Solving the radiative transfer problem in SPH simulations of contact binary mergers using `FluxCal`

by

Roger Hatfull

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Department of Physics
University of Alberta

Abstract

V1309 Sco was observed in 2008 as a luminous red nova and presents as some of the best time–resolved spectral data of a contact binary merger. However, the parameters of its progenitor and the physics behind the morphology of its light curve remain uncertain. We use the Smoothed Particle Hydrodynamics (SPH) code `StarSmasher` to model the progenitor primary star, using a `MESA` model at the base of the red giant branch with $M = 1.52\ M_{\odot}$, $R = 3.715\ R_{\odot}$ and total particle numbers $N = 1 \times 10^5$, $N = 2 \times 10^5$, and $N = 3 \times 10^5$. We vary spatial resolutions in each of the models to find total energy profiles that match the initial `MESA` model well near the surface.

We analyze all three models using a new code we developed called `FluxCal`, which solves the radiative transfer problem in SPH simulations using an envelope fitting method for inherently optically thick particles and a Runge–Kutta ray tracing method in the LTE approximation. `FluxCal` allows for the first time ever the calculation of spectral properties at SPH simulation boundaries with optically thick particles. We observe an accuracy improvement of ~ 2 orders of magnitude in the effective temperature of an SPH stellar model over a traditional ray tracing method.

We simulate a contact binary merger with the $M = 1.52\ M_{\odot}$, $R = 3.715\ R_{\odot}$,

$N = 2 \times 10^5$ red giant primary and a $M = 0.16\ M_\odot$, $R = 0.203\ R_\odot$, $N = 1 \times 10^4$ secondary with `StarSmasher` out to ~ 14 days (≈ 4 months wall time) to model the V1309 Sco merger. We use `FluxCal` to calculate the effective temperatures in 10 hour intervals which we use to calculate luminosities. The merger in our simulations evolved on timescales shorter than expected from observations. This may be due to the primary's envelope artificially expanding during relaxation to a larger radius than the initial `MESA` model, which leads to Roche lobe overflow and loss of angular momentum at larger separations. Calculated luminosities peak at values similar to that observed in V1309 Sco. We require additional simulation time or a simulation with lower resolution to view late–time spectro–temporal dynamics.

Preface

This thesis is original work by Roger Hatfull in collaboration with professor Natalia Ivanova from the University of Alberta. The framework of the presented code `FluxCal` in §5 originates from the unpublished code `SPLOT`, maintained by professor James C. Lombardi at Allegheny College, and is modified significantly in `FluxCal`. The author of the envelope fitting routine in §5.3.2, including the physical concepts involved, was professor Natalia Ivanova. Dr. Jose Nandez (PhD) at the University of Waterloo helped implement a gravity solver for the creation of `FluxCal` input files from `StarSmasher` output files and, in collaboration with professor Natalia Ivanova, developed software used to reduce data used in the presented analysis of SPH stellar profiles in §6.1.

Acknowledgements

Acknowledgements are made for professor Natalia Ivanova for her remarkable guidance and support. Without her enlightening discussions and feedback this thesis could not have been possible. Professor James C. Lombardi is also acknowledged for his invaluable conversations, insightful commentary, and extraordinary helpfulness.

Thanks are given to Dr. Jose Nandez (PhD) who wrote a gravity solver used to create `FluxCal` input files from `StarSmasher` output files and, in collaboration with Natalia Ivanova, wrote a code that allows for comparison of `StarSmasher` stellar model energy profiles to `MESA` energy profiles. His incredible support with running the `StarSmasher` code on ComputeCanada clusters and valuable conversations are also much appreciated.

Acknowledgements go to the professors, faculty, and students of the astrophysics group at the University of Alberta for their support and educational wisdom. In particular, Kenny X. Van, Zhuo Chen, Asma Hattawi, Yue Zhao, and Pavan Hebbar are thanked for their helpful discussions.

Thanks go to professor Daniel Price for developing the code `SPLASH` (Price, 2011) which was used sporadically to attain swift general understandings of spatiotemporal evolution in `StarSmasher` simulations.

Contents

1 Introduction **1**

1.1 Contact Binary Mergers 4

1.1.1 Shrinking the Orbit 4

1.1.2 Ejecting the Common Envelope 5

1.2 V1309 Sco 6

1.2.1 Observations 6

1.2.2 Recreating the Merger 9

2 Radiative Transfer **11**

2.1 Optical Depth 11

2.2 Radiative Transfer Equation 14

2.2.1 Outgoing Intensity 15

2.3 Local Thermodynamic Equilibrium (LTE) 17

2.3.1 Near LTE 17

2.4 Luminosity 18

2.4.1 Visible Luminosity 19

2.5 Radiation Pressure 21

2.5.1 Optical Depth at the Photosurface 22

2.5.2 Photospheric Pressure 25

2.6 Mean Opacity 26

2.6.1 Planck Mean Opacity 26

2.6.2 Rosseland Mean Opacity 27

3 Smoothed Particle Hydrodynamics 29

3.1 Equations of Motion 29

3.1.1 Momentum Equation 31

3.1.2 Energy Equation 31

3.1.3 Artificial Viscosity 32

3.2 Kernel Functions 34

3.3 Spatial Resolution 36

4 StarSmasher 37

4.1 `StarSmasher` Equations of Motion 37

4.1.1 Artificial Viscosity 38

4.2 Radiative Cooling 39

4.2.1 Implementation 42

4.3 Tabulated Equation of State (TEOS) 43

4.3.1 Polynomial Temperature Solution 45

4.4 Overview of Workflow 46

4.5 Relaxation Process 47

4.5.1 Core Particles 48

4.5.2 Optimal `nnopt` 52

5 `FluxCal` 53

5.1 Photosurface Problem 54

5.2 Particle Optical Depths 57

5.3 Optically Thick (Envelope Fitting) 59

5.3.1 Fundamental Envelope Solutions 60

5.3.2 `FluxCal` Envelope Fitting 62

5.4 Optically Thin (Runge–Kutta Integrator) 63

5.4.1 The Integrating Grid 65

5.4.2 Empty Space 66

5.5 Optical Depth Calculation 68

5.6 Visible Luminosity Calculation . . . 69
5.7 Effective Temperature Calculation . . . 69
5.8 `FluxCal` Framework . . . 71
5.9 Other Tools . . . 74

6 Results **76**
6.1 `StarSmasher` Stellar Models . . . 76
6.1.1 Finding the Optimal `nnopt` . . . 82
6.2 Relaxations . . . 84
6.2.1 Surface Optical Depths . . . 84
6.2.2 Model Radii . . . 86
6.2.3 Effective Temperatures . . . 86
6.3 Dynamical Simulation . . . 88
6.4 Light Curve . . . 90

7 Discussion **103**
7.1 Relaxed Stellar Models . . . 103
7.1.1 Expansion During Relaxation . . . 103
7.1.2 Central Cool Spots . . . 104
7.1.3 Ray Tracing at the Surface . . . 104
7.2 Dynamical Simulation . . . 105
7.2.1 Merger Behavior . . . 105
7.2.2 Ejecta Optical Depths . . . 106
7.2.3 Effective Temperatures and LTE . . . 107
7.3 Light Curve Features . . . 107

8 Conclusions **109**

A Appendix **111**
A.1 TEOS `MESA` Inlist . . . 111
A.2 V1309 Sco Primary Star Inlist . . . 112

List of Tables

5.1 Calculated particle optical depths 59

6.1 `StarSmasher` relaxations 84

List of Figures

1.1 Roche lobe potentials 2
1.2 V1309 Sco I magnitude observations 7
1.3 V1309 Sco luminosities 8

2.1 Gas cloud 12
2.2 Intensity within a gas cloud 13
2.3 Viewing angles 20

3.1 Kernel functions 34

4.1 Tabulated equation of state (TEOS) 44
4.2 Relaxation energies 49
4.3 Core clumping 51

5.1 Hexagonal particle packing 55
5.2 Hexagonal lattice order 56
5.3 Stellar envelope solution 61
5.4 Runge–Kutta integration example 64
5.5 Runge–Kutta integration through empty space 67
5.6 `FluxCal` driving and integrating grids 72

6.1 Initial $N = 3 \times 10^5$ model 77
6.2 Final relaxed $N = 1 \times 10^5$ model 79
6.3 Final relaxed $N = 2 \times 10^5$ model 80

6.4 Final relaxed $N = 3 \times 10^5$ model 81
6.5 Optimal `nnopt` comparison 83
6.6 Fully relaxed $N = 3 \times 10^5$ model outer particle optical depths . 85
6.7 Comparison of effective temperatures calculated with only the envelope fitting routine and only the Runge–Kutta integrator . . 87
6.8 Dynamical run energy evolution 91
6.9 Dynamical run particle positions (part 1) 92
6.10 Dynamical run particle positions (part 2) 93
6.11 Dynamical run outer particle optical depths (part 1) 94
6.12 Dynamical run outer particle optical depths (part 2) 95
6.13 Dynamical run effective temperatures (part 1) 96
6.14 Dynamical run effective temperatures (part 2) 97
6.15 Dynamical run effective temperatures with only the Runge–Kutta integrator (part 1) 98
6.16 Dynamical run effective temperatures with only the Runge–Kutta integrator (part 2) 99
6.17 Location of particles on the TEOS after merger 100
6.18 Light curve 101

Chapter 1

Introduction

Dense clouds of molecular gas in the interstellar medium collapse under self–gravity to form either a single star or fragment to form a collection of stars (McKee and Ostriker, 2007). Although the underlying mechanisms for this fragmentation are unclear, both wide and close binary systems have been observed after a protostellar core collapse, with more than $\sim 50\%$ of Sun–like stars having stellar companions (Raghavan et al., 2010), 10–15% of which have separations of 1–10 AU. More than 80% of stars with $M_1 \gtrsim 16\ M_\odot$ are in a binary (Duchêne and Kraus, 2013). From hereafter, we use the term "primary" to refer to the most massive star and "secondary" to refer to the second–to–most massive star.

In the case of a Sun–like primary with a brown dwarf secondary, the primary eventually evolves off the main sequence and onto the red giant branch resulting in its envelope expanding outward. To understand how the system is affected by such an expansion, consider the two stars as point masses in a corotating frame. The net force acting on any small packet of mass is $F = -m\nabla\varphi$, where φ is the effective gravitational potential is written as,

$$\varphi = -\left(\frac{GM_1}{r_1} + \frac{GM_2}{r_2}\right) - \frac{1}{2}\omega_{\rm orb}^2 r_{cm}^2, \tag{1.1}$$

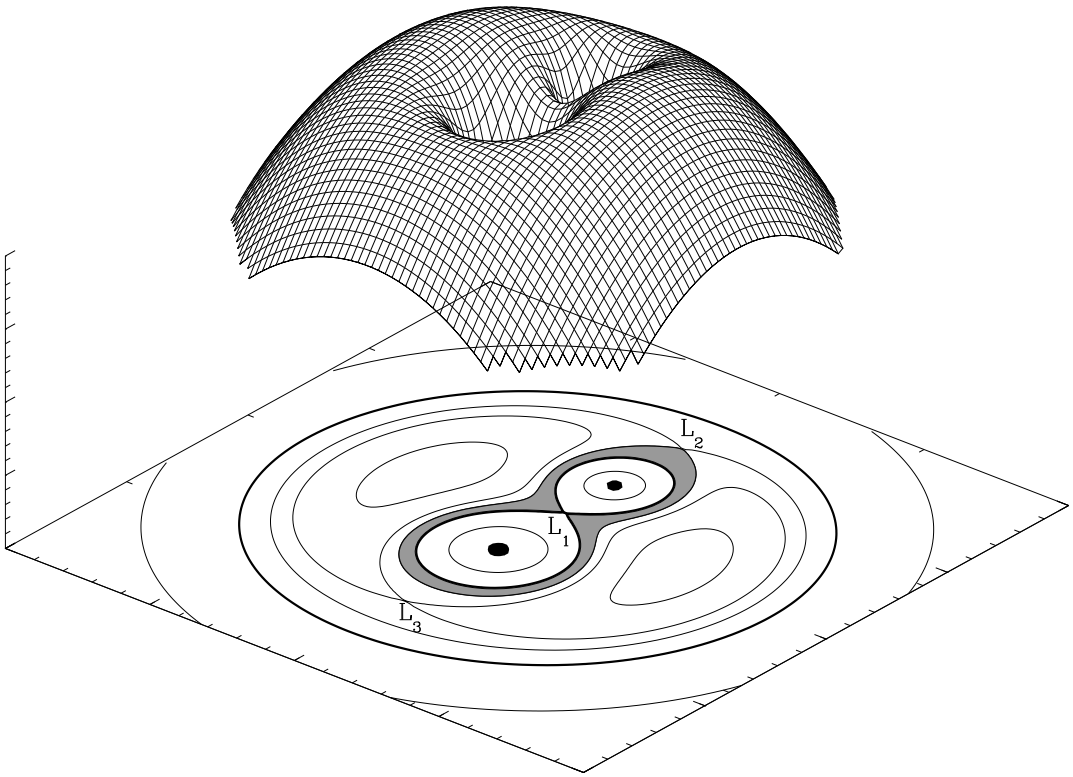

Figure 1.1: The three dimensional effective potential φ (offset mesh) is shown in the vicinity of two point masses (black dots). A cross–section is taken through the equitorial plane and contours of equipotential are traced. The grey shaded region indicates where, for particles with zero velocity, a common envelope can gather. The white region inside the grey shading are the Roche lobes of each component. The point mass with the larger Roche lobe is the primary and the other is the secondary. Image adapted from van der Sluijs (2006).

where G is the gravitational constant, M_1 and M_2 are the masses of the primary and secondary, r_1 and r_2 are the distances to the primary and secondary respectively, $\omega_{\rm orb}$ is the orbital angular velocity, and r_{cm} is the distance from the center of mass. In Fig. 1.1, the locations of local maxima in φ are traced as equipotential contours. The positions where they self–intersect are commonly called the "Lagrange points".

The equipotential that defines the first Lagrange point L_1 also defines the so–called "Roche lobes" of each mass. The 1D volume–equivalent radius of a star at its Roche lobe can be related to its true 3D Roche lobe volume by an approximation derived by Eggleton (1983),

$$r_{L,1} = \frac{0.49(M_1/M_2)^{2/3}a}{0.6(M_1/M_2)^{2/3} + \ln(1 + (M_1/M_2)^{1/3})}, \tag{1.2}$$

where a is the orbital separation, M_1 is the mass of the primary, and M_2 is the mass of the secondary. The secondary's Roche lobe radius is found in the same way by switching all subscripts $1 \rightarrow 2$ and $2 \rightarrow 1$ in Eq. 1.2.

As the primary's envelope expands to larger radii, it may fill its Roche lobe. Once full, mass transfer from the primary to the secondary can occur around the L_1 saddle–point. Prior to the primary's Roche lobe becoming full, the system is called a "detached binary", and "semidetached binary" when one of the Roche lobes is full. Eventually, mass transfer may cause the secondary's Roche lobe to be full as well, forming a "contact binary". Mass at radii larger than the Roche lobe radius is gravitationally bound to the system as a whole, but not to either individual star, forming a common envelope of material, as seen in the gray shaded region in Fig. 1.1. During a common envelope event (CEE hereafter), mass can become gravitationally unbound from the common envelope when the envelope expands past an outer Lagrangian point. Outflow most easily escapes around the L_2 point, as the maximum potential a packet of mass can have within the common envelope is defined by the equipotential at

L_2. It is also possible for outflow to escape the L_3 point and elsewhere if the common envelope expands to sufficient radii.

1.1 Contact Binary Mergers

1.1.1 Shrinking the Orbit

With the two stars in the binary still treated as point masses, the orbital evolution can be understood through analysis of the angular momentum. The total orbital angular momentum of the system is written as,

$$J^2 = G\frac{M_1^2 M_2^2}{M_1 + M_2} a(1 - e^2), \tag{1.3}$$

where G is the gravitational constant, M_1 and M_2 are the masses of the primary and secondary respectively, a is the orbital separation and e is the eccentricity of the orbit. The change in angular momentum over time is calculated by differentiating Eq. 1.3 with respect to time to get,

$$\frac{\dot{J}}{J} = \frac{1}{2}\frac{\dot{a}}{a} + \frac{\dot{M_1}}{M_1} + \frac{\dot{M_2}}{M_2} - \frac{1}{2}\frac{\dot{M_1} + \dot{M_2}}{M_1 + M_2} - \frac{e\dot{e}}{1 - e^2}, \tag{1.4}$$

where dotted values represent the time derivative.

Consider the case of mass transfer from the primary to the secondary ($\dot{M_2} = -\dot{M_1}$) in a circular orbit ($e = 0$) with fully conserved angular momentum ($\dot{J} = 0$). Here, $\dot{a}$ can be found using Eq. 1.4,

$$\frac{\dot{a}}{a} = 2\frac{\dot{M_1}}{M_1}\left(\frac{M_1}{M_2} - 1\right). \tag{1.5}$$

Thus, as the primary loses mass to the secondary, the orbital separation decreases with time for as long as $M_1 > M_2$.

1.1.2 Ejecting the Common Envelope

The ability for a contact binary to eject its common envelope lies in the energy budget of the system. The binding energy of the common envelope was first introduced by Webbink (1984) as proportional to the change in orbital energy,

$$E_{\rm bind} = \alpha_{\rm CE} \left(\frac{GM_{c,1}M_2}{2a_f} - \frac{GM_1M_2}{2a_i} \right), \tag{1.6}$$

where $M_{c,1}$ is the primary's core mass, a_i and a_f are the initial and final orbital separations, and $\alpha_{\rm CE}$ is a dimensionless parameter that describes the efficiency at which energy is transferred from orbital motion to envelope expansion (Livio and Soker, 1988).

The binding energy $E_{\rm bind}$ is equal to negative the total energy of the primary's envelope (Han et al., 1994) and can be calculated as,

$$E_{\rm bind} = \int_{\rm core}^{\rm surface} \left(\frac{Gm}{r} - u \right) dm, \tag{1.7}$$

where m is the mass coordinate, r is the distance from the center of the primary, and u is the specific internal energy. However, in the case of quasi–steady outflow from the common envelope, the binding energy depends on the enthalpy (Ivanova and Chaichenets 2011; Ivanova et al. 2013),

$$E_{\rm bind} = \int_{\rm core}^{\rm surface} \left(\frac{Gm}{r} - \text{ɧ} \right) dm, \tag{1.8}$$

where $\text{ɧ} \equiv u + P/\rho$ is the enthalpy, P is pressure, and ρ is density. A convenient alternative to calculating the binding energy is introduced by de Kool et al. (1987),

$$E_{\rm bind} = \frac{GM_1 \left(M_1 - M_{1,c} \right)}{\lambda r_{L,1}}, \tag{1.9}$$

where λ is a factor of order unity that depends on the density distribution in the primary's stellar envelope (see Wang et al. 2016 for a table of values), $r_{L,1}$

is the radius of the primary's Roche lobe at the onset of spiral–in, as calculated in Eq. 1.2. If the decrease in orbital energy exceeds the binding energy and energy is not lost by other means, the entire common envelope can be ejected.

1.2 V1309 Sco

1.2.1 Observations

The first observations of V1309 Sco were published by Mason et al. (2010) as a classical nova type event. V1309 Sco has a distinctive light curve, as seen in Fig. 1.2, which is thought to be the result of a contact binary merger (Tylenda et al., 2011). Similar observations such as that of V838 Mon, V4332 Sgr, and M31 RV, may also be a result of a merger or collision (Tylenda and Soker, 2006). Objects of this class are most commonly referred to as "Luminous Red Novae" (LRNe), but are also referred to as "intermediate luminosity red transients", "V838 Mon–like events", and "mergebursts", and often present with suddenly increasing luminosities and infrared I and/or visual V magnitudes over days to months, dimming to values lower than pre–outburst over months to years.

V1309 Sco was first identified as a classical nova in 2008 by Nakano et al. (2008) for the American Association of Variable Star Observers (AAVSO hereafter) project[1]. Subsequent observations by Rudy et al. (2008a) discovered narrow Fe II emission lines, suggesting an ejection of an outer mass shell from a white dwarf star. However, observations ~ 40 days later showed no binary companion in the vicinity, making the cause of mass ejection unknown and highly unusual (Rudy et al., 2008b). Tylenda et al. (2011) used archival I band data of V1309 Sco from before the outburst from the third phase of the Optical Gravitational Lensing Experiment (OGLE hereafter) (Udalski, 2003) together

[1]Data available at `https://www.aavso.org/data-download`

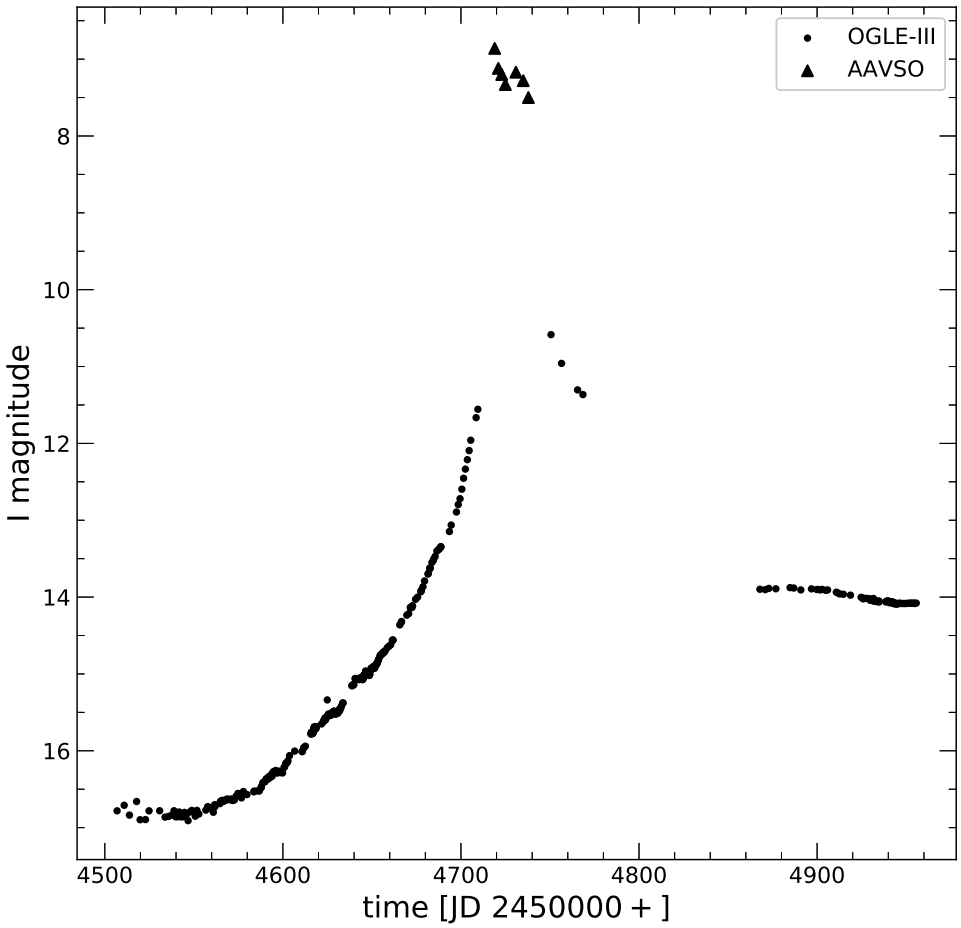

Figure 1.2: Observations of the V1309 Sco outburst in the I filter from the OGLE–III project (Udalski, 2003) are shown as black points and AAVSO data are shown as black triangles. The OGLE data were processed by the standard OGLE procedures (Udalski et al., 2008). The first plateau appears at $\sim$ 2454720 JD, lasts for $\sim$ 20 days, and declines to a secondary plateau at $\sim$ 2454860 JD, which lasts for $\sim$ 100 days. Further observations by Kamiński et al. (2015) show a continued decline to about 17.5 magnitude $\sim$ 1600 days later. Measurement uncertainties are shown, but are smaller than the data points.

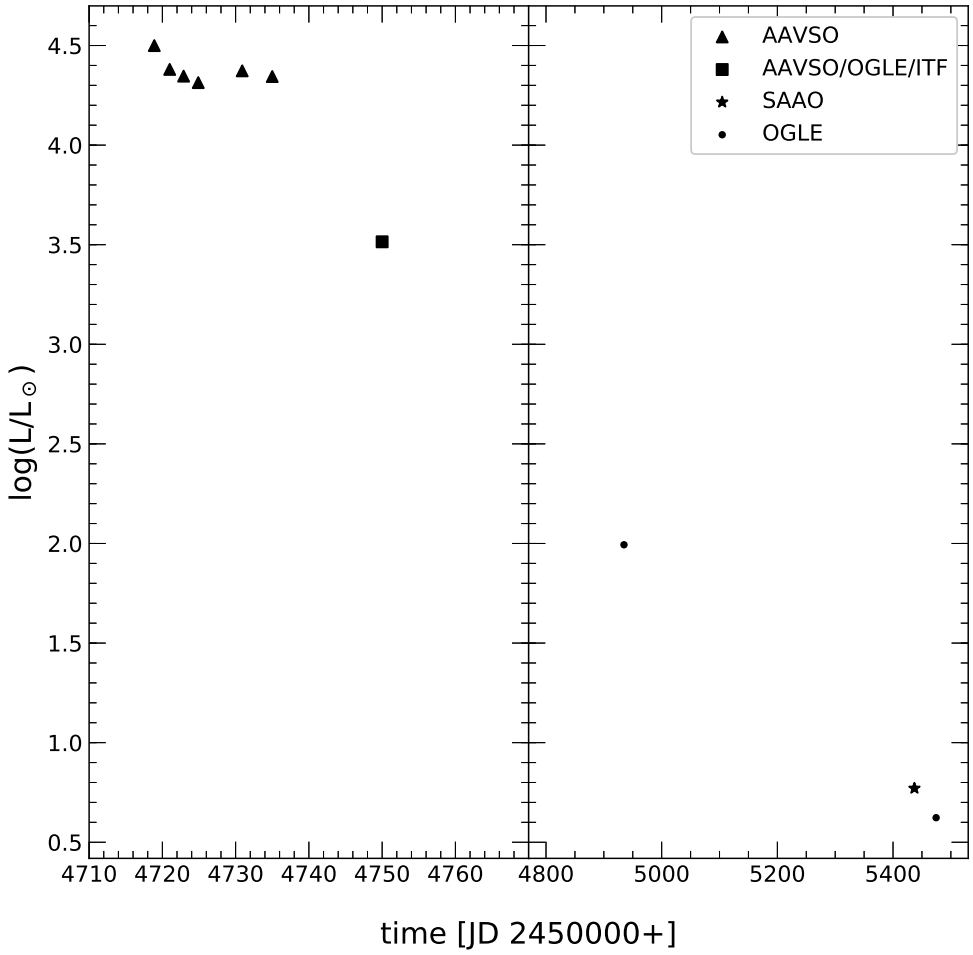

Figure 1.3: Luminosity as calculated by Tylenda et al. (2011) for V1309 Sco using data from AAVSO, OGLE, and SAAO. One data point, shown as a black square, was taken with the NASA Intrared Telescope Facility (shown as ITF, also known as IRTF). Values are calculated using a multiband photometry technique (Tylenda, 2005).

with AAVSO and South African Astronomical Observatory (SAAO) observations, as seen in Fig. 1.2. Prior to outburst, there exists periodicity in the light curve, indicating an eclipsing binary system. The orbital period of the binary was measured as decaying with time and had a period of ~ 1.4 days just prior to outburst (Tylenda et al., 2011). The periodic feature in the light curve is not present post–outburst, indicating a merger.

Tylenda et al. (2011) used multiband photometry (Tylenda, 2005) to calculate the luminosity, as seen in Fig. 1.3, using a distance estimate of 3.0 kpc and reddening $E_{B-V} = 0.8$. Here, photometric magnitudes are compared to a set of reference stellar spectra, with the best fit in terms of least squares is selected (see Tylenda 2005 for more details).

1.2.2 Recreating the Merger

To explain the light curve in Fig. 1.2, the physics involved in the two plateau features must be understood. Prior to outburst, V1309 Sco had an I magnitude of $\sim$ 15. The first plateau occurs at I magnitude $\sim$ 7 at Julian date (JD hereafter) $\sim$ 2454720, and lasts $\sim$ 20 days, while the second plateau occurs at I magnitude $\sim$ 14 at $\sim$ 2454860 JD, and lasts $\sim$ 100 days. The mechanisms driving these features is uncertain, but may be a result of observing a "frozen in space" photosphere in the expanding outflow, where material above the photosphere becomes transparent. Ivanova et al. (2013) discuss the role of recombination in the outflow, as inspired by hydrogen recombination fronts modelled in type IIP supernovae simulations (Kasen and Woosley, 2009). As the recombination front is near the photosphere, recombination energy might be released through radiative flux, possibly affecting the light curve. Ivanova (2018) suggests the recombination energy can not be efficiently transported into convective flux for primary stars with scaled entropies $S/(k_b N_A) < 37$ mol g^{-1} in their envelopes. The role of recombination energy in the features of red novae light curves is still an active debate (Ivanova 2018; Grichener et al. 2018;

Soker et al. 2018).

Chapter 2

Radiative Transfer

Consider an asymmetrical cloud of gas with a source of photons radiating outwards from its center. An observer very far away views the gas cloud along the z axis and can only see the photons that escape the cloud from the surface parallel to the z axis, as seen in Fig. 2.1. As the photons travel through the cloud, they are absorbed and re–emitted by gas particles until they escape. Depending on the chemical composition and number density of the gas, photons may escape easily or not at all. In this chapter, computationally useful approximations are derived for the quantities that define the physics of radiative transfer through the gas cloud – intensity, flux, and luminosity, as well as attenuation, radiation pressure, and photon energy density.

2.1 Optical Depth

The distance a photon with frequency ν travels on average before interacting with a gas particle is called the mean free path,

$$\ell_\nu \equiv \frac{1}{\sigma_\nu n} = \frac{1}{\kappa_\nu \rho}, \tag{2.1}$$

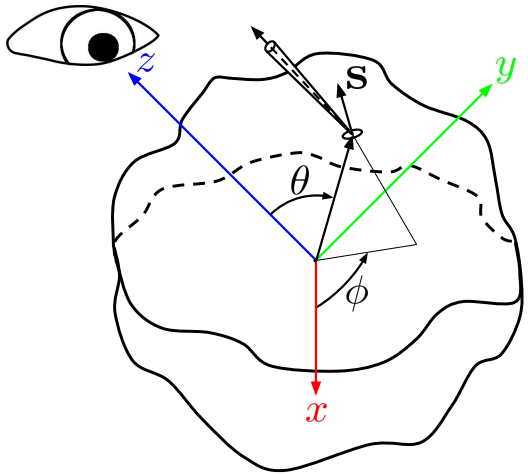

Figure 2.1: A sketch of an asymmetrical gas cloud with the coordinate axis defined by letting z always be toward the observer, shown as an artist rendition of a human eye, regardless of viewing angle. The xy plane is taken to be perpendicular to z with azimuthal angle ϕ measured in xy and θ is the polar angle. A beam of radiation is emitted toward the observer from a small area element in the cloud at (r, θ, ϕ) with area normal $\mathbf{s}$ (see Fig. 2.2).

where σ_ν is the absorption cross–section, n is the particle number density, ρ is the gas mass density, and κ_ν is the mass absorption coefficient, also known as the opacity. A photon is, on average, absorbed by a gas particle when $\ell_\nu \ll |d\varepsilon|$, where $d\varepsilon$ is the distance within the gas cloud along the photon's direction of travel. After being absorbed, a photon may be re–emitted and once again make an attempt to escape. The mean free path depends on the location in the cloud, as σ_ν, n, κ_ν, and ρ are all position dependent. Once the photon enters a region where $\ell_\nu \gg |d\varepsilon|$, it will successfully escape on average. In summary, photons are,

- trapped within the cloud in regions where $|d\varepsilon/\ell_\nu| \gg 1$,
- move mostly unobstructed when $|d\varepsilon/\ell_\nu| \ll 1$.

