times which results in varied physical times.

Owing to the nature of the inflowing shock and relevant time scales, we encountered a possible

Table 3.1. Simulations Parameters

Simulation	Cloud Envelope [a]	χ	$v_{\rm shock}$ (km s^{-1})	$\mathcal{M}$	$t_{\rm cc}$ (yrs)	Cooling [b]
1	Gaussian	100	1000	27.83	31.69	None
2	Gaussian	100	1000	27.83	31.69	SW87
3	Gaussian	100	3000	83.49	10.56	None
4	Gaussian	100	3000	83.49	10.56	SW87
5	Gaussian	100	5000	139.15	6.34	None
6	Gaussian	100	5000	139.15	6.34	SW87
7	Gaussian	1000	1000	8.80	100.21	None
8	Gaussian	1000	1000	8.80	100.21	SW87
9	Gaussian	1000	3000	24.60	33.40	None
10	Gaussian	1000	3000	24.60	33.40	SW87
11	Gaussian	1000	5000	44.00	20.04	None
12	Gaussian	1000	5000	44.00	20.04	SW87
13	Power-law	1000	1000	8.80	100.21	None

[a]For the Gaussian and Power-law envelopes, $r_{\rm core} = 0.7 r_{\rm cloud}$

[b]For the cooling, "SW87" represents analytic formula by Sarazin & White (1987)

simulation limitation in which the cloud material was often blown out of the back end of the computational domain by the shock medium. We solved this limitation and kept the majority of the cloud material in the simulation box for the desired duration by allowing the simulations to run up to a point in which the cloud is nearly comoving with the inflowing shock material and then shift into the inflow frame. We execute this by pausing the calculation at a suitable moment in time, subtracting off the inflow velocity, $v_{\rm shock}$, from the x-velocity of every cell, and resuming the simulation. After the simulation is restarted, the cloud evolves mostly statically in terms of bulk motion. Since the temperatures and densities remain unaltered, the cloud's relative motions are identical to what they would have been in the initial rest frame.

In studying the evolution of these "ejecta-crushing" simulations, we find that at early times the simulations agree with the previously discussed 3D comparison simulation, with the exception that the smoothly varying envelope subdues the interactions between the cloud boundary and the inflowing material. The lack of a sharp density discontinuity decreases the development of features perpendicular to the direction of the inflow and the "wings" present in the comparison simulation are not seen. Instead, the outer envelope material is stripped away, allowing the remaining compressed core to expand adiabatically into the surrounding medium. Also, since we run these simulations for more cloud-crushing times, we see the continued destruction imparted upon the cloud by the constant inflow of material as it gets progressively more shredded. After approximately five cloud-crushing times, the relative velocities between the cloud material and shock material drop to $\sim 0.1 v_{\rm shock}$ as the cloud becomes entrained in the flow. By $t \sim 8 t_{\rm cc}$, the cloud has become well mixed with the inflowing material, with density variations on the few percent level.

One of the most striking differences between the simulations, with and without cooling, is the formation of cold, dense nodules in the simulations with cooling turned on. The nodules appear to be amplifications of over-densities created in the cloud. The regions of high density have sufficiently shortened cooling times and therefore cool and collapse approximately isothermally. As a result, the dense knots are able to persist through their bombardment with inflowing material for a longer fraction of the simulation duration.

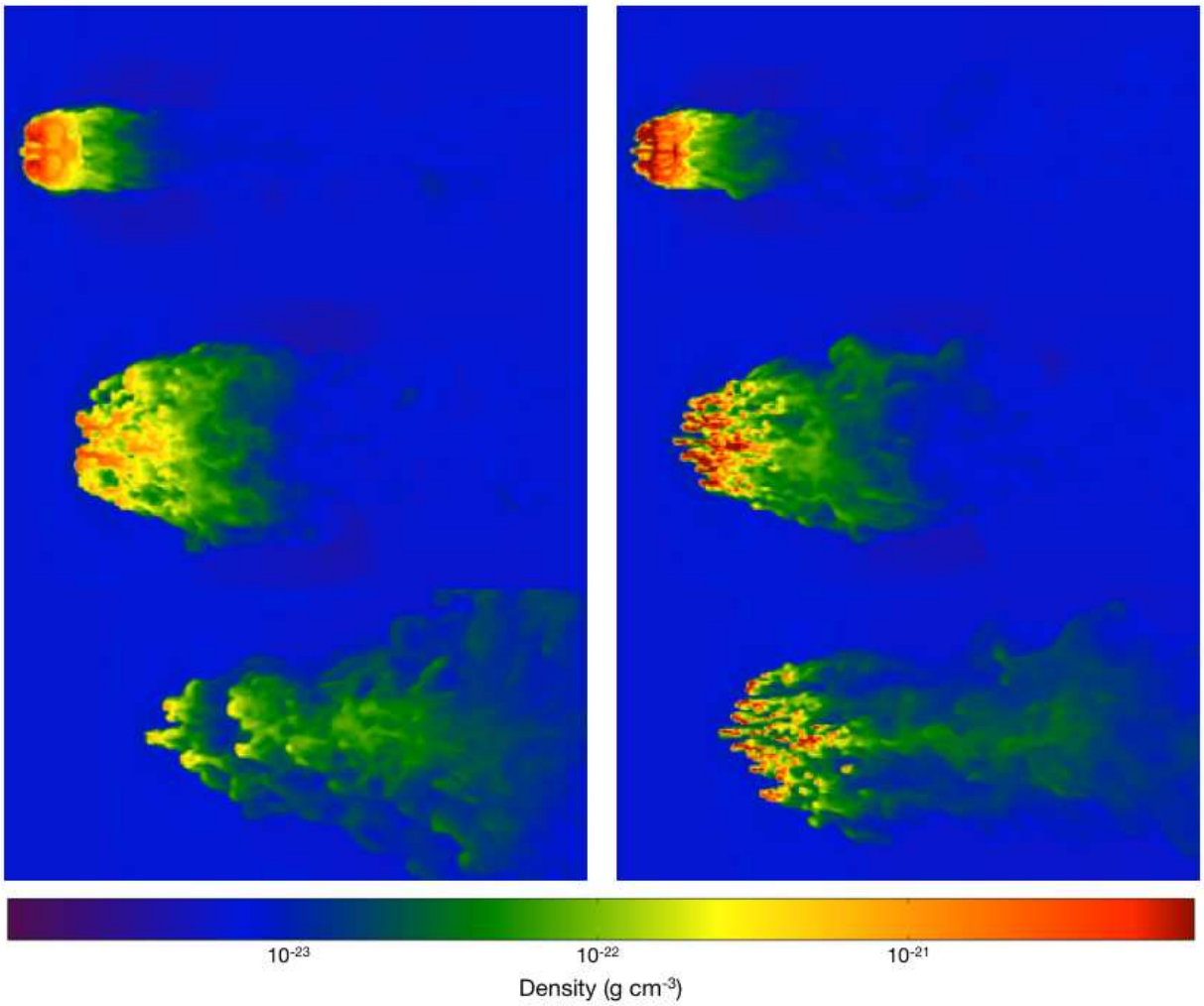


Figure 3.6: Density projections in the x-z plane weighted by cloud material, ξ, for the case of $\chi = 1000$ and $v_{\rm shock} = 1000$ km s^{-1}. The left column shows the evolution for a non-cooling simulation, while the right column shows that for a cooling simulation. From top to bottom, the time progresses in the following order: $t = 1.8t_{\rm cc}$, $t = 2.4t_{\rm cc}$, $t = 3.0t_{\rm cc}$.

Figure 3.6 shows the evolution of the ejecta clump to roughly one-third of the full simulation time. Beyond that point, the cloud continues to expand and mix with the shocked medium. The formation of high density nodules is apparent in the right column of Figure 3.6. We note that Raga et al. (2007) found similar dense protrusions in their 3D simulations of radiative, interstellar bullet flow. The number of nodules that form tends to result from the balance between shock-heating and cooling. For higher shock velocities, and hence higher temperatures, the number of persisting over-densities decreases.

When we run the cooling simulations with an additional level of refinement, we generally find these dense knots to be smaller and higher in number. This result is expected, as the turbulence created by the shock passing through the cloud will naturally cascade down to the smallest physical scale, which is determined by the resolution of the simulation. Additionally, these nodules are constantly being eroded by inflowing material, and at late times their size will be comparable to the grid-cell size. However, despite the difference in size and count, these small knots evolve similarly in density and temperature, which results in nearly identical dust destruction.

3.4 Data Post-processing

3.4.1 Modeling Grain Distributions Using Tracer Particles

As mentioned previously, the clouds in all of our simulations are populated with a uniform distribution of tracer particles. We use 4096 particles in order to adequately sample the cloud while simultaneously keeping the computational requirements of our post-processing analysis manageable. Each individual particle is assigned a full distribution of grain radii and therefore represents a distinct population of dust. The radius distributions used in all of our analysis come from the values calculated by Nozawa et al. (2003) for a core-collapse supernova (CCSN) with a progenitor mass of 20 $M_{\odot}$, which have been reproduced in Figure 3.7. Since we assume that the number density of dust grains in the cloud should scale with the surrounding gas density, we scale the relative abundance of grains in the initial distributions by the density of the surrounding medium that the tracer particles are initially embedded in. Given this assumption, $f(a) = f_0(a)(\rho_{\rm tp}/\rho_{\rm max})$, where a is the grain radius, $f_0(a)$ is the unscaled abundance, $\rho_{\rm tp}$ is the density of the surrounding medium for a given tracer particle, and $\rho_{\rm max}$ is the value for the most dense medium in which a tracer particle is initialized. In general, $\rho_{\rm max} = \rho_{\rm core}$. To determine the distribution of grain radii for all the dust contained in the simulation, we sum the radius distributions for all 4096 tracer particles.

We can also use the tracer particles and their respective grain distributions to calculate a

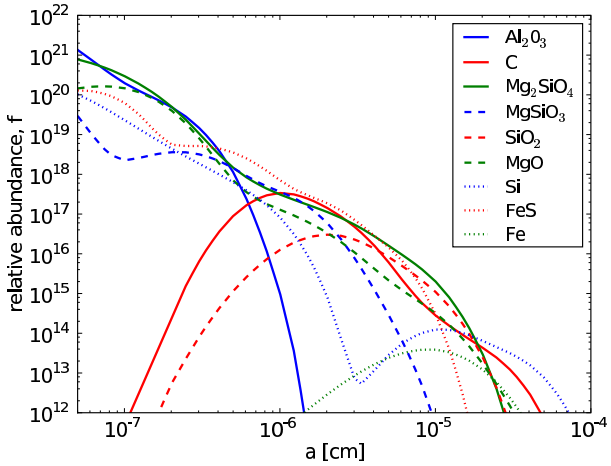


Figure 3.7: Reproduced grain radius distributions for the species expected in the unmixed ejecta model of a core-collapse supernova with a progenitor mass of 20 $M_{\odot}$ as calculated by Nozawa et al. (2003). In situations where data from Nozawa et al. (2003) ended prior to a grain radius of 5×10^{-8} cm, we carried out a linear extrapolation to smaller grain radii to make the data continuous.

proxy for the amount of dust mass in our simulation. First, we define the number density of grains between $a - \delta a$ and a to be $n(a) = f(a)\delta a$, where δa is set by the number of bins used to track the distribution. Second, we convert this number density to the mass proxy by multiplying by a^3. By summing the amount of mass in each radius bin, we can determine the amount of mass contained in a given tracer particle and then sum tracer particles to get the total dust mass in the simulation. Rather than work in absolutes and include the densities of particular grain species, we leave this mass proxy as a relative value and track how it evolves from $t = 0$ to $t = t_{\rm final}$.

In the same way that it was difficult to keep all of the cloud material inside the finite computational domain, some fraction of the tracer particles also flow off the grid, particularly ones that start out in the cloud envelope. Since we have no means of tracking the environment that those tracer particles would exist in after the point where they leave the bounds of the simulation, we exclude them from our sputtering calculations for the full duration of the run. In some simulations the combination of input parameters leads to vigorous cloud expansion that cannot be confined to our computational domain for the desired duration. This results in situations in which large fractions of the tracer particles (>50%) flow beyond the box boundaries. To address this fact, we halt the post-processing analysis at the point when the dust mass contained in the lost tracer

particles exceeds 5% of the initial dust mass. While we would like to carry out our analysis for all simulations on the same timescales, creating simulation domains that are large enough to contain the cloud's expansion for all of parameter space was too computationally expensive to be feasible. Despite this limitation, we still find our results up to the 5% cut-off to be robust.

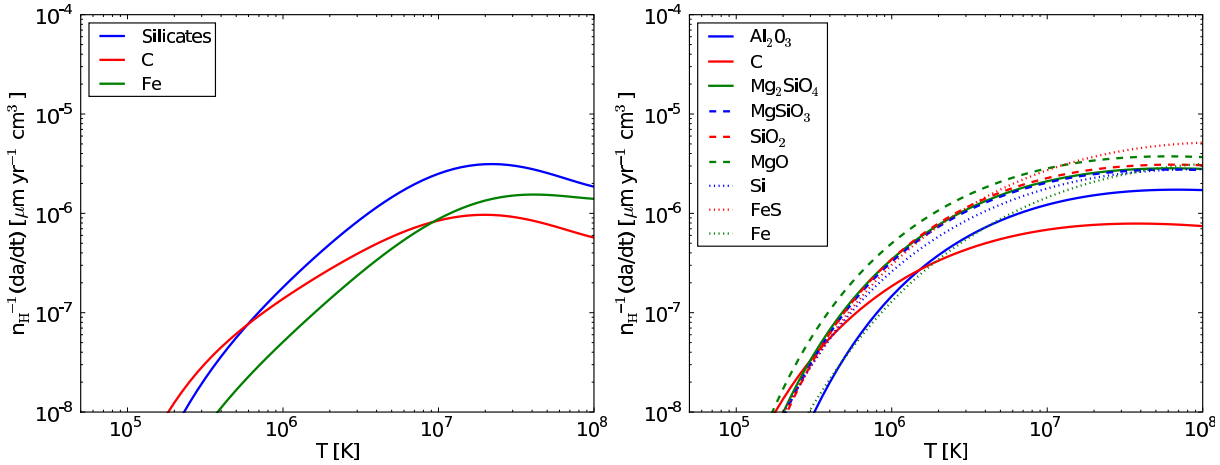


Figure 3.8: Our adopted thermal sputtering rates based on the polynomial fits of Tielens et al. (1994) (top) and for an elemental composition of $Z = Z_\odot$ as calculated by Nozawa et al. (2006) (bottom).

3.4.2 Dust Sputtering

Having established our initial grain distributions, we post-process the grains using generalized fifth-order polynomial fits to thermal sputtering from Tielens et al. (1994) and tabulated thermal sputtering rates. These sputtering rates have been reproduced in Figure 3.8. We pass the density and temperature of the environment of a given tracer particle to one of the sputtering methods and calculate the erosion rate, da/dt, for each time step. We multiply the erosion rate by the length of the time step and subtract that value from the initial grain radius such that $a_{\rm new} = a_{\rm old} - (da/dt)\Delta t$. This method allows each population of dust (each tracer particle) to evolve in direct response to its environment for the duration of the simulation. Using the method mentioned above, we then compute the percentage of dust mass lost as a function of time. While Tielens et al. (1994) provide sputtering rates for SiC and H_2O grains, we do not track them in our analysis because Nozawa

et al. (2003) do not include size distributions for these grains. In addition, since Tielens et al. (1994) used SiO_2 as the basis for the sputtering rates of silicates, we use the grain distribution of SiO_2 as our starting point for silicate sputtering.

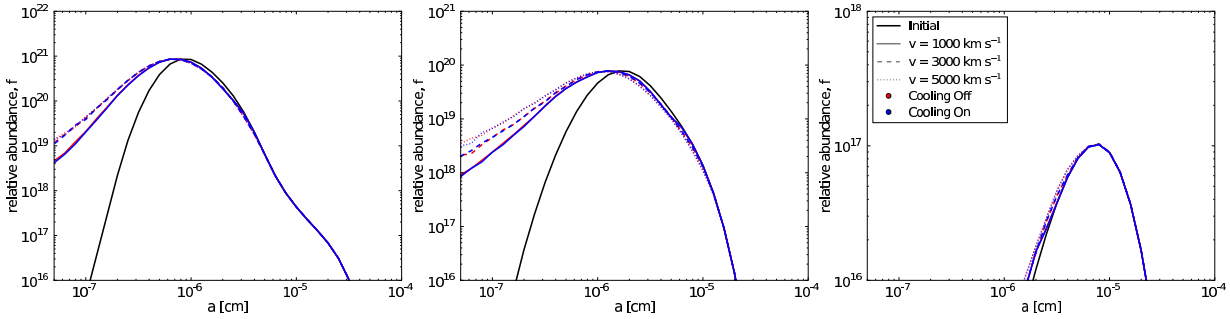
3.5 Results

We present the sputtering results for all 13 simulations listed in Table 3.1. Specifically, we study the changes in grain radius distributions from their starting to ending values and the evolution of the dust mass contained in the tracer particles as a function of time. For a condensed view of the key focus of this paper, dust survival rate, for all combinations of simulations and sputtering methods, we refer the reader to Table 3.2. One caveat is that Simulations 8 ($\chi = 1000$, $v_{shock} = 1000$ km s^{-1}, cooling on), 9 ($\chi = 1000$, $v_{\rm shock} = 3000$ km s^{-1}, cooling off), and 10 ($\chi = 1000$, $v_{shock} = 3000$ km s^{-1}, cooling on) do not span the full range of cloud-crushing times because they ended up losing more than 5% of their dust mass due to tracer particles that left the computational domain. However, since the dust destruction in Simulations 9 and 10 hits a plateau before the simulation analysis ends, the results of these simulations can still be fairly accurately compared. Unfortunately, Simulation 8 terminates well before any such plateau, leading to misleading final dust mass values. In addition, we do not devote any significant discussion to Simulation 13 (power-law density envelope, $\chi = 1000$, $v_{\rm shock} = 1000$ km s^{-1}, cooling off) because the dust mass evolution proved to be nearly identical to Simulation 7 (gaussian density envelope with the same initial conditions), as can be seen in Table 3.2.

3.5.1 Grain Distributions

In order to get a handle on the nature of the dust grain sputtering, we first inspect the changes in the distributions of grain radius. We find that, regardless of species, grains with radii less than a few times 10^{-6} cm are sputtered to much lower radii and are often completely destroyed. However, given the time scale of our simulations, larger grains are capable of surviving the full duration of the sputtering, but are shifted to lower radii. For simplicity, we defined destroyed grains as those

that are sputtered to less than 5×10^{-8} cm.

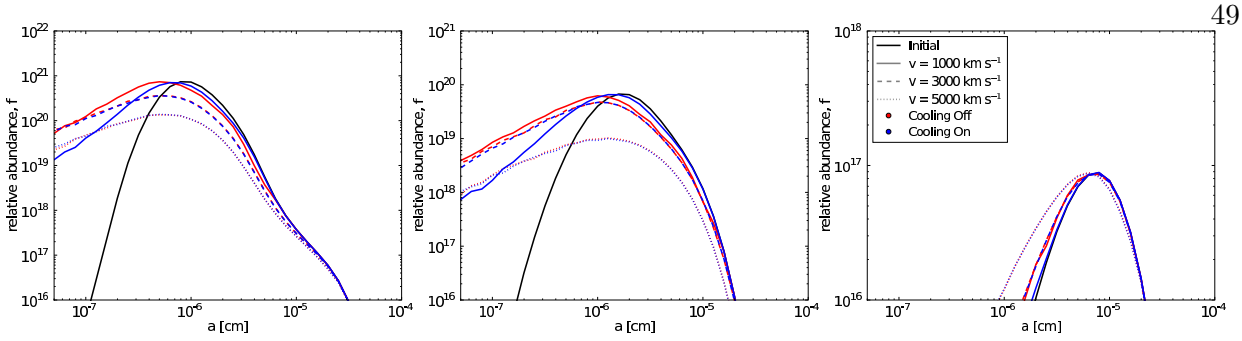


(a) C (b) Silicates (c) Fe

Figure 3.9: Grain radius distributions for three of the grain species in Tielens et al. (1994) for simulations with over-density $\chi = 100$. The black line represents the initial distribution, while the colored lines indicate final distributions. Red signifies simulations with no cooling and blue shows the analytic cooling of Sarazin & White (1987). Solid colored lines indicate simulations with $v_{\rm shock} = 1000$ km s^{-1}, dashed are $v_{\rm shock} = 3000$ km s^{-1}, and dotted are $v_{\rm shock} = 5000$ km s^{-1}. The grains in this figure were sputtered using the rates from Tielens et al. (1994).