The dimensionless quantity "optical depth" τ_ν characterizes the ability for photons to escape a material in a specified direction z. It is defined as,

$$d\tau_\nu \equiv -\kappa_\nu \rho dz, \tag{2.2}$$

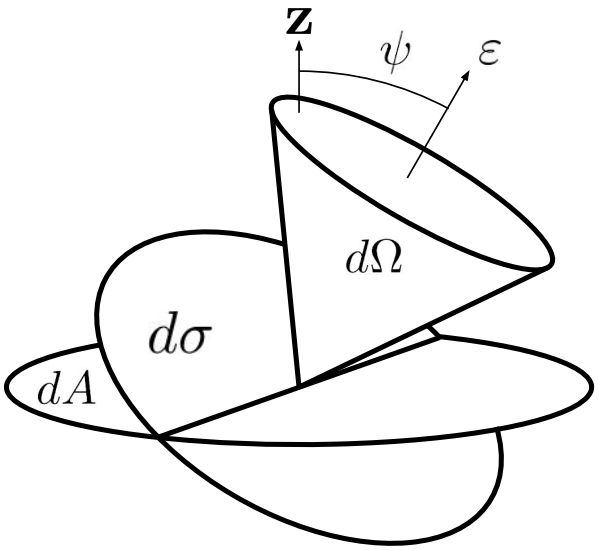

Figure 2.2: A sketch of a beam of intensity inside a gas cloud at some optical depth τ. A single photon traveling in direction ε contributes a small amount of energy to the total intensity beam, defined over the solid angle $d\Omega$. The intensity originates from the small spherical area element dA and the photon travels through an equivalent rotated area element $d\sigma$, angled ψ from the direction of travel.

where both κ_ν and ρ are functions of z. The following terms define the optical properties of the gas cloud,

- "Optically thick" means $\tau_\nu \gg 1$. Photons are unable to escape.
- "Optically thin" means $\tau_\nu \ll 1$. Photons are easily able to escape.
- The surface of the gas cloud, where $\rho = 0$, is where $\tau_\nu = 0$.

The change in optical depth $d\tau_\nu$ can be thought of as the probability of photons to interact with gas particles. At the $\tau_\nu = 0$ surface, there are no gas particles to interact with so the probability is zero. A photon that travels from $\tau_\nu = 0$ to $\tau_\nu \gg 1$ is very likely to interact with a gas particle on its path.

2.2 Radiative Transfer Equation

Consider an area element $dA(\tau_\nu)$ along the line of sight $\mathbf{z}$. Let $\mathbf{z}$ be the surface normal to $dA(\tau_\nu)$ and ε be the direction of travel for a single photon with frequency ν, as seen in Fig. 2.2. The photon passes through area element $d\sigma$, from which $dA(\tau_\nu)$ makes an angle ψ, and contributes to a solid angle $d\Omega$, defined for circular area elements as

$$d\Omega \equiv 2\pi \sin\psi d\psi = -2\pi d\mu, \tag{2.3}$$

where $\mu \equiv \cos(\psi) = dz/d\varepsilon$. The total intensity at optical depth τ_ν in the z direction is written as,

$$I_\nu(\tau_\nu, \mu) \equiv \frac{dE_\nu}{dt}\frac{1}{d\sigma}\frac{1}{d\Omega}, \tag{2.4}$$

where dE_ν is the unit energy per frequency moving through $d\sigma$ in unit time dt.

Photons travel from the source of radiation outwards, diffusing through the gas cloud to the surface. Along a single photon's travel, it interacts with gas particles via absorption and emission processes, contributing to a total intensity at a given location. A single photon makes a small contribution to the intensity $dI_\nu(\tau_\nu, \mu)$ dependent on how that photon is emitted and absorbed,

$$dI_\nu(\tau_\nu, \mu) = dI_\nu^{\mathrm{em}}(\tau_\nu, \mu) + dI_\nu^{\mathrm{abs}}(\tau_\nu, \mu), \tag{2.5}$$

where $dI_\nu^{\mathrm{em}}(\tau_\nu, \mu)$ and $dI_\nu^{\mathrm{abs}}(\tau_\nu, \mu)$ are the small contributions to the emitted and absorbed intensity by a single photon. The absorption component neglects emission effects and the emission component neglects absorption effects.

The absorption component of a single photon along its direction of motion ε is proportional to the total intensity in that direction, where the proportionality coefficient is equal to the probability that it will interact with a gas particle. This is a similar concept to the optical depth in Eq. 2.2 being equivalent to the probability of interaction with a gas particle. The absorption component is

written as,

$$dI_\nu^{\text{abs}}(\tau_\nu, \mu) = -I_\nu(\tau_\nu, \mu)\kappa_\nu \rho d\varepsilon. \tag{2.6}$$

The emission component $dI_\nu^{\text{em}}(\tau_\nu, \mu)$ of a single photon is described by the "source function" $S_\nu(\tau_\nu, \mu)$, which encapsulates the physics of emission within it and has the same dimensions as intensity. As with the absorption component, the emission component is proportional to the source function, where the proportionality constant is again the probability of interacting with a gas particle along $-\varepsilon$,

$$dI_\nu^{\text{em}}(\tau_\nu, \mu) = S_\nu(\tau_\nu, \mu)\kappa_\nu \rho d\varepsilon. \tag{2.7}$$

Hence, Eq. 2.5 can be written using Eq. 2.6 and Eq. 2.7 as,

$$\frac{dI_\nu(\tau_\nu, \mu)}{\kappa_\nu \rho d\varepsilon} = S_\nu(\tau_\nu, \mu) - I_\nu(\tau_\nu, \mu). \tag{2.8}$$

Recall that $d\varepsilon = dz/\mu$, so Eq. 2.8 can be written using the optical depth from Eq. 2.2 as,

$$\mu \frac{dI_\nu(\tau_\nu, \mu)}{d\tau_\nu} = I_\nu(\tau_\nu, \mu) - S_\nu(\tau_\nu, \mu). \tag{2.9}$$

This first–order linear ordinary differential equation can be solved using an integration factor to obtain,

$$I_\nu(\tau_\nu, \mu) = I_\nu(\tau_{\nu,0}, \mu)e^{(\tau_\nu - \tau_{\nu,0})/\mu} + \frac{1}{\mu}\int_{\tau_\nu}^{\tau_{\nu,0}} S_\nu(\tau'_\nu, \mu)e^{(\tau_\nu - \tau'_\nu)/\mu} d\tau'_\nu, \tag{2.10}$$

where $\tau_{\nu,0}$ is the optical depth at which the intensity is considered to originate.

2.2.1 Outgoing Intensity

Consider the case where the intensity originates from an optical depth much larger than the final optical depth, $\tau_{\nu,0} \gg \tau_\nu$. The outgoing intensity is written

using Eq. 2.10 as,

$$I_\nu^+(\tau_\nu, \mu \geq 0) = \frac{1}{\mu} \int_{\tau_\nu}^{\tau_{\nu,0}} S_\nu(\tau'_\nu, \mu \geq 0) e^{(\tau_\nu - \tau'_\nu)/\mu} d\tau'_\nu. \tag{2.11}$$

At the $\tau_\nu = 0$ surface, Eq. 2.11 is written as,

$$I_\nu^+(\tau_\nu = 0, \mu \geq 0) = \frac{1}{\mu} \int_{0}^{\tau_{\nu,0}} S_\nu(\tau'_\nu, \mu \geq 0) e^{-\tau'_\nu/\mu} d\tau'_\nu. \tag{2.12}$$

Consider the Taylor expansion of the source function around $\tau_\nu = 0$,

$$S_\nu(\tau_\nu, \mu \geq 0) \approx S_\nu(\tau_\nu = 0, \mu \geq 0) + \tau_\nu \left(\frac{dS_\nu(\tau_\nu, \mu \geq 0)}{d\tau_\nu} \right)_{\tau_\nu = 0}. \tag{2.13}$$

The outgoing intensity in Eq. 2.12 is hence written as,

$$\begin{aligned} I_\nu^+(\tau_\nu = 0, \mu \geq 0) \approx & S_\nu(\tau_\nu = 0, \mu \geq 0) \left(1 - e^{-\tau_{\nu,0}/\mu}\right) \\ & + \mu \left(\frac{dS_\nu(\tau_\nu, \mu \geq 0)}{d\tau_\nu} \right)_{\tau_\nu = 0} \left[1 - \left(1 + \frac{\tau_{\nu,0}}{\mu} \right) e^{-\tau_{\nu,0}/\mu} \right]. \end{aligned} \tag{2.14}$$

As stated previously, $\tau_{\nu,0} \gg \tau_\nu$, and hence, $\tau_{\nu,0} \gg 0$. Therefore, Eq. 2.14 is written as,

$$I_\nu^+(\tau_\nu = 0, \mu \geq 0) \approx S_\nu(\tau_\nu = 0, \mu \geq 0) + \mu \left(\frac{dS_\nu(\tau_\nu, \mu \geq 0)}{d\tau_\nu} \right)_{\tau_\nu = 0}, \tag{2.15}$$

which is equal to the Taylor expansion of the source function in Eq. 2.13 for $\tau_\nu = \mu$. Thus, the intensity at the $\tau_\nu = 0$ surface, originating at an optically thick region, equals the source function at $\tau_\nu = \mu$,

$$I_\nu^+(\tau_\nu = 0, \mu \geq 0) \approx S_\nu(\tau_\nu = \mu, \mu \geq 0). \tag{2.16}$$

2.3 Local Thermodynamic Equilibrium (LTE)

We adopt that a small region within the gas cloud is in local thermodynamic equilibrium (LTE hereafter) if a single value of temperature can be attributed to all gas particles within that neighborhood, and so radiation produced by those particles is a function of that temperature. In this work, for simplicity, we assume without further proof that at large optical depths LTE is nearly satisfied.

In the case of strict LTE, the source function $S_\nu(\tau_\nu \gg 1, \mu)$ is well described by the Planck blackbody function $B_\nu(T(\tau_\nu \gg 1))$, where $T(\tau_\nu \gg 1)$ is the local temperature in the optically thick region. In LTE, the incoming and outgoing intensities are the same, and hence the change in intensity $dI(\tau_\nu \gg 1, \mu) = 0$. Therefore, the radiative transfer equation from Eq. 2.9 indicates that

$$I_\nu(\tau_\nu \gg 1, \mu) = S_\nu(\tau_\nu \gg 1, \mu) = B_\nu(T(\tau_\nu \gg 1)). \tag{2.17}$$

This implies that the total intensity has no angular dependency. For strict LTE, it can be shown that there is zero net radiation flux as I is angle independent, so in principle strict LTE can not be considered.

2.3.1 Near LTE

If deviation from LTE in §2.3 is small, we can consider the source function as a Taylor expansion around τ_ν, similar to the process in §2.2.1,

$$S_\nu(\tau'_\nu \gg 1, \mu) \approx B_\nu(T(\tau_\nu)) + (\tau'_\nu - \tau_\nu) \left(\frac{dB_\nu(T(\tau_\nu))}{d\tau_\nu} \right)_{\tau_\nu}. \tag{2.18}$$

For large τ, and $\tau_{\nu,0} \gg \tau_\nu$, the outgoing radiation from Eq. 2.11 can be written as,

$$I^+_\nu(\tau_\nu \gg 1, \mu \geq 0) = B_\nu(T(\tau_\nu)) + \mu \left(\frac{dB_\nu(T(\tau_\nu))}{d\tau_\nu} \right)_{\tau_\nu}. \tag{2.19}$$

Similarly, using Eq. 2.18 in Eq. 2.10 gives the incoming intensity as,

$$I_\nu^-(\tau_\nu \gg 1, \mu < 0) = B_\nu(T(\tau_\nu)) + \mu \left(\frac{dB_\nu(T(\tau_\nu))}{d\tau_\nu} \right)_{\tau_\nu}. \tag{2.20}$$

Thus, a general form for the total intensity in optically thick regions is,

$$I_\nu(\tau_\nu \gg 1, \mu) = B_\nu(T(\tau_\nu)) + \mu \left(\frac{dB_\nu(T(\tau_\nu))}{d\tau_\nu} \right)_{\tau_\nu}. \tag{2.21}$$

2.4 Luminosity

The luminosity is defined as the total energy lost by radiation over time in all directions,

$$L \equiv \frac{dE_{\rm tot}}{dt} = \int \frac{dE}{dt}, \tag{2.22}$$

where $dE_{\rm tot}$ is the unit total energy radiated away and dE is the unit energy radiated away through the small area element in §2.2. By integrating the intensity from Eq. 2.4 over all frequencies, Eq. 2.22 can be written as,

$$L = 2\pi \oint_A \int_0^1 I^+(\tau = 0, \mu \geq 0) \mu d\mu dA(\tau = 0), \tag{2.23}$$

where A is the surface area at $\tau = 0$.

We adopt that outgoing radiation from the surface is isotropic at each point on the surface. For isotropic outgoing radiation, the intensity does not depend on μ and can hence be written using Eq. 2.23 as,

$$L = \pi \oint_A I^+(\tau = 0, \mu \geq 0) dA(\tau = 0). \tag{2.24}$$

Let the "effective temperature" $T_{\rm eff}(\mathbf{r})$ be the temperature of a blackbody that would produce the same amount of radiation through the small area element $dA(\tau = 0)$ as the gas cloud. A blackbody satisfies LTE, as described in §2.3, and thus the frequency–integrated outgoing intensity from the gas cloud at its

surface can be written as,

$$I^+(\tau = 0, \mu \geq 0) = \frac{\sigma_{\rm SB}}{\pi} T_{\rm eff}(\mathbf{r})^4. \tag{2.25}$$

Note that this defines $T_{\rm eff}(\mathbf{r})$ as a function of $I^+(\tau = 0, \mu \geq 0)$ and $T_{\rm eff}$ has been written with explicit position $\mathbf{r}$ dependence, as it is dependent on the line of sight.

Let the "photosurface" be an imaginary area which satisfies the above blackbody condition, written as $A_{\rm ph}$. The luminosity at the true surface ($\tau = 0$) of the cloud is calculated as,

$$L = \sigma_{\rm SB} \oint_{A_{\rm ph}} T_{\rm eff}(\mathbf{r})^4 dA(\tau_{\rm ph}), \tag{2.26}$$

where $\tau_{\rm ph}$ is the optical depth at the photosurface.

If the photosurface is spherically symmetric it is called a "photosphere". When $T_{\rm eff}(\mathbf{r})$ is constant over all $\mathbf{r}$, the luminosity in Eq. 2.26 can be written as,

$$L = 4\pi\sigma_{\rm SB} R_{\rm ph}^2 T_{\rm eff}^4, \tag{2.27}$$

where $R_{\rm ph}$ is the radius of the photosphere.

2.4.1 Visible Luminosity

Outside of the gas cloud, assuming that photons do not interact with any other material, the luminosity is a universal quantity. However, a stationary distant observer can only measure the energy escaping the cloud in their direction. Thus, if and only if the photosurface is spherically symmetric and radiation is isotropic can the luminosity be calculated by Eq. 2.27. Instead, consider the "visible luminosity" $\mathscr{L}(\xi, \zeta)$, which is the fraction of the total luminosity a far away observer looking on with polar and azimuthal viewing angles (ξ, ζ) (see Fig. 2.3) can detect, based on the area of the photosurface visible to them. The

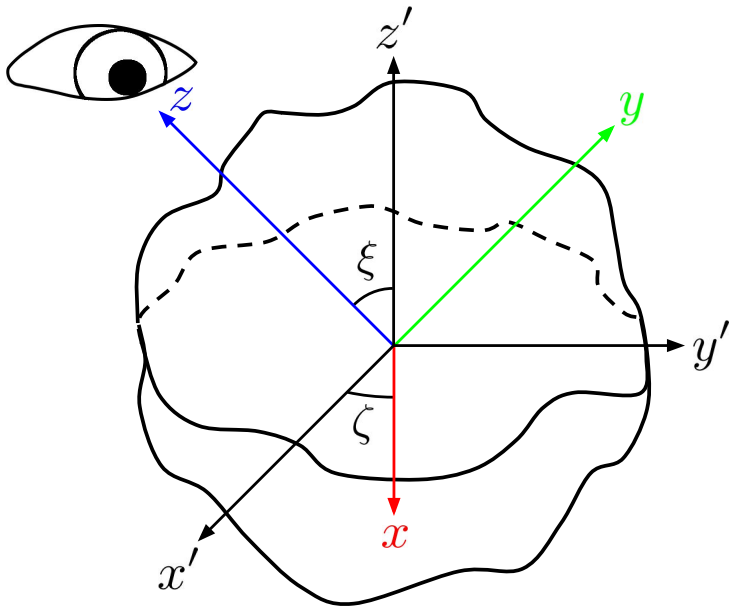

Figure 2.3: The same sketch of the gas cloud from Fig. 2.1, but showing the local coordinates (x', y', z') that describe the orientation of the gas cloud. The cloud's orientation is independent of the viewing angle (ξ, ζ) where ξ is the polar angle and ζ is the azimuthal angle.

visible luminosity is calculated using Eq. 2.23 as,

$$\mathscr{L}(\xi,\zeta) = 2\pi \oint_{Q(\xi,\zeta)} \int_0^1 I^+(\tau = 0, \mu \geq 0)\mu d\mu dq(\tau = 0), \tag{2.28}$$

where $Q(\xi, \zeta)$ is the projected $\tau = 0$ surface area A and can be thought of as the "flattened" 3D $\tau = 0$ surface area.

If radiation is isotropic at the $\tau = 0$ surface, the visible luminosity can be written as,

$$\mathscr{L}(\xi,\zeta) = \pi \oint_{Q(\xi,\zeta)} I^+(\tau = 0, \mu \geq 0)dq(\tau = 0). \tag{2.29}$$

As done in §2.4, let $T_{\rm eff}(\mathbf{r})$ be the temperature of a blackbody that would produce the same amount of visible radiation as the gas cloud through the small projected area element $dq(\tau = 0)$. Here, Eq. 2.29 gives the visible luminosity as,

$$\mathscr{L}(\xi,\zeta) = \sigma_{\rm SB} \oint_{Q_{\rm ph}(\xi,\zeta)} T_{\rm eff}(\mathbf{r})^4 dq(\tau_{\rm ph}), \tag{2.30}$$

where $\tau_{\rm ph}$ is the location at the photosurface and $Q_{\rm ph}(\xi,\zeta)$ is the projected surface area of the photosurface. As done in §2.4, consider the case of a photosphere with constant $T_{\rm eff}$ across its surface. Here, $Q_{\rm ph}(\xi,\zeta)=\pi R_{\rm ph}^2$ and the visible luminosity is hence given by Eq. 2.30 as,

$$\mathscr{L}(\xi,\zeta)=\pi\sigma_{\rm SB}R_{\rm ph}^2T_{\rm eff}^4. \tag{2.31}$$

This differs by the total luminosity in Eq. 2.27 by a factor of 4, which is consistent as the only physical difference between Eq. 2.31 and Eq. 2.27 are the projected and 3D photospheric areas.

2.5 Radiation Pressure

Photons with frequencies ν traveling through $dA(\tau_\nu)$ in direction μ carry momenta $dE_\nu\mu/c$, where c is the speed of light, and thus exert a pressure on $dA(\tau_\nu)$. Pressure is defined as a force per area, which is equivalent to momentum per area per time. Thus, each photon makes a small contribution to the radiative pressure at τ_ν, written as,

$$dP_{{\rm rad},\nu}(\tau_\nu)\equiv\frac{1}{c}\frac{dE_\nu}{dt}\frac{\mu}{dA(\tau_\nu)}, \tag{2.32}$$

where $A(\tau_\nu)$ is the surface area over which the force from a single photon is applied. Using the definition of the intensity per unit frequency in Eq. 2.4, integrating Eq. 2.32 over all angles gives the radiation pressure due to all the photons applying a force on $dA(\tau_\nu)$ as,

$$P_{{\rm rad},\nu}(\tau_\nu)=\frac{2\pi}{c}\int_{-1}^{1}I_\nu(\tau_\nu,\mu)\mu^2d\mu, \tag{2.33}$$

where $I_\nu(\tau_\nu,\mu)$ is the total intensity at the location of τ_ν.

Another way to write the radiation pressure comes from considering the

equation of radiative transfer from Eq. 2.9. In what follows, we write the left hand side of Eq. 2.9 in the same form as the right hand side of Eq. 2.33,

$$\frac{d}{d\tau_\nu}\left[\frac{1}{c}I_\nu(\tau_\nu,\mu)\mu^2\right] = \frac{1}{c}I_\nu(\tau_\nu,\mu)\mu - \frac{1}{c}S_\nu(\tau_\nu,\mu)\mu. \tag{2.34}$$

We integrate both parts over all angles,

$$\begin{aligned}\frac{d}{d\tau_\nu}\left[\frac{2\pi}{c}\int_{-1}^{1} I_\nu(\tau_\nu,\mu)\mu^2 d\mu\right] &= \frac{2\pi}{c}\int_{-1}^{1} I_\nu(\tau_\nu,\mu)\mu d\mu \\ &\quad - \frac{2\pi}{c}\int_{-1}^{1} S_\nu(\tau_\nu,\mu)\mu d\mu.\end{aligned} \tag{2.35}$$

Using Eq. 2.33, Eq. 2.35 can be written as,

$$\frac{dP_{\mathrm{rad},\nu}(\tau_\nu)}{d\tau_\nu} = \frac{2\pi}{c}\int_{-1}^{1} I_\nu(\tau_\nu,\mu)\mu d\mu - \frac{2\pi}{c}\int_{-1}^{1} S_\nu(\tau_\nu,\mu)\mu d\mu. \tag{2.36}$$

If the source function $S_\nu(\tau_\nu,\mu)$ is isotropic, then Eq. 2.36 can be written as,

$$\frac{dP_{\mathrm{rad},\nu}(\tau_\nu)}{d\tau_\nu} = \frac{2\pi}{c}\int_{-1}^{1} I_\nu(\tau_\nu,\mu)\mu d\mu \tag{2.37}$$

2.5.1 Optical Depth at the Photosurface

Consider an imaginary blackbody object (see §2.3) which radiates the same amount of energy over time per area as that radiated away from the photosurface of the gas cloud. Let both the blackbody and gas cloud radiate isotropically such that for each case, Eq. 2.24 can be written as,

$$\pi I^+_{\mathrm{cl}}(\tau_{\mathrm{cl}} = 0, \mu \geq 0) A_{\mathrm{cl}} = L_{\mathrm{cl}}, \tag{2.38}$$

$$\sigma_{\mathrm{SB}} T_{\mathrm{eff}}(\mathbf{r})^4 A_{\mathrm{bb}} = L_{\mathrm{bb}}, \tag{2.39}$$

where subscripts "cl" and "bb" refer to the cloud and blackbody respectively. Equating these then yields the outgoing intensity from the surface of the gas

cloud as,

$$I_{\rm cl}^{+}(\tau_{\rm cl}=0,\mu\geq 0)=\frac{A_{\rm bb}}{A_{\rm cl}}\frac{\sigma_{\rm SB}}{\pi}T_{\rm eff}(\mathbf{r})^4. \tag{2.40}$$

We now introduce the radiation flux as,

$$dF_\nu(\tau_\nu)=\frac{dE_\nu}{dt}\frac{1}{dA(\tau_\nu)}. \tag{2.41}$$

Using the definition of intensity from Eq. 2.4, Eq. 2.41 integrated over all frequencies can be written as,

$$F(\tau)=2\pi\int_{-1}^{1}I(\tau,\mu)\mu d\mu. \tag{2.42}$$

Hence, Eq. 2.37 can be written as,

$$\frac{dP_{\rm rad}(\tau)}{d\tau}=\frac{1}{c}F(\tau). \tag{2.43}$$

Integrating both sides over the surface of the gas cloud yields,

$$\oint_{A_{\rm cl}}\frac{dP_{\rm rad,cl}(\tau_{\rm cl}=0)}{d\tau_{\rm cl}}dA(\tau_{\rm cl}=0)=\frac{1}{c}\oint_{A_{\rm cl}}F_{\rm cl}(\tau_{\rm cl}=0)dA(\tau_{\rm cl}=0), \tag{2.44}$$

where $A_{\rm cl}$ is the $\tau_{\rm cl}=0$ surface area of the gas cloud and $F_{\rm cl}(\tau_{\rm cl}=0)$ is the radiative flux from the surface of the gas cloud. Note the quantity on the right hand side is the same as the luminosity of the cloud as written in Eq. 2.23, as $F_{\rm cl}(\tau_{\rm cl}=0)=F_{\rm cl}^{+}(\tau_{\rm cl}=0)$. Thus, Eq. 2.44 can be written as,

$$\oint_{A_{\rm cl}}\frac{dP_{\rm rad,cl}(\tau_{\rm cl}=0)}{d\tau_{\rm cl}}dA(\tau_{\rm cl}=0)=\frac{1}{c}L_{\rm cl}. \tag{2.45}$$

Let us consider the simple case where $dP_{\rm rad,cl}(\tau_{\rm cl})/d\tau_{\rm cl}$ is constant across $A_{\rm cl}$. Recall that $L_{\rm cl}=L_{\rm bb}$, so Eq. 2.45 can be written using Eq. 2.39 as,

$$\frac{dP_{\rm rad,cl}(\tau_{\rm cl})}{d\tau_{\rm cl}}=\frac{1}{c}\frac{A_{\rm bb}}{A_{\rm cl}}\sigma_{\rm SB}T_{\rm eff}(\mathbf{r})^4. \tag{2.46}$$

In the assumption that $A_{\rm bb} \rightarrow A_{\rm cl}$, separating the variables in Eq. 2.46 and integrating both sides from the photosurface $\tau_{\rm cl} = \tau_{\rm ph}$ to the true surface $\tau_{\rm cl} = 0$ for constant $T_{\rm eff}(\mathbf{r})$ with $\mathbf{r}$ yields,

$$P_{\rm rad,cl}(\tau_{\rm ph}) = \frac{1}{c}\sigma_{\rm SB}T_{\rm eff}^4\tau_{\rm ph} + P_{\rm rad,cl}(\tau_{\rm cl} = 0) \tag{2.47}$$

The pressure at $\tau_{\rm cl} = 0$ in the cloud can be found using Eq. 2.40 in Eq. 2.33 with the assumption that $A_{\rm bb} \rightarrow A_{\rm cl}$ and is written as,

$$P_{\rm rad,cl}(\tau_{\rm cl} = 0) = \frac{2}{3}\frac{\sigma_{\rm SB}}{c}T_{\rm eff}^4. \tag{2.48}$$

From Eq. 2.33, assuming isotropic radiation, the radiative pressure at the photosurface of the cloud can be written as,

$$P_{\rm rad,cl}(\tau_{\rm ph}) = \frac{2\pi}{c}\int_{-1}^{1} I_{\rm cl}(\tau_{\rm ph})\mu^2 d\mu = \frac{4\pi}{3c}I_{\rm cl}(\tau_{\rm ph}) \tag{2.49}$$

The outgoing flux at some location inside the cloud can be written as,

$$F_{\rm cl}^{+}(\tau_{\rm cl}) = 2\pi\int_{0}^{1} I_{\rm cl}(\tau_{\rm cl},\mu)\mu d\mu. \tag{2.50}$$

Thus, by using Eq. 2.50 with isotropic total radiation, the radiative pressure at the photosurface in the cloud in Eq. 2.49 can be written as,

$$P_{\rm rad,cl}(\tau_{\rm ph}) = \frac{4}{3c}F_{\rm cl}^{+}(\tau_{\rm ph}). \tag{2.51}$$

Therefore, Eq. 2.47 can be written using Eq. 2.48 and Eq. 2.51 as,

$$F_{\rm cl}^{+}(\tau_{\rm ph}) = \frac{1}{2}\sigma_{\rm SB}T_{\rm eff}^4\left(\frac{3}{2}\tau_{\rm ph} + 1\right). \tag{2.52}$$

As assumed earlier, $A_{\rm bb} \rightarrow A_{\rm cl}$ and $L_{\rm cl} = L_{\rm bb}$. Therefore, the outgoing flux at the surface of the cloud $F_{\rm cl}^{+}(\tau_{\rm ph}) \rightarrow F_{\rm bb}^{+}(\tau_{\rm cl} = 0)$. For a blackbody radiating

isotropically from its surface, $F^{+}_{\rm bb}(\tau_{\rm cl}=0)=\sigma_{\rm SB}T^{4}_{\rm eff}$ and, thus, from Eq. 2.52, $\tau_{\rm ph}=2/3$.

2.5.2 Photospheric Pressure

Assuming a geometrically isotropic gas cloud, the usual equation of hydrostatic equilibrium $dP/dr=-g\rho$, where g is the local gravitational acceleration, can be combined with the definition of optical depth from Eq. 2.2 along the direction $dz=dr$,

$$\frac{dP(\tau)}{d\tau_{\nu}}=\frac{g}{\kappa_{\nu}}. \tag{2.53}$$

Integrating over all ν, Eq. 2.53 can be written as,

$$\frac{dP(\tau)}{d\tau}=\frac{g}{\bar{\kappa}}, \tag{2.54}$$

where $\bar{\kappa}$ is the frequency mean opacity (see §2.6). Then, taking g and $\bar{\kappa}$ as constants with distance, separating the variables and integrating from the photosphere ($\tau=\tau_{\rm ph}$) to the surface ($\tau=0$) yields,

$$P(\tau_{\rm ph})=\tau_{\rm ph}\frac{g_s}{\bar{\kappa}_s}+P(\tau=0), \tag{2.55}$$

where $g_s=GM/R^2$ is the gravitational acceleration at the surface and $\bar{\kappa}_s$ is the mean opacity at the surface. The pressure at $\tau=0$ can be calculated using Eq. 2.48 and allowing the outgoing intensity to be equal to that of a blackbody. Hence, Eq. 2.55 can be written as,

$$P(\tau_{\rm ph})=\tau_{\rm ph}\frac{g_s}{\bar{\kappa}_s}+\frac{2}{3c}\sigma_{\rm SB}T^{4}_{\rm eff}. \tag{2.56}$$

The equation for the luminosity due to a photosphere in a gas cloud that is radiating isotropically at its surface is given by Eq. 2.27 and, hence, the pressure

at the photosphere in Eq. 2.56 can be written using $\tau_{\rm ph} = 2/3$ as,

$$P(\tau_{\rm ph}) = \frac{2}{3}\frac{g_s}{\bar{\kappa}_s}\left(1 + \frac{L\bar{\kappa}_s}{4\pi cGM}\right). \tag{2.57}$$

For isotropic gas clouds with low luminosities and/or low mean opacities, the above equation is written as,

$$P(\tau_{\rm ph}) \approx \frac{2}{3}\frac{g_s}{\bar{\kappa}_s}, \tag{2.58}$$

where $\tau_{\rm ph} = 2/3$ has been used from §2.5.1. The relation in Eq. 2.58 is utilized later in §5.3.2 to find the pressure at the photosphere of stellar models.

2.6 Mean Opacity

The mean opacity $\bar{\kappa}$ is found by integrating the frequency–dependent opacity κ_ν over all frequencies subject to a weighting function,

$$\bar{\kappa} = \frac{\int_0^\infty \kappa_\nu Z_\nu d\nu}{\int_0^\infty Z_\nu d\nu}, \tag{2.59}$$

where Z_ν is the weighting function.

2.6.1 Planck Mean Opacity

The unit absorbed intensity across all frequencies is calculated by integrating Eq. 2.6 such that,

$$d\left(\int_0^\infty I_\nu^-(\tau_\nu, \mu < 0)d\nu\right) = -\rho d\varepsilon \int_0^\infty \kappa_\nu I_\nu(\tau_\nu, \mu)d\nu. \tag{2.60}$$

Low temperature gases ($T \lesssim 2000$ K) can form clumps of various molecules called dust grains (see Barvainis 1987 for further details). When radiatively heated dust grains collide with gas particles, they transfer enough of their

thermal energy into the gas such that the gas temperature remains constant (Goldreich and Kwan, 1974). Hence, the gas satisfies the strict LTE condition in §2.3 and by Eq. 2.17, the total intensity is equal to the Planck blackbody function. Thus, Eq. 2.60 becomes,

$$d\left(\int_0^\infty I_\nu^-(\tau_\nu, \mu < 0)d\nu\right) = -\rho d\varepsilon \int_0^\infty \kappa_\nu B_\nu(T(\tau_\nu))d\nu. \tag{2.61}$$

By letting the opacity κ_ν equal the Planck mean opacity $\bar{\kappa}_P$, Eq. 2.61 becomes,

$$d\left(\int_0^\infty I_\nu^-(\tau_\nu, \mu < 0)d\nu\right) = -\rho\bar{\kappa}_P d\varepsilon \int_0^\infty B_\nu(T(\tau_\nu))d\nu. \tag{2.62}$$

Hence, the Planck mean opacity is defined by using the Planck blackbody function as the weighting function Z_ν in Eq. 2.59,

$$\bar{\kappa}_P \equiv \frac{\int_0^\infty \kappa_\nu B_\nu(T(\tau_\nu))d\nu}{\int_0^\infty B_\nu(T(\tau_\nu))d\nu} \tag{2.63}$$

This opacity is used for optically thin environments.

2.6.2 Rosseland Mean Opacity

The Rosseland mean opacity follows from the Rosseland flux approximation for high optical depths (Rosseland, 1924). The energy flux at frequency ν at optical depth τ_ν along some line of sight is defined as,

$$F_\nu(\tau_\nu) \equiv \int \frac{dE_\nu}{dt}\frac{1}{dA(\tau_\nu)}. \tag{2.64}$$

Using the definition of intensity from Eq. 2.4, the general form for total intensity in near LTE (see §2.3.1) in Eq. 2.21, and integrating over all angles, the flux in Eq. 2.64 can be written as,

$$F_\nu(\tau_\nu \gg 1) = \frac{4\pi}{3}\frac{dT(\tau_\nu)}{d\tau_\nu}\frac{dB_\nu(T(\tau_\nu))}{dT(\tau_\nu)}. \tag{2.65}$$

By the definition of optical depth in Eq. 2.2, Eq. 2.65 can be written as,

$$F_\nu(\tau_\nu \gg 1) = -\frac{4\pi}{3}\frac{1}{\kappa_\nu \rho}\frac{dT(\tau_\nu)}{dz}\frac{dB_\nu(T(\tau_\nu))}{dT(\tau_\nu)}. \tag{2.66}$$

Integrating over all frequencies gives the total flux as,

$$F(\tau \gg 1) = -\frac{4\pi}{3}\frac{1}{\rho}\frac{dT(\tau)}{dz}\int_0^\infty \frac{1}{\kappa_\nu}\frac{dB_\nu(T(\tau_\nu))}{dT(\tau_\nu)}d\nu. \tag{2.67}$$

By allowing the opacity κ_ν to be equal to an average with frequency such that $\kappa_\nu = \bar{\kappa}_R$ where $\bar{\kappa}_R$ is the Rosseland mean opacity, Eq. 2.67 can be written as,

$$F(\tau \gg 1) = -\frac{4\pi}{3}\frac{1}{\rho}\frac{dT(\tau)}{dz}\frac{1}{\bar{\kappa}_R}\int_0^\infty \frac{dB_\nu(T(\tau_\nu))}{dT(\tau_\nu)}d\nu. \tag{2.68}$$

Hence, the Rosseland mean opacity is defined by using $dB_\nu(T(\tau_\nu))/dT(\tau_\nu)$ as the weighting function Z_ν in Eq. 2.59,

$$\frac{1}{\bar{\kappa}_R} \equiv \frac{\int_0^\infty \frac{1}{\kappa_\nu}\frac{dB_\nu(T(\tau_\nu))}{dT(\tau_\nu)}d\nu}{\int_0^\infty \frac{dB_\nu(T(\tau_\nu))}{dT(\tau_\nu)}d\nu}. \tag{2.69}$$

This opacity is used for optically thick environments.

Chapter 3

Smoothed Particle Hydrodynamics

We model the V1309 Sco progenitor and merger using Smoothed Particle Hydrodynamics (SPH hereafter) methods (Monaghan, 1992). The SPH method is intrinsically Lagrangian, where fluid is considered to consist of overlapping individual particles, and equations are written to follow each particle. Each particle is considered to exist within a spherical region, where gas properties are defined by a "smoothing kernel" with a radius characterized by a quantity called the "smoothing length" h. Here, we discuss the exact equations of motion used in SPH and the most appropriate kernel function for the purposes of this work.

3.1 Equations of Motion

The volume within a particle's smoothing kernel is considered to be a packet of fluid subject to the equations of fluid dynamics. At the most fundamental level, the SPH technique is a method for approximating integrals within fluids using discrete summations. Following the work of Monaghan (1992), the mathematical convolution of some function $f'(\mathbf{r})$ with another function $W(\mathbf{r}, h)$ results in the

function $f(\mathbf{r})$, which is written as,

$$f(\mathbf{r}) = \int f'(\mathbf{r}')W(\mathbf{r}-\mathbf{r}', h)\, d\mathbf{r}'. \tag{3.1}$$

where here, $W(\mathbf{r}-\mathbf{r}', h)$ is known as the "kernel function" and is both normalized to unity over all space and becomes a delta function in the limit as $h \to 0$. Eq. 3.1 can be approximated by summation as,

$$f_s(\mathbf{r}) = \sum_j m_j \frac{f_j}{\rho_j} W_j\,, \tag{3.2}$$

where m_j, ρ_j, and $\mathbf{r}_j$ are the mass, density, and position at the center of particle j, $f_j \equiv f(\mathbf{r}_j)$ is the value of the function f at $\mathbf{r}_j$, and $W_j \equiv W(|\mathbf{r}-\mathbf{r}_j|, h_j)$. While quantities can be calculated at any point in space using Eq. 3.2, the central values of the nearby SPH particles must be known. From here forward, continuous quantities will be written explicitly as a function of position $\mathbf{r}$ and particle central values will be written with subscripts i or j. For example, the density at any position $\mathbf{r}$ is written as,

$$\rho(\mathbf{r}) = \sum_j m_j W_j, \tag{3.3}$$

while the density at the center of a particle i is written as,

$$\rho_i = \sum_j m_j W_{ij}, \tag{3.4}$$

where $W_{ij} \equiv W(|\mathbf{r}_i - \mathbf{r}_j|, h_j)$.

3.1.1 Momentum Equation

The momentum equation of fluid dynamics is written as,

$$\frac{d\mathbf{v}(\mathbf{r})}{dt} = -\frac{\nabla P(\mathbf{r})}{\rho(\mathbf{r})}, \tag{3.5}$$

where $\mathbf{v}$ is the flow velocity, t is time, $P(\mathbf{r})$ is pressure, and $\mathbf{r}$ is the position. This can be rewritten with $\rho(\mathbf{r})$ inside the operators to maintain symmetric formulae,

$$\frac{d\mathbf{v}(\mathbf{r})}{dt} = -\nabla\left(\frac{P(\mathbf{r})}{\rho(\mathbf{r})}\right) - \frac{P(\mathbf{r})}{\rho(\mathbf{r})^2}\nabla\rho(\mathbf{r}). \tag{3.6}$$

From Eq. 3.2, Eq. 3.6 is calculated at the center of a particle i as,

$$\frac{d\mathbf{v}_i}{dt} = -\nabla\left(\sum_j m_j \frac{P_j}{\rho_j^2} W_{ij}\right) - \frac{P_i}{\rho_i^2}\sum_j m_j \nabla W_{ij}\,, \tag{3.7}$$

$$\frac{d\mathbf{v}_i}{dt} = -\sum_j m_j \left(\frac{P_j}{\rho_j^2} + \frac{P_i}{\rho_i^2}\right)\nabla W_{ij}\,, \tag{3.8}$$

where particle central values are taken as constants.

3.1.2 Energy Equation

The energy equation of fluid dynamics is written as,

$$\frac{du(\mathbf{r})}{dt} = -\left(\frac{P(\mathbf{r})}{\rho(\mathbf{r})}\right)\nabla\cdot\mathbf{v}(\mathbf{r}), \tag{3.9}$$

where $u(\mathbf{r})$ is the specific internal energy, defined as the total internal energy divided by mass, at location $\mathbf{r}$. Symmetrizing the equation as before in Eq. 3.6, Eq. 3.9 can be written as,

$$\frac{du(\mathbf{r})}{dt} = \mathbf{v}(\mathbf{r})\cdot\nabla\left(\frac{P(\mathbf{r})}{\rho(\mathbf{r})}\right) - \nabla\cdot\left(\frac{P(\mathbf{r})}{\rho(\mathbf{r})}\mathbf{v}(\mathbf{r})\right). \tag{3.10}$$

To turn this into an SPH summation, Eq. 3.2 is used as,

$$\frac{du_i}{dt} = \sum_j m_j \frac{P_j}{\rho_j^2} \mathbf{v}_i \cdot \nabla W_{ij} - \sum_j m_j \frac{P_j}{\rho_j^2} \mathbf{v}_j \cdot \nabla W_{ij}, \tag{3.11}$$

$$\frac{du_i}{dt} = \sum_j m_j \frac{P_j}{\rho_j^2} \left(\mathbf{v}_i - \mathbf{v}_j\right) \cdot \nabla W_{ij}. \tag{3.12}$$

The term $\nabla \cdot \mathbf{v}(\mathbf{r})$ in Eq. 3.9 is written symmetrically by placing $\rho(\mathbf{r})$ inside the operators to get,

$$\nabla \cdot \mathbf{v}(\mathbf{r}) = \frac{1}{\rho(\mathbf{r})} \Big[\nabla \cdot \big[\rho(\mathbf{r}) \mathbf{v}(\mathbf{r})\big] - \mathbf{v}_i \cdot \nabla \rho(\mathbf{r}) \Big] \tag{3.13}$$

$$\nabla \cdot \mathbf{v}(\mathbf{r}) = \frac{1}{\rho_i} \sum_j m_j \left(\mathbf{v}_j - \mathbf{v}_i\right) \cdot \nabla W_{ij}. \tag{3.14}$$

Therefore, Eq. 3.9 becomes,

$$\frac{du_i}{dt} = \sum_j m_j \frac{P_i}{\rho_i^2} \left(\mathbf{v}_i - \mathbf{v}_j\right) \cdot \nabla W_{ij}. \tag{3.15}$$

Averaging Eq. 3.12 and Eq. 3.15 gives,

$$\frac{du_i}{dt} = \frac{1}{2} \sum_j m_j \left(\frac{P_j}{\rho_j^2} + \frac{P_i}{\rho_i^2} \right) \left(\mathbf{v}_i - \mathbf{v}_j\right) \cdot \nabla W_{ij}. \tag{3.16}$$

Notice that Eq. 3.16 has the same symmetric terms as the momentum equation, which maintains continuity between two particles.

3.1.3 Artificial Viscosity

We have intentionally neglected the viscosity term in the above equations for simplicity, but it must be included as it contributes significantly to the loss of angular momentum within a fluid. While viscosity is thought to be essentially some internal shear between fluid interfaces, the mechanism by which it operates cannot be uniformly modeled. Instead, a new variable is introduced to the

particle equations of motion in Eq. 3.8 and Eq. 3.16 such that,

$$\frac{d\mathbf{v}_i}{dt} = -\sum_j m_j \left(\frac{P_j}{\rho_j^2} + \frac{P_i}{\rho_i^2} + \Pi_{ij} \right) \nabla W_{ij} \,, \tag{3.17}$$

$$\frac{du_i}{dt} = \frac{1}{2} \sum_j m_j \left(\frac{P_j}{\rho_j^2} + \frac{P_i}{\rho_i^2} + \Pi_{ij} \right) (\mathbf{v}_i - \mathbf{v}_j) \cdot \nabla W_{ij} \tag{3.18}$$

where Π_{ij} is the artificial viscosity term, introduced by Monaghan and Gingold (1983) as,

$$\Pi_{ij} = \begin{cases} \frac{-\alpha \bar{c}_{ij} \mu_{ij} + \beta \mu_{ij}^2}{\bar{\rho}_{ij}} & (\mathbf{v}_i - \mathbf{v}_j) \cdot (\mathbf{r}_i - \mathbf{r}_j) > 0, \\ 0, & \text{otherwise}, \end{cases} \tag{3.19}$$

$$\mu_{ij} = \frac{\bar{h}_{ij} \, (\mathbf{v}_i - \mathbf{v}_j) \cdot (\mathbf{r}_i - \mathbf{r}_j)}{(\mathbf{r}_i - \mathbf{r}_j)^2 + \eta^2}, \tag{3.20}$$

where $\bar{c}_{ij}$, $\bar{\rho}_{ij}$, and $\bar{h}_{ij}$ are the arithmetic means of the sound speed, density, and smoothing length, respectively, and η is chosen to be $\eta \approx 0.1 \bar{h}_{ij}$ to prevent Π_{ij} from becoming infinite when $\mathbf{v}_i - \mathbf{v}_j \neq 0$ and separation $\mathbf{r}_i - \mathbf{r}_j \to 0$. The coefficients α and β are similar to the Navier-Stokes first and second order viscosity coefficients and are typically given values of $\alpha = 0.5$ and $\beta = 1$, with $\alpha = 1$ and $\beta = 2$ for situations with strong shocks (Steinmetz, 1996).

While there are many proposed alternatives to Eq. 3.19 (Hosono et al., 2016), the Balsara correction (Balsara, 1995) is among the most common, as it prevents regions with low particle numbers from being stripped of their angular momentum too quickly (Steinmetz, 1996). Following the work of Steinmetz (1996), it is written as,

$$\bar{\Pi}_{ij} = \Pi_{ij} \, (f_i + f_j) \,, \tag{3.21}$$

$$f_i = \frac{|\nabla \cdot \mathbf{v}|_i}{|\nabla \cdot \mathbf{v}|_i + |\nabla \times \mathbf{v}|_i + 0.0001 c_i / h_i}. \tag{3.22}$$

where $\bar{\Pi}_{ij}$ is the Balsara–corrected artificial viscosity term, $|\nabla \cdot \mathbf{v}|_i$ and $|\nabla \times \mathbf{v}|_i$

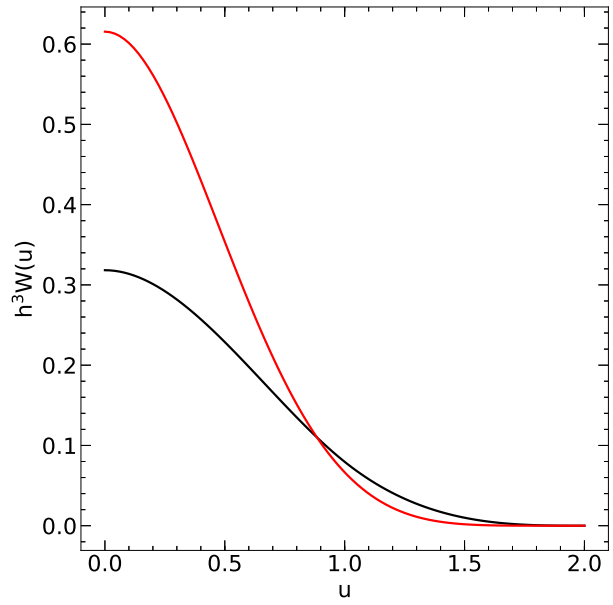

Figure 3.1: Dimensionless quantity $h^3W(u)$ is plotted against $u = r/h$ for the cubic spline function (black) as described by Eq. 3.23 and the Wendland C^4 function (red), as written in Eq. 3.24. Both kernel functions have compact support over $2h$.

are the divergence and curl of the velocity evaluated at the position of particle i, and $0.0001c_i/h_i$ prevents divergences, where c_i and h_i are the sound speed and smoothing length of particle i respectively. For the purposes of this work, we use the artificial viscosity prescription in the SPH code `StarSmasher`, which is different than that described here (see §4.1.1).

3.2 Kernel Functions

Using kernels with compact support, meaning they equal zero at the surface of the kernel, is a hallmark of SPH codes. By holding the number of neighboring particles N_{nb} fixed over time and allowing the smoothing lengths h vary for each particle, a trade off is made between numerical convergence and computation time by choosing N_{nb}.

A popular kernel function is the cubic spline (Monaghan and Lattanzio,

1985),

$$W(u) = \frac{1}{\pi h^3} \begin{cases} 1 - \frac{3}{2}u^2 + \frac{3}{4}u^3 & 0 \leq u \leq 1 \\ \frac{1}{4}(2-u)^3 & 1 < u \leq 2 \\ 0 & \text{otherwise} \end{cases} \tag{3.23}$$

where $u = r/h$ is the ratio between the distance from the center of the particle and the smoothing length. While this does provide a smooth function from center to surface, as seen in Fig. 3.1, it also gives way to clumping instabilities, also known as pairing instabilities. This occurs when two or more particles get very close to one another and can not separate. Dehnen and Aly (2012) show this instability occurs when the total internal energy for some number of particles enters a local equilibrium based on their configuration rather than the hydrostatic equilibrium condition. This tends to happen with functions that have negative Fourier transformations, such as the cubic spline, and do not for other functions, such as the Wendland family (Wendland, 1995). In particular, the Wendland functions perform well in terms of keeping the Poisson shot noise low when calculating summation interpolants in Eq. 3.1 compared to other functions (Rosswog, 2015).

We use the 3D Wendland C^4 function (see Fig. 3.1) for the purposes of this work,

$$W(u) = \frac{1}{\pi h^3}\frac{495}{256}\left(1 - \frac{1}{2}u\right)^6\left(\frac{35}{12}u^2 + 3u + 1\right). \tag{3.24}$$

The Wendland C^4 kernel is free from clumping instabilities (Dehnen and Aly, 2012) and yields superior numerical convergence over the traditional cubic spline (Zhu et al., 2015). It is significantly steeper than the cubic spline near the center of the kernel, but also approaches zero more gradually near the surface, as seen in Fig. 3.1. This property also helps prevent clumping by keeping the kernel gradient less flat around the center such that small fluctuations produce larger forces, as in Eq. 3.8.

3.3 Spatial Resolution

The centers of SPH particles are the only locations where physical values are calculated by the process described in §3.1. At each of these positions the number of neighbors requirement described in §3.2 is strictly met and thus these are the only locations where physical values are guaranteed some level of accuracy dependent on the number of neighbors and smoothing lengths. For all other positions in the simulation space, this accuracy level is not guaranteed but can still be achieved with a sufficient number of overlapping kernels. Thus, the "spatial resolution" within some volume at some location in an SPH simulation is defined as the particle number density,

$$n(\mathbf{r}) = \frac{N(V)}{V}, \tag{3.25}$$

where $N(V)$ is the number of particle centers in volume V at position $\mathbf{r}$. At the center of particle i,

$$n_i = \frac{3}{4\pi}\frac{N_{nb,i}}{h_i^3}, \tag{3.26}$$

where $N_{nb,i}$ is the number of neighbors for particle i and h_i is its smoothing length.

Chapter 4

StarSmasher

We use the SPH code `StarSmasher` (Gaburov et al., 2010) to simulate the V1309 Sco progenitor and merger, a code specifically designed for contact binary merger dynamics. `StarSmasher` can generate 3D stellar models from 1D inputs and is free from shocks that form when particles from regions with significantly different mass resolutions mix. The code calculates gravitational accelerations using the `TREESPH` method (Hernquist and Katz, 1989). We use the Wendland C^4 kernel as described in §3.2 at all times. In this chapter, we discuss the equations of motion, equation of state, and radiative cooling prescription in `StarSmasher` as well as the creation of 3D stellar models and general workflow for simulating binary mergers.

4.1 `StarSmasher` Equations of Motion

The equations of motion in `StarSmasher` are different from that in §3.1 in that particle smoothing lengths h_i vary with the nearby mass resolution. Following Appendix A of Gaburov et al. (2010), the user is given control of the parameter `nnopt`, defined as,

$$\texttt{nnopt} \equiv \sum_{j}^{N_{nb,i}} G\left(r_{ij}, h_i\right), \tag{4.1}$$

where $N_{nb,i}$ is the number of neighbors for particle i, $r_{ij} \equiv |\mathbf{r}_i - \mathbf{r}_j|$ is the distance from the center of particle i to particle j, and the weighting function $G\left(r_{ij}, h_i\right)$ is written as,

$$G\left(r_{ij}, h_i\right) \equiv V(4h_i - 4|r_{ij} - h_i|, h_i), \tag{4.2}$$

$$V(x_{ij}, h_i) \equiv 4\pi \int_0^{x_{ij}} x_{ij}'^2 W(x_{ij}', h_i) dx_{ij}', \tag{4.3}$$

where $0 \leq x_{ij} < 2h_i$, and $W(x_{ij}, h_i)$ is the kernel function with compact support of $2h_i$. When the summation in Eq. 4.1 is less than the value of `nnopt`, the smoothing length h_i is set such that $N_{nb,i}$ is incremented by 1. In other words, h_i is increased so that the kernel of particle i includes the next closest particle. Then, the following condition is checked,

$$\sum_j^{N_{nb,i}} G\left(r_{ij}, h_i\right) \geq \texttt{nnopt}. \tag{4.4}$$

The particle i incrementally gains more neighbors until Eq. 4.4 is satisfied, and this defines its number of neighbors $N_{nb,i}$.

4.1.1 Artificial Viscosity

The acceleration and change in internal energy of particle i due to artificial viscosity in `StarSmasher` is written as in the appendix of Hwang et al. (2015) as,

$$\left(\frac{d\mathbf{v}_i}{dt}\right)_{\text{AV}} = -\sum_j \frac{1}{2} m_j \Big[\Pi_{ij} \nabla W_{ij}(h_i) + \Pi_{ji} \nabla W_{ij}(h_j)\Big], \tag{4.5}$$

$$\left(\frac{du_i}{dt}\right)_{\text{AV}} = \sum_j \frac{1}{2} m_j \Pi_{ij} \left(\mathbf{v}_i - \mathbf{v}_j\right) \cdot \nabla W_{ij}(h_i). \tag{4.6}$$

This is similar to the approach described in §3.1.3, but here Π_{ij} has different dimensions and indices i and j are not interchangeable, so $\Pi_{ij} \neq \Pi_{ji}$. The

artificial viscosity term Π_{ij} used in `StarSmasher` is written as,

$$\Pi_{ij} = 2\frac{P_i}{\rho_i^2}\left(-\alpha\mu_{ij} + \beta\mu_{ij}^2\right), \tag{4.7}$$

where,

$$\mu_{ij} = \begin{cases} \frac{(\mathbf{v}_i - \mathbf{v}_j)\cdot(\mathbf{r}_i - \mathbf{r}_j)}{c_i|\mathbf{r}_i - \mathbf{r}_j|} f_i & \text{if } (\mathbf{v}_i - \mathbf{v}_j)\cdot(\mathbf{r}_i - \mathbf{r}_j) < 0, \\ 0 & \text{if } (\mathbf{v}_i - \mathbf{v}_j)\cdot(\mathbf{r}_i - \mathbf{r}_j) \geq 0, \end{cases} \tag{4.8}$$

$$f_i = \frac{|\nabla\cdot\mathbf{v}|_i}{|\nabla\cdot\mathbf{v}|_i + |\nabla\times\mathbf{v}|_i + 0.00001c_i/h_i}. \tag{4.9}$$

In the current work, parameters $\alpha = 1$ and $\beta = 2$ are always used (see §3.1.3).

4.2 Radiative Cooling

Modeling the physics of radiative transfer as described in §2 in SPH codes requires characterizing the absorption and emission characteristics of the fluid, which varies with direction and frequency, as seen in the derivation of the radiative transfer equation in §2.2. Radiative cooling is modeled in `StarSmasher` by treating particles as polytropic pseudo–clouds, as in Stamatellos et al. (2007), and applying a "pressure scale height method" (Lombardi et al., 2015).

Following the work of Stamatellos et al. (2007), consider an SPH particle i located within a pseudo–cloud at radius $r = \xi_r R_0$, where ξ_r is a dimensionless radius and R_0 is a scale–length. The cloud has a central density ρ_c and is chosen, along with R_0 to reproduce ρ_i and gravitational potential Φ_i at ξ_r,

$$\rho_i = \rho_c\Theta(\xi_r)^n, \tag{4.10}$$

$$\Phi_i = -\varphi_r G\rho_c 4\pi R_0^2, \tag{4.11}$$

where $\Theta(\xi_R)$ is the Lane–Emden function for index n (Chandrasekhar, 1939),

$$\varphi_r = -\xi_R \frac{d\Theta}{d\xi_r}(\xi_R) + \Theta(\xi_r), \tag{4.12}$$

where ξ_R is the boundary of the polytrope.

The pressure scale height is defined as,

$$H_P \equiv \frac{P(\mathbf{r})}{|\nabla P(\mathbf{r})|}. \tag{4.13}$$

Using the polytropic equation of state, $P(\mathbf{r}) = K\rho(\mathbf{r})^{1+1/n}$ with Eq. 4.10 gives,

$$H_{P,i} = \frac{R_0}{n+1} \frac{\Theta(\xi_r)}{|d\Theta(\xi_r)/d\xi_r|} \tag{4.14}$$

The column–density of particle i on a radial line from ξ_r to ξ_R is given by Stamatellos et al. (2007),

$$\Sigma_i(\xi_r) = \int_{\xi_r}^{\xi_R} \rho_c \Theta(\xi_r')^n R_0 d\xi_r', \tag{4.15}$$

$$= \frac{\rho_i}{\Theta(\xi_r)^n} \int_{\xi_r}^{\xi_R} \Theta(\xi_r')^n R_0 d\xi_r'. \tag{4.16}$$

For SPH codes, ρ_i in Eq. 4.10 is known, and $H_{P,i}$ can be calculated by rewriting Eq. 4.13 using the fluid momentum equation in Eq. 3.5,

$$H_{P,i} = \frac{P_i}{\rho_i |d\mathbf{v}_i/dt|}. \tag{4.17}$$

Thus, writing Eq. 4.16 using terms ρ_i and $H_{P,i}$ yields,

$$\Sigma_i(\xi_r) = \frac{(n+1)\rho_i H_{P,i}}{\Theta(\xi_r)^{n+1}} \left| \frac{d\Theta(\xi_r)}{d\xi_r} \right| \int_{\xi_r}^{\xi_R} \Theta(\xi_r')^n d\xi_r' \tag{4.18}$$

As in Stamatellos et al. (2007), the pseudo–mean column–density is calcu-

lated by taking a mass–weighted average of $\Sigma_i(\xi_r)$ over all ξ_r,

$$\bar{\Sigma}_i = \frac{\int_0^{\xi_R} \Sigma_i(\xi_r)\Theta(\xi_r)^n \xi_r^2 d\xi_r}{\int_0^{\xi_R} \Theta(\xi_r)^n \xi_r^2 d\xi_r}. \tag{4.19}$$

$$= \zeta' \rho_i H_{p,i}, \tag{4.20}$$

where ζ' is a dimensionless quantity that varies with n (see Lombardi et al. 2015 for a table of values and further discussion).

The pseudo–mean column–density is related to the optical depth and hence describes the ability for radiation to escape. As radiation escapes, it carries away specific internal energy, defined by Stamatellos et al. (2007) as,

$$\left.\frac{du_i}{dt}\right|_{\rm RAD} = \frac{4\sigma_{\rm SB}\left[T_0(\mathbf{r}_i)^4 - T_i^4\right]}{\bar{\Sigma}_i^2 \bar{\kappa}_R(\rho_i, T_i) + \bar{\kappa}_P(\rho_i, T_i)^{-1}}, \tag{4.21}$$

where $T_0(\mathbf{r}_i)$ represents the background effective temperature, ρ_i and T_i are the density and temperature of particle i, $\bar{\kappa}_R(\rho_i, T_i)$ is the pseudo–mean opacity (see Eq. 23 of Stamatellos et al. 2007), and $\kappa_P(\rho_i, T_i)$ is the Planck mean opacity (see §2.6). The background effective temperature is an artificial consideration that prevents particles from cooling beyond some characteristic temperature due to radiative processes alone. It can be thought of as the temperature of the physical surrounding region within which the `StarSmasher` simulation is being performed, such as the interstellar medium for example. Its value is specific to the physical context within which the simulation is being performed and we use $T_0 = 100$ K for our dynamical simulation in §6.3, as done in Lombardi et al. (2015) for modeling the V838 Mon red nova. While $\bar{\kappa}_R(\rho_i, T_i)$ and $\kappa_P(\rho_i, T_i)$ are calculable for a polytrope (see Stamatellos et al. 2007 and Lombardi et al. 2015 for details), tabulated values are used instead.

4.2.1 Implementation

In this section we provide a summary of the implementation of radiative cooling in `StarSmasher` by Lombardi et al. (2015). Specific internal energies are updated for each timestep,

$$u_i(t+\Delta t) = u_i(t)e^{-\Delta t/t_{\mathrm{therm},i}} + u_{0,i}\left[1-e^{-\Delta t/t_{\mathrm{therm},i}}\right] + \left.\frac{du_i}{dt}\right|_{\mathrm{HYDRO}}\Delta t, \quad (4.22)$$

where t is time, Δt is the timestep, $t_{\mathrm{therm},i}$ is the thermalization timescale of particle i, $\left.\frac{du_i}{dt}\right|_{\mathrm{HYDRO}}$ is the change in u_i due to only hydrodynamical effects, and $u_{0,i}$ is the background specific internal energy, which describes the specific internal energy of the surrounding region, as with background effective temperature discussed in §4.2. The background temperature is set such that $u_{0,i} = u(\rho_i, T_0)$, where $u(\rho_i, T_0)$ is calculated using the tabulated equation of state described in §4.3. The thermalization timescale for particle i $t_{\mathrm{therm},i}$ is written as,

$$t_{\mathrm{therm},i} = \frac{u_{0,i}-u_i}{du_i/dt|_{\mathrm{RAD}}}. \quad (4.23)$$

Thus, radiative cooling is calculated by `StarSmasher` for each particle by the following 5–step process:

1. Pseudo–mean column density $\bar{\Sigma}_i$ from Eq. 4.19 is calculated using ρ_i and $H_{P,i}$ from Eq. 4.17.

2. Pseudo–mean and Planck mean opacities $\bar{\kappa}_R(\rho_i, T_i)$ and $\bar{\kappa}_P(\rho_i, T_i)$ are calculated from pre–computed tables of values (see Eq. 15 of Lombardi et al. 2015 for pseudo–mean opacity calculation).

3. Radiative cooling rate $du_i/dt|_{\mathrm{RAD}}$ is calculated using Eq. 4.21 as well as the hydrodynamical heating rate $du_i/dt|_{\mathrm{HYDRO}}$ from Eq. 3.18 using the artificial viscosity term from §4.1.

4. Thermalization timescale $t_{\mathrm{therm},i}$ is calculated by Eq. 4.23.

5. Specific internal energy u_i is calculated using Eq. 4.22.

The total energy lost due to outgoing radiation in a `StarSmasher` simulation is calculated as,

$$E_{\rm rad} = \sum_{i}^{N} \left(u_{i,\rm HYDRO} - u_i \right), \tag{4.24}$$

where $u_{i,\rm HYDRO}$ is the specific internal energy particle i would have achieved if there was no radiative cooling and u_i includes the radiative cooling term. Values for $E_{\rm rad}$ are shown in §6.3 for a dynamical simulation of a stellar merger (see §4.4) and the change in $E_{\rm rad}$ with time for the same simulation is shown in Fig. 6.18.