In the figures that follow we look at the evolution of grain radius distribution for various locations in parameter space and for two of the three sputtering rate options. In each panel, the solid black line illustrates the initial grain distributions as set by the distributions from Nozawa et al. (2003). For the colored lines, solid lines correspond to simulations with the slowest shock velocities, $v_{\rm shock} = 1000$ km s^{-1}, dashed lines are the simulations with shock velocities of $v_{\rm shock} = 3000$ km s^{-1}, and dotted lines are the simulations with the highest velocity shocks, $v_{\rm shock} = 5000$ km s^{-1}. The colors represent our different options for cooling, where red highlights runs with no cooling and blue highlights runs with the analytic cooling function of Sarazin & White (1987). All figures adhere to the same scheme.

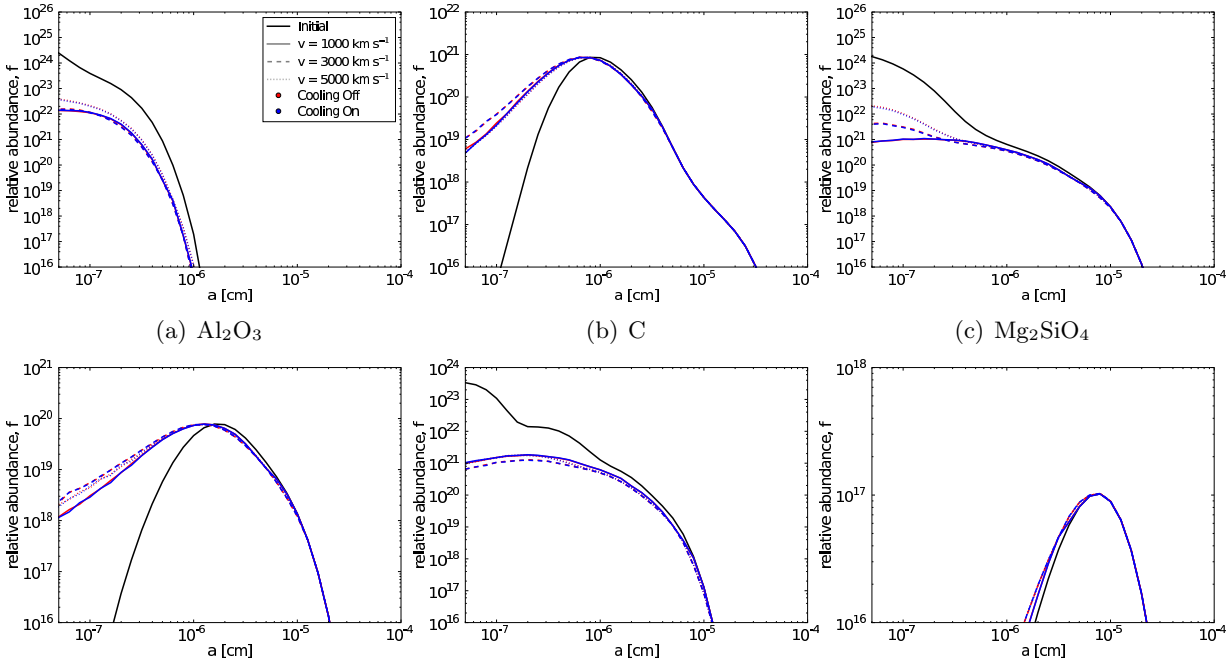
In Figures 3.9 and 3.10 we see the evolution in three grain species using the sputtering rates from Tielens et al. (1994) for density contrasts $\chi = 100$ and $\chi = 1000$ respectively. In both the low-density and high-density clouds, the higher shock velocities are more effective at reducing the size of large grains. In addition, for $\chi = 1000$, the increased density leads to even more dust destruction, with a severe drop off in the small grains as a function of shock velocity. When we inspect the grain distributions as a function of cooling (red vs. blue), we see little difference between the simulations.



(a) C (b) Silicates (c) Fe

Figure 3.10: Same as Figure 3.9, but for over-density $\chi = 1000$.

Since the differences are more apparent when looking at dust mass, we discuss these differences in the following section.



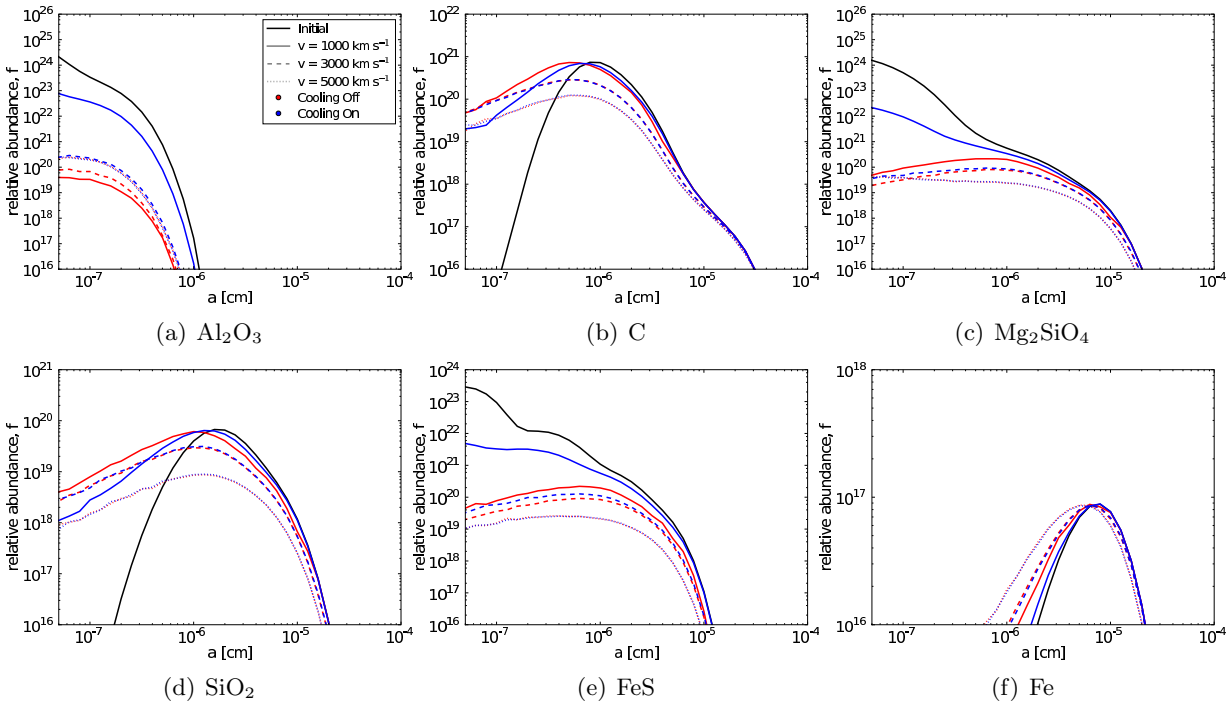
(a) Al_2O_3 (b) C (c) Mg_2SiO_4

(d) SiO_2 (e) FeS (f) Fe

Figure 3.11: Grain radius distributions for the six of the grain species in Nozawa et al. (2003) for simulations with $\chi = 100$. The line styles and colors have the same meaning as Figure 3.9. The grains in this figure were sputtered using the $Z = Z_\odot$ rates from Nozawa et al. (2006).

We show the results of sputtering using the rates from Nozawa et al. (2006) for over-densities of 100 and 100 in Figures 3.11 and 3.12 respectively. In the case of $\chi = 100$, the number of small

grains sputtered out of existence does not show a clear correlation with shock velocity. In some cases, a slower shock destroys more grains than a faster shock. This is likely a result of the cloud being shredded more rapidly in the high-velocity cases, which reduces the density of the cloud material and shuts off sputtering. There appears to be no significant dependence on whether or not cooling was employed in the simulation. However, we predict that, with higher metal abundances, the differences between cooling and non-cooling simulations should become more pronounced. For the simulations with $\chi = 1000$, grain sputtering depends strongly on shock velocity, as all grains sizes are sputtered more heavily by higher velocity shocks. Again, we do not see a clear distinction between cooling and non-cooling simulations, even in higher density regimes. There does seem to be a large discrepancy for the slowest shock velocity, but this stems from the fact that the cooling simulation for that shock velocity did not make it to ten cloud-crushing times. The differences between simulations with and without cooling can better be seen in Section 3.5.2, when we look at evolution in dust mass.



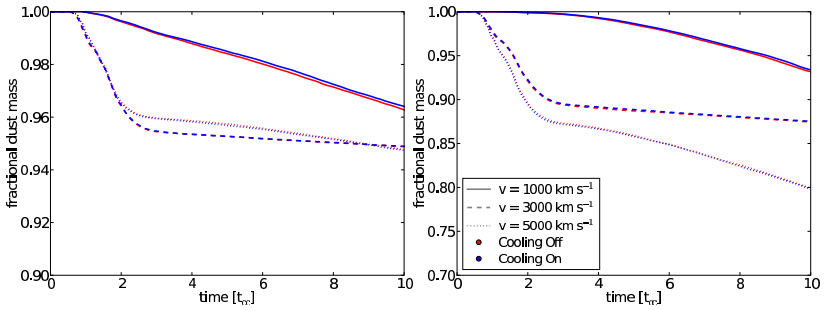
(a) Al_2O_3 (b) C (c) Mg_2SiO_4

(d) SiO_2 (e) FeS (f) Fe

Figure 3.12: Same as Figure 3.11, but for over-density $\chi = 1000$.

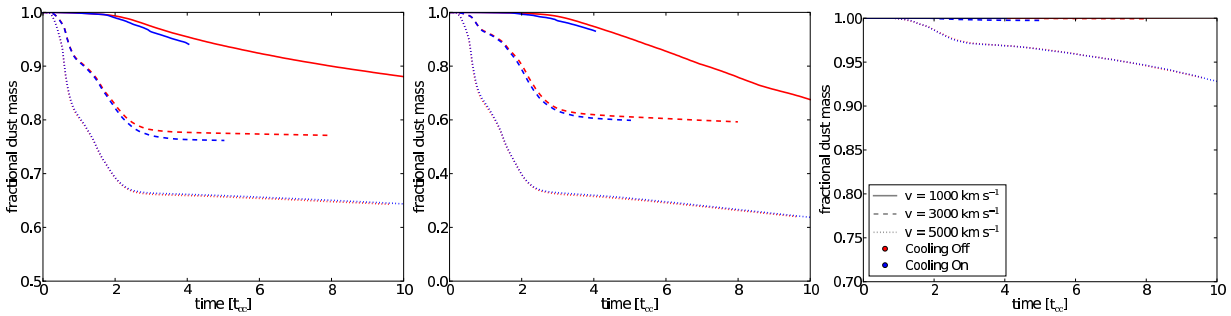
3.5.2 Dust Mass Evolution

Using the sputtered grain distributions, we convert them into relative dust masses using the method described in Section 3.4.1. In the following figures, we continue with the same color and line style convention, but instead of radius distributions we consider the fractional dust mass of each grain species as a function of cloud-crushing times. In these plots, any lines that end before 10 cloud-crushing times are terminated because more than 5% of the original dust mass contained in the particles was lost due to particles that crossed the simulation boundaries. In addition, for the lower density cloud, effectively zero sputtering of iron grains occurred, and we have omitted those plots.



(a) C (b) Silicates

Figure 3.13: Dust mass evolution for two of the grain species in Tielens et al. (1994) for simulations with $\chi = 100$. Colors and lines styles are the same as Figure 3.9. The grains in this figure were sputtered using the rates from Tielens et al. (1994).

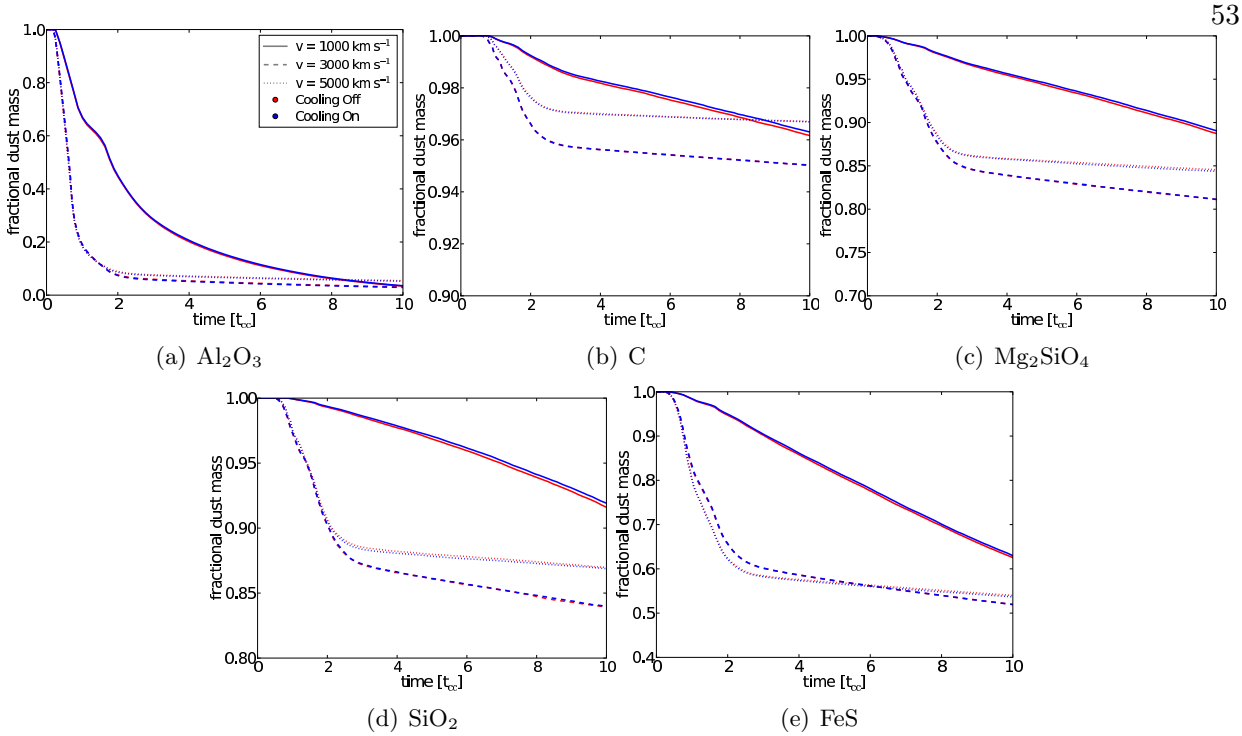


(a) C (b) Silicates (c) Fe

Figure 3.14: Same as Figure 3.13, but for over-density $\chi = 1000$. In addition, we include the evolution of Fe grains.

We again look first at the results when we use the sputtering rates of Tielens et al. (1994). Following the previous order, Figure 3.13 shows the mass evolution for $\chi = 100$, and Figure 3.14 shows the same for $\chi = 1000$. For the low over-density cloud, with the exception of Fe grains, which were negligibly sputtered, we find the scatter in percentage of dust remaining at the end of the simulation is rather small. However, the path that each simulation takes to get to its final dust mass varies based on the shock velocity. The high-velocity shock runs show a rapid drop in dust mass and then dust destruction slows considerably, while the dust mass in the low velocity shock runs gradually falls to its final level. In the higher over-density cloud, the dependence on shock velocity is much more pronounced. Higher shock velocities lead to greater dust destruction – as much as 50% additional destruction for certain grain species. Comparing the red and blue lines in the cooling versus non-cooling simulations, we only see differences in the higher density cloud, after the initial drop in dust mass. At later times, we detect a slight splitting in the dust mass evolution tracks in which the simulations with cooling turned on appear to destroy a small fraction of additional dust. Our explanation for the diverging behavior is that if the medium is capable of cooling, it will slightly drop the pressure in the expanding cloud and keep the medium in which dust is embedded at a higher density. Since the cloud has been raised to high shock temperatures at late times, where the sputtering rates are relatively constant with temperature, differences in density will play the dominant role in controlling sputtering.

We carry out the same inspection for Nozawa et al. (2006) in Figure 3.15 ($\chi = 100$) and Figure 3.16 ($\chi = 1000$). We immediately see that, similar to the results from the polynomial sputtering rates, the variation in final dust mass is fairly small for the low-density cloud. We also see that the trend of faster sputtering at early times for higher velocities shocks holds true in these cases as well. One particular difference between the results from these sputtering rates and those of Tielens et al. (1994) is that, for the high-velocity shocks, the fractional dust mass seems to plateau at late times, as where in the earlier figures the dust mass evolution still has a negative slope at the end of the simulation. This may indicate that longer simulation run times are required for the simulations that did not stop evolving. For the higher density cloud, we again see that the final



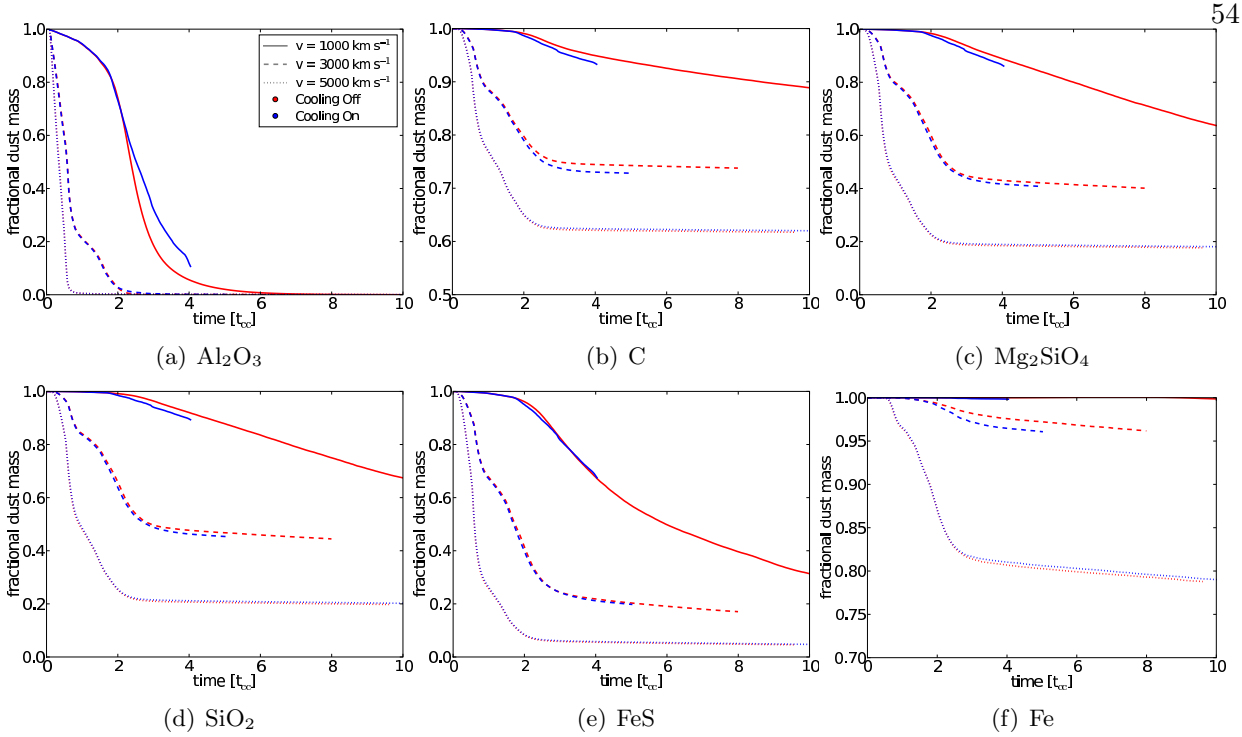
(a) Al_2O_3 (b) C (c) Mg_2SiO_4

(d) SiO_2 (e) FeS

Figure 3.15: Dust mass evolution for five of the grain species in Nozawa et al. (2003) for simulations with $\chi = 100$. Colors and line styles are the same as Figure 3.13. The grains in this figure were sputtered using the $Z = Z_\odot$ rates from Nozawa et al. (2006).

dust mass fractions are much more widely varied. For the sputtering rates of Nozawa et al. (2006), we see a non-negligible amount of iron sputtering for the medium velocity shock simulations of the high-density cloud which is not present in the polynomial sputtering rate calculations. We also see the same effects of cooling in the higher density cloud mentioned previously. In addition, one interesting difference between the non-cooling and cooling simulations shows up most prominently in the evolution of Al_2O_3 (see Figure 3.16). The non-cooling simulation appears to have a more rapid fall off in dust mass for the case of $v_{\rm shock} = 1000$ km s^{-1} than the cooling simulation. However, even though the cooling simulation did not reach ten cloud-crushing times, it seems unlikely that it would not have also arrived at total Al_2O_3 grain destruction. Since this particular dust species consists of only small grains, it is possible that the slight drop in plasma temperature caused by the cooling is able to slow the grain destruction, despite the increased density effect mentioned above.