4.3 Tabulated Equation of State (TEOS)

We use the same tabulated equation of state (TEOS hereafter) as in Nandez et al. (2015), which was created using the `MESA-EOS` module (see §4.2 of Paxton et al. 2011) for solar chemical composition $X = 0.7$, $Y = 0.28$, and $Z = 0.02$ to describe the relationship between density ρ, specific internal energy u, and temperature T, as seen in Fig. 4.1. The TEOS is created by Nandez et al. (2015) using `MESA` version 6208. The `MESA inlist` is provided in Appendix A.1. Outside the temperature range $8.2 \leq \log T \leq 3$, values are extrapolated using the method described in §4.3.1 with proportionality constants defined by the boundary values in the table. The values in the TEOS are hence based on the equation of state of `MESA`, which uses OPAL tables (Rogers and Nayfonov, 2002) for stars with masses $M \geq 0.1\ M_\odot$ and SCVH tables (Saumon et al., 1995) for low–mass stars and giant planets with masses $M < 1\ M_\odot$. Due to the high resolution of the TEOS, `StarSmasher` uses a bi–linear interpolation method to calculate physical values.

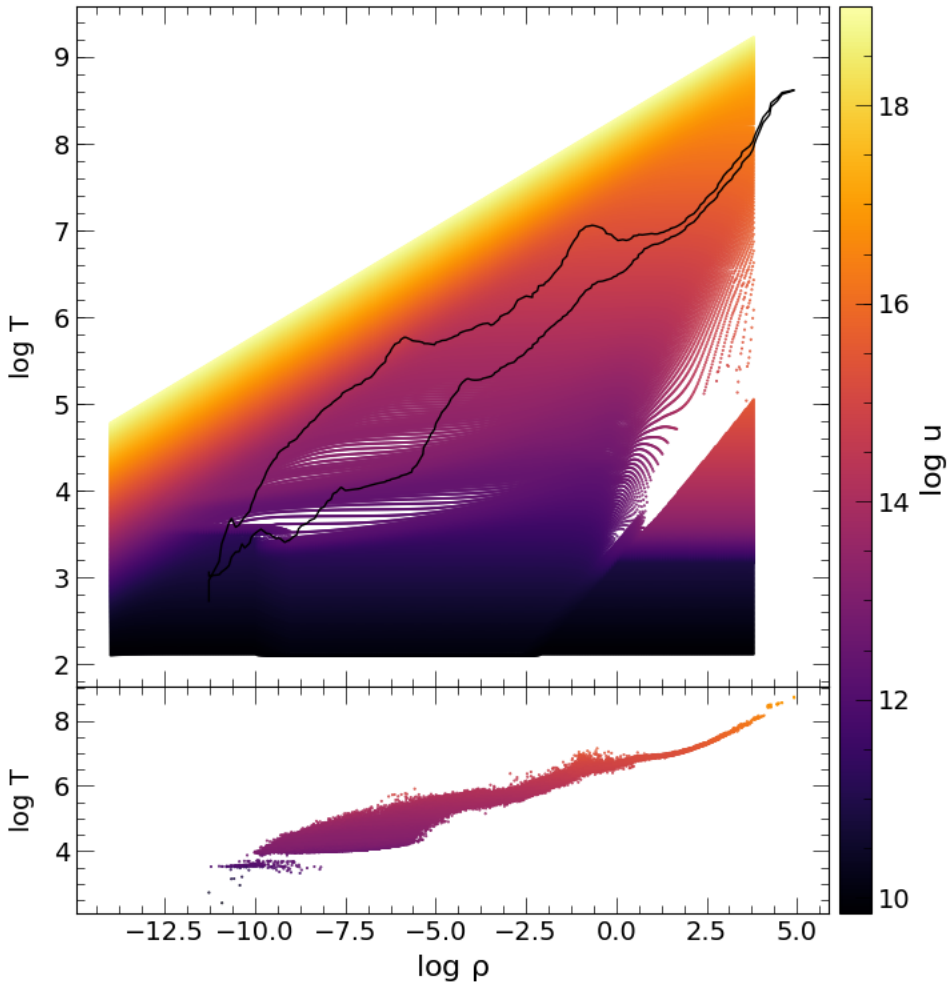

Figure 4.1: We show values of specific internal energy values in color for the range of $\log \rho$ and $\log T$ values covered by the TEOS discussed in §4.3. The black contour shows approximately the space occupied by particles in the dynamical simulation in §6.3 at $t \approx 8.33$ days. The bottom plot shows the particle ρ, T, and u values within the contour. `StarSmasher` interpolates across the TEOS using a bi–linear method within the TEOS bounds to calculate temperatures and a polynomial solution elsewhere, as discussed in §4.3.1. All values are in cgs units.

4.3.1 Polynomial Temperature Solution

When an attempt is made to use the TEOS to calculate a temperature with ρ and u values that are outside its domain, the temperature is instead calculated by solving a 4^{th} order polynomial equation of state constructed from the energy density terms of the ideal gas law and radiation (Hwang et al., 2015). By the equipartition theorem in three dimensions, the energy density of an ideal gas is,

$$u_{\text{ideal}} = \frac{3}{2}\frac{k_B T}{\mu m_H}, \tag{4.25}$$

where k_B is the Boltzmann constant and μm_H is the mean molecular weight. The energy due to radiation is the usual $U_{\text{rad}} = aT^4$. Thus, the specific internal energy is,

$$u_{\text{rad}} = \frac{aT^4}{\rho}, \tag{4.26}$$

where ρ is the mass density. The total energy density therefore follows from combining Eq. 4.25 with Eq. 4.26 to get,

$$u = \frac{3}{2}\frac{k_B T}{\mu m_H} + \frac{aT^4}{\rho}. \tag{4.27}$$

The general solution to this 4^{th} order polynomial follows from pages 55, 57, and 58 of Stillwell (1989) as,

$$T = -\sqrt{\frac{\gamma}{2}} + \frac{1}{\sqrt{2}}\sqrt{\frac{q}{\sqrt{2\gamma}} - \gamma}, \tag{4.28}$$

$$\gamma = \sqrt[3]{\frac{q^2}{16} + \sqrt{\left(\frac{q^2}{16}\right)^2 - \left(\frac{r}{3}\right)^3}} + \sqrt[3]{\frac{q^2}{16} - \sqrt{\left(\frac{q^2}{16}\right)^2 - \left(\frac{r}{3}\right)^3}}, \tag{4.29}$$

$$q = \frac{3}{2}\frac{k}{\mu m_H}\frac{1}{a}\rho, \tag{4.30}$$

$$r = -\frac{1}{a}\rho u. \tag{4.31}$$

4.4 Overview of Workflow

The first step in creating a contact binary merger simulation is to generate a full 3D SPH model of each star. However, `StarSmasher` does not include the necessary physics to perform stellar evolution, such as nuclear reactions and energy transfer, for example. Additionally, stellar evolution for Sun–like stars occurs on timescales of tens of billions of years, which is impractical to simulate using SPH methods, as SPH codes take timesteps on dynamical timescales, which can go down to seconds. 3D stellar models are instead initialized directly from 1D `MESA` output and relaxed dynamically until hydrostatic equilibrium is achieved with radiative cooling turned off (see §4.5). During this time, the only damping force present is that created by the artificial viscosity, as described in §6.1.

The gravitational force experienced by a small packet of mass m near the surface of a stellar model is approximated by treating the rest of the model as a point mass such that,

$$F_{g,m} = -\frac{GMm}{R^2}, \tag{4.32}$$

where G is the gravitational constant, $M \gg m$ is the mass of the point mass, and R is the radius of the stellar model, equal to the separation of m and M. If a second point mass of mass $M' \gg m$ is instantaneously placed some distance r away from m, the instantaneous change in the total gravitational force on m is,

$$\delta F_{g,m} = -\frac{GM'm}{r^2}. \tag{4.33}$$

To prevent significantly disrupting the hydrostatic equilibrium of each fully relaxed SPH stellar model by the artificial gravitational shock in Eq. 4.33, we create a binary system in a corotating frame by placing each model a large separation r apart such that $\delta F_{g,m}$ is small. The angular velocity that defines the corotating frame is that which allows the net centrifugal and gravitational

accelerations of the stars to balance and is written as (Gaburov et al., 2010),

$$\omega_{\rm orb}^2 = -\frac{1}{2}\left(\frac{\sum_{i=1}^{N_1} m_i|\dot{\mathbf{v}}_{x,i}|}{\sum_{i=1}^{N_1} m_i x_i} + \frac{\sum_{i=N_1}^{N} m_i|\dot{\mathbf{v}}_{x,i}|}{\sum_{i=N_1}^{N} m_i x_i}\right), \tag{4.34}$$

where $\omega_{\rm orb}$ is the orbital angular velocity, $|\dot{\mathbf{v}}_{x,i}|$ is the magnitude of the change in velocity of particle i in the direction x towards the center of mass with time, and x_i is the distance of particle i away from the center of mass. The particle index i is ordered such that all particles which originated in the primary star have $i = 1, \ldots, N_1$ and for the secondary star $i = N_1, \ldots, N$ where N is the total number of particles.

We relax the binary system with radiative cooling turned off until hydrostatic equilibrium is achieved in both stars in what is hereafter called a "binary scan". `StarSmasher` gradually places the stars closer to the center of mass to obtain hydrostatic equilibrium at smaller separations. This step is purely artificial, and the particles are not moved by applying a global force, but rather by instantaneously changing their positions with each time step. We stop the binary evolution when the desired separation is reached.

After the binary scan, we set the stars to interact dynamically in an inertial (non–rotating) frame in what is hereafter called a "dynamical run" or "dynamical simulation". We activate radiative cooling during the dynamical run to allow the system to lose energy by the process described in §4.2.

4.5 Relaxation Process

With a 1D `MESA` model as input, `StarSmasher` initializes a 3D model by placing particles on a close–packed hexagonal lattice (hcp hereafter) to best approximate the lowest possible initial energy configuration (Lombardi et al., 2006). The 3D model is initialized with a constant number density and each particle is assigned a physical value based on its radial position relative to the mass shells

in the original 1D model. After initialization, the 3D model is then set free to interact and its energy values are monitored to determine how well relaxed it is. The monitored energies are calculated as the following summations,

$$U = \sum_i m_i u_i \tag{4.35}$$

$$W = \frac{1}{2} \sum_i m_i \varphi_i \tag{4.36}$$

$$T = \frac{1}{2} \sum_i m_i v_i^2 \tag{4.37}$$

$$E = U + W + T, \tag{4.38}$$

where U is the total internal energy, W is the total gravitational potential energy, T is the total kinetic energy, E is the total energy, and φ_i is the gravitational potential of particle i (see Appendix A of Gaburov et al. 2010 for details on φ). Total internal energy U is calculated by summing over all non–core particles (see §4.5.1) while W and T are calculated with core particles inclusive.

As a model relaxes, its envelope exchanges internal energy and gravitational potential energy in search of a lower energy configuration. This forms oscillations with decreasing amplitude over time, as seen in Fig. 4.2. The oscillations are dampened by the artificial viscosity prescription in §4.1. Once oscillations are negligible (total internal energy $|\Delta U| \lesssim 10^{41}$ ergs s^{-1} or total gravitational potential energy $|\Delta W| \lesssim 10^{41}$ ergs s^{-1}), the model is considered to be relaxed.

4.5.1 Core Particles

Density changes very quickly over short length scales in dense stellar cores. The only way to spatially resolve the density in the traditional SPH way is to use a very large number of particles. As a result of the initial hcp lattice discussed in §6.1, a particle must always be placed at the center of the model. Let a regular SPH particle c have a mass $m_c^* = \rho_c^* V_c^*$, where V_c^* is its volume, determined by

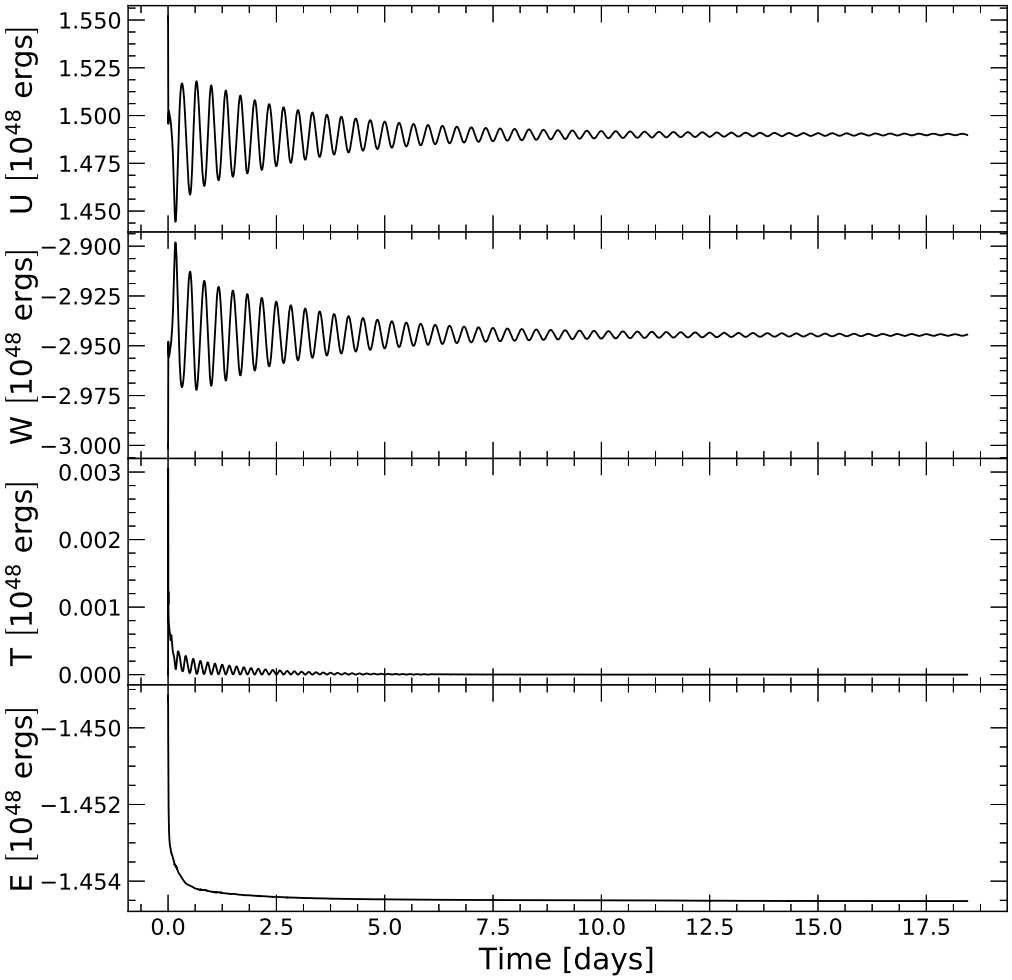

Figure 4.2: Energy fluctuations in the envelope of a $N = 3 \times 10^5$ `StarSmasher` relaxation for a `MESA` model with $M = 1.52\ M_\odot$ and $R = 3.72\ R_\odot$ evolved to an age of 2.54 Gyr, placing it near the base of the red giant branch (see §6.2). Here, U, W, and T are the total internal, gravitational potential, and kinetic energies respectively, with E being the total energy. See the beginning of §4.5 for how they are calculated. Damping is due to the artificial viscosity prescription.

its smoothing length. The only possible physical values at the center of the model are those at the center of the `MESA` model, so particle c is assigned the central `MESA` density. Recall the number density is constant, as discussed in §4.5, so it is therefore possible for particle c to have a volume for which its mass exceeds the total mass of the star.

To show this, note that the number density at the core is equal to the total number density of the model, as number density is constant throughout,

$$\frac{1}{V_c} = \frac{N}{V}, \tag{4.39}$$

where N is the number of particles and V is the total volume of the model. Therefore, the central particle must have mass,

$$m_c^* = \frac{\rho_c^* V}{N}. \tag{4.40}$$

For a given value of N, it might be that $m_c^* \geq M$ where M is the total mass of the star. For example, for $N = 2 \times 10^5$, $R = 3.715\ R_\odot$, $\rho_c^* \approx 6 \times 10^4$ g cm^{-3}, and $M = 1.52\ M_\odot$, $m_c^* = 10.91\ M_\odot$, which is much too large. Thus, `StarSmasher` uses a special "core particle" when the following condition is met,

$$\rho_c^* \frac{V}{N} \geq \omega M, \tag{4.41}$$

where ρ_c^* is the central density of the `MESA` model and the dimensionless parameter $0 < \omega < 1$ is inserted to ensure the centrally placed particle's mass does not exceed ωM. The value $\omega = 0.25$ is always adopted, independent of the `MESA` model.

A core particle is different from regular particles in the sense that they interact with other particles gravitationally only. Gravitational interactions with core particles are calculated identically as with regular particles. However, core particle smoothing lengths are fixed upon initialization and are not subject to

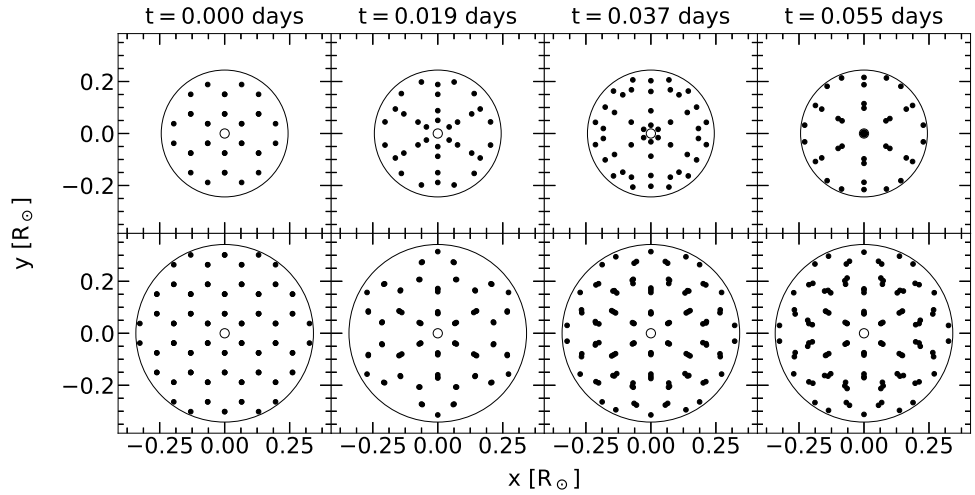

Figure 4.3: Early time evolution of the particle center positions (filled circles) within the kernel (large circle) of the core particle (small circle) in two `StarSmasher` relaxations of a $N = 5 \times 10^4$, $M = 1.52\ M_\odot$, `nnopt`$= 69$ model, initialized from the same `MESA` model described in §6.1. The core particle kernel has a smoothing length in the top panel of $h_c = 0.122\ R_\odot$ and $h_c = 1.71\ R_\odot$ in the bottom panel. As seen in the top panel, nearby particles clump together with the core when the core particle smoothing length is sufficiently small.

the `nnopt` criterion described in §4.1. When a `StarSmasher` model is initialized with a core particle, the core particle is the last to be assigned values. The mass of a core particle m_c thus equals the difference between the total mass of the `MESA` model and the total mass of all the initialized particles.

A core particle can have a smoothing length that causes the gravitational force between the core particle and its neighbors to be very large. This causes the core particle neighbors to clump around it and form a pairing instability (see §3.2), as seen in Fig. 4.3. This generally occurs for small core particle smoothing lengths, and by assigning a larger smoothing length, clumping no longer occurs. This is due to the core particle mass being distributed over a larger volume and hence the force due to gravity changes more gradually near the core particle, which prevents clumping. All presented models in this work have core smoothing lengths that do not produce clumping in the vicinity of the core particle, which we find by increasing the smoothing length in small

increments until we no longer observe clumping near the core throughout the entire relaxation.

4.5.2 Optimal `nnopt`

When the number of neighbors parameter `nnopt` is varied, it changes the spatial resolution of the simulation by altering the particles' smoothing lengths (see §3.3). While intuitively a larger `nnopt` should produce a uniformly more accurate simulation by increasing the number of neighbors for each particle $N_{nb,i}$, this is not necessarily the case. Particles near the surface of the model have their neighbors distributed unevenly, with more neighbors located inward toward the core than outward toward the surface. Thus, the smoothing lengths of particles near the surface must be larger than those deeper within the envelope. The local spatial resolution $n(\mathbf{r}_i)$, as written in Eq. 3.26, is significantly lower in the outer kernels of the surface particles than the average through the envelope.

Lowering the value of `nnopt` allows for smaller smoothing lengths everywhere, including at the surface. Ultimately, lower `nnopt` gives better local spatial resolution n_i, as $n_i \propto N_{nb,i}/h_i^3$, but lower accuracy in the summation interpolants in §3.1. Thus, there exists some unique optimal `nnopt` value for a given N for which the spatial resolution and the accuracy in physical values calculated for the surface particles produce a profile that best matches the initial `MESA` model profile. See §6.1.1 for how `nnopt` can be found for given particle resolutions.

Chapter 5

FluxCal[1]

Even with a sufficiently accurate initial 3D stellar model of the V1309 Sco progenitor, simply merging it with a secondary in a dynamical simulation and retrieving SPH particle temperatures is not sufficient to make comparisons with the real observations in §1.2.1. The radiative transfer equation from §2.2 must be solved in the SPH simulation space to acquire spectral data which may then be compared to real observations. Previous attempts at solving the radiative transfer problem for SPH codes have all resulted in ray tracing algorithms which use either a Monte–Carlo or Runge–Kutta integration scheme, such as `SPAMCART` (Lomax and Whitworth, 2016), `MOCCASIN` (Hubber et al., 2016), `DART-Ray` (Natale et al., 2015), `Urchin` (Altay and Theuns, 2013), and `SPHRAY` (Altay et al., 2008), among others.

In a basic sense, ray tracing methods calculate the intensity by solving the equation of radiative transfer in Eq. 2.9 using a numerical integration method along many lines of sight through the simulated fluid. Each ray travels through particle smoothing kernels until the condition for optical thickness in §2.1 is met, such that additional contributions to the intensity are considered negligible by the user. However, a ray may encounter an optically thick region with

[1]Available at `https://github.com/hatfullr/fluxcal`

poor spatial resolution, as discussed in §3.3. In this case, the physical values calculated in the usual SPH way in §3.1 are not consistently accurate. These regions are common in simulations involving stellar objects, where there exist high density, high opacity particles at the photosurface.

In this chapter, we introduce a new code we developed called `FluxCal` as an alternative to other ray tracing methods in solving the radiative transfer problem for SPH simulations. `FluxCal` uses both a novel envelope fitting technique (§5.3) and a Runge–Kutta ray tracer (§5.4) to calculate a grid of effective temperatures from a viewing angle. We discuss the limitations of ray tracing in SPH simulations with regards to stellar models in §5.1, the method by which `FluxCal` calculates particle optical depths in §5.2, and in §5.3 and §5.4 how this information is used to overcome the limitations. The visible luminosity and effective temperatures in SPH simulation space are calculated by `FluxCal` by the method described in §5.6 and §5.7, respectively. Finally, we provide a summary of how `FluxCal` operates in §5.8 and discuss additional tools `FluxCal` provides for exploring and/or debugging a simulation in §5.9.

5.1 Photosurface Problem

To evaluate the efficacy of ray–tracing algorithms in SPH simulations, consider a relaxed SPH stellar model with a photosphere located at a radius $R_{\rm ph}$, as defined in §2.4. Each particle in an SPH simulation must have some number of neighboring particles, as discussed in §3.1. The center of each particle is where the accuracy of physical values is guaranteed (see §3.3). Thus, the location farthest from the center of the model where physical values are accurate is the center of the outermost particle. Hence, to make an accurate calculation at the photosphere, the minimum requirement is that the outermost particle's center is located at the photosphere. This condition is true in general any time a photosurface is located in a a region of poor spatial resolution, not only

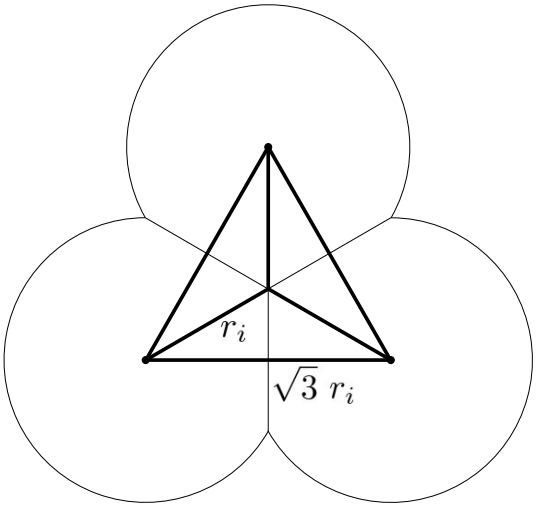

Figure 5.1: A sketch of three SPH particles with the largest separation from center–to–center allowed without forming a gap in the middle. An equilateral triangle is formed by tracing a line from each particle center to another, and the particles have radii that extend to the center of this triangle. It is a matter of trivial geometry to show the maximum separation from center–to–center is equal to $\sqrt{3}\ r_i$.

photospheres, and the following discussion applies to photosurfaces of any shape. To avoid having simulated fluid beyond the true surface of the initial model (where the density is zero), the outermost particle must have a physical radius r_i equal to the photospheric depth $R - R_{\rm ph}$, where R is the radius at the true surface.

Consider the SPH stellar model has some number of photospheric particles $N_{\rm ph}$ all with radii $r_i = R - R_{\rm ph}$ with centers located at the photospheric radius $R_{\rm ph}$. Allow these particles to be positioned in such a way that they are not neighbors with one another and such that there are no gaps between their kernels, as in Fig. 5.1. For this to hold true and still be valid in the SPH sense, their neighbors must all be located deeper within the model. When the particles are maximally separated, they form a hexagonal lattice such that their center–to–center separations are all equal to $\sqrt{3}\ r_i$, as seen in Fig. 5.1. To calculate the value of $N_{\rm ph}$, consider the number density of particles at the photosphere. The surface area within which there are three particles is the

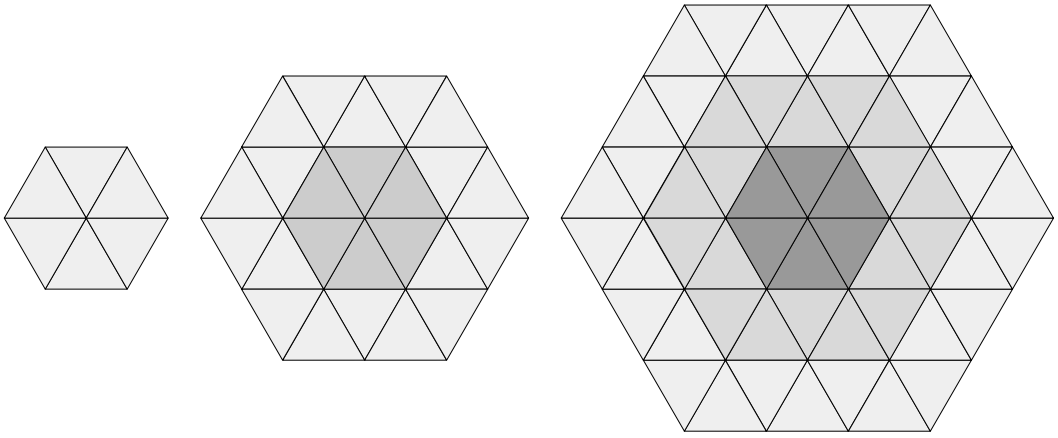

Figure 5.2: A sketch of hexagonal lattices of order $k = 1$ (left), $k = 2$ (middle), and $k = 3$ (right).

surface area of the equilateral triangle seen in Fig. 5.1, written as,

$$A_{\triangle} = \frac{3\sqrt{3}}{4} r_i^2. \tag{5.1}$$

Imagine constructing a complete hexagonal lattice of order k. The number of particles required to construct a $k = 1$, $k = 2$, and $k = 3$ lattice respectively, as seen in Fig. 5.2, as well as the surface area of each order is,

$$N_k = \{7, 19, 37\}, \tag{5.2}$$

$$A_k = \{6A_{\triangle}, 24A_{\triangle}, 54A_{\triangle}\}. \tag{5.3}$$

It is relatively straightforward to see that the total area of the k^{th} lattice is written as,

$$A_k = 6k^2 A_{\triangle}. \tag{5.4}$$

It is less straightforward to see that the number of particles in the k^{th} lattice is written as,

$$N_k = 1 + 6\sum_{j=1}^{k} j = 1 + 3k(k+1). \tag{5.5}$$

Eq. 5.5 can be made easier to understand by analyzing the $k = 4$ lattice, but

we leave this as an exercise for the reader.

By equating Eq. 5.4 to the total area of the photosphere, we can solve for k,

$$6k^2 A_{\triangle} = 4\pi R_{\rm ph}^2, \tag{5.6}$$

$$k = \sqrt{\frac{2\pi}{3A_{\triangle}}} R_{\rm ph}, \tag{5.7}$$

$$k = \frac{\sqrt{8\pi}}{3^{5/4}} \frac{R_{\rm ph}}{r_i}. \tag{5.8}$$

Thus, the total number of particles required at the photosphere is calculated by combining Eq. 5.5 and Eq. 5.8 to get,

$$N_{\rm ph} \geq 1 + \frac{\sqrt{8\pi}}{3^{5/4}} \frac{R_{\rm ph}}{r_i} \left(\frac{\sqrt{8\pi}}{3^{5/4}} \frac{R_{\rm ph}}{r_i} + 1 \right), \tag{5.9}$$

$$N_{\rm ph} \gtrsim 1 + 1.27 \frac{R_{\rm ph}}{r_i} \left(1.27 \frac{R_{\rm ph}}{r_i} + 1 \right). \tag{5.10}$$

For the V1309 Sco progenitor primary, we calculate the value of $r_i = R - R_{\rm ph}$ using the definition of optical depth in Eq. 2.2 with constant $\rho \sim 10^{-7}$ g cm^{-3} and $\bar{\kappa} \sim 0.25$ cm^2 g^{-1} near the photosphere taken from the `MESA` model, and letting $\tau_{\rm ph} = 2/3$. Here, $r_i \sim 4 \times 10^{-4}\ R_{\odot}$, and $R_{\rm ph} = 3.715\ R_{\odot}$ (also from the `MESA` model), so $N_{\rm ph} \gtrsim 10^8$ particles, which is impractical with contemporary technology.

5.2 Particle Optical Depths

The optical depth is found by integrating the optical depth from Eq. 2.2 over all frequencies,

$$d\tau(w) \equiv -\rho(w)\bar{\kappa}(w)dw, \tag{5.11}$$

where $\rho(w)$ is the mass density, $\bar{\kappa}(w)$ is the mean opacity (see §2.6), and w is an axis along a line of sight. When calculating the optical depth, `FluxCal`

uses tabulated Rosseland and Planck mean opacities, as in §2.6. We use the Rosseland mean opacity table from MESA (see §4.3 of Paxton et al. 2011) and the Planck mean opacity table from Semenov et al. (2003). We use Rosseland mean opacities for temperatures above 1500 K, and Planck mean opacities below.