Finally, in an effort to summarize the results of all 13 simulations, Table 3.2 lists the final



(a) Al_2O_3 (b) C (c) Mg_2SiO_4

(d) SiO_2 (e) FeS (f) Fe

Figure 3.16: Same as Figure 3.15, but for over-density $\chi = 1000$. In addition, we include the evolution of Fe grains.

dust mass fraction for all species for all simulations. In reviewing these results, we find that grain species that have initial distributions weighted toward large grain radii have higher survival rates over 10 cloud-crushing times. Furthermore, since the sputtering rates scale linearly with density, simulations with higher density clouds all show substantial sputtering. The higher density clouds also tend to show a much stronger scaling with shock velocity than do their low-density counterparts. There appears to be no strong dependence on the two sputtering rate methods, most differences are on the tenths of a percent to few percent level. While the simulations including cooling did show small deviations from those simulations without cooling in the dust mass evolution plots, in most cases the end results for surviving dust mass fraction are indistinguishable.

3.6 Summary and Future Work

We summarize the above the results into a few key points:

Table 3.2. Final Dust Mass Fraction

Sim.	Al_2O_3	C	Mg_2SiO_4	$MgSiO_3$	SiO_2	MgO	Si	FeS	Fe
1	0.034	0.963 / 0.962	0.888	0.423	0.932 / 0.916	0.838	0.998	0.626	1.000 / 1.000
2	0.036	0.964 / 0.963	0.891	0.428	0.934 / 0.919	0.842	0.998	0.631	1.000 / 1.000
3	0.030	0.949 / 0.950	0.811	0.334	0.875 / 0.839	0.774	0.998	0.520	1.000 / 1.000
4	0.029	0.949 / 0.950	0.811	0.333	0.875 / 0.840	0.775	0.998	0.519	1.000 / 1.000
5	0.054	0.948 / 0.967	0.846	0.394	0.799 / 0.870	0.811	0.998	0.540	1.000 / 1.000
6	0.052	0.947 / 0.967	0.844	0.392	0.798 / 0.869	0.809	0.998	0.537	1.000 / 1.000
7	0.000	0.880 / 0.889	0.637	0.139	0.676 / 0.674	0.615	0.997	0.314	1.000 / 0.999
8*	0.107	0.941 / 0.933	0.861	0.452	0.930 / 0.893	0.813	0.996	0.674	1.000 / 0.998
9	0.000	0.771 / 0.738	0.401	0.044	0.593 / 0.378	0.445	0.971	0.171	0.999 / 0.962
10	0.001	0.762 / 0.728	0.408	0.054	0.598 / 0.453	0.382	0.965	0.198	0.997 / 0.961
11	0.001	0.643 / 0.617	0.177	0.014	0.240 / 0.197	0.181	0.853	0.046	0.932 / 0.789
12	0.001	0.644 / 0.620	0.181	0.015	0.237 / 0.202	0.185	0.854	0.047	0.928 / 0.790
13	0.000	0.882 / 0.890	0.637	0.140	0.676 / 0.675	0.615	0.998	0.315	1.000 / 0.999

Note. — For species with two values, they show results from sputtering rates of Tielens et al. (1994) and Nozawa et al. (2006), respectively. Species with only one value show just the Nozawa et al. (2006) sputtering rates. In the context of the Tielens et al. (1994) sputtering, SiO_2 represents "silicates".

*The high final dust mass values in Simulation 8 are a result of the analysis being truncated prior to ten full cloud-crushing times, as mentioned in the text.

- Owing to linear density dependence in the sputtering rates, the high density clouds destroy significantly higher percentages of dust. In some cases, as much as 70% more dust is destroyed, with total destruction fractions as high as 80-100%.

- The relative velocity between the reverse shock and the ejecta clump strongly influences the total dust destruction. For high-density ejecta clumps, the highest velocity shock results in an additional ∼50% mass loss compared to the lowest velocity shock for some grain species.

- The initial size distribution of dust grains has a considerable impact on the survival rate. Species with appreciable mass in grains above 0.1 μm have the highest survival rate, losing only 30% or less of their mass. Grains that start out at less than 0.1 μm are often completely destroyed.

- Total dust destruction varies widely across grain species and can be high for some portions of parameter space. For high-density ejecta clumps and high-velocity shocks, the often studied species of C, SiO_2, and Fe show 62%, 20%, and 80% survival, respectively.

From an observational perspective, these results may have interesting implications for the IR emission seen in supernova remnants, which is usually attributed to dust that has been formed and processed by the supernova explosion. Rho et al. (2008) modeled detailed mid-IR spectra of Cas A to determine the populations of dust that are likely present in the remnant. They find a reasonably high abundance of Si dust, which agrees with the high survival rate of those grains. Additionally, Sandstrom et al. (2009) analyzed mid-IR spectra for SNR 1E0102-7219 and found both a lack of Al_2O_3 grains as well as a large population of amorphous carbon grains with radii of 0.1 μm, in good agreement with our results. Such observations are excellent indicators that the results from our simulations could be used to predict emission features in future remnant observations.

Since the plasma in these shock-cloud interactions will become enriched with high-mass ions as a result of dust-grain sputtering, we will investigate, in future work, the additional effects that this might have. We have preliminary findings that indicate that for cooling simulations with

enhanced metallicity ($Z \gtrsim 10\ Z_{\odot}$), the results diverge further from the non-cooling simulations. Specifically, the simulations with increased cooling owing to higher metal abundances tend to show a greater amount of dust destruction. This likely stems from the previously mentioned effect in which the shorter cooling time leads to the formation and collapse of localized cloud over-densities that persist for a great fraction of the simulation time. We also predict that, with accurate sputtering rates for enhanced metal abundances, we should see greater dust destruction by high mass ions, since the sputtering yields for these ions should will be factors of 10-100 higher.

While our current method of post-processing the tracer particles is sufficient as a first step in tracking dust populations in an ejecta clump, we would like to employ more active means of following dust erosion. Therefore, to improve upon current simulations, we plan to sputter the dust grains "on-the-fly" so that dust destruction occurs during simulation run time. This should provide something closer to an instantaneous sputtering rate rather than the current low time resolution da/dt, which is applied only at each discrete data dump. In addition, it will allow for a means of accounting for feedback effects that dust sputtering may have on the system. As the dust grains are eroded, they reintroduce metals into the gas phase which would allow them to act as additional coolants and ions for grain sputtering.

Computing the feedback of metals by grain sputtering also motivates us to improve our cooling method by tracking the exact abundances of a variety of metals and then using Cloudy to get cooling rates that directly scale with the metal abundance pattern. If we accurately track the abundances of the metals, we can increase the metal abundances in the hot plasma as the grains are sputtered. Providing the link between grain sputtering and gas cooling could lead to a possible equilibrium between grain sputtering and plasma cooling. At this point, collapsing gas drives up the sputtering rate, which leads to increased metal abundances that cool the gas even further, perhaps to low enough temperatures to shut off the thermal sputtering.

Finally, given the nature of the tracer particles used to track dust populations, we have only been able to model the dust as being directly coupled to the cloud material. Therefore, we have only applied thermal sputtering rates and neglect non-thermal sputtering by grain motion through

the plasma. Dust coupling to the cloud may not be an unreasonable assumption if the grains are charged and tied to magnetic fields lines that might be present in the ejecta material. It would be ideal to decouple the grains from the fluid and account for the effects of non-thermal sputtering as well as possible grain-grain collisions. Additionally, if magnetic field lines are present, it would be important to investigate the effects of betatron acceleration (Shull, 1977, 1978) as it might enhance sputtering of large grains. Methods for including these factors, as well as those mentioned in the preceding paragraphs, are being considered for inclusion in a second paper on this topic.

Chapter 4

Numerical Simulations of Supernova Dust Destruction. Metal-Enriched Ejecta Knots

ABSTRACT

Following our previous work, we investigate through hydrodynamic simulations the destruction of newly-formed dust grains by sputtering in the reverse shocks of supernova remnants. Using an idealized setup of a planar shock impacting a dense, spherical clump, we implant a population of Lagrangian particles into the clump to represent a distribution of dust grains in size and composition. We vary the relative velocity between the reverse shock and ejecta clump to explore the effects of shock-heating and cloud compression. Because supernova ejecta will be metal-enriched, we consider gas metallicities from $Z/Z_{\odot} = 1$ to 100 and their influence on cooling properties of the cloud and the thermal sputtering rates of embedded dust grains. We post-process the simulation output to calculate grain sputtering for a variety of species and size distributions. In the metallicity regime considered in this paper, the balance between increased radiative cooling and increased grain erosion depends on the impact velocity of the reverse shock. For slow shocks ($v_{\rm shock} \leq 3000$ km s^{-1}), the amount of dust destruction is comparable across metallicities, or in some cases is decreased with increased metallicity. For higher shock velocities ($v_{\rm shock} \geq 5000$ km s^{-1}), an increase in metallicity from $Z/Z_{\odot} = 10$ to 100 can lead to an additional 24% destruction of the initial dust mass. While the total dust destruction varies widely across grain species and simulation parameters, our most extreme cases result in complete destruction for some grain species and only 44% dust mass survival for the most robust species. These survival rates are important in understanding how early supernovae contribute to the observed dust masses in high-redshift galaxies.

4.1 Introduction

Over the last two decades, far-infrared (FIR) and millimeter observations of high-redshift quasars ($z > 6$) have produced estimates for galactic dust masses as high as 10^8 $M_\odot$ (Smail et al., 1997; Hughes et al., 1998; Bertoldi et al., 2003; Wang et al., 2008). In order to explain the formation of this large quantity of dust within the short lifetime ($\sim$1 Gyr) of the universe at this epoch, a mechanism must exist which is capable of both significant and rapid dust production. One recently pursued solution is that the majority, if not all, of this dust comes from core-collapse supernova (CCSN) explosions of the first generations of stars (Morgan & Edmunds, 2003; Maiolino et al., 2004; Hirashita et al., 2005).

While various theoretical work (Kozasa et al., 1989, 1991; Todini & Ferrara, 2001; Nozawa et al., 2003; Bianchi & Schneider, 2007; Nozawa et al., 2010) has suggested that $\sim$0.1-0.3 $M_\odot$ of dust could be formed per supernova event, which is in rough agreement with the $\sim$0.2-1.0 $M_\odot$ required to explain high-redshift dust (Morgan & Edmunds, 2003; Dwek et al., 2007; Dwek & Cherchneff, 2011; Gall et al., 2011). However, observational efforts focused on local supernova remnants (SNRs) often fall orders of magnitude short of these values (Stanimirović et al., 2005; Williams et al., 2006; Meikle et al., 2007; Rho et al., 2008, 2009; Kotak et al., 2009). More recent observations aimed at finding colder dust ($T < 40$ K) using AKARI and the Balloon-borne Large Aperature Submillimeter Telescope (BLAST) by Sibthorpe et al. (2010), as well as *Herschel* by Barlow et al. (2010) and Matsuura et al. (2011), find larger dust masses. Matsuura et al. (2011) find $\sim$0.6 $M_\odot$ of dust in SN1987A. Such estimates could make the argument for SNe as dust factories more plausible.

One question that arises from these SNR studies is what fraction of the freshly formed dust predicted by theory will survive the interaction between the reverse shocks and the ejecta. As the fast-moving dust ($V_{ej} \geq 1000$ km s^{-1}) is impacted by the reverse shock, it will be subject to sputtering and grain-grain collisions as the density and temperature of the dust-enriched gas are increased.

In our previous work (Silvia et al., 2010, hereafter Paper I), we investigated, through hydrodynamic simulations, the destruction of newly formed dust grains by sputtering in the reverse shocks of SNRs. Using "cloud-crushing" simulations (Woodward, 1976; Mac Low et al., 1994; Klein et al., 1994), we found that the degree of dust destruction depends heavily on the initial radius distribution of the dust grains as well as the initial density of the ejecta cloud and relative velocity between the reverse shock and the cloud. In the most extreme cases, we found grain destruction to vary from 20-100% depending on grain species. We also found morphological similarities within these simulations to the observational studies of Cassiopeia A presented in Fesen et al. (2011), specifically in the scenarios where the effects of cooling led to the fragmentation of ejecta clouds.

However, our earlier simulations computed the radiative cooling and sputtering rates for approximately solar metallicity ($Z_{\odot}$) gas. In reality, the gas contained in supernova ejecta will be highly metal-enriched. The question arises as to the balance between the enhanced radiative cooling and the increase in sputtering yields in this high-metallicity regime. This balance will also be influenced by the changing density of the metal-enriched gas as the cloud is shredded and the plasma thins out.

In the current paper, we perform additional cloud-crushing simulations to study the evolution of dust mass contained within an idealized ejecta knot as it is impacted by a supernova reverse shock. In Section 4.2 we give a brief review of the code used to carry out these simulations, the methods for tracking our dust populations, and the changes made from our previous work. In Section 4.3, we describe the simulations unique to this paper, specifically aimed at probing the higher metallicity regimes expected to be present in supernova ejecta. We present the results of these simulations in Section 4.4 and conclude in Section 4.5 with a summary and discussion of the implications of these results.

4.2 Methodology

4.2.1 Code and Simulation Setup

As in Paper I, we use the Eulerian adaptive mesh refinement (AMR), hydrodynamics + N-body code, `Enzo` (Bryan & Norman, 1997; Norman & Bryan, 1999; O'Shea et al., 2005). We do not make use of any of the cosmological or gravity-solving components of the code owing to the idealized nature of our problem. We follow the same cloud-crushing setup as outlined in Paper I and refer the reader to that work for a detailed description.

The user-supplied parameters required to initialize the simulation are: cloud radius, $r_{\rm cloud}$; cloud temperature, $T_{\rm cloud}$; density of the ambient medium, $\rho_{\rm m}$; initial over-density, χ, of the cloud with respect to the ambient medium ($\rho_{\rm cloud} = \chi\rho_{\rm m}$); and the velocity of the shock relative to the stationary cloud, $v_{\rm shock}$. From these input parameters, the following values must be derived in order to completely initialize the simulation: temperature of the ambient medium, $T_{\rm m}$, post-shock density, $\rho_{\rm shock}$, and post-shock temperature, $T_{\rm shock}$. We set $T_{\rm m}$ so that the cloud remains in pressure equilibrium, while the shock-related values are calculated using the Rankine-Hugoniot jump conditions.

During runtime, we take advantage of the AMR capabilities of `Enzo` by employing the same refinement criterion as described in Paper I, increasing resolution in areas of the simulation that contain a significant fraction of cloud material. Cells that are initially enclosed within the cloud radius are assigned a "cloud material" value, ξ, that is advected with the flow in the same way as density. When a cell exceeds a pre-defined cloud material "mass", $m_\xi = \xi V_{\rm cell}$, where $V_{\rm cell}$ is the cell volume, the resolution of that cell is doubled. A more thorough description of this refinement process can be found in Paper I.

4.2.2 Dust Tracking and Post-processing

Our dust-grain populations continue to be tracked through tracer particles embedded in the flowing gas. As before, we post-process the density and temperature histories of each tracer particle

to compute dust survival rates. For the initial distributions in grain radii, we again made use of the values calculated by Nozawa et al. (2003) for a CCSN with a progenitor mass of 20 $M_{\odot}$ (see Figure 4.1). We follow the evolution of all nine grain species included in the unmixed grain model of Nozawa et al. (2003). To determine the evolution in dust mass, we use the same mass proxy outlined in Paper I, which involves tracking the changes in $n(a) \times a^3$, where $n(a)$ is the number density of grains and a is the grain radius. However, the erosion rates used to sputter the grains with each successive time step are different than those used in Paper I, as described in the following section. We note that we only account for dust grain destruction in this work and do not include a prescription for possible grain growth that might occur at high densities and low temperatures. For the majority of our simulations, the amount of time that the dust grains spend in an environment conducive to grain growth is relatively brief and any potential increase in dust mass should be minimal.

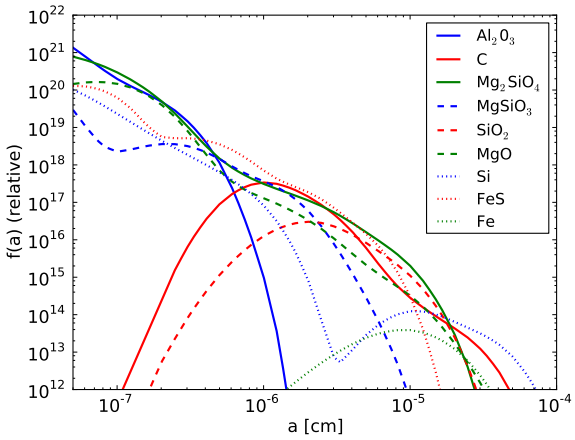


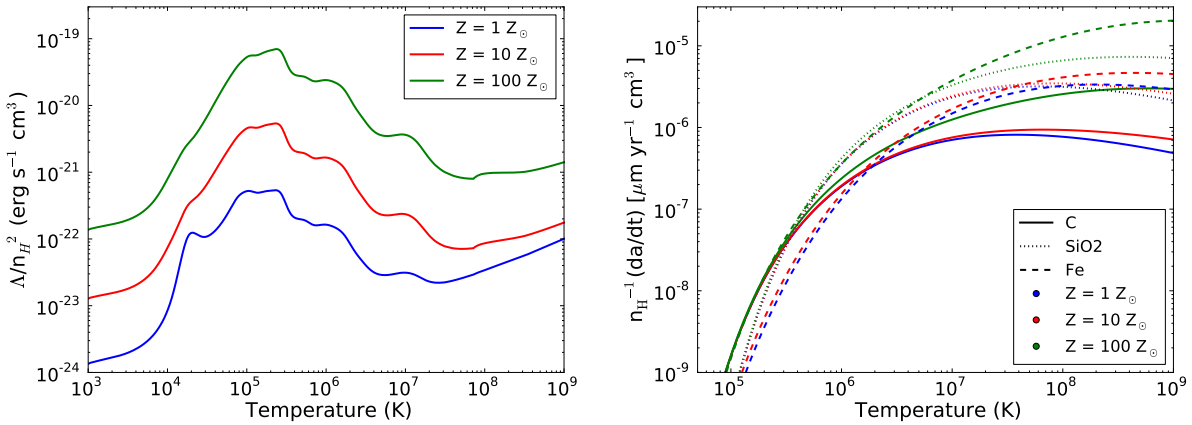
Figure 4.1: Reproduced grain radius distributions for the species expected in the unmixed ejecta model of a core-collapse supernova with a progenitor mass of 20 $M_{\odot}$ as calculated by Nozawa et al. (2003). The y-axis is the abundance of grains of a given radius, $f(a)$, such that the number density of grains between a and $a + \delta a$ is $n(a) = f(a)\delta a$, where δa is set by the number of bins used to track the distribution.

4.2.3 Modifications and Additions

To account for the fact that we expect the supernova ejecta clumps to be metal-enriched, we implement both new cooling curves and grain erosion rates for high-metallicity gas. We calculate the cooling rates using `Cloudy` (Ferland et al., 1998) for gas assumed to be in ionization equilibrium. In addition, these rates assume that both the electrons and the different ion species have the same temperature. Cooling due to the thermal emission of dust grains is not included.

For this work, we compute rates for metallicities of $Z/Z_{\odot} = 1$, 10, and 100. The element abundances for solar metallicity gas come from the composition listed in the documentation for `Cloudy`, where all abundances are specified by number relative to hydrogen. We define the metallicities with values greater than unity to mean that the abundances for metals are increased by factors of 10 and 100 from their solar values. The abundance of hydrogen and helium remain the same for all three metallicities. To simplify the notation, we refer to these metallicities as 1 $Z_{\odot}$, 10 $Z_{\odot}$, and 100 $Z_{\odot}$ for the remainder of the paper. We also note that the values for the metal abundances remain static for the duration of our simulations, and any potential increase in metal abundance due to sputtered dust grains is omitted. We comment on this omission in the final section of the paper. Figure 4.2a shows the cooling rate coefficients for 1, 10, and 100 $Z_{\odot}$ gas as a function of temperature. To understand the contributions that hydrogen, helium, and heavier elements make on the cooling curves, we refer the reader to Gnat & Sternberg (2007).

In contrast to Paper I, in which the cooling rates were applied to all cells within the simulation domain, we only cool those cells that contain cloud material. This reduces the computational time required to calculate the cooling rates and accounts for the fact that, while the cloud itself can be highly metal-enriched, the ambient medium is at much lower densities and metallicities. To ensure that important information is not lost by cooling only the cloud, we consider that for an ambient medium with a metallicity of $Z = 1\ Z_{\odot}$, a number density of $n = 1$ cm^{-3}, and a temperature of $T = 10^6$ K, the cooling time is a few $\times\ 10^4$ yrs, much longer than the time scale for any of the simulations included in this work.



(a) Metallicity-dependent cooling curves

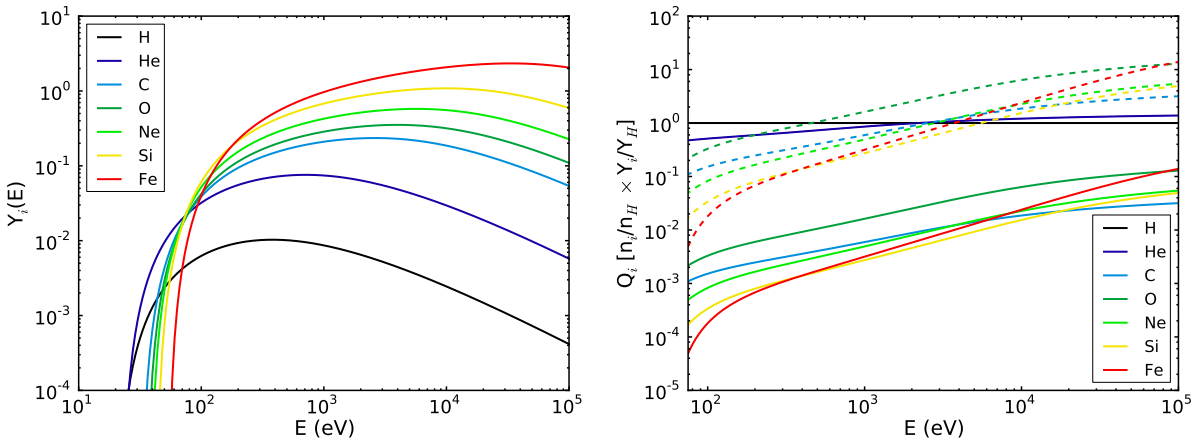
(b) Metallicity-dependent erosion rates

Figure 4.2: (a) Radiative cooling curves for solar-scaled metal abundances $Z = 1\ Z_{\odot}$, $10\ Z_{\odot}$, and $100\ Z_{\odot}$. (b) Dust grain erosion rates for C, SiO_2, and Fe grains.

We calculate new erosion rates using the formula provided by Nozawa et al. (2006) for the limit in which thermal sputtering is the dominant mechanism for grain erosion. This is appropriate, since the tracer particles used to track the dust are embedded in the flow. In computing these erosion rates, we use the same scaled solar metal abundance ratios that were used to produce our new cooling functions. Computing rates based on these enhanced metal abundances is an important step. The sputtering yield, $Y(E)$ where E is the energy of the impacting ion, is strongly dependent on the atomic mass of the impactor; high-mass ions can have orders of magnitude higher yields at high energies. The differences in erosion rates as a function of metallicity are shown for carbonaceous, silicate, and ferrous grains in Figure 4.2b. The differences do not become significant until the temperature exceeds a few times 10^6 K and, even then, only for the highest metallicity. In order to simplify Figure 4.2b, we have omitted the other six grain species studied in this work. However, the erosion rates for the omitted species are comparable to the three presented species.

We include Figure 4.3a to show the nature of the sputtering yields for a sample of impacting ions as a function of energy and Figure 4.3b to show the contribution of various impacting ions to overall grain sputtering for gas with both a metallicity of 1 $Z_{\odot}$ and 100 $Z_{\odot}$. For simplicity, we only present these quantities for C grains, but note that the figures are qualitatively similar for

the other grain species. From Figure 4.3b, it is evident that the contributions from the sputtering yields of high-mass ions do not begin to dominate over hydrogen and helium until the metallicity reaches 100 $Z_{\odot}$, which agrees with the trend observed in the erosion rates presented in Figure 4.2b.



(a) Sputtering yields

(b) Metallicity-dependent contributions to sputtering

Figure 4.3: (a) Sputtering yields, $Y_{\rm i}(E)$, of C grains versus projectile energy, E, for various incident ions; (b) The contribution of various incident ions to the sputtering of dust grains, $Q_{\rm i}$, where $Q_{\rm i}$ is defined to be the product of the element abundance by number relative to hydrogen, $n_{\rm i}/n_{\rm H}$, and the sputtering yield normalized to that of hydrogen, $Y_{\rm i}/Y_{\rm H}$, as a function of energy. We show the contributions based on solar abundance ratios (solid lines) and abundance ratios for $Z = 100\ Z_{\odot}$ (dashed lines).

When these new cooling and sputtering rates are implemented within our simulations, we find that, for the high metallicity ($Z = 100\ Z_{\odot}$) simulations, the densest cells end up with extremely short cooling times, often shorter than the time step set by the Courant-Friedrichs-Lewy (CFL) condition. This tends to result in cells that over-cool during a given time step and can lead to negative cell energies. To avoid this issue, we make two modifications to `Enzo`. First, we add a temperature floor, $T_{\rm floor} = 1000$ K, to all of our simulations to prevent clouds from becoming unrealistically cold. Given the energetic environment of the supernova remnant and the background radiation from shock-heated, x-ray emitting gas in the remnant's shell, such a floor is physically reasonable. Second, we modify the time-step calculation such that it is never more than 25% of the cooling time. While this prevents negative cell energies, it can become computationally expensive, as the cloud is compressed to high densities and the cooling time becomes very short in

some regions of the cloud. It is this computational limitation that prevents us from exploring even higher metallicities, at least within the bounds of our computational resources.

4.3 Simulations

All of the simulations presented in this work have root-grid dimensions of 512 × 256 × 256 and a physical resolution of 1.25×10^{15} cm (∼84 AU) per cell edge. As in our previous work, we initialize each cloud to have a radius of $r_{\rm cloud} = 10^{16}$ cm, which spans 8 cells at the root-grid resolution. We also use the same Gaussian envelope formulation as in Paper I, with the density fall-off occurring at $r = 0.7\ r_{\rm cloud}$.

We allow the simulation domain to be refined up to three additional levels such that the highest resolution cells will be 8 times smaller than the root-grid. This differs from our previous work, where we only allowed for two additional levels of resolution. The third level of refinement was required specifically in the simulations with high metallicity, to follow the collapse of some fragments to much higher densities and smaller spatial scales than in the low-metallicity cases. When we allow for a fourth level of refinement, the final dust masses do not change significantly; we therefore limit ourselves to three levels to save computational resources. As mentioned above, our cell refinement is based on the amount of cloud material in a given cell.

For this work, we primarily focus our exploration of parameter space to the relative velocity between the reverse shock and the ejecta cloud and the metallicity of the gas contained within the cloud. Specifically, we investigate shock velocities of 10^3 km s^{-1}, $3{\times}10^3$ km s^{-1}, $5{\times}10^3$ km s^{-1}, and 10^4 km s^{-1} and metallicities of 1 $Z_\odot$, 10 $Z_\odot$, and 100 $Z_\odot$. For ease of reference, we provide a numbered list of all simulations in Table 4.1. For all simulations, the cloud constrast is $\chi = 1000$, the initial cloud temperature is set equal to our temperature floor, $T_{\rm cloud} = T_{\rm floor} = 1000$ K, and the total runtime is $4.2t_{\rm cc}$, where $t_{\rm cc} = \chi^{1/2} r_{\rm cloud}/v_{\rm shock}$ is the cloud-crushing time (Klein et al., 1994).

Figure 4.4 shows the evolution in density of the shock-cloud interaction from the initial impact to $t \sim 3\ t_{\rm cc}$ for $v_{\rm shock} = 5{\times}10^3$ km s^{-1} and $Z = 100\ Z_\odot$. Notable features include the tail of ablated

Table 4.1. Simulation Parameters

Simulation	χ	$v_{\rm shock}$ (km s^{-1})	$t_{\rm cc}$ (yrs)	Metallicity ($Z/Z_{\odot}$)
1	1000	1000	100.2	1
2	1000	1000	100.2	10
3	1000	1000	100.2	100
4	1000	3000	33.4	1
5	1000	3000	33.4	10
6	1000	3000	33.4	100
7	1000	5000	20.0	1
8	1000	5000	20.0	10
9	1000	5000	20.0	100
10	1000	10000	10.0	1
11	1000	10000	10.0	10
12	1000	10000	10.0	100

Note. — Parameter definitions: χ is the initial overdensity of the cloud with respect to the ambient medium, $v_{\rm shock}$ is the relative velocity between the inflowing shock and the stationary cloud, and $t_{\rm cc}$ is the cloud-crushing time.

material produced as the shock washes over the cloud, and the numerous cold, dense fragments that form a result of the high cooling rates for the metal-enriched gas. A handful of these fragments persist for a large fraction of the total simulation time.

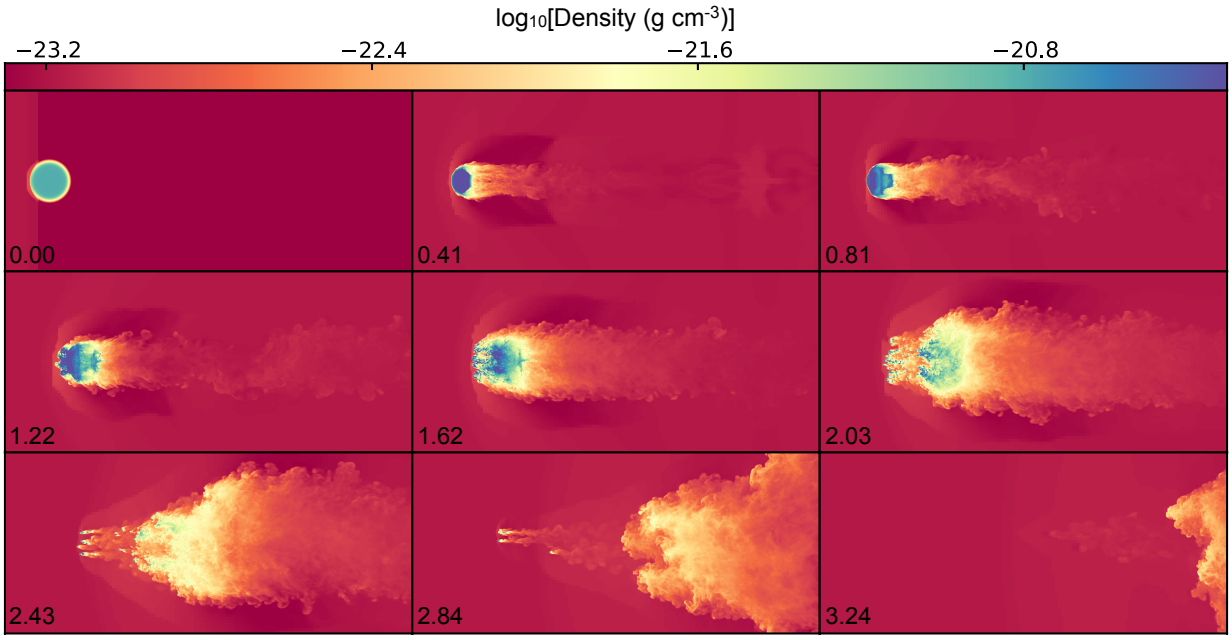


Figure 4.4: Density projections in the x-z plane weighted by cloud material for the case of $v_{\rm shock} = 5000$ km s^{-1} and $Z = 100\ Z_{\odot}$. The number in the lower left corner of each panel indicates the time of the snapshot in units of the cloud-crushing time, $t_{\rm cc}$. Projections were made using the software analysis package, `yt` (yt-project.org; Turk et al., 2011b).

While our previous work ran all simulations for a total of ten cloud-crushing times, we found that in our new Simulations 2 and 3 (slow shock and high metallicities) it was difficult to keep the majority of the dust particles within the computational domain much beyond four cloud-crushing times without significantly increasing the size of the root grid. Though possible, we deemed this solution to be a poor use of our computational resources. As will be seen in Section 4.4, the dust masses are still evolving in the relatively slow-shock ($v_{\rm shock} = 10^3$ km s^{-1} and $3{\times}10^3$ km s^{-1}) simulations at the point of termination. However, the fast-shock simulations do not have significant changes in dust mass beyond that time. Therefore, while we do see clear trends in the slow-shock simulations as a function of metallicity, we cannot say definitively how these trends would behave at later times.

4.4 Results

We present the results for the twelve simulations listed in Table 4.1, with a focus on the evolution in total dust mass and the time spent by the post-processed dust particles in various areas of density-temperature phase-space.

4.4.1 Dust Mass Evolution

The evolution of dust mass for each of the nine dust species and for all twelve simulations can be seen in Figure 4.5, and the final dust masses can be located in Table 4.2. The most notable difference between the simulations is the drastic increase in dust destruction between the slowest shock velocity, $v_{\rm shock} = 10^3$ km s^{-1}, and the highest shock velocity, $v_{\rm shock} = 10^4$ km s^{-1}, in some cases a difference of >70%. For the intermediate shock velocities, $v_{\rm shock} = 3 \times 10^3$ km s^{-1} and 5×10^3 km s^{-1}, the evolution in dust mass depends on the grain species. For many species, the simulations with $v_{\rm shock} = 5 \times 10^3$ km s^{-1} show comparable dust destruction to the highest velocity shocks, while the simulations with $v_{\rm shock} = 3 \times 10^3$ km s^{-1} exhibit only moderate dust destruction. There does appear to be a fundamental difference between the simulations with the slowest shock velocities and those with faster shocks at early times. For nearly all grain species, there is a sharp drop in dust mass within one cloud-crushing time for shock velocities of 3×10^3 km s^{-1} and higher, with the magnitude of the drop increasing with shock velocity. This suggests that there is a threshold velocity somewhere between 10^3 and 3×10^3 km s^{-1} at which the initial impact of the shock into the cloud results in a degree of shock heating and compression that significantly influences the overall dust destruction in the cloud. For slower shocks, the majority of the dust destruction occurs at much later times, as the cloud is shredded and incorporated into in the hot, post-shock gas.

In our study of the effects of metallicity for a given shock velocity, we find that only in the slowest shocks does an increase in metallicity consistently lead to a decrease in dust destruction (see the results of Simulation 3). In the case of $v_{\rm shock} = 3 \times 10^3$ km s^{-1} and $Z = 100\ Z_{\odot}$, the amount

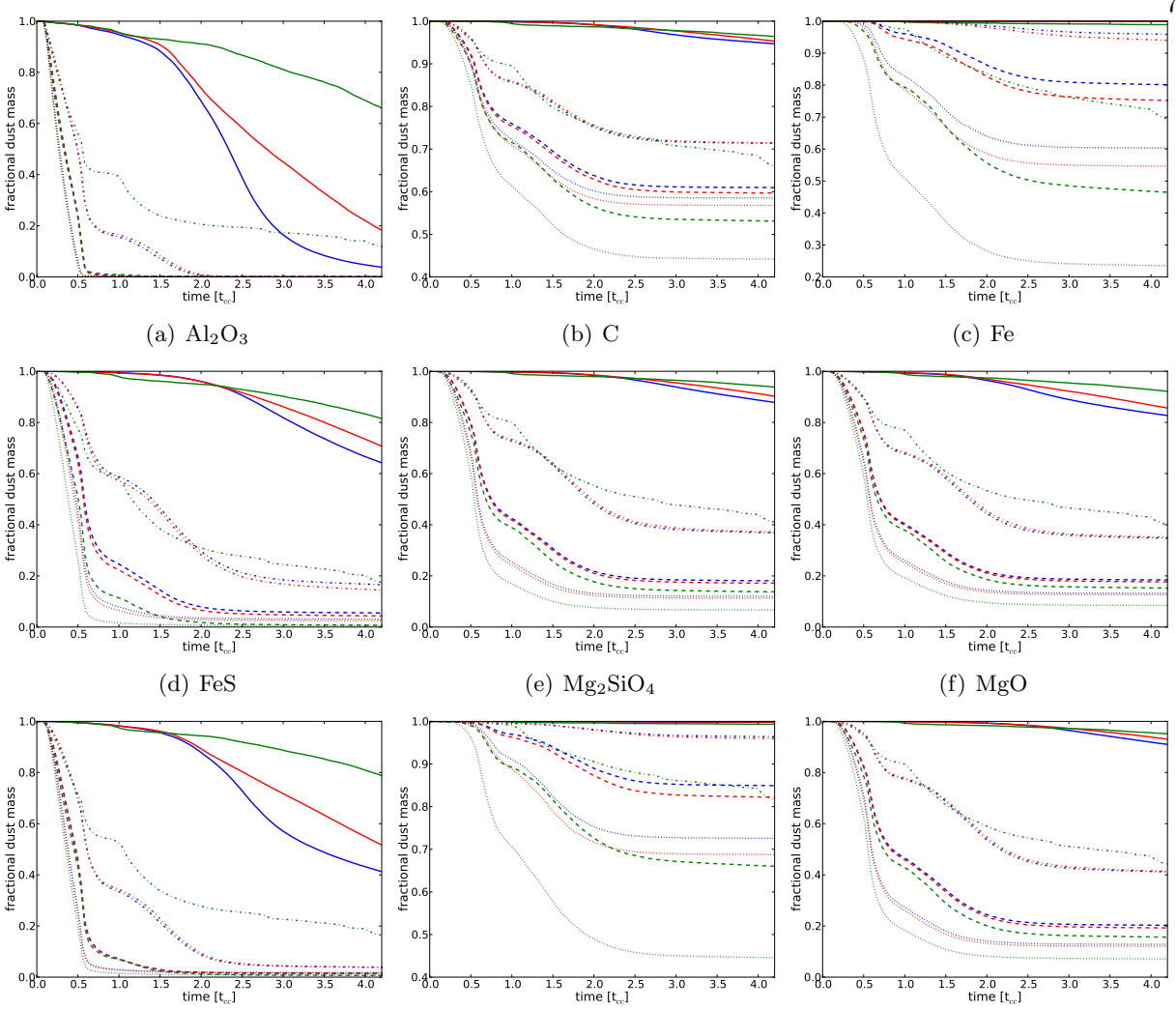
Table 4.2. Final Dust-mass Survival Fraction

Simulation	Al_2O_3	C	Mg_2SiO_4	$MgSiO_3$	SiO_2	MgO	Si	FeS	Fe
1	0.039	0.947	0.879	0.413	0.910	0.827	0.998	0.643	1.000
2	0.184	0.953	0.902	0.518	0.931	0.857	0.998	0.708	1.000
3	0.661	0.964	0.938	0.789	0.951	0.922	0.993	0.816	0.989
4	0.001	0.714	0.369	0.038	0.411	0.347	0.964	0.165	0.959
5	0.001	0.714	0.371	0.039	0.413	0.351	0.960	0.145	0.940
6	0.118	0.659	0.409	0.162	0.441	0.399	0.815	0.173	0.693
7	0.001	0.610	0.181	0.014	0.203	0.184	0.849	0.055	0.801
8	0.001	0.597	0.171	0.014	0.192	0.177	0.822	0.044	0.752
9	0.000	0.531	0.137	0.007	0.156	0.152	0.660	0.007	0.465
10	0.002	0.585	0.120	0.018	0.128	0.132	0.726	0.031	0.603
11	0.002	0.568	0.113	0.016	0.121	0.126	0.688	0.025	0.547
12	0.001	0.442	0.067	0.008	0.072	0.084	0.446	0.005	0.235

Note. — For simulation parameters, refer to Table 4.1

of dust destruction is noticeably less for some grain species (Al_2O_3, $MgSiO_3$) and considerably more for others (Fe, Si). This suggests that, for some grain species, the increased cooling due to the enhanced metallicity is able to counter-balance the shock-heating and keep the gas in a regime of decreased sputtering. For the grain species with lower survival rates, a shock velocity of 3×10^3 km s^{-1} is sufficient to drive up the amount of dust destruction. However, we again note that this is only based on the results at $t = 4.2t_{\rm cc}$. At this velocity, the simulations with $Z = 1\ Z_\odot$ and $10\ Z_\odot$ appear to have hit a dust mass plateau, but for $Z = 100\ Z_\odot$ the evolution in dust mass still has a clear downward slope. For $v_{\rm shock} = 5 \times 10^3$ km s^{-1} and 10^4 km s^{-1}, an increase in metallicity leads directly to an increase in total dust destruction. The most drastic change occurs when the metallicity is increased from $Z = 10\ Z_\odot$ to $100\ Z_\odot$; $\sim$20% more of the initial dust is lost with this change in metallicity for Fe and Si grains in the highest velocity scenario. For all simulations in which the evolution in dust mass plateaus at late times, the cloud has been significantly shredded and the density of the gas has been reduced to a level that does not allow for appreciable grain sputtering.