The summation interpolant of Eq. 5.11 is calculated by the same method in §3.1 as,

$$d\tau(w) = -\sum_j m_j \bar{\kappa}_j W(|\mathbf{r} - \mathbf{r}_j|, h_j) dw, \tag{5.12}$$

where m_j and $\bar{\kappa}_j$ are the mass and mean opacity at the center of particle j respectively, h_j is the smoothing length of particle j, $\mathbf{r}$ is the position and $\mathbf{r}_j$ is the position of particle j. For a single isolated particle i,

$$d\tau(w) = -m_i \bar{\kappa}_i W(|w - w_i|, h_i) dw, \tag{5.13}$$

where here w is a distance along some axis that passes through the center of particle i. The optical depth of particle i is hence calculated by integrating Eq. 5.13 from the $\tau = 0$ surface of the kernel to the $\tau = \tau_i$ center. For a kernel function with compact support over $2h$, as described in §3.2,

$$\tau_i = m_i \bar{\kappa}_i \int_{w_i}^{w_i + 2h_i} W(|w - w_i|, 2h_i) dw. \tag{5.14}$$

Making the substitution $u = (w - w_i)/h_i$, $du = dw/h_i$,

$$\tau_i = m_i \bar{\kappa}_i h_i \int_0^2 W(u) du. \tag{5.15}$$

Performing the integral on the right–hand side results in some constant divided by the smoothing length cubed. Thus, Eq. 5.15 is written as,

$$\tau_i = \frac{m_i \bar{\kappa}_i}{h_i^2} C_\tau, \tag{5.16}$$

$\mathbf{m_i}$	$\mathbf{h_i}$	$\mathbf{T_i}$	$\rho_\mathbf{i}$	$\kappa_\mathbf{i}$	$\tau_\mathbf{i}$
$[10^{26}$ g]	$[10^{12}$ cm]	[K]	[g cm^{-2}]	[cm^2 g^{-1}]	[–]
0.687	1.131	3.398×10^3	6.270×10^{-10}	3.720×10^{-4}	7.287×10^{-2}
0.702	2.410	3.591×10^3	1.055×10^{-10}	1.984×10^{-4}	8.753×10^{-3}
0.859	3.083	3.492×10^3	3.103×10^{-11}	1.089×10^{-4}	3.589×10^{-3}
6.888	0.026	4.315×10^5	4.725×10^{-4}	6.992×10^1	2.582×10^8
5.354	0.017	5.585×10^5	1.995×10^{-3}	5.106×10^0	3.484×10^7
5.766	0.023	5.190×10^5	5.658×10^{-4}	5.381×10^1	2.181×10^8

Table 5.1: The masses m_i, smoothing lengths h_i, temperatures T_i, densities ρ_i, opacities κ_i, and optical depths τ_i as calculated by Eq. 5.16 for a selection of particles ≈ 8.33 days after merger in the `StarSmasher` dynamical simulation described in §6.3. Optically thin particles are shown on top and optically thick are shown on bottom. We use the Wendland C^4 kernel function in this simulation.

where $C_\tau = 3/(4\pi) \approx 0.24$ for the cubic spline function from Eq. 3.23 and $C_\tau = 55/(48\pi) \approx 0.36$ for the Wendland C^4 function from Eq. 3.24.

We show in Table 5.1 optical depths calculated by Eq. 5.16 for particles with the Wendland C^4 kernel function from our `StarSmasher` dynamical simulation of a contact binary merger discussed later in §6.3. There exists a wide range of optical depths in our dynamical simulation, as some particles are cool and diffuse in the outflow while others are hot and dense near the core.

5.3 Optically Thick (Envelope Fitting)

Any optically thick SPH particle at the surface of the gas cloud has a photosphere somewhere within its kernel, as discussed in §2.1. Tracing a ray through the kernel of an optically thick particle to find the location of its photosphere is inaccurate, as described in §3.3 and §5.8. Thus, we employ a fitting routine to match particle central temperatures T_i, pressures P_i, and local gravitational accelerations g_i to stellar envelope solutions.

5.3.1 Fundamental Envelope Solutions

Temperature T and pressure P profiles in stellar envelopes can be described as a family of solutions determined by surface gravity and effective temperature. For simple parameterized envelopes, this can be demonstrated analytically using the method described by Kippenhahn and Weigert (1994). Let the mean opacity $\bar{\kappa}$ scale as a power law with P and T such that $\bar{\kappa} = \kappa_0 P^a T^b$. The usual equation of radiative transport is hence written as,

$$\frac{\partial T}{\partial P} = \frac{3\kappa_0}{64\pi\sigma_{\rm SB} G} \frac{l}{mT^3} P^a T^b, \tag{5.17}$$

where G is the gravitational constant, $l \equiv 4\pi r^2 F$ is the luminosity and F is the local flux, m is the enclosed mass, and r is the radial position. It is assumed that $l \approx L$ and $m \approx M$, which are the total luminosity and total mass respectively, both of which are constants. Separating the variables in Eq. 5.17 and integrating yields,

$$T^{4-b} = BP^{(a+1)} + C, \tag{5.18}$$

where C is the integration constant and,

$$B \equiv \frac{4-b}{a+1} \frac{3\kappa_0}{64\pi\sigma_{\rm SB} G} \frac{L}{M}. \tag{5.19}$$

For any given values of a and b, the solution in Eq. 5.18 has a slope,

$$\nabla \equiv \frac{\partial \ln T}{\partial \ln P}. \tag{5.20}$$

The approximation of $\bar{\kappa}$ as a power law with P and T and setting $l \approx L$ and $m \approx M$ as constants restricts how far inwards the solution for any given a and b may be extended.

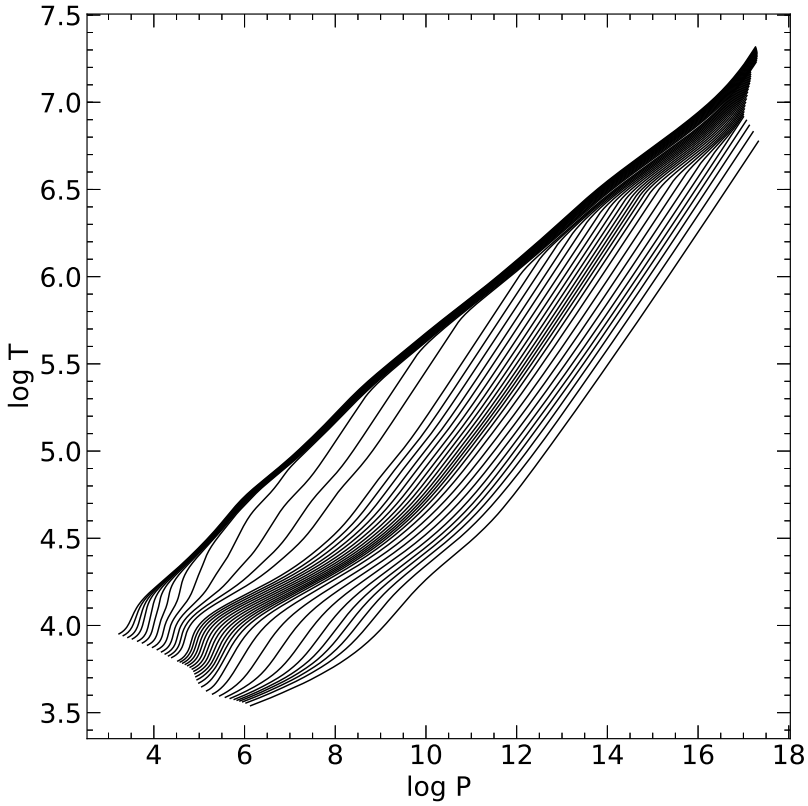

Figure 5.3: A set of MESA models in the range 0.1 $M_\odot \leq M < 2.0\ M_\odot$ evolved to the zero age main sequence. The lowest mass model is at the very bottom of the plot, with increasing mass moving upwards. These models are representative of the envelope solutions described by Kippenhahn and Weigert (1994).

5.3.2 FluxCal Envelope Fitting

We create stellar models and evolve them to the zero–age main sequence using default MESA parameters as seen in Fig. 5.3. We then evolve these to create a grid of models with surface gravitational accelerations g over a range of $-4 \leq \log g \leq 4.6$ in increments of 0.1. We tabulate values for ∇ through the temperature range $3.3 \leq \log T \leq 6$ in increments of 0.01 within the envelope for each model. For given particle central values T_i, and g_i, we calculate ∇ from the table by simple interpolation.

The pressure at the photosphere of low luminosity and/or low surface opacity stars is given by Eq. 2.58,

$$P_{\rm ph} \approx \frac{2}{3}\frac{g_i}{\bar{\kappa}_s}. \tag{5.21}$$

Here, $\bar{\kappa}_s$ is the mean opacity at the surface and is computed using a table of low temperature opacities (Ferguson et al., 2005). The fitted pressure is calculated as,

$$P = P_i \left(\frac{T}{T_i}\right)^{1/\nabla}, \tag{5.22}$$

where P_i is the particle central pressure.

We use a simple midpoint method to calculate T at $P = P_{\rm ph}$, where the lower bound is set to the lowest possible $\log T$ value in the ∇ table, the upper bound is set to $\log T_i$, and the midpoint is the numerical average of the two. For the upper bound, midpoint, and lower bound, FluxCal calculates P as in Eq. 5.22 and $P_{\rm ph}$ as in Eq. 5.21. From hereafter, we denote these points with a subscript u for upper bound, m for midpoint, and l for lower bound. When $P_u \geq P_{ph,u}$ and $P_m \leq P_{ph,m}$, T_l is given the value at T_m, otherwise $T_u = T_m$. The midpoint is recalculated as the numerical average of T_u and T_l, P and $P_{\rm ph}$ are recalculated at all three points, and the process is repeated until T_m converges to $T_{\rm ph}$. We use the temperature at the photosphere $T_{\rm ph}$ to calculate the effective temperature $T_{\rm eff}$ in §5.7.

5.4 Optically Thin (Runge–Kutta Integrator)

`FluxCal` traces a ray in the negative z direction using a 4^{th} order Runge–Kutta adaptive step size integrator (Press et al., 1992) to calculate the optical depth in Eq. 2.2. The integrator calculates optical depth at each step as,

$$\tau_{m+1} = \tau_m + \frac{1}{6}k_1 + \frac{1}{3}k_2 + \frac{1}{3}k_3 + \frac{1}{6}k_4, \tag{5.23}$$

where $m = 0, 1, 2, 3, \ldots$ is an index of the steps taken for which $\tau_{m=0} = 0$, and k_1, k_2, k_3, and k_4 are increments defined as,

$$k_1 \equiv s\left(\frac{d\tau}{dz}\right)_{z_m,\tau_m}, \tag{5.24}$$

$$k_2 \equiv s\left(\frac{d\tau}{dz}\right)_{z_m+h/2,\tau_m+k_1/2}, \tag{5.25}$$

$$k_3 \equiv s\left(\frac{d\tau}{dz}\right)_{z_m+h/2,\tau_m+k_2/2}, \tag{5.26}$$

$$k_4 \equiv s\left(\frac{d\tau}{dz}\right)_{z_m+h,\tau_m+k_3}, \tag{5.27}$$

where z_m is the physical z location at index m and s is the step size. In the adaptive Runge–Kutta method, $\delta\tau \equiv \tau_{m+1} - \tau_m$ is calculated at each step and s is altered depending on some defined accuracy parameter a_*. If $\delta\tau > a_*$, the integrator discards τ_{m+1} and the step is repeated from τ_m with a smaller s, and if $\delta\tau \ll a_*$, s is increased to save computational time.

By default, `FluxCal` takes the integration bounds as from the maximum z value $z_{\max}$ for which fluid exists ($\tau = 0$) along the line of sight, to the minimum z value $z_{\min}$ for which fluid exists. The integrator stops before reaching $z_{\min}$ if the accumulated $\tau \geq \tau_{\text{thick}}$, where τ_{thick} is a user–controlled quantity, or if an optically thick particle's kernel has been entered.

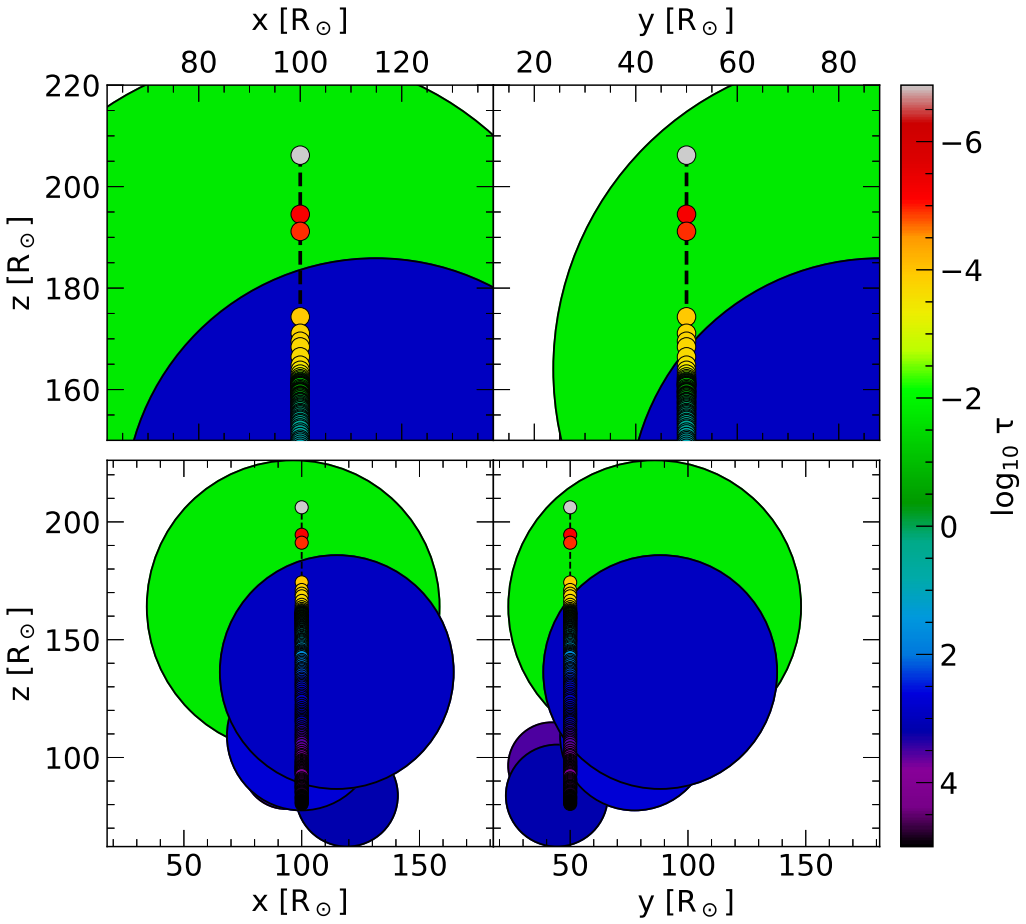

Figure 5.4: We show an example of the steps taken by the Runge–Kutta adaptive step size integrator (colored points), as discussed in §5.4, on an integration ray (black dashed line) through the data from the `StarSmasher` dynamical simulation shown in Fig. 6.11 and Fig. 6.12 and discussed in §6.3. Particles (shown as circles) have radii equal to twice their smoothing lengths h_i and are shaded according to their optical depths calculated by Eq. 5.16 with $C_\tau \approx 0.36$ (Wendland C^4 kernel). Points of entry into each particle kernel are not visible here, nor for any 2D projection as a consequence of creating the projection. Here, we do not use the envelope fitting routine described in §5.3 and the definition of optically thick is arbitrarily set to $\tau_{\rm thick} = 1 \times 10^5$ to show the behavior of the $4^{\rm th}$ order Runge–Kutta adaptive step size integrator.

5.4.1 The Integrating Grid

Although an intuitive solution to obtaining physical values for the Runge–Kutta integrator is to use the SPH method at each step, it is computationally more efficient to create a grid of values and calculate a distance weighted average between the two nearest vertices. From here forward, we call this grid the "integrating grid". Let H_z be the z spacing between integrating grid vertices along the line of sight such that,

$$H_z \equiv \frac{z_{\max} - z_{\min}}{N_z - 1}, \tag{5.28}$$

where $z_{\max}$ and $z_{\min}$ are the maximum and minimum z values for which fluid exists respectively and N_z is the number of grid vertices, equal to 416 by default in `FluxCal` for all lines of sight. We calculate the densities and temperatures for all integrating grid vertex indices K in the usual SPH way (see §3.1),

$$\rho(K) = \sum_j m_j W_j, \tag{5.29}$$

$$T(K) = \sum_j \frac{m_j}{\rho(K)} T_j W_j, \tag{5.30}$$

where W_j is the kernel function for particle j evaluated at K. We take the density term $\rho(K)$ in the summation interpolant Eq. 5.30 as a constant for all particles j by a symmetry argument.

For any physical location $z_{\min} \leq z \leq z_{\max}$, we define the real–valued index location $\mathcal{K}$ as,

$$\mathcal{K} \equiv \frac{z - z_{\min}}{H_z}. \tag{5.31}$$

We calculate the values of $\rho(\mathcal{K})$ and $T(\mathcal{K})$ as a distance weighted average,

$$\rho(\mathcal{K}) = \rho(\lfloor \mathcal{K} \rfloor)\big(\lfloor \mathcal{K} \rfloor + 1 - \mathcal{K}\big) + \rho(\lfloor \mathcal{K} \rfloor + 1)\big(\mathcal{K} - \lfloor \mathcal{K} \rfloor\big), \tag{5.32}$$

$$T(\mathcal{K}) = T(\lfloor \mathcal{K} \rfloor)\big(\lfloor \mathcal{K} \rfloor + 1 - \mathcal{K}\big) + T(\lfloor \mathcal{K} \rfloor + 1)\big(\mathcal{K} - \lfloor \mathcal{K} \rfloor\big), \tag{5.33}$$

where $\lfloor \mathcal{K} \rfloor$ is the floor value of $\mathcal{K}$. `FluxCal` uses both Eq. 5.32 and Eq. 5.33 to calculate the optical depth in §5.5 and the effective temperature in §5.7 along the line of sight.

5.4.2 Empty Space

When physical values are calculated using a distance weighted average, it is possible to obtain non–zero density values when outside SPH particle kernels. For example, consider a pair of integrating grid vertices K and $K+1$ where K has one intersecting kernel due to particle i and $K+1$ has none. Let there be some location,

$$\lfloor \mathcal{K} \rfloor < \mathcal{K}_i^* < \lfloor \mathcal{K} \rfloor + 1, \tag{5.34}$$

where $\lfloor \mathcal{K} \rfloor$ is the real–valued index location of K, $\mathcal{K}_i^*$ is the boundary of the kernel for particle i, and $\lfloor \mathcal{K} \rfloor + 1$ is the location of $K+1$. In the region $\mathcal{K}_i^* < \lfloor \mathcal{K} \rfloor + 1$, the distance weighted average density (Eq. 5.32) $\rho(\mathcal{K}) > 0$ when it should be equal to zero, as $\mathcal{K}_i^* < \lfloor \mathcal{K} \rfloor + 1$ is empty space. We show this situation with a sketch in Fig. 5.5.

To maintain a density of zero in empty space, we write the density $\rho(\mathcal{K})$

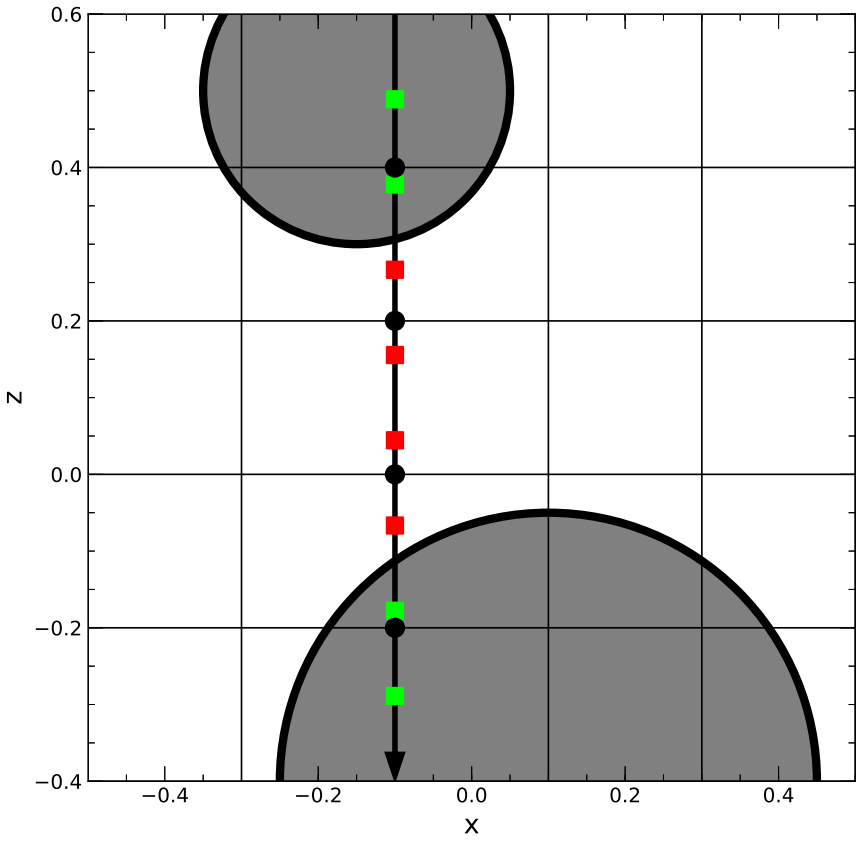

Figure 5.5: We show a sketch of the Runge–Kutta integrator exiting a particle kernel into empty space before re–entering another kernel. The integrator moves along the line of sight (black arrow) into the integration grid (black lines) taking incremental steps (green and red squares). We show the steps in which the integrator is not in empty space as green squares, and red squares where it is. Physical values are calculated by `FluxCal` as the distance weighted average of the nearest integration grid cell values (filled black circles). The integrator is located in empty space ($\rho = 0$) at the red squares, and in non–empty space ($\rho > 0$) at the green squares. The plotting bounds are arbitrary, the particles shown are not real SPH particles, the integration grid was not taken from `FluxCal` output, and the integration steps shown are not indicative of the real Runge–Kutta integrator behavior.

along line of sight as,

$$\rho(\mathcal{K}) = \begin{cases} m_i W_i, & \text{if } |\mathbf{r}_\mathcal{K} - \mathbf{r}_i| < 2h_i \text{ and either } \begin{pmatrix} \rho(\lfloor\mathcal{K}\rfloor) = 0 \\ \rho(\lfloor\mathcal{K}\rfloor + 1) > 0 \\ \text{or} \\ \rho(\lfloor\mathcal{K}\rfloor) > 0 \\ \rho(\lfloor\mathcal{K}\rfloor + 1) = 0 \end{pmatrix}, \\ 0, & \text{if } |\mathbf{r}_\mathcal{K} - \mathbf{r}_i| \geq 2h_i \text{ and either } \begin{pmatrix} \rho(\lfloor\mathcal{K}\rfloor) = 0 \\ \rho(\lfloor\mathcal{K}\rfloor + 1) > 0 \\ \text{or} \\ \rho(\lfloor\mathcal{K}\rfloor) > 0 \\ \rho(\lfloor\mathcal{K}\rfloor + 1) = 0 \end{pmatrix}, \\ \rho(\mathcal{K}), & \text{otherwise (see Eq. 5.32),} \end{cases} \tag{5.35}$$

where $\mathbf{r}_\mathcal{K}$ is the physical position at $\mathcal{K}$ and $\mathbf{r}_i$ is the physical location of the most–recently–entered particle. We calculate the temperature similar to the density, except the top–most case in Eq. 5.35 has the quantity $m_i T_i W_i / \rho(\mathcal{K})$, as in Eq. 5.30.

5.5 Optical Depth Calculation

Using the values of $\rho(\mathcal{K})$ and $T(\mathcal{K})$ from Eq. 5.32 and Eq. 5.33 respectively, `FluxCal` calculates the mean opacity $\bar{\kappa}(\mathcal{K})$ using the table of ρ, T, $\bar{\kappa}$ values described in §5.2. The frequency–integrated optical depth can be written by integrating both sides of Eq. 2.2 from $\tau = 0$ to τ,

$$\tau = \int_{z_{\min}}^{z_{\max}} \bar{\kappa}(\mathcal{K}) \rho(\mathcal{K}) dz, \tag{5.36}$$

where $\bar{\kappa}$ is the mean opacity (see §2.6). `FluxCal` uses the Runge–Kutta method

described in §5.4 to perform this integral and the result is used later in calculating the effective temperature along the line of sight in §5.7.

5.6 Visible Luminosity Calculation

We first assume the SPH simulation radiates isotropically from its surface. Hence, we calculate the visible luminosity using Eq. 2.30, written here again as,

$$\mathscr{L}(\xi,\zeta) = \oint_{Q_{\rm ph}(\xi,\zeta)} T_{\rm eff}(\mathbf{r})^4 dq(\mathbf{r}_{\rm ph}). \tag{5.37}$$

The effective temperature $T_{\rm eff}(\mathbf{r})$ must be calculated at all visible $dq(\mathbf{r}_{\rm ph})$ to evaluate $\mathscr{L}(\xi,\zeta)$. `FluxCal` accomplishes this by applying both the optically thick and thin methods for calculating temperatures to the Runge–Kutta integrator, a process we detail in §5.7.

5.7 Effective Temperature Calculation

Recall from §2.4 the effective temperature $T_{\rm eff}$ is the temperature of a blackbody with the same outgoing energy per time as from the surface of a given radiating gas cloud. The effective temperature along a line of sight can be calculated using Eq. 2.25 as,

$$T_{\rm eff}^4 = \frac{\pi}{\sigma_{\rm SB}} I^+(\tau = 0). \tag{5.38}$$

We assume that both the outgoing radiation and the source function are isotropic for all τ.

As we described in §5.5, the optical depth is calculated as a function of dz, and thus has dependency on μ. For a far away observer, all rays emitted from a gas cloud are parallel with each other and the line of sight, so $\mu = 1$ for all rays. With this in mind, the frequency–integrated solution to the radiative transfer

equation in Eq. 2.10 can be written as,

$$I(\tau) = I_0(\tau_0)e^{\tau-\tau_0} + \int_{\tau}^{\tau_0} S(\tau')e^{-\tau'}d\tau'. \tag{5.39}$$

Thus the outgoing intensity at the $\tau = 0$ surface is written as,

$$I^+(\tau = 0) = I_0^+(\tau)e^{-\tau} + \int_0^{\tau} S(\tau')e^{-\tau'}d\tau', \tag{5.40}$$

where we have written τ_0 as τ after setting τ to zero. Here, $\tau = 0$ is the surface location and τ is the accumulated optical depth calculated by the Runge–Kutta integrator in §5.4. We calculate the originating outgoing intensity $I_0^+(\tau)$ as,

$$I_0^+(\tau) = \frac{\sigma_{\rm SB}}{\pi} \begin{cases} T_{\rm ph}^4, & \text{thick particle} \\ 0, & \text{otherwise} \end{cases}, \tag{5.41}$$

where $T_{\rm ph}$ is calculated using the envelope fitting method in §5.3.2 only if an optically thick particle (see §5.5) exists along the line of sight.

We calculate the source function $S(\tau')$ only for optically thin particles as

$$S(\tau') = \frac{\sigma_{\rm SB}}{\pi} T(\tau')^4. \tag{5.42}$$

Thus, combining Eq. 5.40, Eq. 5.41, and Eq. 5.42 with Eq. 5.38 yields the effective temperature as,

$$T_{\rm eff} = \begin{cases} T_{\rm ph}, & \text{only optically thick} \\ \left[\int_0^{\tau} T(\tau')^4 e^{-\tau'}d\tau'\right]^{1/4}, & \text{only optically thin} \\ \left[T_{\rm ph}^4 e^{-\tau} + \int_0^{\tau} T(\tau')^4 e^{-\tau'}d\tau'\right]^{1/4}, & \text{optically thick \& thin} \\ 0, & \text{no particles on line of sight} \end{cases}. \tag{5.43}$$

From top–to–bottom, the conditions apply to the z line of sight when (1) the

first particle visible is optically thick, (2) there are optically thin particles but no optically thick particles, (3) there are optically thin particles in front of an optically thick particle, and (4) there are no particles.

5.8 `FluxCal` Framework

The user provides data files containing physical times and particle: positions, masses, smoothing lengths, densities, velocities, internal energies, mean molecular weights, and local gravitational accelerations. `FluxCal` calculates particle central temperatures and pressures using either the high resolution TEOS or the polynomial solution as described in §4.3 from `StarSmasher`. To use `FluxCal` with other codes, a TEOS with a similar resolution should be provided to maintain precision in the bi–linear interpolation method. For best possible results, the TEOS used in the SPH simulation and the TEOS used in `FluxCal` should be at least similar so that `FluxCal` performs calculations in a similar physical framework.

`FluxCal` rotates the particles to the desired viewing angle (ξ, ζ), as seen in Fig. 2.3. It then defines a 3D coordinate system with the same origin as in the data set, but with the positive z direction pointed toward an observer who is infinitely far away, as shown by the red, blue, and green arrows in Fig. 5.6. `FluxCal` then creates a 2D grid, called the "driving grid" with an arbitrary initial resolution of 3×3 on the xy plane with a z location large enough to encapsulate all fluid. The driving grid has a surface area equal to the largest possible cross–section with the line of sight and hence encapsulates all particles and their kernels on the xy plane. For each vertex on the driving grid, `FluxCal` constructs the integrating grid as described in §5.4.1 and integration is performed along the negative z direction, as discussed in §5.4. In summary, the integrator begins at the surface of the first visible particle kernel, travels iteratively in the negative z direction, and terminates when any of the following

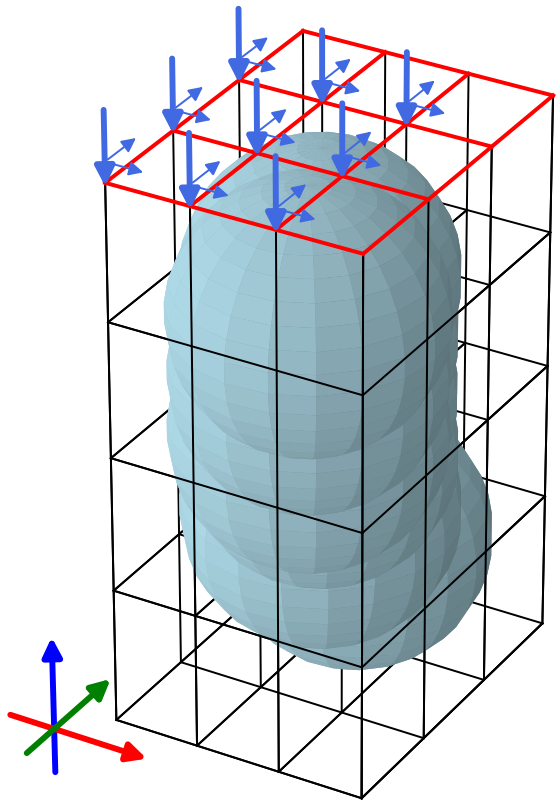

Figure 5.6: We show a sketch of the driving and integrating grids used by `FluxCal` as described in §5.8 for a set of particles (gray spheres). The driving grid (shown in red) is initialized with an arbitrary resolution of 3×3 on the xy plane and the integrating grid (shown in black) is constructed on the xyz plane. The integrating grid here has a uniform number of cells for visual clarity and do not start at $z_{\max}$ or end at $z_{\min}$ (see §5.4.1), which is not representative of the actual `FluxCal` code. We show the orientation of the coordinate axes in the bottom left with the standard corresponding colors red, green, and blue for x, y, and z respectively. We indicate integration in the $-z$ direction with thick light blue arrows. The value of T_{eff} at a vertex fills a corresponding cell area, which we show with thin light blue arrows extending perpendicular to the thick light blue arrows. That is, T_{eff} is constant and equal to the vertex value across one entire driving grid cell. The grids we show here have very low resolutions and are not representative of typical `FluxCal` resolutions.

conditions are met:

- The accumulated optical depth from integrating through optically thin particles along the line of sight constitutes an optically thick region. `FluxCal` calculates the effective temperature $T_{\rm eff}$ using a distance weighted average of nearby particles, as described in §5.4.1.
- The surface of an optically thick particle kernel has been reached. `FluxCal` calculates the value of $T_{\rm eff}$ using both envelope fitting and ray tracing.
- There are no particle kernels in the negative z direction. `FluxCal` takes the value of $T_{\rm eff}$ to be zero.