In studying the final (surviving) dust masses for each grain species, we find that the values are



(a) Al_2O_3 (b) C (c) Fe

(d) FeS (e) Mg_2SiO_4 (f) MgO

(g) $MgSiO_3$ (h) Si (i) SiO_2

Figure 4.5: Dust mass evolution (surviving mass) for the nine dust species tracked in this work versus time in units of cloud-crushing time. The line styles correspond to the four different shock velocities, 10^3 km s^{-1} (solid), 3×10^3 km s^{-1} (dot-dashed), 5×10^3 km s^{-1} (dashed), and 10^4 km s^{-1} (dotted). The colors correspond to the three different metallicities, $Z = 1\ Z_\odot$ (blue), 10 $Z_\odot$ (red), 100 $Z_\odot$ (green). Panels (b), (c), and (h) do not fully span the range 0 to 1 on the y-axis.

widely scattered for the simulations with the slower shock velocities. The higher velocity simulations show dust survival of less than 20% for six of the nine species, with C, Fe, and Si grains being the three species with the least destruction. Although Fe grains had a higher survival rate than other dust species, the amount of Fe dust destruction was considerably higher than any of the simulations in our previous work. In the most extreme case, only 24% of the Fe dust mass survived. If we look

at dust survival for two other commonly studied grain species for the same extreme case, 44% and 7% of the initial dust mass remains for carbonaceous (C) and silicate (SiO_2) grains, respectively.

For example, consider the differences in dust survival across grain species. The difference, $\Delta_{\rm survival}$, between most destroyed grains and least destroyed grains ranges from as high as 96% in Simulations 4 and 5 (Al_2O_3 compared to Si) to as low as 45% in Simulation 12 (again, Al_2O_3 compared to Si). This variation stems primarily from the initial distributions in grain radius. A high degree of destruction is observed in the species with a majority of their mass locked up in small grains. Destruction is reduced significantly for those species with appreciable mass in large grains. As noted in Paper 1, grains with initial radii less than 0.1 μm are easily obliterated, while larger grains take a considerable duration to be substantially sputtered. If the actual size distributions of dust grains in SNRs were greatly disparate from the distributions assumed here, the final dust mass survival fractions might change considerably.

4.4.2 Density-Temperature Phase-Space

In an attempt to understand the differences in the physical environments between the twelve simulations, we present phase-space diagrams of density and temperature to illustrate where the dust particles spend their time (Figure 4.6). We also over-plot erosion rate contours for C grains as a function of density and temperature to determine where in phase-space the majority of the dust destruction occurs. The most apparent difference between simulations is the rise in gas temperatures with increasing shock velocity, a direct result of shock-heating. The increase in temperature pushes the gas into a regime of higher grain erosion, especially when the gas is still compressed to relatively high densities. We also note that the large number of time steps at high temperature and low-density seen across all simulations is a result of the simulation end-state, at which point the cloud has been shock-heated and shredded.

In reviewing some of the subtle differences between simulations, we identify features that help to explain the differences between final dust masses. First, we believe the high number of time steps spent by particles at high densities but very low temperatures in Simulation 3 ($v_{\rm shock}$ =

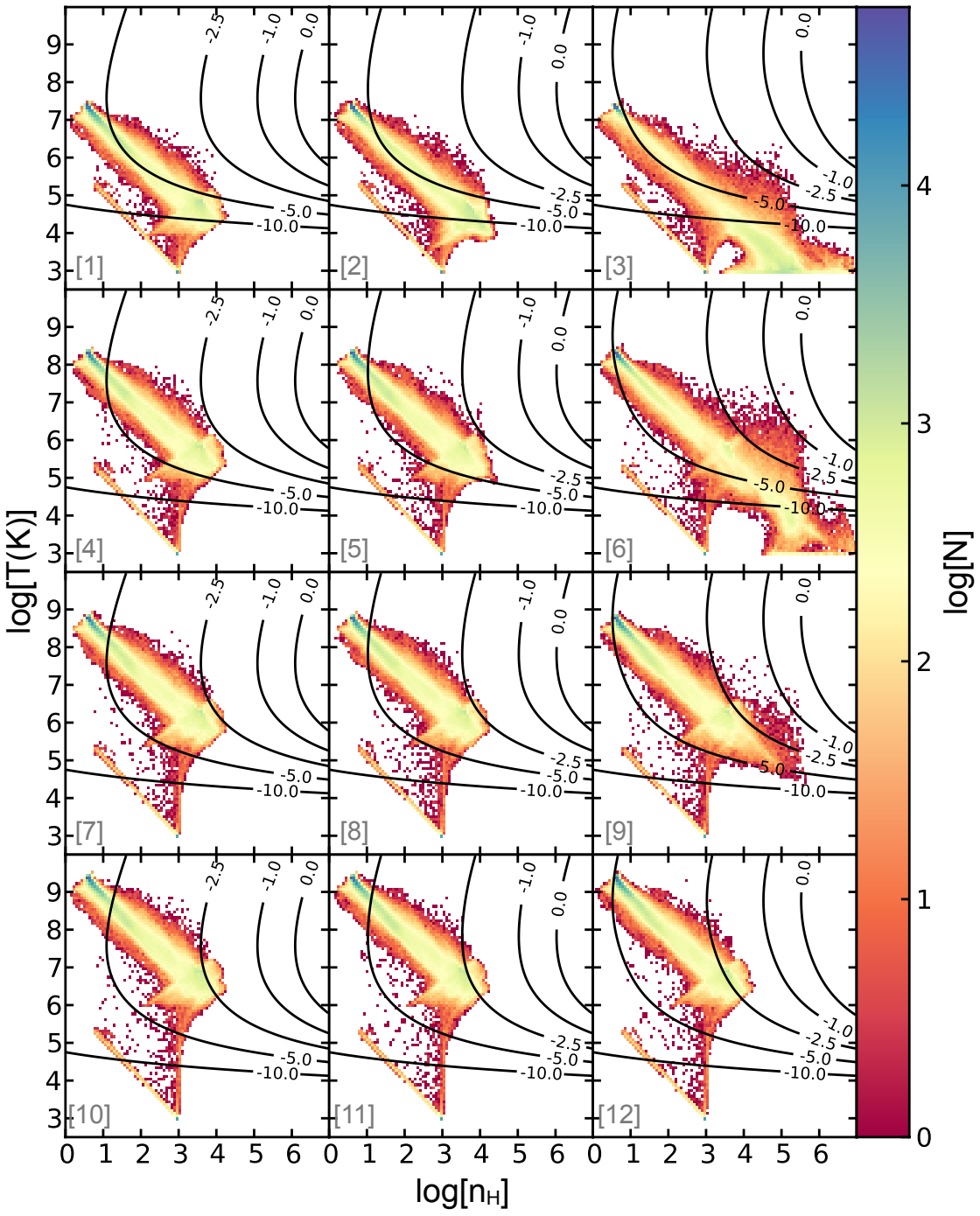


Figure 4.6: Phase plots of hydrogen number density ($n_{\rm H}$ in cm^{-3}) vs. temperature (T in K). The color scale represents the total number of times steps (N) spent by all dust particles in the simulation. The contour lines show the dust destruction rate, $\log(da/dt)$, in units of μm yr^{-1} for C grains. Each panel is one of the 12 simulations, labeled by the number in brackets [Table 4.1].

10^3 km s^{-1}, $Z = 100\ Z_\odot$) is likely the dominating factor in the decreased dust destruction when compared to the lower metallicity simulations (1 and 2). In this scenario, the temperatures of the gas drop low enough from the radiative cooling to slow grain erosion. Although Simulation 6 ($v_{\rm shock} = 3 \times 10^3$ km s^{-1}, $Z = 100\ Z_\odot$) shows a similar population of high-density, low-temperature dust particle environments, it is not sufficient to produce the same consistent decrease in dust destruction, owing primarily to the overall increase in gas temperature. The pile up of dust particles at $T = 1000$ K, seen in Simulations 3 and 6, is a direct result of the temperature floor described in Section 4.2.3. For $v_{\rm shock} = 5 \times 10^3$ km s^{-1}, there is a distinct feature present in Simulation 9 with $Z = 100\ Z_\odot$ that separates it from the lower metallicity cases. As a result of the enhanced metal cooling, the gas is able to cool and condense beyond a hydrogen number density of $n_{\rm H} = 10^4$ cm^{-3}, but it remains within a temperature regime to allow for appreciable grain erosion, explaining the increased dust destruction observed in the dust mass evolution.

In the case of the highest velocity shock, the intense heating appears to overwhelm the effects of radiative cooling, even for the highest metallicity simulations, as the phase plots for Simulations 10-12 appear nearly identical in the qualitative sense. Given the similarity between the physical environments experienced by the dust particles, it would appear that in this regime the dominant cause for increased dust destruction at high metallicity is the increase in the erosion rates. This is not an unreasonable explanation, since the dust particles spend the majority of their time above 10^6 K, the point at which the dust destruction rates begin to vary significantly for higher metal abundances.

4.5 Summary and Discussion

The above results can be summarized as follows:

- The degree of destruction varies widely across the explored parameter space. For the least destructive cases (low shock velocity, high gas metallicity), we see near-negligible grain destruction for some species (Fe and Si) and the highest survival rates for all other

species. In the maximally destructive cases (high shock velocity, high gas metallicity), we find complete to near-complete destruction for multiple grain species (Al_2O_3, FeS, and $MgSiO_3$).

- The relative velocity between the reverse shock and the cloud of ejecta material is a dominant factor in determining the survival rate of dust grains; an increase in velocity correlates directly with an increase in dust destruction. The most extreme example is for silicate (SiO_2) grains, where an additional 88% of the initial dust mass is destroyed between the simulations with $v_{\rm shock} = 10^3$ km s^{-1} and those with $v_{\rm shock} = 10^4$ km s^{-1}.

- There appears to be a threshold shock velocity between 1000 and 3000 km s^{-1} at which the amount of dust destruction at early times ($t < t_{\rm cc}$) is increased considerably.

- In the physically expected, highly metal-enriched simulations, we find ranges of dust mass survival fraction of 44-96%, 7-95%, and 24-99% for the often studied grain species of C, SiO_2, and Fe, respectively.

- As in Paper I, the initial distribution in grain radius greatly influences the overall survival of a given species, with grains with initial radii below 0.1 μm being destroyed in most simulations.

- For the high-metallicity ($Z = 100\ Z_\odot$) simulations, the balance between the increased radiative cooling and increased sputtering yield depends strongly on the speed of the shock. When the shock velocity is below 3000 km s^{-1}, the increased cooling lowers the temperature of the gas enough to suppress dust destruction. At high velocities, the gas temperature is driven up to a high-erosion regime.

In comparing this new suite of simulations to that of Paper I, the key differences are the higher metallicity cooling functions, the erosion rates for enhanced metal abundances for all elements up to iron, the exploration of higher shock velocities ($v_{\rm shock} = 10^4$ km s^{-1}), and the suppressed cooling in the ambient medium surrounding the cloud. As a consistency check, we find that the simulations

with $Z = Z_{\odot}$ for shock velocities of $v_{\rm shock} = 1000$ km s^{-1}, 3000 km $^{-1}$, and 5000 km s^{-1} produce very similar dust mass survival fractions, indicating that these changes do not invalidate our previous results. In fact, the degree of dust survival does not begin to deviate significantly from Paper 1 until the metallicity is increased to $Z = 100\ Z_{\odot}$, as expected.

We also find that our dust survival fractions remain in agreement with observations of IR emission in SNRs. Specifically, the continued survival of Si grains matches the high abundances found in Cas A by Rho et al. (2008), while complete destruction of Al_2O_3 grains and survival of amorphous carbon grains match observations of SNR 1E0102-7219 by Sandstrom et al. (2009).

Ideally, we would have liked to explore a metallicity regime in which the gas in the ejecta cloud is saturated with metals, effectively mimicking clouds of pure oxygen, silicon, sulfur, argon, or calcium. Using oxygen as an example, an abundance ratio of order one would require a metallicity of $Z \sim 1000\ Z_{\odot}$ since the solar abundance ratio is $n_{\rm O}/n_{\rm H} \sim 5 \times 10^{-4}$. Unfortunately, due to the extremely short cooling times, this proved computationally prohibitive. The simplest solution is to limit the simulation time-step by the cooling time and allow the simulation to run for a longer physical time, but it is also the most computationally expensive. In order to probe the highest metallicity regimes, a more resource-efficient method would need to be implemented without sacrificing the validity of the simulation. Such a solution is left to future work.

Currently, we still operate in the limit of strictly thermal sputtering, since our dust grains are directly coupled to the flow of the gas. Because large grains would require a greater transfer of momentum from the gas before becoming completely entrained in the flow, it is possible that the dominant mode of sputtering for such grains would be in the non-thermal limit, as high velocity gas washes over them. In order to decouple the dust particles from the motions of the gas, we would have to track separated grain populations of varying radii. The number of radius bins required to accurately track the entire dust mass for all nine grain species is uncertain, but it could be computationally significant. As mentioned in Paper I, it may not be unreasonable for the dust particles to be coupled to the flow if the grains are charged and magnetic fields permeate the ejecta material. However, magnetized ejecta clouds would require an investigation of the effects of

betatron acceleration on charged grains (Shull, 1977, 1978).

Finally, we investigated the possibility of transitioning from post-processing of our dust particle histories to compute erosion to an "on-the-fly" method that would sputter the grains as the simulation ran. This would allow metals that are sputtered off the grains to feed back into the ambient medium and increase the metallicity. However, if we assume that the gas within the ejecta cloud is already heavily metal-enriched, the amount of metals released into the gas in this way would not alter the metallicity enough to make a substantial difference. As we have shown, it takes a large change in the metal abundance to create an significant difference in the amount of dust that survives the shock-cloud interaction.

Chapter 5

Non-equilibrium Ionization Chemistry

Accurately solving the non-equilibrium (NEQ) state of ion species in metal-enriched gas via numerical simulations is a considerable undertaking, but one that is validated by significant payoffs. Not only do physically realistic estimates of ion abundances afford the opportunity to precisely calculate radiative cooling rates, which can have great impact on the thermodynamic properties of the gas, but they also increase the viability of comparing computational results to astronomical observables. The development of `Dengo`, a software package for solving complex chemical networks, aims to provide this capability to any and all simulation codes currently in use in the astrophysical community.

One of the distinct advantages of `Dengo` is that each of the required components for solving a chemical network is separately created and defined. This allows for components to be modified without requiring the entire solver to be rewritten. Dengo's flexibility can be particularly important when defining not only the species that need to be tracked for a given astrophysical scenario, but the rates that control their formation and destruction. In the case of primordial star formation, Turk et al. (2011a) found that variations in the rate that determines the formation of molecular hydrogen had significant impacts of the final state of their simulations. Therefore, the ability to swap rates in and out of `Dengo` with minimal effort enables simplified parameter space studies.

For this work, we limit our studies to include only atomic species and leave the inclusion of molecular species to other works. To calculate the ion populations of hydrogen and helium, we use the rates presented in Abel et al. (1997); for the solutions that incorporate radiative cooling,

we use the rates compiled by Anninos et al. (1997). For all other atomic species, the ionization and recombination rates come from the CHIANTI atomic database (Dere et al., 2009). Radiative cooling rates for those species are provided by the tabulated ion-by-ion cooling efficiencies calculated by Gnat & Ferland (2012), which are publicly available[1] . We note that these rate tables only span a temperature range of 10^4 K to 10^8 K; future work will involve compiling rates to expand this range, especially for lower temperatures.

With all of the machinery in place, the species selected, and the rates compiled, the first step was to test `Dengo` as a stand-alone solver, the results of which are presented in Section 5.1. Once we confirmed that `Dengo` could accurately evolve a full network of ion species, we proceeded to run the "first light", `Dengo`-enhanced simulations of the intergalactic medium (IGM). These simulations are the first step in understanding the impact of including NEQ calculations in numerical studies of the WHIM and making connections to observations of the IGM. The details of these simulations and their results are included in Section 5.2.

5.1 Testing `Dengo`

For the tests in this section, we include a limited subset of all possible atomic species to serve as a proxy for a full network. Specifically, we include H, He, C, N, O, Ne, Mg, Si, and S. These species were chosen due to their connection to commonly observed elements and because they provide a reasonably large network (86 species with the inclusion of electrons) to test the capabilities of `Dengo`. Since the current implementation of `Dengo` in `Enzo` solves each cell separately, the presented tests are also for individual cells. However, `Dengo` has the additional capability of solving multiple cells simultaneously. The tests check that `Dengo` is capable of accurately reaching the equilibrium solution and self-consistently cooling a ionized gas based on the individual ion abundances. In both tests, it is assumed that the abundance ratios of all metals with respect to hydrogen follow the abundances from Caffau et al. (2011) and Lodders et al. (2009), a set that comes bundled with the CHIANTI for $Z = Z_{\odot}$.

[1] http://wise-obs.tau.ac.il/∼orlyg/ion_by_ion/

5.1.1 Evolving to Equilibrium

The first test we present ensures that, when `Dengo` is provided the initial conditions for a gas that is out of equilibrium, it correctly arrives at the values expected for collisional ionization equilibrium. To confirm this, we initialize a single cell of gas with a hydrogen number density of $n_H = 1$ cm^{-3} in a neutral state at a temperature of 5×10^6 K, where we define "neutral state" to represent the scenario in which only elements in their first (neutral) ionization state are initially populated. From this point, we evolve the cell forward in time until all of the atomic species reach their equilibrium values. Figure 5.1 shows the time evolution for temperature and five of the nine atomic species included in the test (H, He, C, N, and O). We only include five of the nine species because as the atomic number increases, the number of ion lines on the plots become difficult to distinguish. Note that the changes in temperature are due to the increase in the number density of electrons because we assume that the gas remains in pressure equilibrium. Radiative cooling is not included in this test case, and the gas energy of the cell remains constant. When the final ion fractions are compared to the CIE values from CHIANTI, there is complete agreement for all ion species.