`FluxCal` calculates the visible luminosity $\mathscr{L}(\xi,\zeta)$ (see §5.6) from the viewing angle (ξ,ζ) under the assumption that the outgoing radiation is isotropic,

$$\mathscr{L}(\xi,\zeta) = A_{\rm cell}\sigma_{\rm SB}\sum_{\rm cells} T^4_{\rm eff,cell}, \tag{5.44}$$

where $A_{\rm cell}$ is the area of each driving grid cell and $T_{\rm eff}$ is the effective temperature, calculated by the method in §5.7.

To ensure some level of precision in $\mathscr{L}(\xi,\zeta)$, `FluxCal` calculates the visible flux from the driving grid and compares it to the previously calculated value. `FluxCal` calculates the visible flux as,

$$\mathscr{F}(\xi,\zeta) = \sigma_{\rm SB}\sum_{\rm cells} T^4_{\rm eff,cell}, \tag{5.45}$$

where the summation is performed over all driving grid cells. `FluxCal` incrementally increases the driving grid resolution if the following condition is met,

$$\mathscr{F}_n(\xi,\zeta) - \mathscr{F}_{n-1}(\xi,\zeta) > p^*\mathscr{F}_n(\xi,\zeta), \tag{5.46}$$

where subscript n denotes the total number of iterations for which $\mathscr{F}(\xi,\zeta)$ has been calculated for a given time–slice and p^* is a quantity that describes the

precision of $\mathscr{F}(\xi, \zeta)$ and is set by the user (called "`fracaccuracy`" in the code). In this way, `FluxCal` achieves numerical convergence in the visible flux, which is a quantity that depends on both $\mathscr{L}(\xi, \zeta)$ and A_{cell}, as in Eq. 5.44. Hence, the user defines the level of desired precision in $\mathscr{L}(\xi, \zeta)$ by setting `fracaccuracy`, and that precision is achieved once the condition in Eq. 5.46 is met. For all our `FluxCal` calculations performed in this work, we use `fracaccuracy` $= 0.01$.

5.9 Other Tools

Spectral physics in SPH codes are often difficult to debug as there are many dependencies that come into play. To help tackle this, `FluxCal` has several useful debugging tools which operate separately from the main $\mathscr{L}(\xi, \zeta)$ calculator.

For example, it is often useful to know the properties of the particles nearest to the observer along the line of sight. As in Eq. 5.39, optically thin particles attenuate intensity from optically thick regions differently for each driving grid vertex. `FluxCal` can find the properties of particles closest to the observer at each driving grid vertex to give deeper insight on the behavior of outer atmospheres. Particles with kernels that span more than one driving grid cell may produce duplicate entries. `FluxCal` removes such duplicate entries in the final output.

`FluxCal` can also give the integration results from any of the rays traced through the driving grid vertices, or through any desired location without a grid. This is particularly useful for understanding how any given line of sight in the simulation affects the value of $\mathscr{L}(\xi, \zeta)$, or for identifying regions of interest, such as grid vertices that have relatively high or low effective temperatures.

`FluxCal` is able to obtain all available information on any number of particles over their spatiotemporal evolution through the simulation. This is very desirable for analyzing contact binary merger simulations, as particles that contribute to the outflow can be tracked and their origins identified.

A typical problem that arises when there are optically thick particles in an SPH simulation is having a single, large, optically thick particle taking up most of the view during some time after the start of the simulation. Using `FluxCal`, it is straightforward to find that particle's index and quickly trace its evolution back to the start of the simulation to verify if its evolution is physical. This may show, for example, that it originated from a dense, hot region and was artificially kicked out by a numerical instability.

Chapter 6

Results

6.1 `StarSmasher` Stellar Models

We model a primary star using three particle resolutions $N = 1{\times}10^5$, $N = 2{\times}10^5$, and $N = 3{\times}10^5$ for a `MESA` model with $M = 1.52\ M_\odot$ and $R = 3.715\ R_\odot$ evolved to an age of 2.54 Gyr. Our `MESA` model has a central density of $\rho_c^* \approx 6 \times 10^4$ g cm^{-3} and a core mass of 0.189 $M_\odot$. We use `MESA` version 9793 with the inlist provided in Appendix A.2. This star is near the base of the red giant branch on its evolution and is intended to approximate the V1309 Sco progenitor primary (Stępień, 2011).

The models each have a core mass m_c that depends on N and `nnopt`, as seen in Table 6.1, as the core particle is simply assigned the difference between the `MESA` total mass and the total mass of the other particles, as discussed in §4.5.1. We show the initial $N = 3 \times 10^5$ model in Fig. 6.1 and the final relaxed $N = 1 \times 10^5$, $N = 2 \times 10^5$, and $N = 3 \times 10^5$ models in Fig. 6.2, Fig. 6.3, and Fig. 6.4. Each model has the optimal `nnopt` value for its resolution, as discussed in §4.5.2. In all figures, quantities with a leading δ_m are the relative fractional errors by mass between the `StarSmasher` model and the `MESA` model. For example,

$$\delta_m \rho_i = \frac{\rho_{\texttt{MESA}}(M_i) - \rho_i}{\rho_{\texttt{MESA}}(r_i)}, \tag{6.1}$$

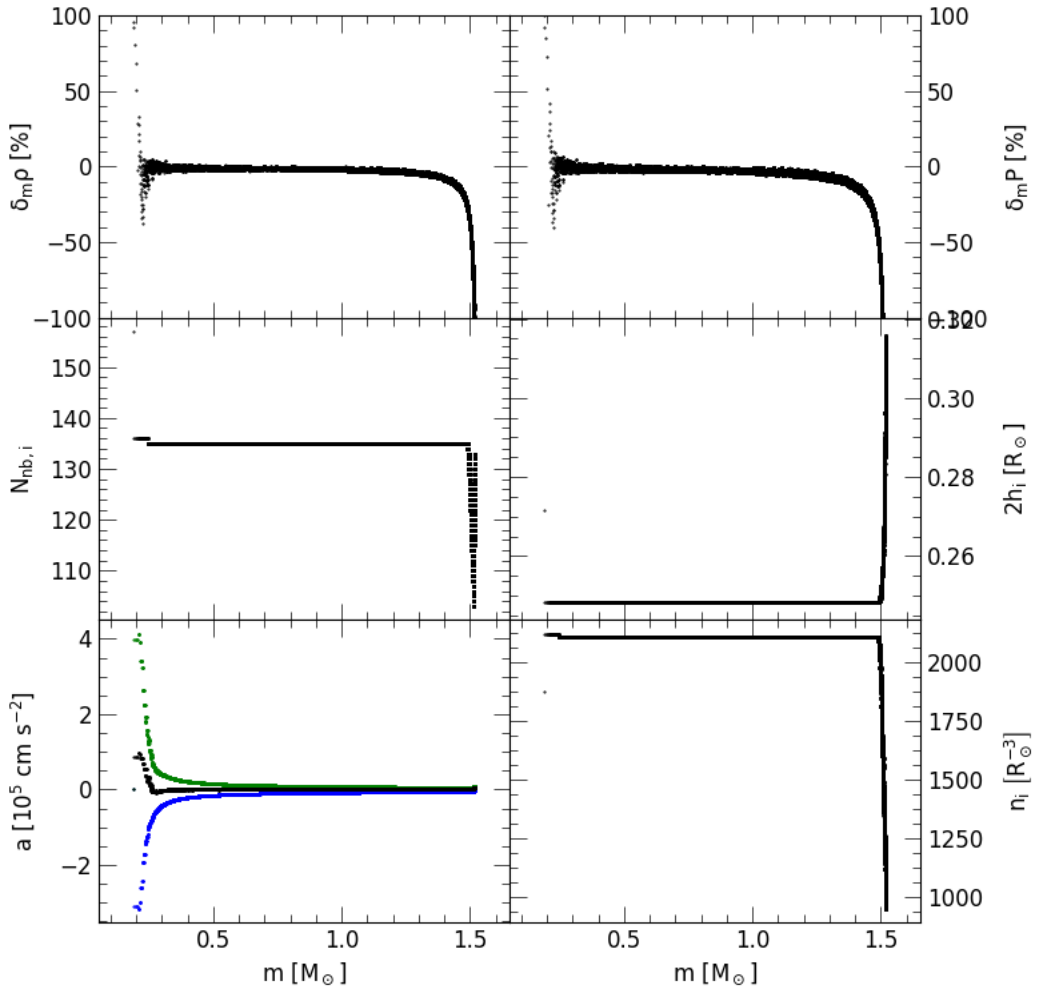

Figure 6.1: Initial profile of the `StarSmasher` $N = 3 \times 10^5$ model before relaxation. We show the relative fractional errors by mass between the `MESA` and `StarSmasher` models for the density $\delta_m \rho$ (top left) and pressure $\delta_m P$ (top right). Also we show the number of neighbors $N_{nb,i}$ (middle left), particle radii $2h_i$ (middle right), particle acceleration components (bottom left), and local spatial resolutions n_i (bottom right). The acceleration components of gravity (green) and hydrodynamics (blue) form the net acceleration (black). We omit from this image values of $\delta_m \rho$ and $\delta_m P$ that represent errors larger than 100%.

where $\delta_m \rho_i$ is the relative fractional error in the density at the center of particle i located at mass coordinate M_i, ρ_i is the density of particle i, and $\rho_{\texttt{MESA}}(M_i)$ is the density of the shell nearest to r_i in the `MESA` model. We calculate the mass coordinate for each particle using a numerical sum over all particles from the core to the surface,

$$M_i = \sum_j^i m_j, \tag{6.2}$$

where m_j is the mass of particle j.

We calculate the local spatial resolution at each particle using Eq. 3.26 with smoothing lengths of $2h_i$. At both the beginning and the end of each relaxation, the spatial resolution varies by approximately an order of magnitude in the top $\sim 8\%$ by mass. Resolution is the worst for the outermost particles and improves moving inward through the envelope. In all our models, both the worst and best spatial resolutions in the entire envelope are achieved in the top $\sim 8\%$ of mass. This variance depends on the smoothing lengths h_i exponentially and on the neighbor number $N_{nb,i}$ linearly. In Fig. 6.2, Fig. 6.3, and Fig. 6.4, there is a clear change in n_i near the core as a near direct result of $N_{nb,i}$, as h_i is approximately constant there.

The initial pressure and density profile of the $N = 3 \times 10^5$ model, as seen in Fig. 6.1, matches to within $\sim 5\%$ through most of the envelope until the surface region at $m \gtrsim 1.45\ M_\odot$. There, particle radii $2h_i$ are significantly larger than that of particles deeper within the envelope. This is a result of the $N_{nb,i}$ criterion described in §4.1 and occurs in the top $\sim 0.8\%$ of the envelope by mass, compromising the spatial resolution. We find similar results for the $N = 1 \times 10^5$ and $N = 2 \times 10^5$ models, as seen in Fig. 6.2 and Fig. 6.3 respectively, though the surfaces are not as well modelled as in the $N = 3 \times 10^5$ case.

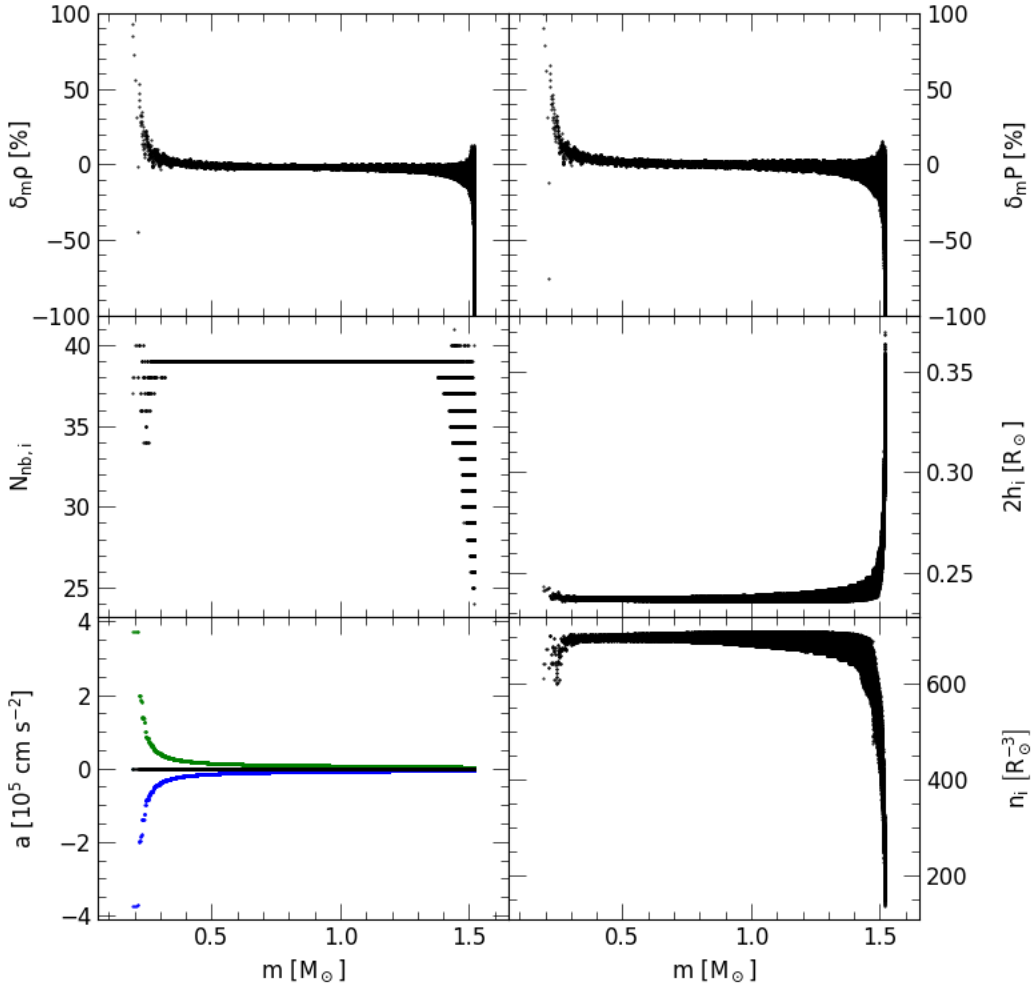

Figure 6.2: The same plot as in Fig. 6.1 but for the fully relaxed $N = 1 \times 10^5$ model after ~ 18 days. Values of $\delta_m \rho$ and $\delta_m P$ that represent errors larger than 100% are not shown.

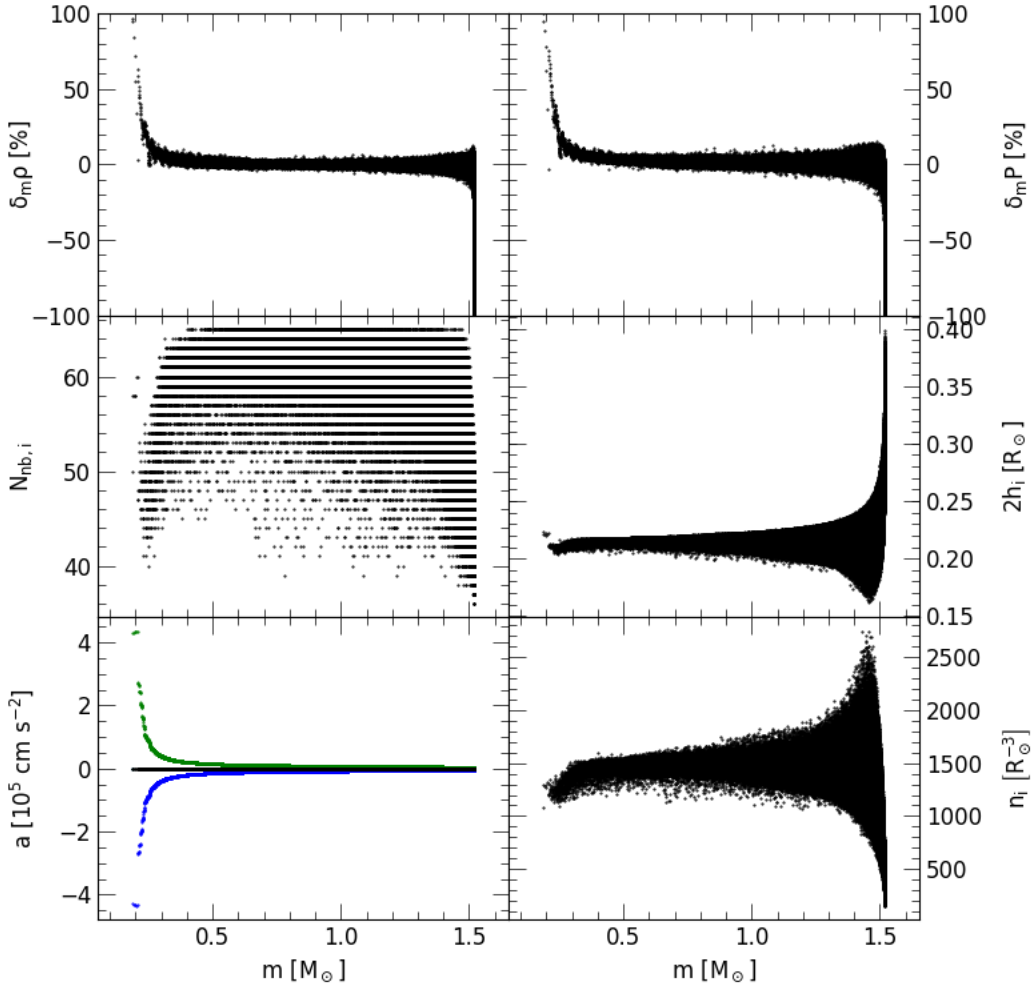

Figure 6.3: The same plot as in Fig. 6.1 but for the fully relaxed $N = 2 \times 10^5$ model after ~ 18 days. Values of $\delta_m \rho$ and $\delta_m P$ that represent errors larger than 100% are not shown.

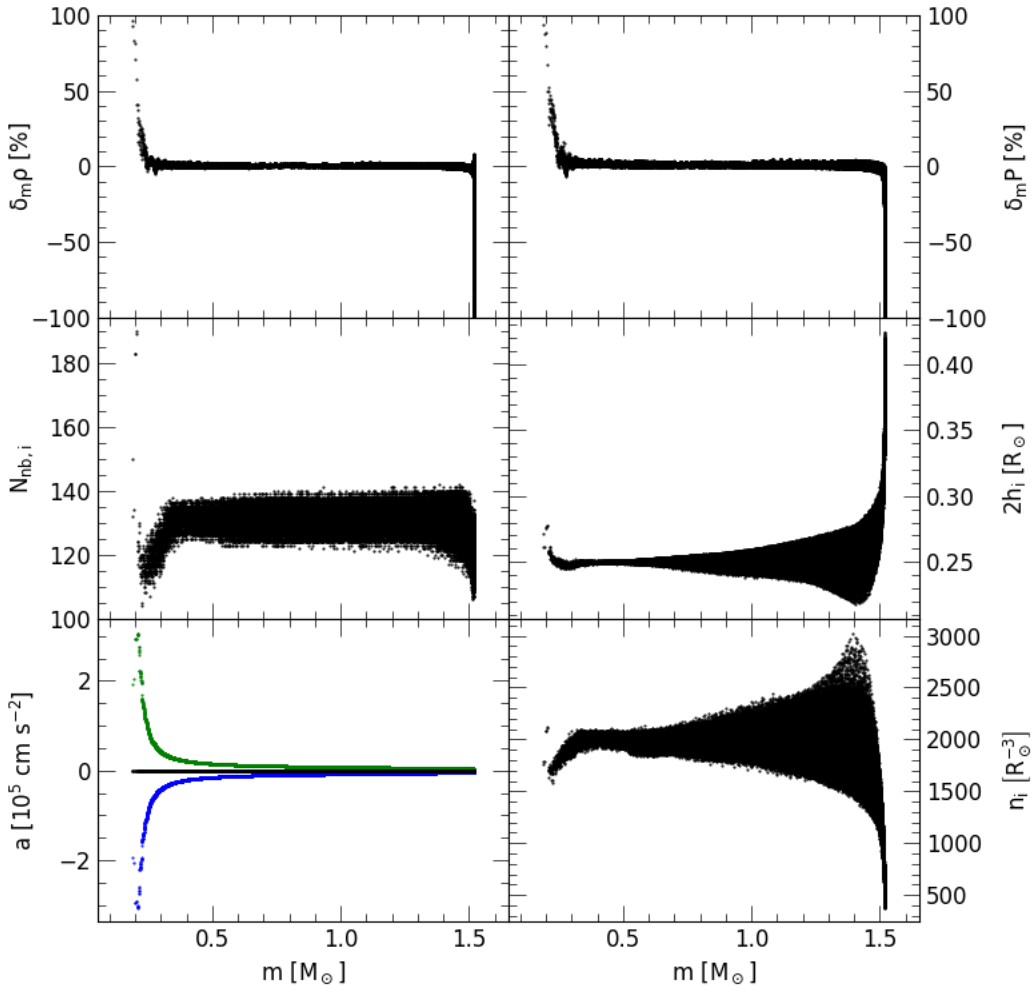

Figure 6.4: The same plot as in Fig. 6.1 but for the fully relaxed $N = 3 \times 10^5$ model after ~ 18 days. Values of $\delta_m \rho$ and $\delta_m P$ that represent errors larger than 100% are not shown.

6.1.1 Finding the Optimal `nnopt`

As discussed in §4.5.2, there exists an `nnopt` value which allows the total energy profile of any given `StarSmasher` relaxation to best match the initial `MESA` model. We first create the $N = 1 \times 10^5$, $N = 2 \times 10^5$, and $N = 3 \times 10^5$ models with `nnopt` $= 10$ and calculate the relative fractional error in the total energy profiles by mass $\delta_m E$ as,

$$\delta_m E(m) = \frac{E_m(m) - E_{\rm SPH}(m)}{E_m(m)}, \tag{6.3}$$

where m is the mass coordinate, $E_m(m)$ is the integrated total energy from the surface inwards at m in the `MESA` model, and $E_{\rm SPH}(m)$ is the integrated total energy at m in the SPH model. We analyze the $\delta_m E$ profile in the outer 0.1 $M_\odot$, shown in Fig. 6.5, by eye to determine goodness of fit. This is sufficient, as profiles typically vary by only $\sim 1\%$ in this mass range. In what follows, we obtain only a close approximation to the best possible `nnopt` value, to within a few. Choosing an `nnopt` that differs by a few to what is found here results in a difference in $\delta_m E$ by less than 1%. In situations where small changes in `nnopt` give large changes in the $\delta_m E$ profile, the following method can still be used, but a numerical metric describing the goodness of fit should be employed, such as a least–squares method or χ^2 value.

For the given N, we relax additional `StarSmasher` models with `nnopt` values increasing in increments of 10 until the $\delta_m E$ profile has a slope ~ 0 and values approaching 0. Generally, with increasing `nnopt`, the slope becomes more negative and values become more positive. Once we find an `nnopt` for which the slope has become negative and values are close to zero throughout, we explore ever smaller perturbations in `nnopt` around that value to find the δE profile that has the values and slope closest to zero. We show in Fig. 6.5 and Table 6.1 the optimal `nnopt` values for the $N = 1 \times 10^5$, $N = 2 \times 10^5$, and $N = 3 \times 10^5$ models as 20, 30, and 87 respectively.

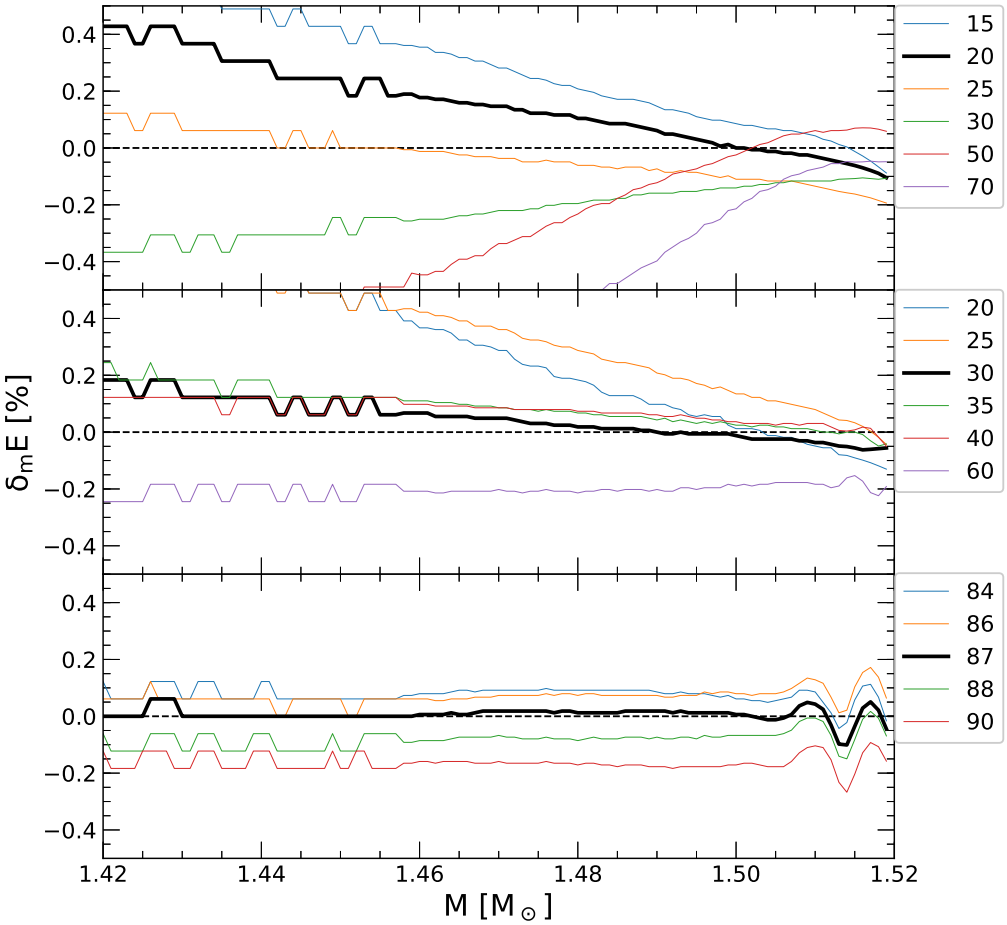

Figure 6.5: Relative fractional errors in total energy δE between the fully relaxed `StarSmasher` $N = 1\times10^5$ (top), $N = 2\times10^5$ (middle), and $N = 3\times10^5$ (bottom) models from Table 6.1, normalized to mass coordinate $M = 1.44\ M_\odot$. We find the optimal `nnopt` values (black) by the method described in §6.1.1 for all three models as 20, 30, and 87 respectively.

N	nnopt	$\mathbf{m_c}$	**R**	$\langle \mathbf{T_{eff}} \rangle$	$\langle \tau_\mathbf{s} \rangle$	$\mathbf{4\mathscr{L}}$	**L**	$\langle \mathbf{T_{eff,RK}} \rangle$	$\langle \mathbf{2h_s} \rangle$
–	–	$[M_\odot]$	$[R_\odot]$	[K]	$[10^{10}]$	$[L_\odot]$	$[L_\odot]$	$[10^5$ K]	$[R_\odot]$
MESA	–	0.189	3.715	4973	–	–	7.59	–	–
1×10^5	20	0.191	3.947	5252	2.08	10.17	10.66	1.363	0.336
2×10^5	30	0.186	3.982	5288	1.59	10.58	11.14	1.121	0.324
3×10^5	87	0.188	3.991	5448	0.62	12.37	12.62	1.284	0.374

Table 6.1: Fully relaxed StarSmasher models for varying particle resolutions at optimal nnopt (see §6.1). Shown are the core masses m_c, radii R ($\tau = 0$ surface), average effective temperatures $\langle T_{\rm eff} \rangle$ found by the method described in §5.7, average optical depths of the particles closest to the observer $\langle \tau_s \rangle$, four times the visible luminosities $4\mathscr{L}(\xi, \zeta)$, and total luminosities $L = 4\pi\sigma_{\rm SB} R^2 \langle T_{\rm eff} \rangle^4$. Also shown are the average effective temperatures $\langle T_{\rm eff,RK} \rangle$ calculated by the Runge–Kutta integrator only (see §5.4) for optical thickness $\tau_{\rm thick} = 10$ and the average surface particle radii $\langle 2h_s \rangle$, as defined in §6.2.2. Visible luminosities are multiplied by a factor of 4 to approximate the total luminosities L in the assumption that the star is spherically symmetric, as discussed in §2.4.1.

6.2 Relaxations

We process the three StarSmasher relaxations from §6.1 with FluxCal to find values of $T_{\rm eff}$ both with and without the envelope fitting method described in §5.3.2. Our results from the envelope fitting turned off represents the expected outcome for traditional ray tracing algorithms. We calculate the average optical depths of the outermost particles in Table 6.1 to describe the behavior of FluxCal. Finally, we calculate the visible and theoretical luminosities and compare their values. Our methods for calculating these quantities are hereby detailed.

6.2.1 Surface Optical Depths

We calculate the average surface particle optical depth as,

$$\langle \tau_s \rangle \equiv \frac{1}{N_s} \sum_i^{N_s} \tau_i, \tag{6.4}$$

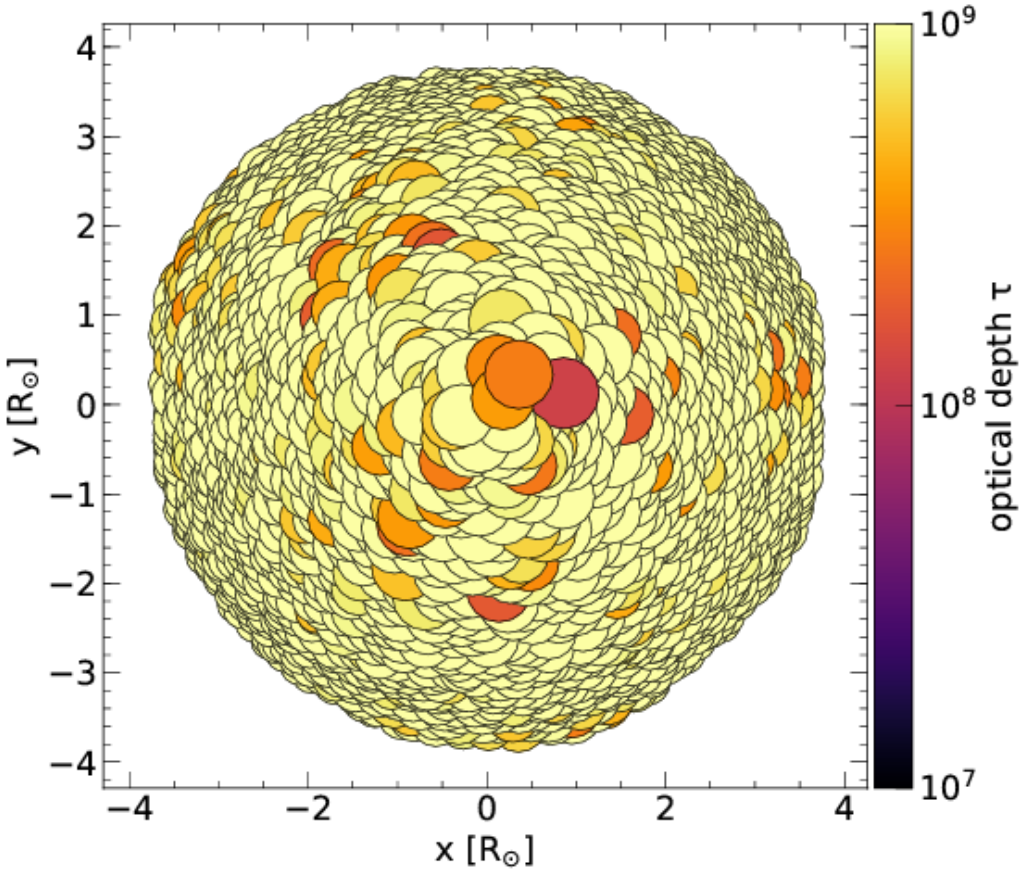

Figure 6.6: The optical depths of the first visible particles along driving grid vertices for our $N = 3 \times 10^5$ model in Table 6.1, as calculated with Eq. 5.16. The driving grid has a resolution of 63×93 cells from $(x, y) \approx (-3.94, -3.95)$ to $(x, y) \approx (3.94, 3.96)$. We show particles as filled circles with radii equal to twice their smoothing lengths, in order of distance to the observer along the z axis.

where N_s is the total number of particles with kernels that are visible to the observer and τ_i is the optical depth of particle i, as calculated by Eq. 5.16. We find $\langle \tau_s \rangle$ decreases with increasing N, which is consistent with better spatial resolutions at the surface, as described in §3.3 and §4.5.2. Our results in Fig. 6.6 show the spatial distribution of the surface particle optical depths in the $N = 3 \times 10^5$ particle model. We identify the surface particles by finding the first visible smoothing kernel to the observer for each driving grid vertex, as described in §5.8. We show the surface particles as cross sections with the xy plane in Fig. 6.6 in order of position on the z axis.