5.1.2 Counteracting Ionization with Cooling

For the second `Dengo` test included in this work, we take a gas that is initially at the expected CIE values and allow it to radiatively cool and recombine. Again, we initialize a single cell of gas with $n_H = 1$ cm^{-3} and $T = 5 \times 10^6$ K. At this temperature, all of the atomic species included in this test start out with an ion fraction $X_s \sim 1$ for the completely ionized state, where s denotes the ion species. When we evolve the test forward in time, the temperature drops gradually at first until the abundances of ions with high radiative cooling efficiencies (e.g. O VI) begin to rise. At this point, the gas cools much more rapidly until it hits the temperature floor of 10^4 K present in the metal species cooling tables of Gnat & Ferland (2012). We run the test just a bit past the steep drop in temperature to allow the metal species to almost fully recombine. Figure 5.2 provides a

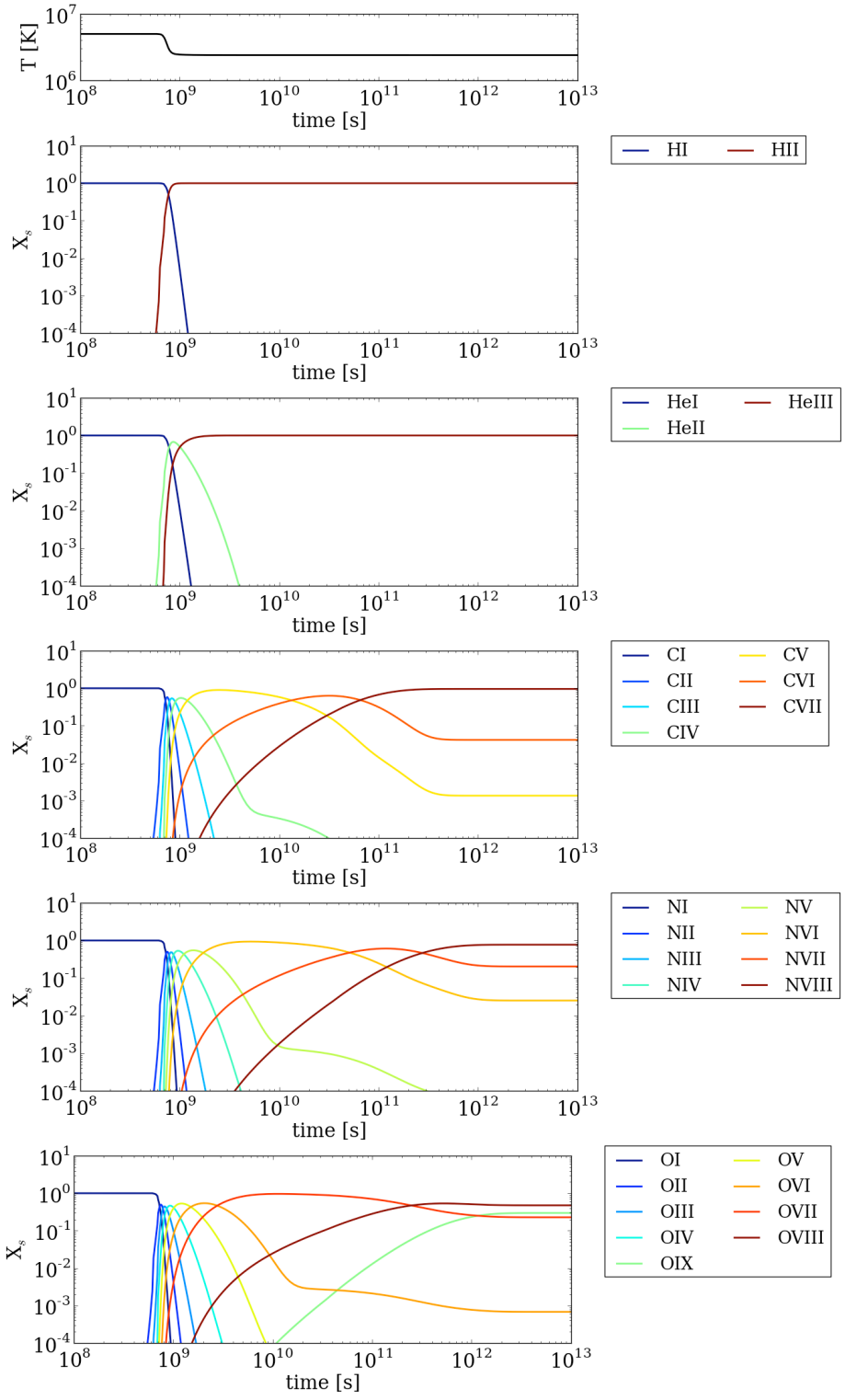


Figure 5.1: The temperature (top panel) and ion fractions, X_s, for H, He, C, N, and O (bottom five panels) as a function of time for the `Dengo` ionization test. All of the atomic species are initialized to be in the neutral state.

visual evolution of this test.

We also use this test to compare to the time-dependent ionization results of Gnat & Sternberg (2007), who found that, for a radiatively cooling gas, the ion fractions can shift far from their CIE values. In our test, we find similar results, which are presented in Figure 5.3 for three ion species (C IV, O VI, and Ne VIII) commonly observed in UV quasar absorption-line studies and important radiative coolants via their 2s→2p transitions. The primary point to note is that all three of the Lithium-like ions shown in Figure 5.3 account for a significant fraction of the total atomic species over a much larger range in temperature than they would if they were in equilibrium. This outcome is driven by the rapid drop in the cooling time as the gas reaches a temperature where these ion species become powerful coolants. As the cooling time approaches the recombination time, the ions do not recombine fast enough to reach their CIE values. The variations in the shape of the ion fraction plateaus that form at low temperature for C, O, and Ne are due to the inherent differences in the recombination rates of these ions. These results highlight the importance of including NEQ calculations in simulations of the IGM.

5.2 Applications to the Intergalactic Medium

Having confirmed that `Dengo` can accurately solve complex chemical networks for single-cell tests, the next step in this thesis work was to apply `Dengo` to full-scale simulations of the intergalactic medium (IGM). This required coupling the `Dengo` solver to `Enzo` in such a way that the ionization state of the gas could be calculated in real time along with the standard hydrodynamic and gravitational evolution of the Universe. This was accomplished by allowing `Dengo` to sub-cycle within a given `Enzo` time step, and to reach an ionization solution before allowing `Enzo` to take another time step. While this increases the computational resources required to run a cosmological simulation to completion, we do not find the addition of `Dengo` to be prohibitively expensive. In various tests of our `Enzo`+`Dengo` hybrid code, we find that our simulations take anywhere from four to twelve times more CPU hours than `Enzo`-only simulations. Given that these slow-downs are present in a code base that has not yet been fully optimized for speed, we deem the additional

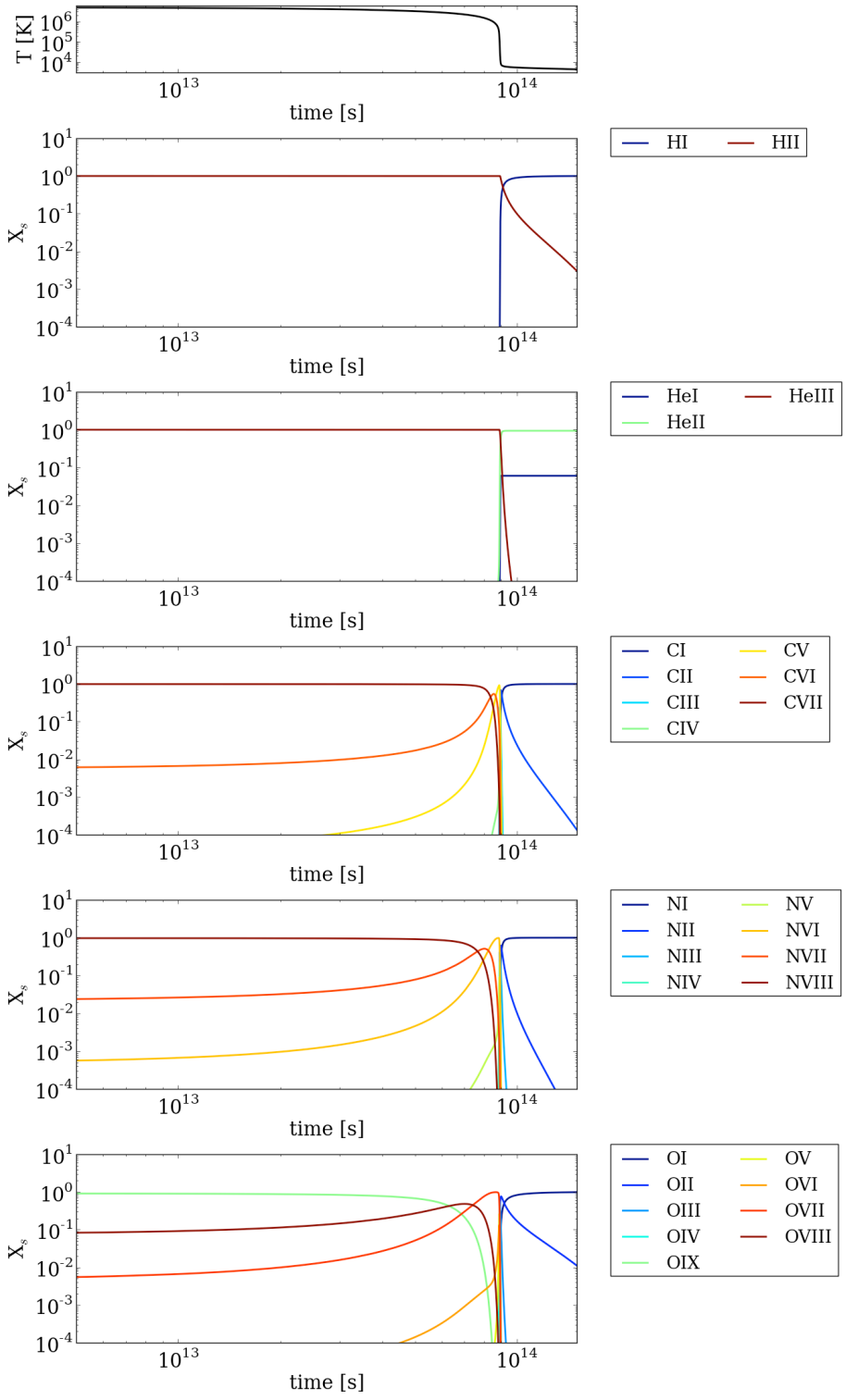


Figure 5.2: The temperature (top panel) and ion fractions, X_s, for H, He, C, N, and O (bottom five panels) as a function of time for the `Dengo` cooling test. The ion fractions are initialized at their collisional ionization equilibrium (CIE) values at $T_0 = 5 \times 10^6$ K.

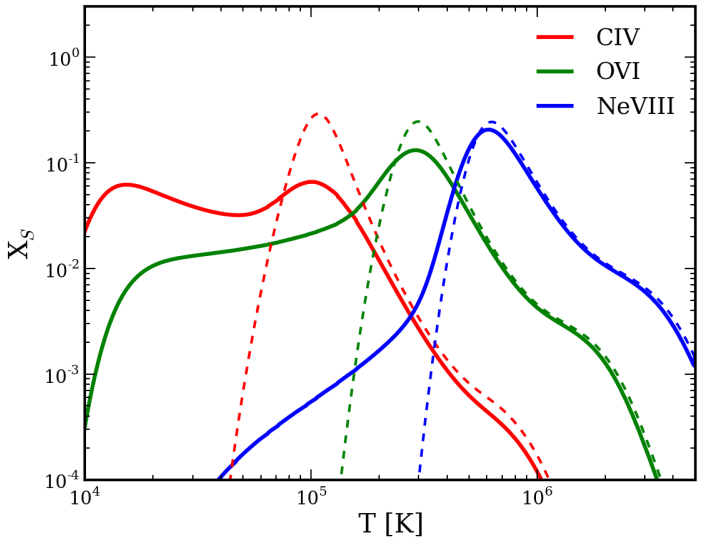


Figure 5.3: The ion fractions, X_s, of C IV (red), O VI (green), and Ne VIII (blue), three commonly studied ion species in UV quasar spectra, as a function of temperature for the `Dengo` cooling test. The solid lines presents the values output by `Dengo`, while the dashed lines show the collisional ionization equilibrium (CIE) values.

computational costs to be reasonable.

In the sections that follow, we focus on the simulations that have been run to highlight the effects of including NEQ calculations in simulations of the IGM. Section 5.2.1 provides the details of the simulations, while Section 5.2.2 explains the results.

5.2.1 The Simulations

The IGM simulations discussed in this section are intentionally designed to be as similar as possible to the simulations of Smith et al. (2011). As such, they are initialized at a redshift of $z = 99$ with a power spectrum of density fluctuations that follows Eisenstein & Hu (1999) for a ΛCDM universe. We use a set of cosmological parameters that are in approximate agreement with the WMAP-7 results; $(\Omega_b,\ \Omega_{\rm CDM},\ \Omega_\Lambda,\ h,\ \sigma_8,\ n_s) = (0.0441,\ 0.2239,\ 0.732,\ 0.704,\ 0.82,\ 1)$. As mentioned in Section 2.1.2, we also include a star formation prescription that follows Cen & Ostriker (1992) and a metagalactic UV background that evolves in redshift based on Haardt & Madau (2001). Finally, the one area where we diverge from Smith et al. (2011) is in our mechanism

for computing the cooling rate of the gas. Whereas Smith et al. (2011) used a multidimensional lookup table of heating and cooling rates produced by `Cloudy` coupled to the NEQ primordial chemistry in Enzo, we use the static lookup table that follows the analytic formula of Sarazin & White (1987) for $Z/Z_{\odot} = 0.5$ gas. This chosen method for computing the cooling rates is temporary and will ultimately be replaced by the ion-by-ion cooling rates calculated by `Dengo`.

Thus far, we have run two simulations with identical box sizes of 25 comoving Mpc h^{-1}, each initialized with the same random seed so that the final large-scale structure is the same. We limited our first unigrid simulation to consist of 256^3 cells and 256^3 dark matter particles. This provides a comoving cell resolution of $\Delta x = 98$ kpc h^{-1} and a dark matter particle mass of $m_{dm} = 6 \times 10^7\ M_{\odot}$ h^{-1}. Since this simulation was intended to be a test bed for the addition of `Dengo`, it was chosen to be the smallest possible computational volume that still resulted in appreciable star formation and significant metal injection. If the cell resolution is decreased much beyond this level, few cells manage to meet the criteria necessary to produce stars.

Once we ensured that `Dengo` was running correctly within the 256^3 simulation, we increased the number of cells and the particle count to 512^3. This results in a comoving cell resolution of $\Delta x = 49$ kpc h^{-1} and a dark matter particle mass of $m_{dm} = 7 \times 10^6\ M_{\odot}$ h^{-1}. All of the other simulation parameters are identical to the smaller computational volume. Since this larger simulation better resolves the physical structure of evolving universe, we restrict all of the analysis presented in this work to the datasets it produced. In future work, we will pursue a more careful investigation of the effects that resolution may have on the NEQ properties of the gas.

Finally, in this initial investigation of the impact of doing a proper NEQ treatment of the ionization chemistry in the IGM, we use `Dengo` to track H (H I and H II), He (He I, He II, and He III), and O (O I through O IX) in our simulations. While we have the ability to track additional elements, we chose to restrict ourselves to only these atomic species in order to minimize the computational resources required to store the additional data fields and focus our attention on one of the most commonly studied metal species. Again, future work will expand on the number of species included in our simulations.

The sections that follow will discuss the "first light" results of the 512^3 NEQ IGM simulation. The primary focus of these results is to compare the broad-stroke differences between assuming that all ions are in CIE and computing the ionization state of the gas in real time. For all of this work, we use the CIE fractions from Sutherland & Dopita (1993).

Since the long-term goal of this work is to compare our results to observational data, we couch these comparisons in the context of how things change for O VI, O VII, and O VIII, primary observable species in UV and X-ray absorption-line studies. At the end of this chapter we briefly discuss some of the bulk properties of what is arguably the favorite ion of UV absorption-line studies, O VI.

5.2.2 The Results

For the majority of the results in this section we focus on the final, $z = 0$, state of the simulation. In order to get a handle on the physical structure of the computational volume, we refer the reader to Figure 5.4, where we've included projections through the entire simulation domain of mass-weighted density, mass-weighted temperature (T), Mach number ($\mathcal{M}$), and mass-weighted metallicity (Z). These projections and the ones included in Figure 5.5 were created using the simulation analysis package `yt`[2] (Turk et al., 2011b). For the weighted projections, the projected values are computed as $\sum[v \times w] \;/\; \sum w$, where v is the value of the cell and w is the value of the weighting field. As an example, the mass-weighted projection of temperature is calculated such that $T_{proj} = \sum_i [T_i \times \rho_i] \;/\; \sum_i \rho_i$, where i denotes the cell number.

Features of note include the expected filamentary structure, commonly referred to as the "Cosmic Web", the high temperatures local to cosmological haloes, and the dispersal of metals driven by the feedback mechanisms employed by star particles. Additionally, the projection of $\mathcal{M}$, calculated using the shock-finding algorithm of Skillman et al. (2008), shows high values in the cold, under-dense cosmological voids and lower values in the stellar feedback regions within the haloes. This somewhat counter-intuitive result is due to the fact that cold gas from voids that accretes

[2] http://yt-project.org

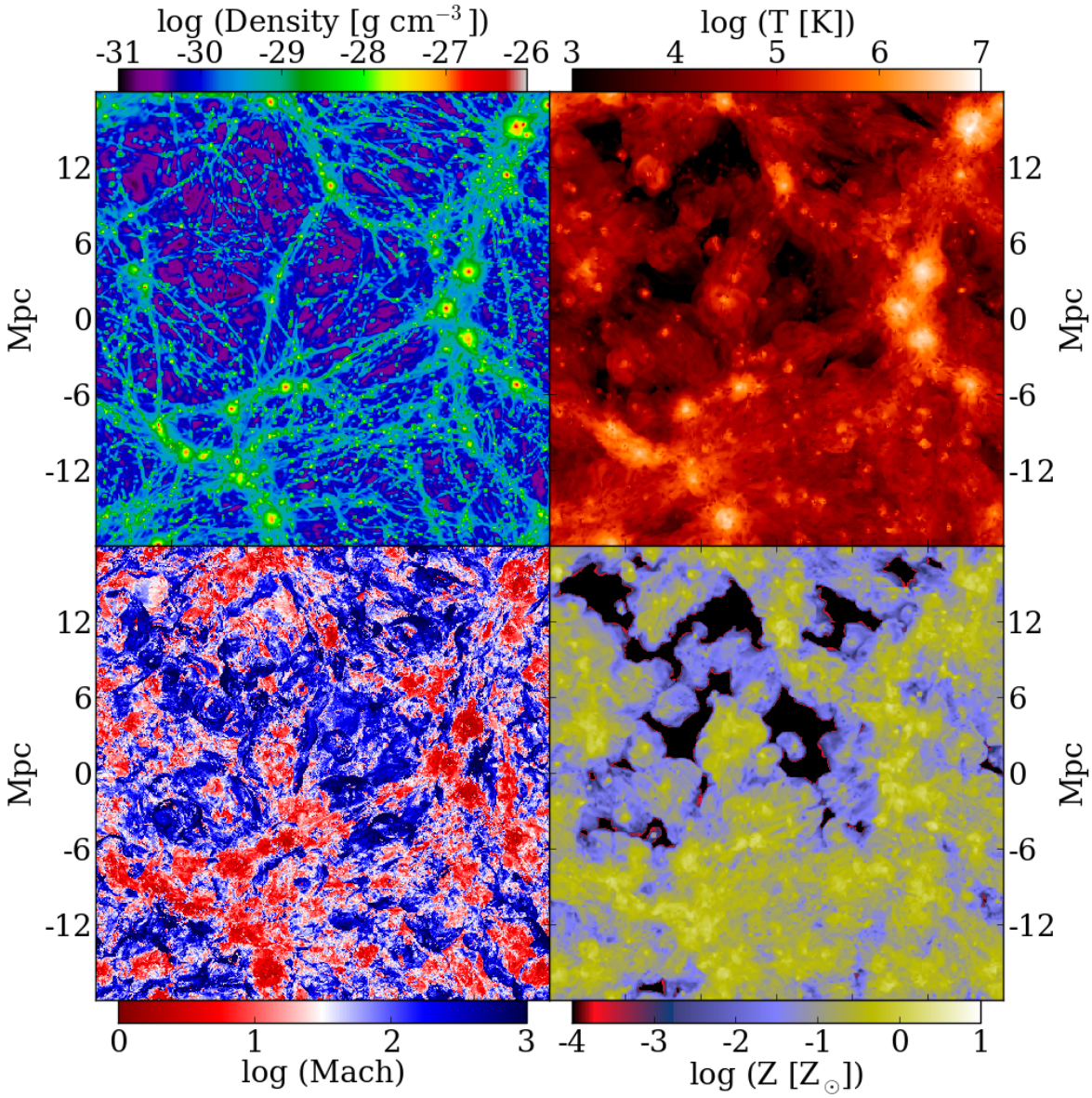


Figure 5.4: Projections through the full computational domain (512^3, 25 comoving Mpc h^{-1}) of mass-weighted density (upper left), mass-weighted temperature (upper right), energy-weighted Mach number (lower left), and mass-weighted metallicity (lower right) at $z = 0$.

onto cosmological filaments and galaxy clusters is moving at high velocities with an inherently low sound speed while hot gas running into other hot gas inside haloes has lower relative velocities and temperature contrasts. It is also important to note which regions contain appreciable abundances of metals, as these will be the same regions capable of harboring the ion species of oxygen.