6.2.2 Model Radii

We calculate the radius at the $\tau = 0$ surface as,

$$R \equiv \max(|\mathbf{r}_i| + 2h_i), \tag{6.5}$$

where $|\mathbf{r}_i|$ is the position of particle i and $2h_i$ is its radius. As seen in Table 6.1, R increases with `nnopt`. The consequence of larger `nnopt` is larger smoothing lengths and, hence, larger radii R, as described in §5.1 and §4.5.2. We define the average surface particle radius $\langle 2h_s \rangle$ as,

$$\langle 2h_s \rangle \equiv \frac{2}{N_s} \sum_{i}^{N_s} h_i. \tag{6.6}$$

We present values for $\langle 2h_s \rangle$ in Table 6.1.

6.2.3 Effective Temperatures

We calculate effective temperatures for the $N = 1 \times 10^5$, $N = 2 \times 10^5$, and $N = 3 \times 10^5$ models using `FluxCal` by the process described in §5.7 and average them across the driving grid to find $\langle T_{\rm eff} \rangle$ in Table 6.1. The surface particles have very large optical depths, as seen in Table 6.1, Fig. 6.6, and discussed in

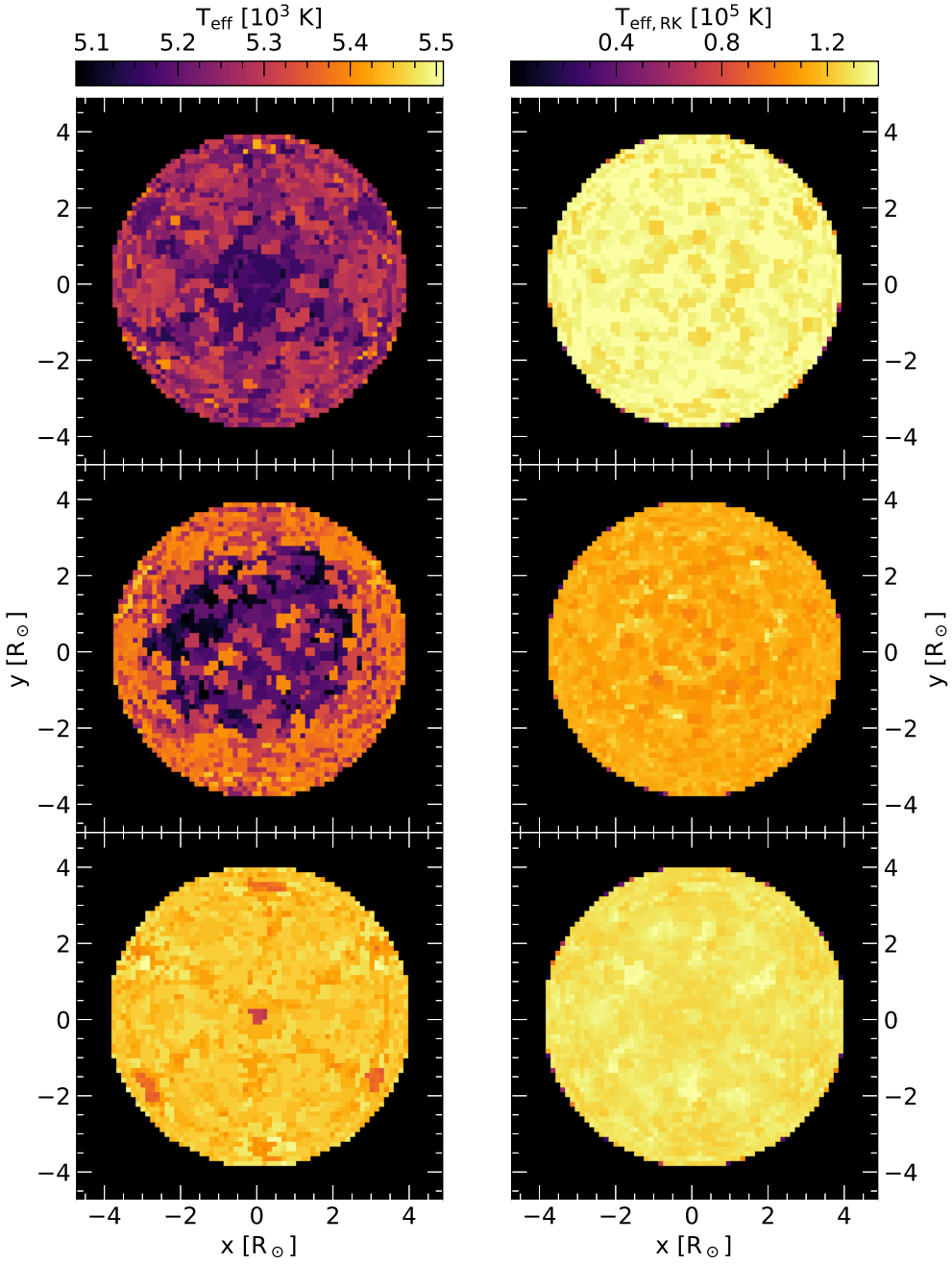

Figure 6.7: Effective temperatures calculated by `FluxCal` for our $N = 1 \times 10^5$ (top), $N = 2 \times 10^5$ (middle), and $N = 3 \times 10^5$ (bottom) models in Table 6.1. On the left side is the effective temperature calculated normally, while the right side uses only the Runge–Kutta integrator with optical thickness set to $\tau_{\text{thick}} = 10$.

§6.2.1, and thus the Runge–Kutta integrator, as described in §5.4, is not used by `FluxCal` for any of the driving grid vertices.

We turn off the envelope fitting routine and use only the Runge–Kutta integrator to calculate the effective temperatures $T_{\text{eff,RK}}$ with $\tau_{\text{thick}} = 10$. We average these values across the driving grid to find $\langle T_{\text{eff,RK}} \rangle$, which are ~ 2 orders of magnitude larger than $\langle T_{\text{eff}} \rangle$ in all three models.

We show the spatial distributions of $T_{\text{eff,RK}}$ and T_{eff} for the $N = 1 \times 10^5$, $N = 2 \times 10^5$, and $N = 3 \times 10^5$ models in Fig. 6.7 for the viewing angle $(\xi, \zeta) = (0, 0)$. A cool spot is present in the center of all three models, the largest of which is in the $N = 2 \times 10^5$ model, and appears like an opposite limb darkening effect. `FluxCal` cannot capture the physics of limb darkening, as the angle μ is kept constant and thus the origin of this feature is unclear. The feature may be a result of the initial hcp lattice discussed in §4.5 and insufficient particle mixing (see §7.1.2), but more work is required to draw any conclusions.

Our values of $\langle T_{\text{eff,RK}} \rangle$ do not scale monotonically with N, while $\langle T_{\text{eff}} \rangle$ do. `FluxCal` does not use the Runge–Kutta integrator to calculate $\langle T_{\text{eff}} \rangle$ as in Eq. 5.43 due to the high optical depths of the outermost particles ($\langle \tau_s \rangle \sim 10^{10}$). Thus, values of $\langle T_{\text{eff,RK}} \rangle$ are subject to the SPH photosurface problem discussed in §5.1. This is corroborated by the poor spatial resolution n_i near the surface of all models, seen in Fig. 6.2, Fig. 6.3, and Fig. 6.4. The envelope fitting solution is unaffected by the poor outer spatial resolution, as it calculates quantities using only central values, as opposed to the kernel–smoothed quantities used by the Runge–Kutta integrator (see §5.4).

6.3 Dynamical Simulation

We simulate V1309 Sco as a contact binary merger with `StarSmasher` by the process described in §4.4 with our $N = 2 \times 10^5$ particle model as the primary, a $M = 0.16\ M_\odot$, $R = 0.203\ R_\odot$, 2.54 Gyr, $N = 1 \times 10^4$ brown dwarf model

as the secondary (Stępień, 2011), with an initial orbital period of 1.44 days (Tylenda et al., 2011). We do this to show the behavior of `FluxCal` in a regime of both optically thick and optically thin particles and as an attempt to model the observations of V1309 Sco in §1.2.1. We ran the simulation for ≈ 4 months wall time with 16 NVIDIA P100 Pascal GPUs and 96 Intel E5-2683 v4 "Broadwell" CPUs on the ComputeCanada `Cedar` supercluster.

Outflow begins at $t = 0$ days until merger at $t \sim 5$ days and, as seen in Fig. 6.8, $m_{\rm eje} \sim 0.06\ M_\odot$ is ejected. From the particle positions in Fig. 6.9 and Fig. 6.10 as well as optical depths in Fig. 6.11 and Fig. 6.12, we find that no visible outflow originates from the secondary.

We show the energies associated with the merger in Fig. 6.8, which are calculated as,

$$E_{\rm int} = \sum_i m_i u_i \tag{6.7}$$

$$E_{\rm pot} = \frac{1}{2} \sum_i m_i \varphi_i \tag{6.8}$$

$$E_{\rm kin} = \frac{1}{2} \sum_i m_i |\mathbf{v}_i|^2, \tag{6.9}$$

$$E_{\rm SPH} = E_{\rm int} + E_{\rm pot} + E_{\rm kin} \tag{6.10}$$

$$E_{\rm tot} \equiv E_{\rm tot} + E_{\rm rad} \tag{6.11}$$

where u_i, φ_i, and $\mathbf{v}_i$ are the specific internal energies, specific gravitational potential energies, and velocities of particle i respectively, and $E_{\rm SPH}$ and $E_{\rm tot}$ are the total energy from SPH hydrodynamics and total energy – including radiative cooling $E_{\rm rad}$ – respectively. We calculate the total internal energy $E_{\rm int}$ by summing over all non–core particles, while the total potential and kinetic energies $E_{\rm pot}$ and $E_{\rm kin}$ are calculated by summing over all particles inclusive of core–particles. The average total energy of our simulation is $\langle E_{\rm tot} \rangle \approx -1.73 \times 10^{48}$ erg, the change in total energy over the entire evolution is $\Delta E_{\rm tot} \approx -1.68 \times 10^{45}$

erg, and the total energy radiated away is $\Delta E_{\rm rad} \approx 4.19 \times 10^{44}$ erg. We find $\Delta E_{\rm SPH} \approx -1.1 \times 10^{45}$ erg and a numerical precision of,

$$\left(1 - \left|\frac{\Delta E_{\rm SPH}}{E_{\rm SPH}(t=0)}\right|\right) \times 100 \approx 99.88\%. \tag{6.12}$$

We calculate the specific total energy of a single particle by summing over that particle's specific energies,

$$e_i \equiv \frac{1}{2}|\mathbf{v}_i|^2 + u_i + \frac{1}{2}\varphi_i. \tag{6.13}$$

The particles that have $e_i \geq 0$ are unbound from the system. Thus, the total energies of the unbound particles are found in the same way as Eq. 6.7 through Eq. 6.11, but summing only over particles with $e_i \geq 0$.

The energy radiated away by cooling processes $E_{\rm rad}$, as calculated by Eq. 4.24, increases with ejected mass $m_{\rm eje}$ and continues to increase at a constant rate some time after mass ejection ceases.

We show the locations of all particles in the dynamical run at $t \approx 14.25$ days in Fig. 6.17. Particles with low densities fall out of the range of the TEOS at $t = 12.375$ days, many of which are the outermost particles in the outflow. Thus, the outermost particles at late times have temperatures calculated by the polynomial solution in §4.3.1, not the TEOS.

6.4 Light Curve

As seen in Fig. 6.13 and Fig. 6.14, the shape of the outflow is approximately symmetric in the ζ direction. Thus, we calculate visible luminosities using `FluxCal` by the process described in §5.6 from a set of ξ values to show how $\mathscr{L}(\xi)$ varies with viewing angle. We multiply $\mathscr{L}(\xi)$ by a factor of 4 as a geometric isotropic assumption for the photosurface, which is equivalent to the isotropic assumption made by Tylenda et al. (2011). As seen in Fig. 6.18, the

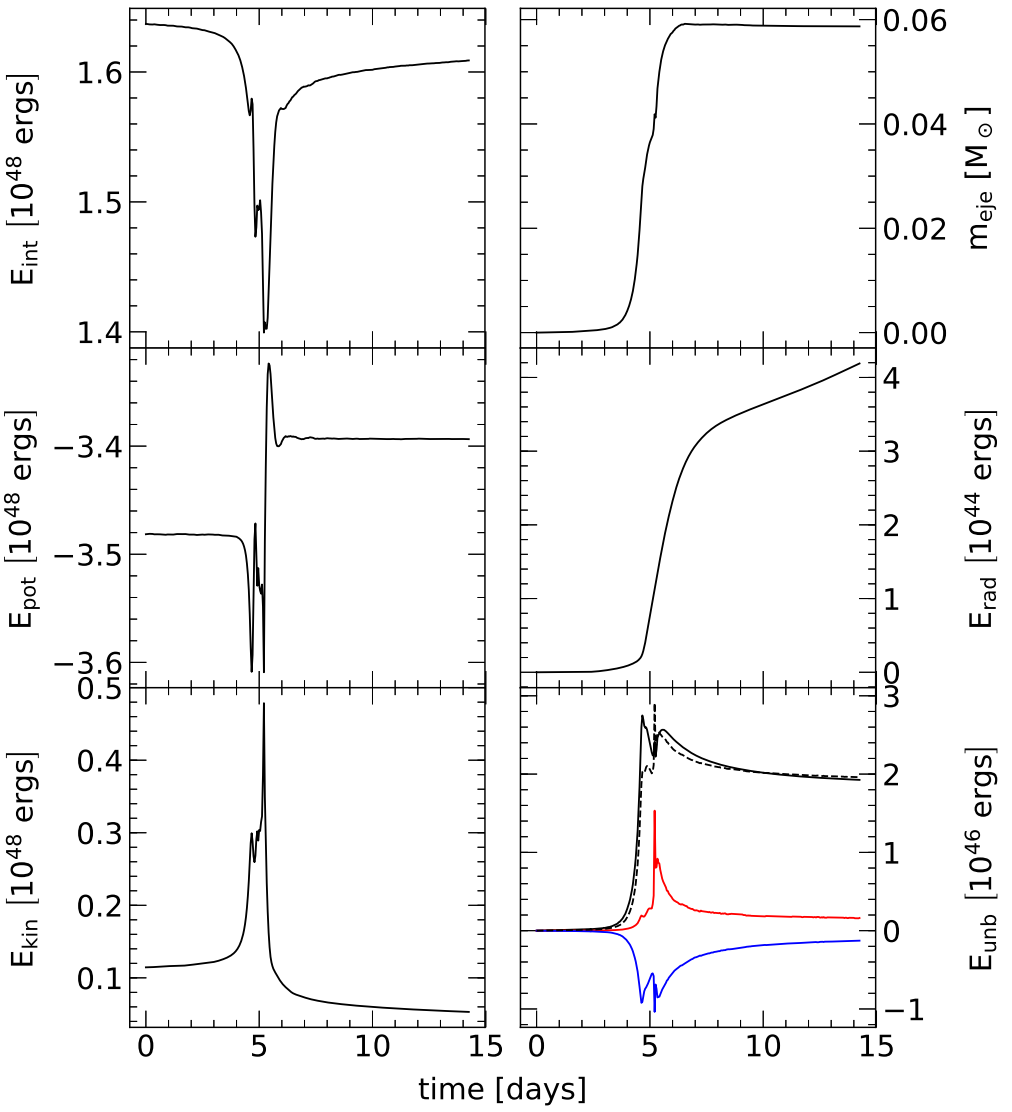

Figure 6.8: Energies of the bound and unbound material over time from the dynamical simulation of a merger between our $N = 2 \times 10^5$ primary model and a $N = 1 \times 10^4$, $M = 0.16\ M_\odot$, $R = 0.203\ R_\odot$ secondary (Stępień, 2011). The simulation begins at $t = 0$ days when the orbital period is 1.44 days (Tylenda et al., 2011). We show the total internal energy $E_{\rm int}$, total potential energy $E_{\rm pot}$ and total kinetic energy $E_{\rm kin}$ in the left column, while showing the total mass of the ejecta $m_{\rm eje}$, energy lost by radiative cooling $E_{\rm rad}$ calculated by Eq. 4.24, and the unbound kinetic (black), internal (red), potential (blue), and total (black dashed) energies in the right column. We calculate all quantities using `StarSmasher` output. The total energy of the simulation stays conserved to within 99.88% over the entire evolution.

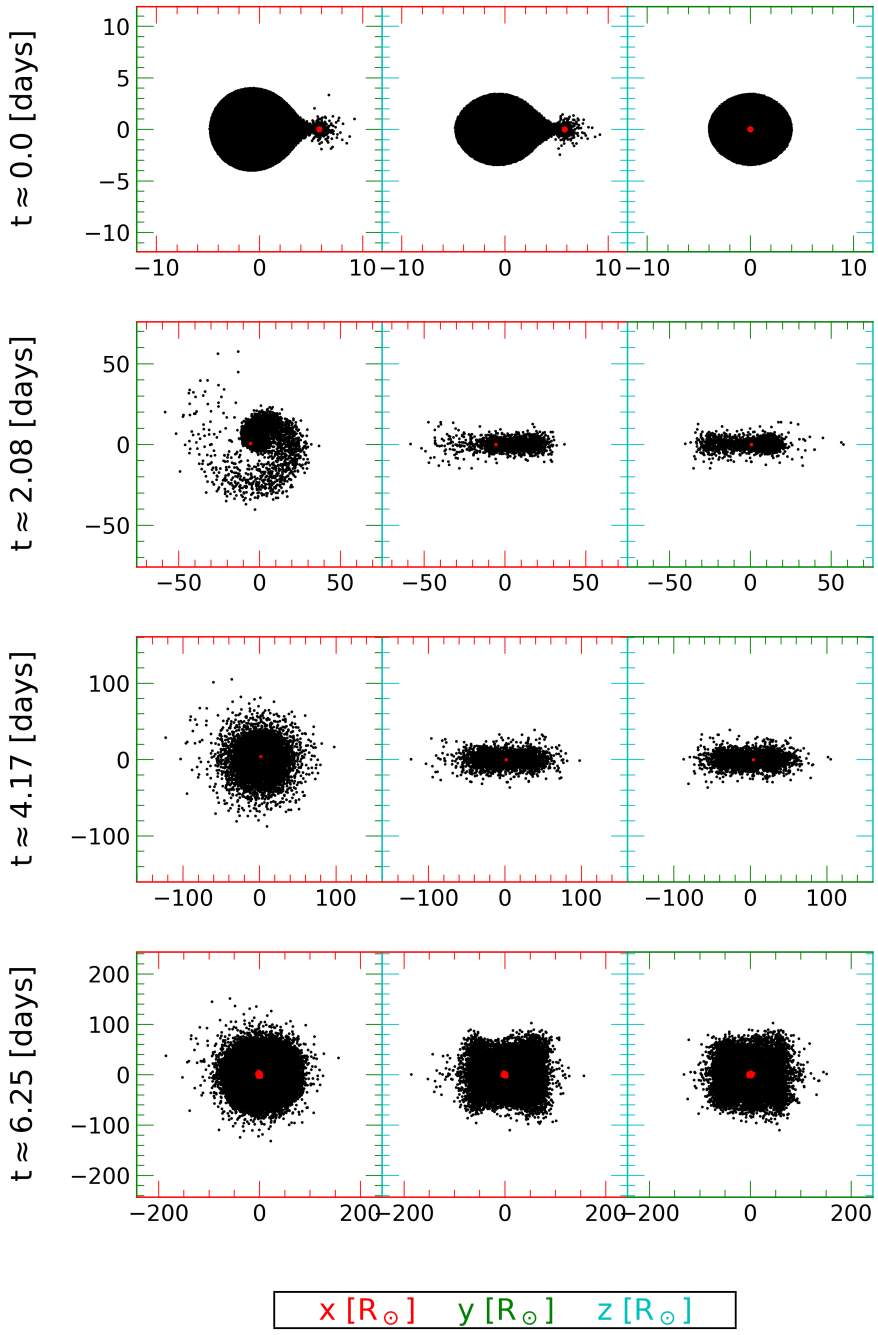

Figure 6.9: Particle positions in the dynamical `StarSmasher` simulation seen in Fig. 6.8 and discussed in §6.3. The left, middle, and right columns have lines of sight down the z, y, and x axes respectively, corresponding with viewing angles ($\xi = 0°$, $\zeta = 0°$), ($\xi = 90°$, $\zeta = 0°$), and ($\xi = 90°$, $\zeta = 90°$). We color the axes to indicate spatial orientation.

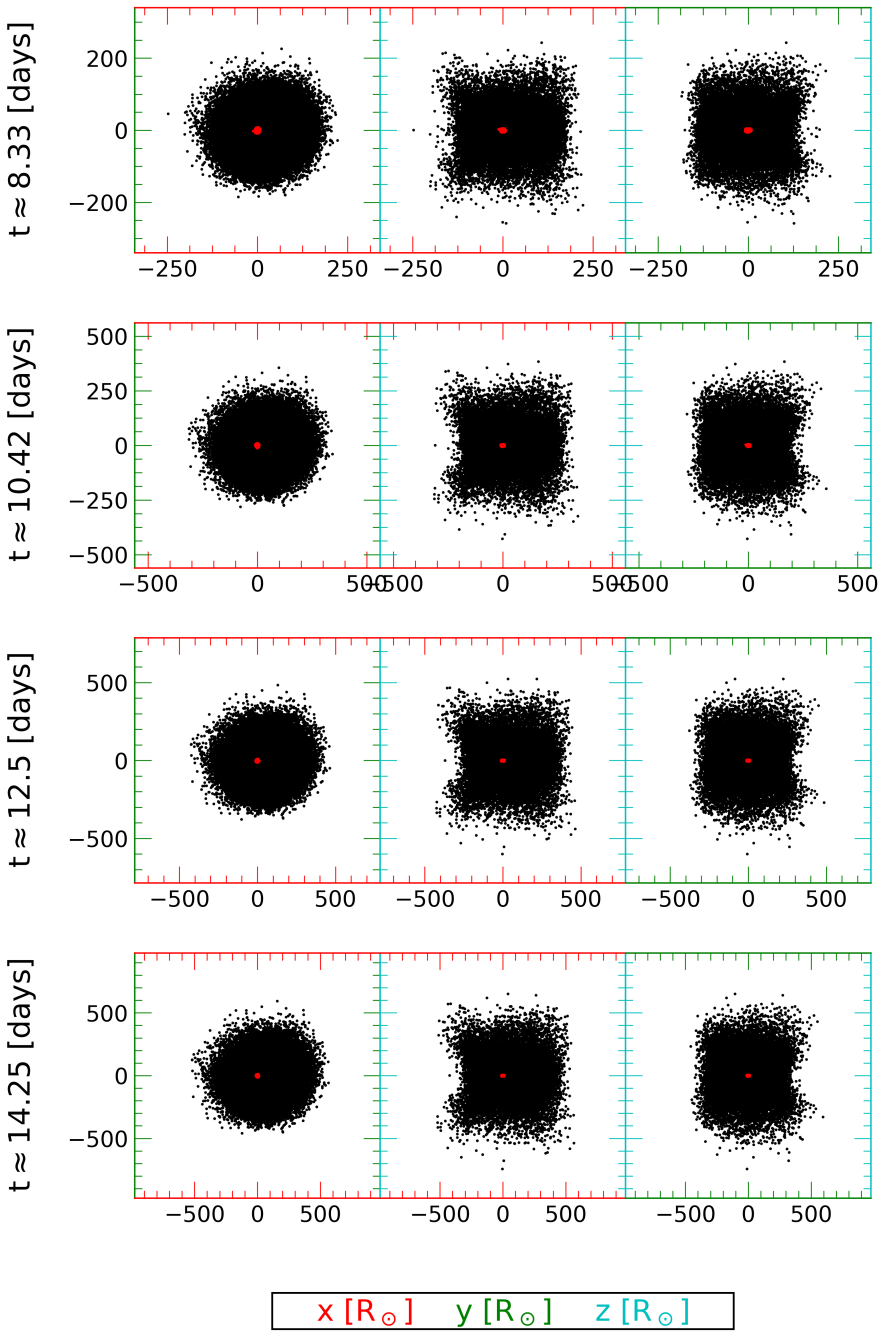

Figure 6.10: Particle positions as continued from Fig. 6.9.

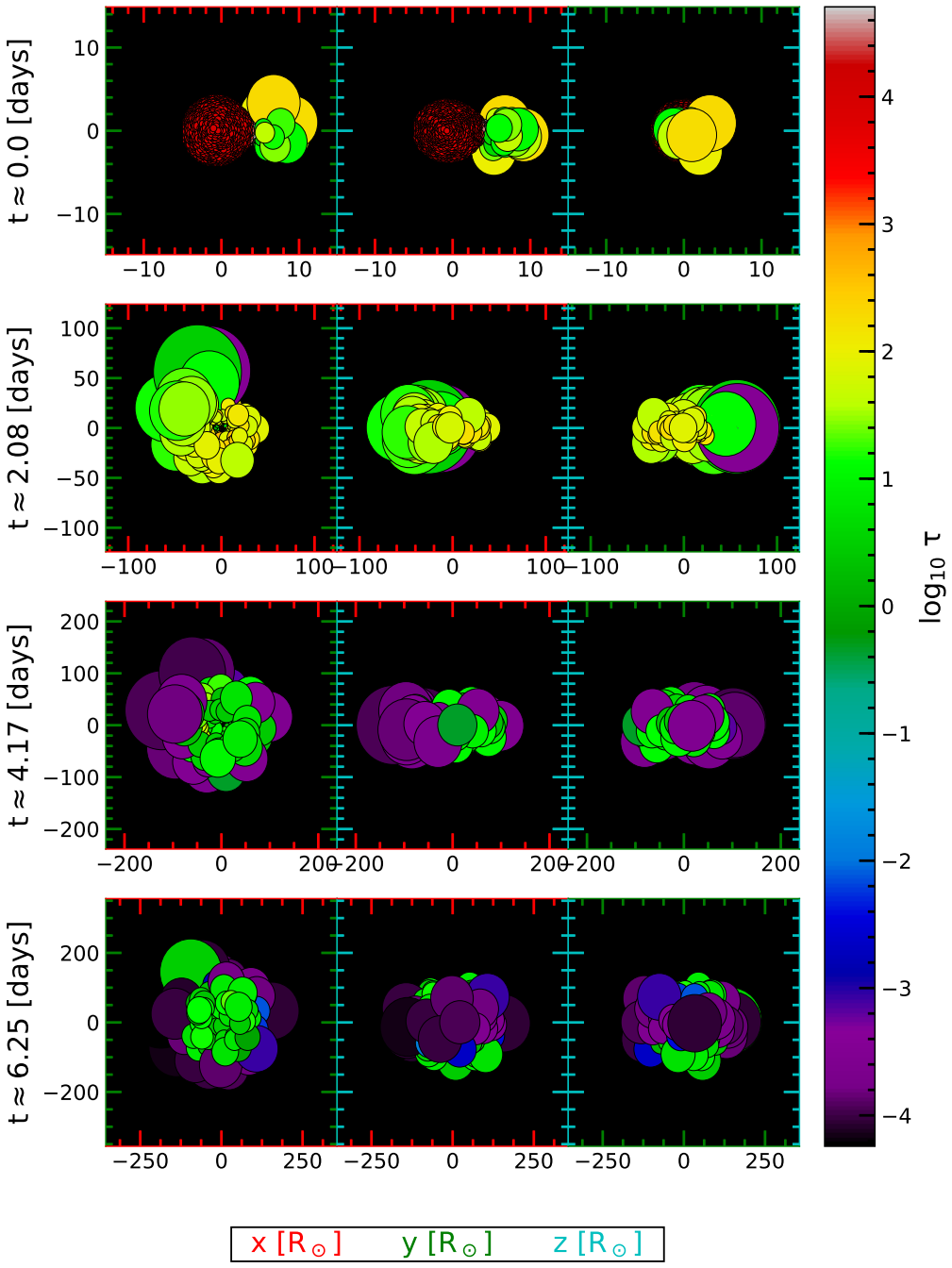

Figure 6.11: Optical depths of the nearest particles to the observer, as calculated by `FluxCal` (see §5.2), for the dynamical run, plotted in the same fashion as Fig. 6.6. The outflow becomes more optically thin as it expands and cools. This plot is continued in Fig. 6.12.

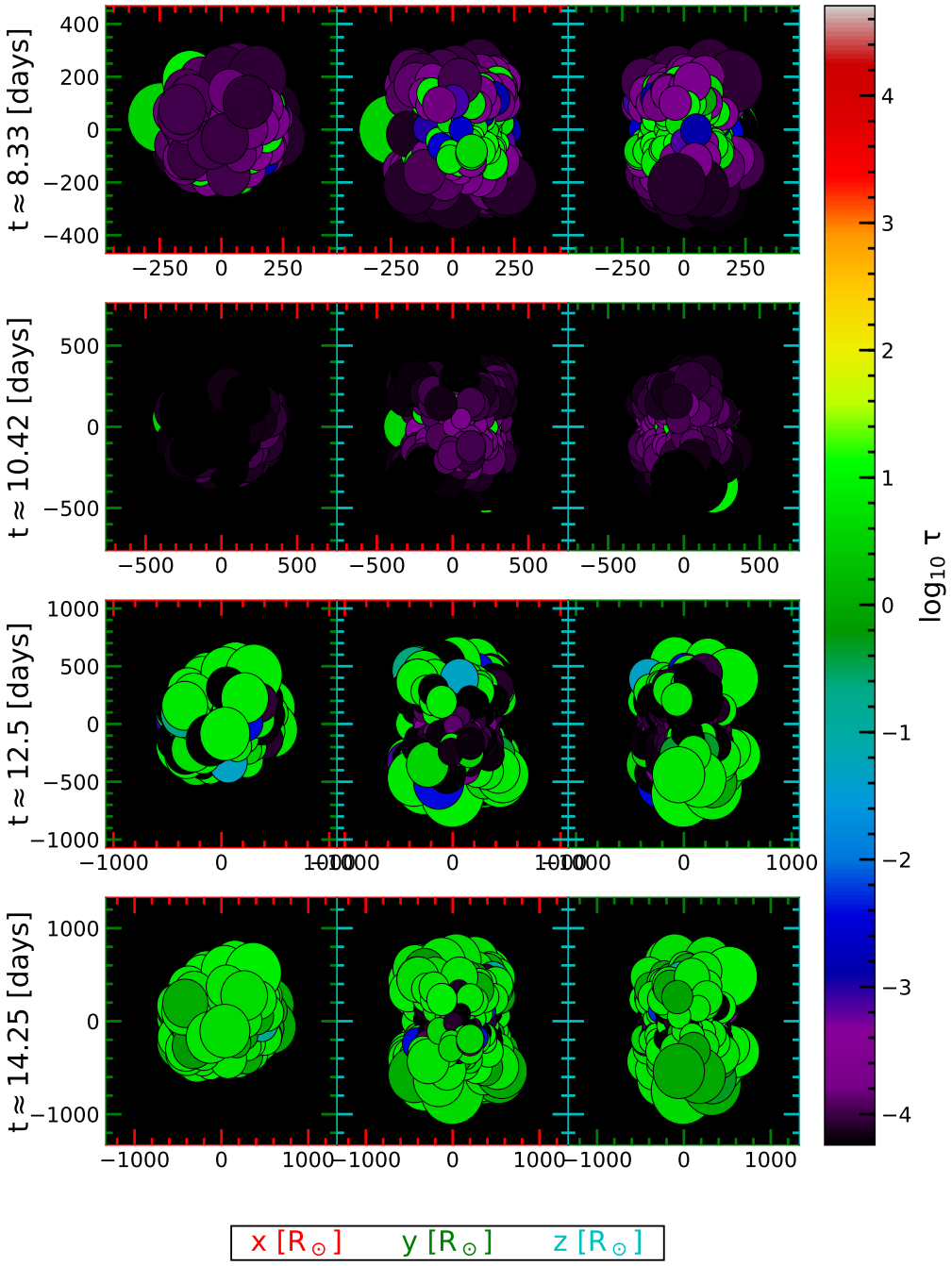

Figure 6.12: Optical depths of the dynamical run continued from Fig. 6.11.