5.2.2.1 CIE vs. NEQ

Having acquired a sense of the overall physical properties of the simulation, we now focus our attention of the differences between a post-processed, CIE calculation of the ion fractions and the new, `Dengo`-enabled NEQ ion fractions computed during run time. First, in Figure 5.5, we present projections of both the CIE-based and NEQ-calculated number densities of O VI, O VII, and O VIII in the same orientation as Figure 5.4 for $z = 0$. By projecting the number density of these species along one of the simulation axes, we get a proxy for the column density of these ions because the projected values are calculated as the sum over all 512 cells, $\sum n_{ion} \Delta l$, where n_{ion} is the number density of the selected ion in that cell and Δl is the cell length. At $z = 0$ the total path length through the box is 35.3 Mpc. A simple by-eye inspection of the NEQ projections shows substantially more ion presence at greater distances from the centers of haloes than the CIE values for all three species. Currently, we have not yet quantified how much further out on average these species tend to exist, but such work is in progress.

In order to understand the thermodynamic state of the excess ion abundances, we show the fractional amount of ion mass in a given baryon overdensity and temperature bin in Figure 5.6. From this perspective, it becomes clear that there is a considerable amount of mass outside of the expected CIE values. In particular, all three ions (O VI, O VII, and O VIII) show appreciable mass for a wide range in baryon overdensity down to temperatures of order 10^4 K, well below the point where CIE would dictate that they should have recombined into lower ion states. However, in the regime of high overdensities and high temperatures, the NEQ simulation shows strong agreement with the post-processed CIE value. This confirms that `Dengo` is correctly calculating the ion abundances and cells in this region of phase space. When the ionization and recombination timescales are short, the ion abundacnes are evolving toward their CIE values.

To get a better handle on how the NEQ solution differs from the CIE calculations, Figure 5.7 shows the amount of ion mass at a particular ion fraction for the same baryon overdensity and temperature phase space (though now separated). The right side of Figure 5.7 is particularly

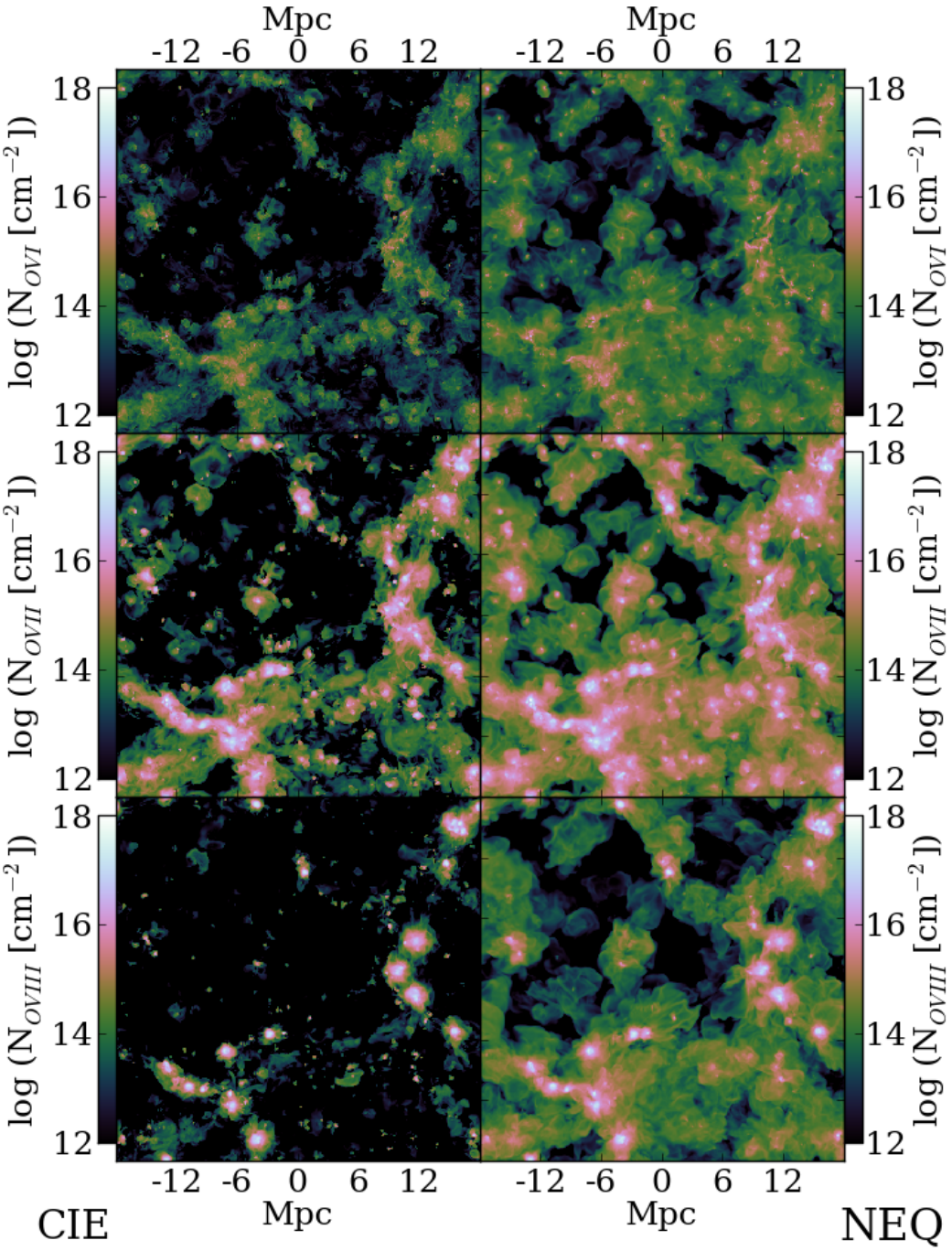


Figure 5.5: Projections through the full computational domain (512^3, 25 comoving Mpc h^{-1}) of number density for O VI, O VII, and O VIII at $z = 0$. At this redshift, the full path length through the box is 35.3 Mpc. The left panels use the post-processed collisional ionization equilibrium (CIE) values and the right panels use the non-equilibrium (NEQ) values computed in the simulation.

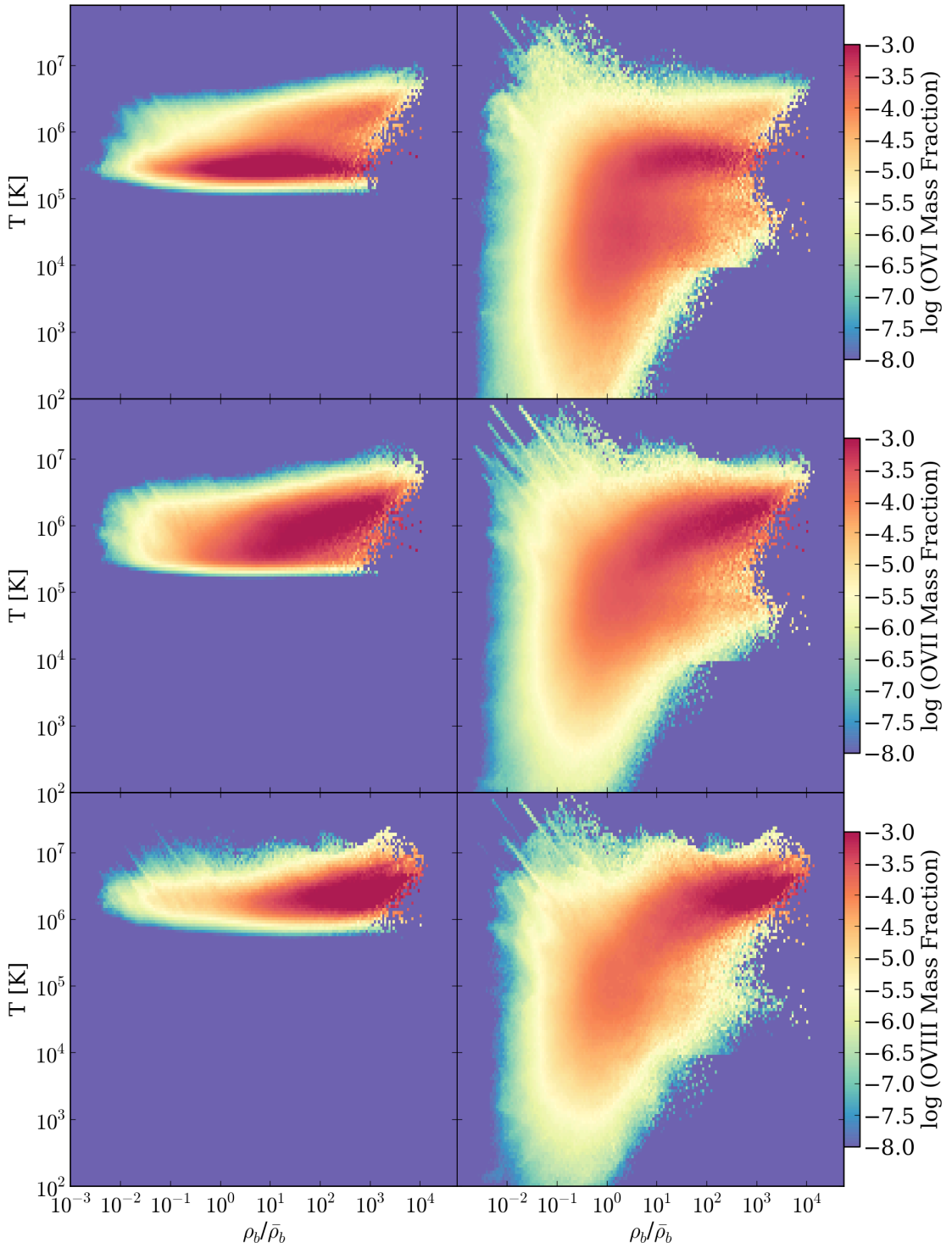


Figure 5.6: Fraction of total O VI (top), O VII (middle), and O VIII (bottom) mass as a function of baryon overdensity (x-axis) and temperature (y-axis). The panels on the left show the values based on the collisional ionization equilibrium (CIE) solution, and the panels on the right show the values for the non-equilibrium (NEQ) solution.

informative, allowing us to compare directly with what the CIE values for the ion faction should be as a function of temperature. In performing this comparison, it becomes clear, that, although there is a population of gas that traces the CIE fractions, there is another very apparent population that maintains high ion fractions down to low temperatures. We interpret this population as gas that is originally heated via stellar feedback, which proceeds to adiabatically expand and cool into a regime with long recombination times, effectively "freezing in" the ion fractions.

In addition to this adiabatic freeze-in effect, there is also an effect that is analogous to the results from the `Dengo` test discussed in Section 5.1.2, which produces additional highly ionized species at low temperatures, but at higher overdensities. At these higher overdensities, the cooling rates remain high enough such that the cooling time approaches the recombination time. This additional population, created by radiative freeze-in, can be observed in Figure 5.6 at overdensities ranging from 10 to 1000 and temperatures below 10^5 K - 10^6 K. It is important to note that since our current treatment of the cooling uses a static look-up table for a constant metallicity, the nature of this population will likely change when we enable `Dengo`'s full functionality and directly couple the cooling to the ionization state of the gas.

The combination of freeze-in effects results in not only more highly ionized species in a greater region of the density-temperature phase space, but also more total oxygen mass at these ionization levels. To understand exactly how much more mass is contained in these species when compared to the CIE values, we multiply the mass density of the ion, ρ_{ion}, in a given cell by the cell volume, V_{cell}, and then sum those masses over the entire computational domain. We carry out this calculation for both the NEQ values from the simulation and the associated CIE-calculated values. Table 5.1 displays the results of these calculations as well as the ratio of the NEQ values to the CIE values. The jump in O VI mass is the most drastic, owing to the rather narrow temperature range and low peak of its CIE fractions, but O VII and O VIII show appreciable increases as well. This result could have significant implications when comparing simulations to observed ion abundances and in producing an accurate baryon census.

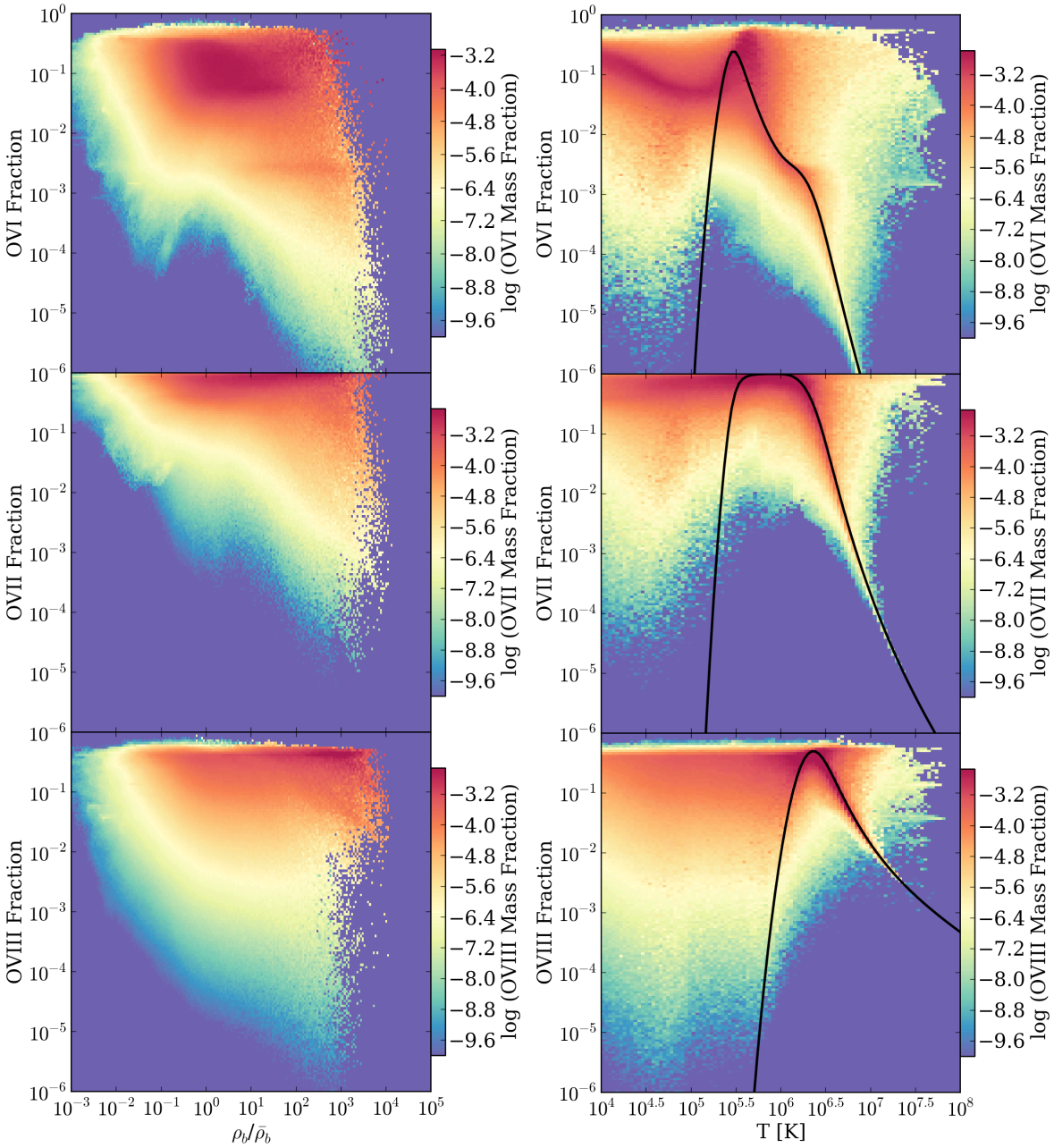


Figure 5.7: Fraction of total O VI (top), O VII (middle), and O VIII (bottom) at a particular ion fraction as a function of baryon overdensity (left panels) and temperature (right panels). For the temperature plots, the collisional ionization equilibrium (CIE) values for the ion fractions are overplotted as the solid black lines. This provides a useful means of determining where the non-equilibrium (NEQ) solution diverges from CIE.

5.2.2.2 The Infamous O VI

For the final section of this chapter, we take our first stab at producing results that eventually will be directly comparable to observations. In particular, we have made estimates of the number

Table 5.1. Ion Masses

Species	M_{ion} $[M_\odot]$ (CIE)	M_{ion} $[M_\odot]$ (NEQ)	NEQ/CIE
O VI	8.29×10^9	2.81×10^{10}	3.39
O VII	2.06×10^{11}	3.75×10^{11}	1.82
O VIII	9.22×10^{10}	1.72×10^{11}	1.86

Note. — The first two columns show the total amount of mass contained with in the computational domain for O VI, O VII, and O VIII for both the (CIE) solution and the non-equilibrium solution (NEQ). The third column shows the ratio of these two values.

of O VI absorbers that should be expected observationally based on the O VI densities present in our simulation. To achieve this, we take our simulation and generate 200 random AGN sight lines that traverse a range in redshift from $z = 0$ to $z = 0.4$, following the same method used by Smith et al. (2011). This range in redshift is of particular use for comparison with QSO absorption-line surveys, like those presented by Danforth & Shull (2008) and Tilton et al. (2012).

As an individual ray travels through the simulation domain, we record the number density of O VI for each computational cell that it intersects and the path length, dl, it takes through that cell. With this information, we are able to compute the column density of O VI for each line element, $N_{OVI} = n_{OVI} \times dl$. Once we have these values, we can attribute contiguous line elements with sufficient column densities as coherent structures capable of producing a single absorption line. This allows use to carry out the same sort of absorber count statistics as Danforth & Shull (2008) and Tilton et al. (2012). Specifically, we have calculated the cumulative number of O VI absorbers, $d\mathcal{N}(> N)/dz$, above a given column density N as a function O VI column density. We have done this calculation for the NEQ values pulled directly from our simulations, for the post-processed CIE values, and for the CIE + photoionization (C+P) post-processing method used by Smith et al. (2011). Although we limited our previous investigations to the differences between NEQ and CIE, we have included also the C+P results as a means of highlighting how the NEQ distribution might change when we eventually add photoionization support to `Dengo`. The results of these calculations

are presented in Figure 5.8.

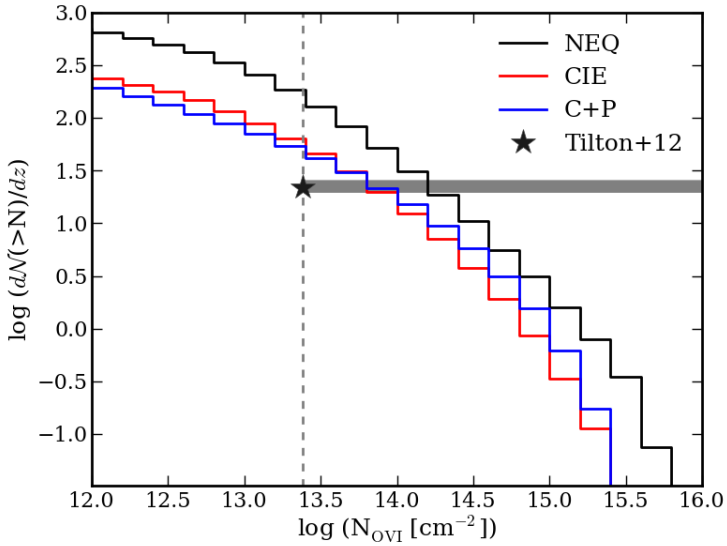


Figure 5.8: The cumulative number density of O VI absorbers above column density N per unit redshift as a function of absorber column density as computed from 200 random AGN sight lines that span $z = 0$ to $z = 0.4$. The black line shows the results based on the number density of O VI for the non-equilibrium (NEQ) solution as computed by `Dengo`. The colored lines show the same quantities for the post-processing methods used by Smith et al. (2011) for collisional ionization equilibrium (CIE; red) and the combination of CIE and photoionization (C+P; blue). The $\star$ indicates the value of $d\mathcal{N}/dz$ when Tilton et al. (2012) integrated down to $\log(N_{OVI}) \approx 13.38$. The vertical dashed line highlights the column density of this value while the shaded region shows the 1-σ errors on the value of $d\mathcal{N}/dz$. The discrepancy between the simulated and observed values is discussed in Section 5.2.2.2.

The clear difference between the NEQ method and the CIE assumption is that there are more O VI absorbers at all column densities. Although significant, this result is not entirely unexpected and directly follows from the overall increase in O VI mass indicated by Table 5.1. By comparing the CIE values to those of the C+P method, we see that including photoionization decreases the number of low column density absorbers while boosting the high column density ones. Naively, one would expect to see similar shifts in the absorber count for an NEQ+P solution, but given the drastic change in results that stem from a time-dependent treatment of collisional ionization, it is not entirely clear what effects a time-dependent treatment of photoionization may have. Coupling the photoionizing background to the ionization rates in `Dengo` will serve as a first step in determining the nature of these effects. This step is, of course, not without complications, as the nature of the

metagalactic background is highly uncertain in the extreme-UV regime (Shull et al., 2012). It is this regime that produces photons that have sufficient energy to ionize O V and O VI, which require 113.87 eV and 138.08 eV photons, respectively.

We note that the absolute values shown in Figure 5.8 are all higher than what is generally found observationally. Specifically, if we compare our results to the Tilton et al. (2012) point (the star in Figure 5.8), where they found $d\mathcal{N}/dz = 22.2$ when they integrated down to $\log(N_{OVI}) \approx 13.38$ cm^{-2}, we see that our results are high by roughly an order of magnitude. Again, this is likely due to the overly simplified method used for calculating the cooling rate of the gas. The look-up table assumes uniform metallicity for all gas, but in reality, some of that gas will be at much lower metallicities. This gas will overcool, drive up the star formation rate, and increase the consequent injection of metals. Therefore, the abundance of metals throughout the simulation volume is likely higher than it would be in the observed IGM, and the column densities for all metal absorbers are correspondingly boosted.

5.3 Discussion

While we are still in the preliminary stages of extracting observables from these NEQ simulations of the IGM, we find the overarching differences between the NEQ solutions and expected CIE values to be very compelling. Furthermore, given the magnitude of disparity in certain regimes, including a NEQ chemistry solver for the metal species in IGM simulations is critical for understanding the true ionization state of the medium. Now that `Dengo` is confirmed to be functional and accurate, we can make such simulations a reality. While further development of the software package is still required, it should prove to be a powerful tool as we move forward in our investigation of the ionization structure of the IGM. Furthermore, as the resolution of our simulations improves, especially with a carefully crafted AMR criterion, we will be able to extend our NEQ solutions into studies of the circumgalactic medium around galaxies.

Chapter 6

Conclusion

6.1 Summary

My investigations into the chemical evolution in the Universe via numerical simulations can be summarized as follows:

- Dust destruction in supernova remnants
  - Cloud-crushing simulations that model the interaction between supernova ejecta and the reverse shock of the supernova remnant can be a useful tool for estimating the amount of dust destruction that occurs as the metal-enriched gas travels through the remnant and into the interstellar medium.
  - The relative velocity between the ejecta knot and the reverse shock is a dominant factor in determining the total dust destruction that occurs during the ejecta-shock interaction.
  - For high metallicity ($Z/Z_{\odot} = 100$) gas, there exists two competing factors that influence the total amount dust destruction: increased radiative cooling and increased thermal sputtering. In the case of low relative shock velocities, the increased cooling decreases the temperature enough to reduce the amount of dust destruction while high relative velocities destroy significantly more dust, owing to the increased sputtering.
  - Although the amount of dust destruction varies significantly over the explored parameter space, in the metal enriched cases, the amount of dust destruction reaches over

50% for all grain species with total destruction for some. This brings into question whether or not supernova alone can produce the necessary dust masses ($\sim$1 $M_{\odot}$ per supernova event) to match high-redshift observations.

- Non-equilibrium ionization chemistry
  - The new `Dengo` software package for solving chemical networks has been tested for accuracy and implemented within cosmological `Enzo` simulations of the IGM.
  - By using `Dengo` to track the real-time ionization state of oxygen in these simulations, it is clear that the non-equilibrium (NEQ) state of the gas diverges significantly from the values of gas that is in collisional ionization equilibrium (CIE).
  - The oxygen ion species commonly studied in UV and X-ray studies, O VI, O VII, and O VIII, exist at a larger spatial extent around metal-producing cosmological haloes than CIE calculations predict and efforts to quantify exactly how much larger are in progress.
  - The primary effects that drive the oxygen ion fractions out of equilibrium are adiabatic and radiative "freeze-in". In the adiabatic case, gas that is heated and ionized by stellar feedback expands and cools more rapidly than the highly ionized species are able to recombine. This results in large ion fractions for O VI, O VII, and O VIII at temperatures where they would not normally be present at all. In the radiative case, higher density gas ends up with cooling times that begin to approach the recombination times for highly ionized species and, again, drive up the ion fractions for these species at low temperatures. These frozen in populations may change when the effects of photoionization are included.
  - Not only do NEQ calculations increase the spatial extent of highly ionized species, but they also result in greater total oxygen masses at these ionization states. This could have a significant impact on baryon census calculations.

* Early calculations of observable quantities show that NEQ results produce significantly more O VI absorbers than CIE calculations. While this result may change when the effects of photoionization are included, it has interesting implications in comparing simulations to observations and warrants further investigation.

6.2 Plans for future work

As I move forward in my research efforts, my primary focus for the immediate future is to improve and expand the capabilities of `Dengo` and use this new software package to produce state-of-the-art simulations of the intergalactic and circumgalactic media. The specific goals of this future work include:

- Enabling the components of `Dengo` that calculate the radiative cooling rates so that the internal energy of the gas evolves self-consistently with the ion fractions of the metal species.

- Adding photoionization terms to `Dengo`, which will allow for a more direct comparison with the collisional ionization equilibrium + photoionization results of Smith et al. 2011. As a first step, the photoionization terms will be coupled to the uniform metagalactic UV background. The long term goal will be to use one of the radiation transport modules currently in `Enzo` to provide these terms.

- Using the PPM solver for evolving the hydrodynamics of the gas. While the ZEUS solver is sufficient for the work presented here, the PPM solver is better at accurately resolving shocks and will help to ensure that the physical structure of the IGM in the simulations is a true representation of what is expected in the Universe.

- Using tracer particles to track the thermodynamic history of the gas. Currently, SPH simulations have the ability to show exactly where gas that ends up in a particular phase at $z = 0$ originated from, while our grid-based simulations do not. This can be a very powerful tool for understanding how gas that resides in the WHIM phase arrived there.

Tracer particles would provide this capability without having to sacrifice the advantages provided by using an Eulerian code.

- Compiling a more complete set of rate coefficients for use in `Dengo`. In particular, ion-specific radiative cooling rates that span a larger range in temperature are needed.

- Using AMR to drill down to higher physical resolutions so that the nature of circumgalactic medium can be investigated. This will likely require defining a new set of refinement criteria in order to balance the increased resolution with the computational cost. One such criterion may involve refining the simulation domain based on the second derivative of the non-equilibrium ion species. This would provide high resolution in any areas of the computational domain where ion species of particular interest are being created or destroyed.

- Calculating more quantities that can be directly compared to observations. By creating synthetic observables, high quality simulations of the IGM that incorporate the above changes will be able to confirm the results of current observations and predict the results of future ones.

While some of these goals may require considerable time and effort to ensure that the simulation results are accurate and robust, others will be possible on relatively short time scales. In particular, more accurately calculating the radiative cooling rates and coupling the ion species to a uniform photoionizing background is on the very near horizon. Even minor additions such as these will be a large step forward in producing more physically realistic simulations of the IGM and should help to validate the synthetic observables that can be extracted.

Finally, it is my hope that as `Dengo` becomes an evermore powerful software package for handling complex chemical networks it will be adopted and used by a large community of computational astrophysicists. Not only will this serve to enhance simulations of a diverse range of astrophysical phenomena, but it will drive further development of the code which undoubtedly improve its overall

utility. Community feedback and collaborative development of open source software is a process that enables the rapid scientific advancements in our field and a process that I have greatly enjoyed being a part of while producing this thesis.

Bibliography

Abel, T., Anninos, P., Zhang, Y., & Norman, M. L. 1997, New Astron., 2, 181

Abel, T., Bryan, G. L., & Norman, M. L. 2002, Science, 295, 93

Anninos, P., Zhang, Y., Abel, T., & Norman, M. L. 1997, New Astron., 2, 209

Barlow, M. J. et al. 2010, A&A, 518, L138

Bedogni, R. & Woodward, P. R. 1990, A&A, 231, 481

Behrisch, R. 1981, Sputtering by Particle Bombardment, Vol. 1, Physical sputtering of single-element solids (Springer-Verlag)

Berger, M. J. & Colella, P. 1989, Journal of Computational Physics, 82, 64

Bertoldi, F. et al. 2003, A&A, 406, L55

Bianchi, S. & Schneider, R. 2007, MNRAS, 378, 973

Bregman, J. N. 2007, ARA&A, 45, 221

Bromm, V., Coppi, P. S., & Larson, R. B. 2002, ApJ, 564, 23

Bryan, G. L. & Norman, M. L. 1997, in ASP Conf. Ser., Vol. 123, Computational Astrophysics; 12th Kingston Meeting on Theoretical Astrophysics, ed. D. A. Clarke & M. J. West, 363

Caffau, E., Ludwig, H.-G., Steffen, M., Freytag, B., & Bonifacio, P. 2011, SoPh, 268, 255

Cen, R. & Fang, T. 2006, ApJ, 650, 573

Cen, R. & Ostriker, J. P. 1992, ApJL, 399, L113

—. 1999, ApJ, 514, 1

Cen, R., Tripp, T. M., Ostriker, J. P., & Jenkins, E. B. 2001, ApJL, 559, L5

Chen, X., Weinberg, D. H., Katz, N., & Davé, R. 2003, ApJ, 594, 42

Clark, P. C., Glover, S. C. O., & Klessen, R. S. 2008, ApJ, 672, 757

Colella, P. & Woodward, P. R. 1984, Journal of Computational Physics, 54, 174

Collins, D. C., Xu, H., Norman, M. L., Li, H., & Li, S. 2010, ApJS, 186, 308

Danforth, C. W. & Shull, J. M. 2005, ApJ, 624, 555

—. 2008, ApJ, 679, 194

Danforth, C. W., Shull, J. M., Rosenberg, J. L., & Stocke, J. T. 2006, ApJ, 640, 716

Davé, R. et al. 2001, ApJ, 552, 473

Davé, R., Hernquist, L., Katz, N., & Weinberg, D. H. 1999, ApJ, 511, 521

Dere, K. P., Landi, E., Young, P. R., Del Zanna, G., Landini, M., & Mason, H. E. 2009, A&A, 498, 915

Dunne, L., Eales, S., Ivison, R., Morgan, H., & Edmunds, M. 2003, Nature, 424, 285

Dwek, E. 2006, Science, 313, 178

Dwek, E. & Arendt, R. G. 2007, in AIP Conf. Ser., Vol. 937, Supernova 1987A: 20 Years After: Supernovae and Gamma-Ray Bursters, ed. S. Immler, K. Weiler, & R. McCray, 58

Dwek, E. & Cherchneff, I. 2011, ApJ, 727, 63

Dwek, E., Galliano, F., & Jones, A. P. 2007, ApJ, 662, 927

Dwek, E. & Scalo, J. M. 1980, ApJ, 239, 193

Eisenstein, D. J. & Hu, W. 1999, ApJ, 511, 5

Fang, T. et al. 2005, ApJ, 623, 612

Ferland, G. J., Korista, K. T., Verner, D. A., Ferguson, J. W., Kingdon, J. B., & Verner, E. M. 1998, PASP, 110, 761

Fesen, R. A. et al. 2006, ApJ, 645, 283

Fesen, R. A., Zastrow, J. A., Hammell, M. C., Shull, J. M., & Silvia, D. W. 2011, ApJ, 736, 109

Gall, C., Andersen, A. C., & Hjorth, J. 2011, A&A, 528, A14+

Gnat, O. & Ferland, G. J. 2012, ApJS, 199, 20

Gnat, O. & Sternberg, A. 2007, ApJS, 168, 213

Gupta, A., Mathur, S., Krongold, Y., Nicastro, F., & Galeazzi, M. 2012, ApJL, 756, L8

Haardt, F. & Madau, P. 1996, ApJ, 461, 20

Haardt, F. & Madau, P. 2001, in Clusters of Galaxies and the High Redshift Universe Observed in X-rays, ed. D. M. Neumann & J. T. V. Tran

Hammell, M. C. & Fesen, R. A. 2008, ApJS, 179, 195

Hirashita, H., Nozawa, T., Kozasa, T., Ishii, T. T., & Takeuchi, T. T. 2005, MNRAS, 357, 1077

Hughes, D. H. et al. 1998, Nature, 394, 241

Klein, R. I., McKee, C. F., & Colella, P. 1994, ApJ, 420, 213

Komatsu, E. et al. 2011, ApJS, 192, 18

Kotak, R. et al. 2009, ApJ, 704, 306

Kozasa, T. & Hasegawa, H. 1987, Prog. Theor. Phys., 77, 1402

Kozasa, T., Hasegawa, H., & Nomoto, K. 1989, ApJ, 344, 325

—. 1991, A&A, 249, 474

Krause, O., Birkmann, S. M., Rieke, G. H., Lemke, D., Klaas, U., Hines, D. C., & Gordon, K. D. 2004, Nature, 432, 596

Ledoux, C., Bergeron, J., & Petitjean, P. 2002, A&A, 385, 802

Lehnert, M. D. et al. 2010, Nature, 467, 940

Lodders, K., Palme, H., & Gail, H.-P. 2009, Landolt Börnstein, 44

Mac Low, M.-M., McKee, C. F., Klein, R. I., Stone, J. M., & Norman, M. L. 1994, ApJ, 433, 757

Maiolino, R. et al. 2004, Nature, 431, 533

Matsuura, M. et al. 2011, Science, 333, 1258

McCray, R. 1993, ARA&A, 31, 175

McCray, R. 2007, in AIP Conf. Ser., Vol. 937, Supernova 1987A: 20 Years After: Supernovae and Gamma-Ray Bursters, ed. S. Immler, K. Weiler, & R. McCray, 3–14

Meikle, W. P. S. et al. 2007, ApJ, 665, 608

Morgan, H. L. & Edmunds, M. G. 2003, MNRAS, 343, 427

Nakamura, F., McKee, C. F., Klein, R. I., & Fisher, R. T. 2006, ApJS, 164, 477

Nath, B. B., Laskar, T., & Shull, J. M. 2008, ApJ, 682, 1055

Norman, M. L. & Bryan, G. L. 1999, in ASSL, Vol. 240, Numerical Astrophysics: Proceedings of the International Conference on NAP98, ed. S. M. Miyama, K. Tomisaka, & T. Hanawa (Boston, MA: Kluwer Academic), 19

Nozawa, T., Kozasa, T., & Habe, A. 2006, ApJ, 648, 435

Nozawa, T. et al. 2007, ApJ, 666, 955

Nozawa, T., Kozasa, T., Tominaga, N., Maeda, K., Umeda, H., Nomoto, K., & Krause, O. 2010, ApJ, 713, 356

Nozawa, T., Kozasa, T., Umeda, H., Maeda, K., & Nomoto, K. 2003, ApJ, 598, 785

Omukai, K., Tsuribe, T., Schneider, R., & Ferrara, A. 2005, ApJ, 626, 627

Oppenheimer, B. D. & Davé, R. 2009, MNRAS, 395, 1875

Orlando, S., Peres, G., Reale, F., Bocchino, F., Rosner, R., Plewa, T., & Siegel, A. 2005, A&A, 444, 505

O'Shea, B. W., Bryan, G., Bordner, J., Norman, M. L., Abel, T., Harkness, R., & Kritsuk, A. 2005, in Adaptive Mesh Refinement - Theory and Applications (Berlin: Springer)

O'Shea, B. W. & Norman, M. L. 2007, ApJ, 654, 66

—. 2008, ApJ, 673, 14

Patnaude, D. J. & Fesen, R. A. 2005, ApJ, 633, 240

Pettini, M., Smith, L. J., Hunstead, R. W., & King, D. L. 1994, ApJ, 426, 79

Raga, A. C., Esquivel, A., Riera, A., & Velázquez, P. F. 2007, ApJ, 668, 310

Reynolds, D. R., Hayes, J. C., Paschos, P., & Norman, M. L. 2009, Journal of Computational Physics, 228, 6833

Rho, J. et al. 2008, ApJ, 673, 271

Rho, J., Reach, W. T., Tappe, A., Hwang, U., Slavin, J. D., Kozasa, T., & Dunne, L. 2009, ApJ, 700, 579

Sandstrom, K. M., Bolatto, A. D., Stanimirović, S., van Loon, J. T., & Smith, J. D. T. 2009, ApJ, 696, 2138

Santoro, F. & Shull, J. M. 2006, ApJ, 643, 26

Sarazin, C. L. & White, R. E. 1987, ApJ, 320, 32

Schneider, R., Ferrara, A., & Salvaterra, R. 2004, MNRAS, 351, 1379

Shull, J. M. 1977, ApJ, 215, 805

—. 1978, ApJ, 226, 858

Shull, J. M. 2003, in Astrophysics and Space Science Library, Vol. 281, The IGM/Galaxy Connection. The Distribution of Baryons at z=0, ed. J. L. Rosenberg & M. E. Putman

Shull, J. M., Smith, B. D., & Danforth, C. W. 2012, ApJ, 759, 23

Sibthorpe, B. et al. 2009, ArXiv e-prints

—. 2010, ApJ, 719, 1553

Silvia, D. W., Smith, B. D., & Shull, J. M. 2010, ApJ, 715, 1575

—. 2012, ApJ, 748, 12

Skillman, S. W., O'Shea, B. W., Hallman, E. J., Burns, J. O., & Norman, M. L. 2008, ApJ, 689, 1063

Smail, I., Ivison, R. J., & Blain, A. W. 1997, ApJL, 490, L5

Smith, B., Sigurdsson, S., & Abel, T. 2008, MNRAS, 385, 1443

Smith, B. D., Hallman, E. J., Shull, J. M., & O'Shea, B. W. 2011, ApJ, 731, 6

Smith, B. D., Turk, M. J., Sigurdsson, S., O'Shea, B. W., & Norman, M. L. 2009, ApJ, 691, 441

Stanimirović, S. et al. 2005, ApJL, 632, L103

Stocke, J. T. et al. 2013, ApJ, 763, 148

Stone, J. M. & Norman, M. L. 1992a, ApJL, 390, L17

—. 1992b, ApJS, 80, 753

Sugerman, B. E. K. et al. 2006, Science, 313, 196

Sutherland, R. S. & Dopita, M. A. 1993, ApJS, 88, 253

The Enzo Collaboration et al. 2013, ArXiv e-prints

Tielens, A. G. G. M., McKee, C. F., Seab, C. G., & Hollenbach, D. J. 1994, ApJ, 431, 321

Tilton, E. M., Danforth, C. W., Shull, J. M., & Ross, T. L. 2012, ApJ, 759, 112

Todini, P. & Ferrara, A. 2001, MNRAS, 325, 726

Tripp, T. M., Savage, B. D., & Jenkins, E. B. 2000, ApJL, 534, L1

Tripp, T. M., Sembach, K. R., Bowen, D. V., Savage, B. D., Jenkins, E. B., Lehner, N., & Richter, P. 2008, ApJS, 177, 39

Tumlinson, J. et al. 2011, Science, 334, 948

Turk, M. J., Abel, T., & O'Shea, B. 2009, Science, 325, 601

Turk, M. J., Clark, P., Glover, S. C. O., Greif, T. H., Abel, T., Klessen, R., & Bromm, V. 2011a, ApJ, 726, 55

Turk, M. J., Smith, B. D., Oishi, J. S., Skory, S., Skillman, S. W., Abel, T., & Norman, M. L. 2011b, ApJS, 192, 9

Wang, P. & Abel, T. 2009, ApJ, 696, 96

Wang, R. et al. 2008, ApJ, 687, 848

Whittet, D. C. B. 1992, Dust in the galactic environment (Bristol: IOP), 306

Williams, B. J. et al. 2006, ApJL, 652, L33

Wise, J. H. & Abel, T. 2011, MNRAS, 414, 3458

Woodward, P. R. 1976, ApJ, 207, 484