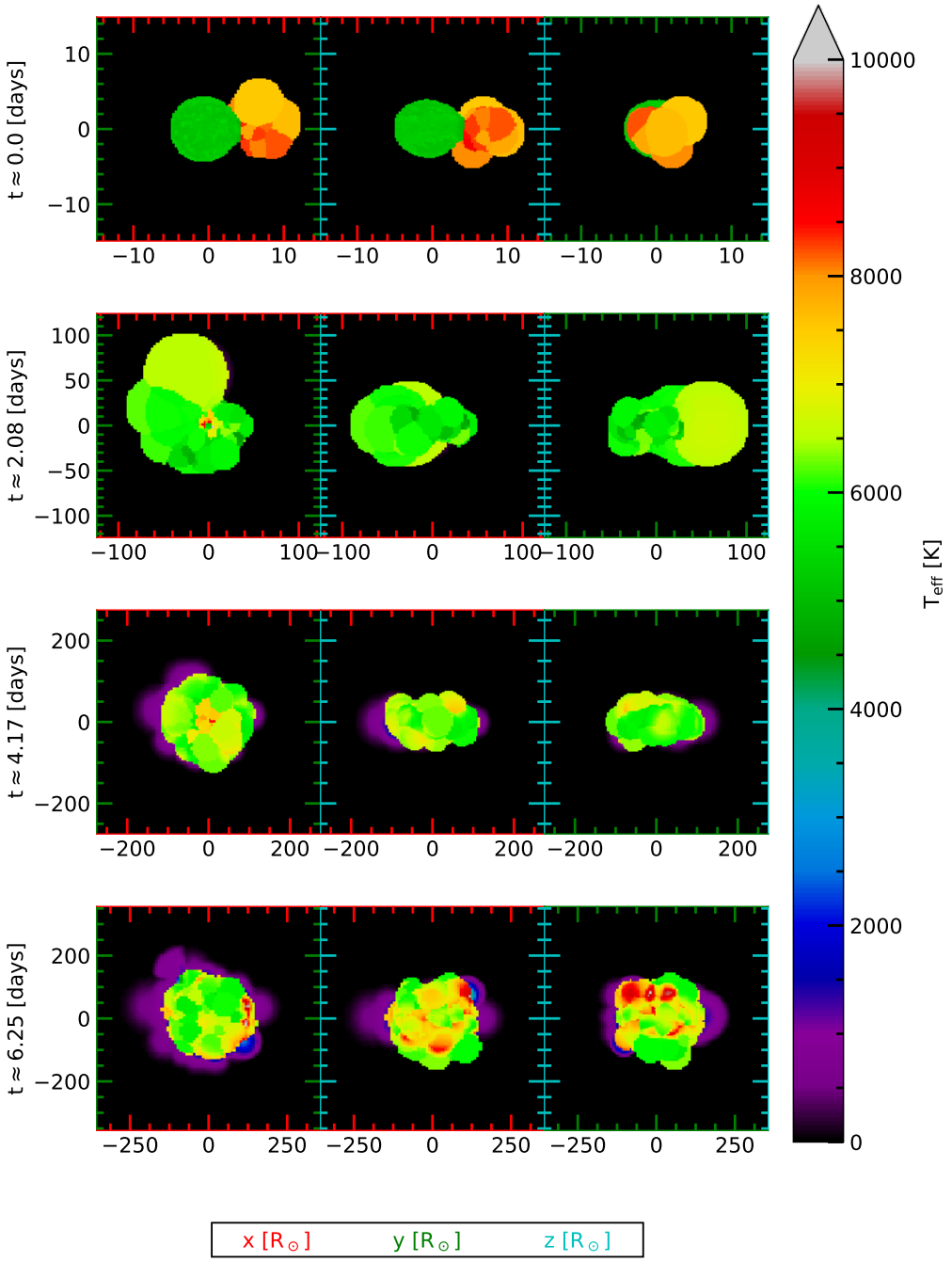

Figure 6.13: Effective temperatures calculated by `FluxCal` (see §5.7) for the dynamical run. This plot is continued in Fig. 6.14.

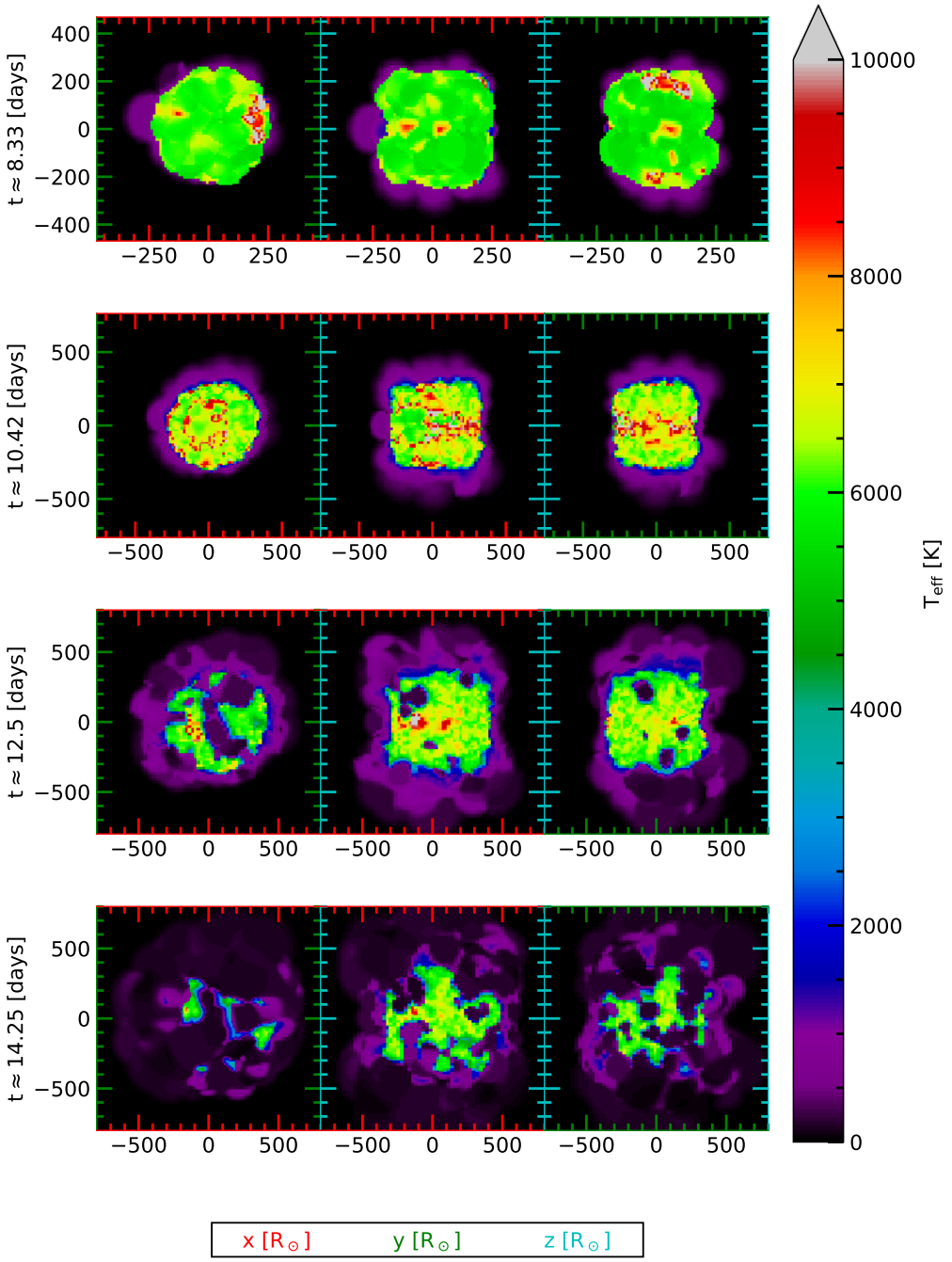

Figure 6.14: Effective temperatures of the dynamical run continued from Fig. 6.13.

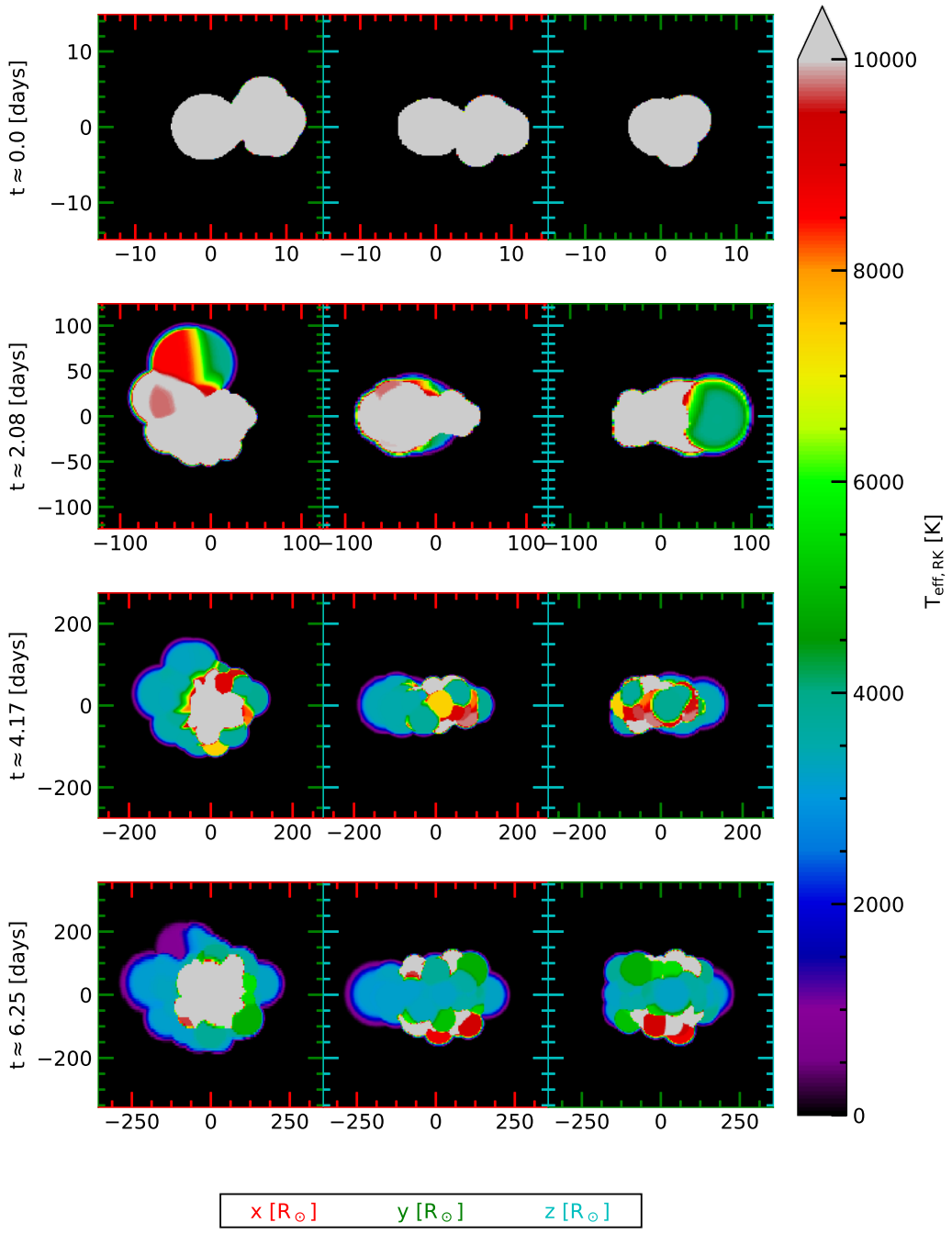

Figure 6.15: The same as Fig. 6.13, but with the envelope fitting routine for optically thick particles turned off. The Runge–Kutta adaptive step size integrator (see §5.4) determines all effective temperatures and hence the driving grid is forced to very high resolutions at early times. This plot is continued in Fig. 6.16.

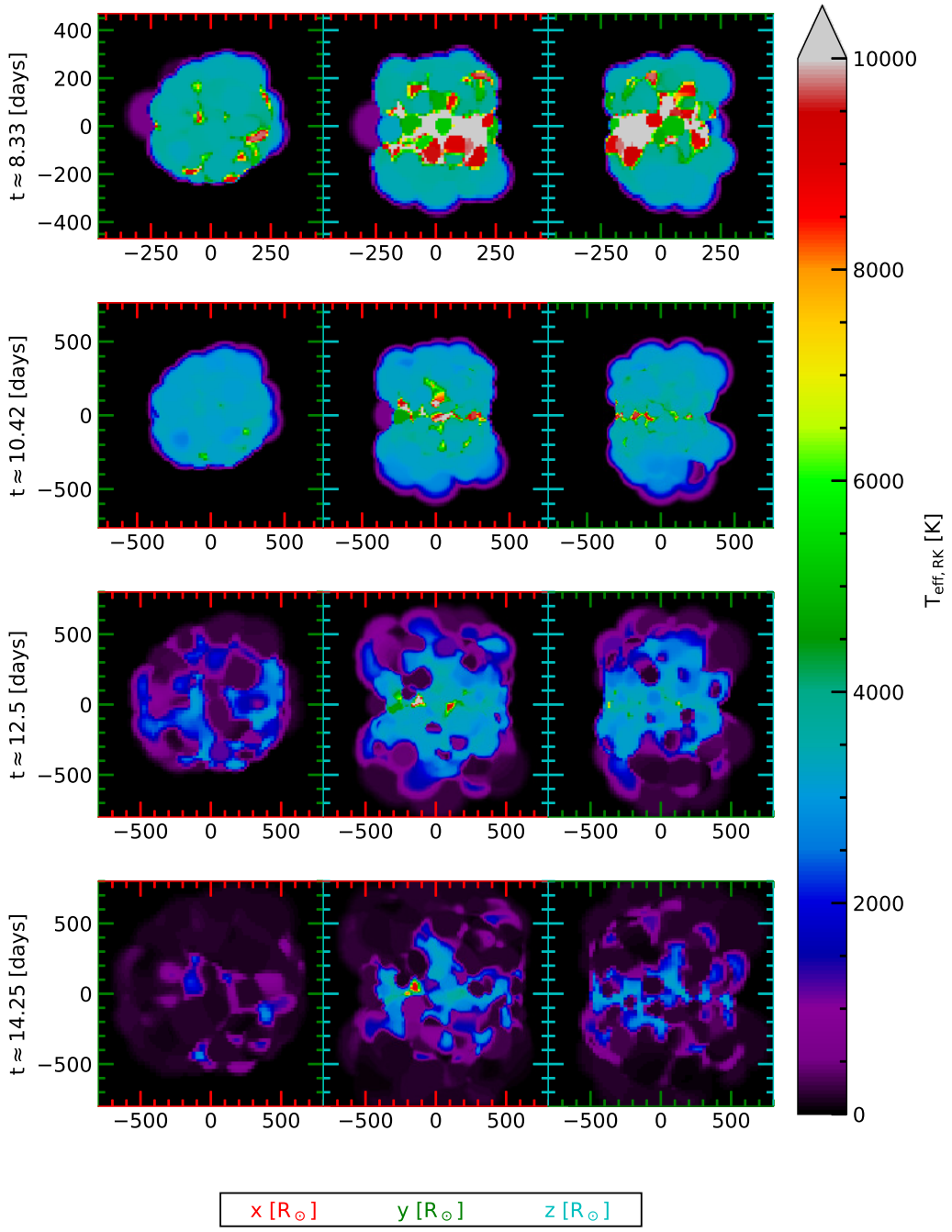

Figure 6.16: The same as Fig. 6.14, but continued from Fig. 6.15.

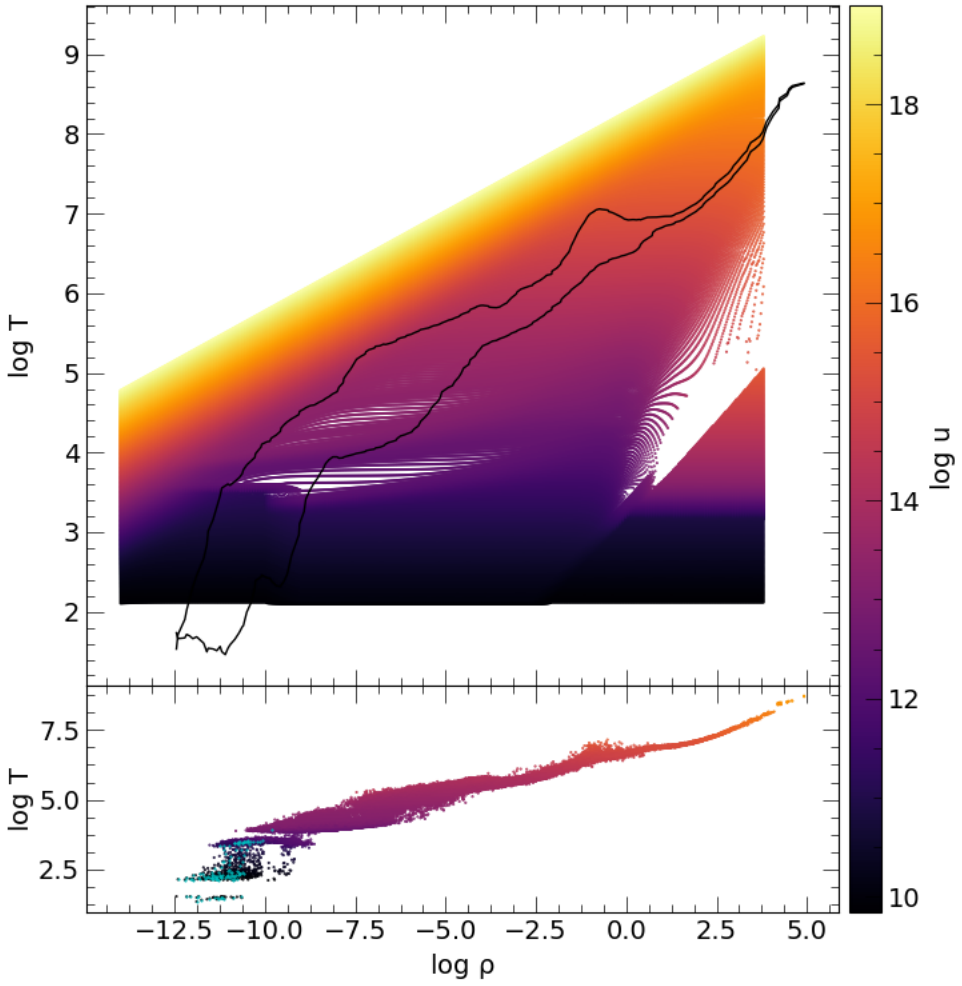

Figure 6.17: The location of all particles at $t \approx 14.25$ days in the dynamical simulation on the TEOS diagram, plotted in the same way as in Fig. 4.1. We show the particles that are closest to the observer at a $(\xi, \zeta) = (90°, 0°)$ viewing angle in cyan.

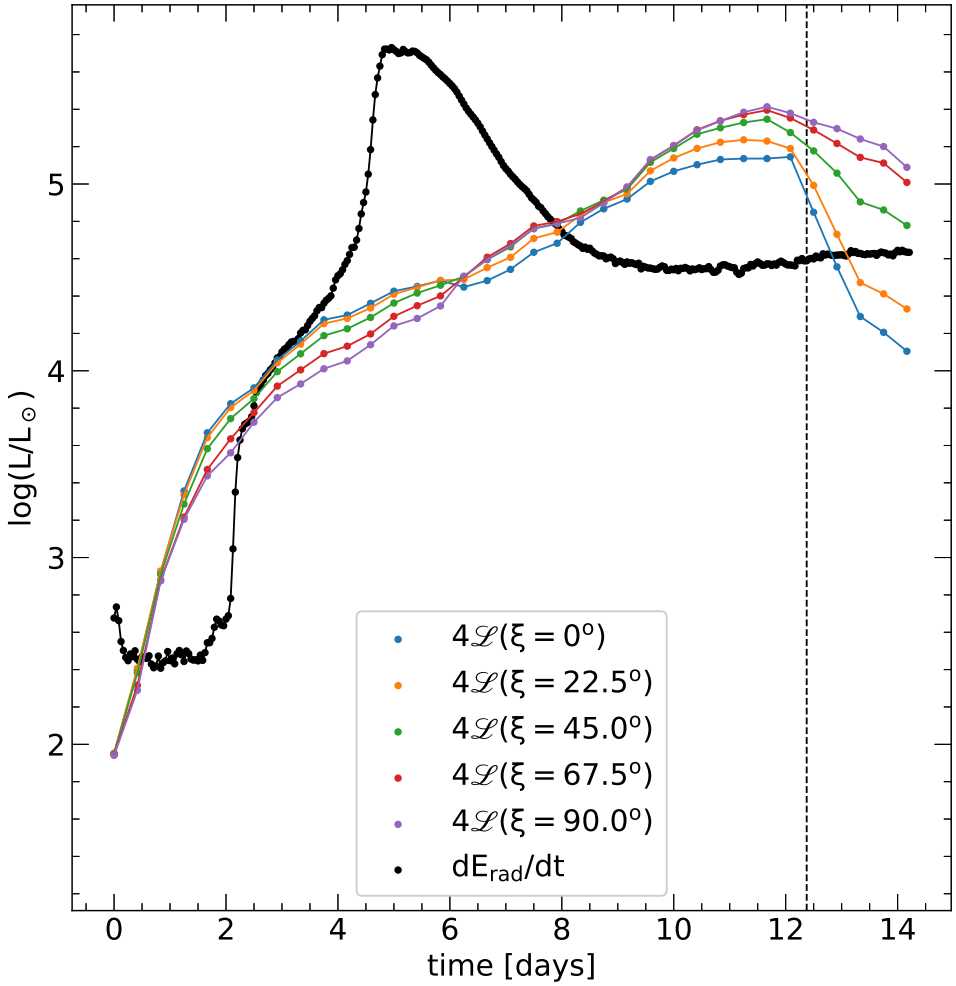

Figure 6.18: Light curves from multiple viewing angles of the dynamical run. We calculate visible luminosities (colored) using `FluxCal` (see §5.6) and multiply them by a factor of 4 as an isotropic assumption to the total luminosity. We show the derivative with time of the energy lost due to `StarSmasher` radiative cooling $E_{\rm rad}$ from Fig. 6.8 in black. We also show the time $t = 12.375$ days when particles of low density first fall out of range of the TEOS as a dashed black line. We assume a geometrically isotropic photosurface to imitate the isotropic assumption made by Tylenda et al. (2011) in calculating their luminosities. Error bars are smaller than the data points.

simulation increases in luminosity by ~ 4 orders of magnitude over ~ 10 days, which is an order of magnitude faster than the V1309 Sco I magnitude outburst, as seen in Fig. 1.2. For each viewing angle, the luminosity drops rapidly after the peak, over timescales shorter than that observed for V1309 Sco by about an order of magnitude.

We calculate the change in energy radiated away with time due to radiative cooling in `StarSmasher` (see §4.2) as,

$$\frac{dE_{\mathrm{rad},n}}{dt} = \frac{E_{\mathrm{rad},n+1} - E_{\mathrm{rad},n}}{t_{n+1} + t_n}, \tag{6.14}$$

where n is the index of the data and the final data point is omitted.

Our light curves in Fig. 6.18 have a constant time resolution of 10 hours while dE_{rad}/dt has a constant time resolution of 1 hour. We calculate the total energy radiated away from $t = 0$ days to $t \approx 14.25$ days as,

$$\begin{aligned}
\int_{t=0}^{t=14.25 \text{ days}} 4\mathscr{L}(\xi = 0^\circ)dt &= 2.283 \times 10^{44} \text{ erg}, \\
\int_{t=0}^{t=14.25 \text{ days}} 4\mathscr{L}(\xi = 22.5^\circ)dt &= 2.677 \times 10^{44} \text{ erg}, \\
\int_{t=0}^{t=14.25 \text{ days}} 4\mathscr{L}(\xi = 45^\circ)dt &= 3.317 \times 10^{44} \text{ erg}, \\
\int_{t=0}^{t=14.25 \text{ days}} 4\mathscr{L}(\xi = 67.5^\circ)dt &= 3.777 \times 10^{44} \text{ erg}, \\
\int_{t=0}^{t=14.25 \text{ days}} 4\mathscr{L}(\xi = 90^\circ)dt &= 3.961 \times 10^{44} \text{ erg}, \\
E_{\mathrm{rad}}(t = 14.25 \text{ days}) &= 4.120 \times 10^{44} \text{ erg}.
\end{aligned}$$

The time when particles fall out of the range of the TEOS follows the peak in luminosity. The slope of $\log(4\mathscr{L}/L_\odot)$ from each viewing angle appears unaffected by the outer particles exiting the TEOS range.

Chapter 7

Discussion

7.1 Relaxed Stellar Models

7.1.1 Expansion During Relaxation

Particles closest to the core are initialized by `StarSmasher` with densities ρ and pressures P smaller than the initial `MESA` model, and their neighbors are initialized with ρ and P larger than the initial model as seen in Fig. 6.1. In our $N = 1 \times 10^5$, $N = 2 \times 10^5$, and $N = 3 \times 10^5$ initial models, there are more particles at the core with high $\delta\rho$ and δP than there are low $\delta\rho$ and δP, resulting in an overall high $\delta\rho$ and δP core region. This creates an outward expansion wave as the inner particles move to lower ρ and P regions. The expansion wave propagates from the core to the surface and results in significantly expanded relaxed models. Our $N = 1 \times 10^5$, $N = 2 \times 10^5$, and $N = 3 \times 10^5$ models have larger radii than expected as a result, as seen in Table 6.1.

This may be a result of core particles being assigned masses that are smaller than the core of the `MESA` model. As mentioned in §4.4, `StarSmasher` is not equipped to handle stellar core physics, such as nuclear reactions and energy transfer. Core particles are assigned masses based only on the nearby number density, and thus it is possible to have a core particle with a mass smaller than

the MESA core, as seen in Table 6.1. By disallowing a core particle to have a mass less than the MESA core, the nearby particles and their neighbors may be initialized with proper ρ and P values, though more work is needed for affirmation.

7.1.2 Central Cool Spots

Cool spots appear in the center of our $N = 1\times10^5$, $N = 2\times10^5$, and $N = 3\times10^5$ models, the most pronounced of which is in the $N = 2 \times 10^5$ model. We observe a hexagonally symmetric pattern on the relaxed $N = 1 \times 10^5$ and $N = 3 \times 10^5$ models, as seen in Fig. 6.7, whose origin is unclear but may be a result of the initial hcp lattice. In a 3D hcp lattice, there are symmetry axes with which aligned particles may have disproportionate gravitational accelerations in some directions, ultimately leading to smaller changes in entropy relative to other, non–aligned particles. In this case, the hexagonal pattern is visible as bands of lower effective temperatures due to the smaller change in entropy there. However, such bands are not clearly visible for the $N = 2 \times 10^5$ model and there instead exists a large cool spot. The method by which this cool spot is created is uncertain and further work is required to understand the underlying mechanism.

7.1.3 Ray Tracing at the Surface

We calculate the average particle optical depths $\langle\tau_s\rangle$ using Eq. 5.16 at the surface of the relaxed models and find values $\sim 10^{10}$, which is consistent with §5.1 and §4.5.2. We use the Runge–Kutta integrator and no envelope fitting to calculate $\langle T_{\rm eff,RK}\rangle$, which results in values that are ~ 2 orders of magnitude greater than $\langle T_{\rm eff}\rangle$ and have no observable consistent behavior; they do not scale monotonically. The values of $\langle T_{\rm eff,}{}_{RK}\rangle$ also do not scale with the spatial resolution of the outermost particles which we estimate as $n_i \lesssim 600\ R_\odot^{-3}$,

$n_i \lesssim 1000\ R_\odot^{-3}$, and $n_i \lesssim 1500\ R_\odot^{-3}$ for the $N = 1 \times 10^5$, $N = 2 \times 10^5$, and $N = 3 \times 10^5$ models respectively (see Fig. 6.2, Fig. 6.3, and Fig. 6.4). This indicates the spatial resolution at the optically thick surface is sufficiently low that the Runge–Kutta integrator calculates values with very large error margins.

7.2 Dynamical Simulation

7.2.1 Merger Behavior

The ejected mass at $t \gtrsim 6.25$ days forms a bipolar topography similar to V4332 Sgr (Kamiński et al., 2018), as seen in Fig. 6.9 and Fig. 6.10. As the surrounding envelope expands, the outflowing particles become more optically thin, as seen in Fig. 6.11, until $t \gtrsim 10.42$ days, when particle optical depths increase, as seen in Fig. 6.12. This may be a result of dust formation in the outer particles. Indeed, the outer most particles at the final time have temperatures $T \lesssim 1500$ K (see §2.6.1), as seen in Fig. 6.17, indicating dust has formed or is forming at the end of the simulation.

Outflow material is hotter than expected at the initial time $t = 0$ days, by ~ 3500 K. This is a result of the expanded $N = 2 \times 10^5$ primary envelope, as discussed in §7.1.1, which leads to contact between the primary and secondary at a larger separation than expected. This results in overflow of the primary's envelope into the common envelope and thus loss of orbital angular momentum (see §1.1.1) and subsequent outflow at separations larger than with a non–expanded envelope. The net effect is a merger that evolves over shorter timescales than with a non–expanded envelope. Solving the problem of envelope expansion during relaxation may allow the merger to occur on longer dynamical timescales. With a smaller primary envelope size, the primary would fill its Roche lobe at smaller separations, and thus the common envelope phase would occur at smaller separations. Angular momentum would not be lost due to mass transfer

from the primary to the secondary as early–on as in our expanded case, which would result in a merger over longer dynamical timescales.

Less than 5% of the initial primary's envelope mass is ejected by the final time, so full ejection by recombination runaway as described by Ivanova et al. (2015) does not apply in our case.

7.2.2 Ejecta Optical Depths

The polynomial temperature solution described in §4.3.1 is used for low density ($\log \rho \lesssim -10$), low temperature ($\log T \leq 3$) ejecta at late times ($t > 12.375$ days) in the dynamical simulation for the outermost material, as seen in Fig. 6.17, and high density ($\log \rho \gtrsim 3$), high temperature ($\log T \geq 8.2$) inner material throughout, as seen in Fig. 4.1. This assumes a constant adiabatic index of $\Gamma = 5/3$ for an ideal monatomic gas, which is correct for both atomic and molecular hydrogen. However, as discussed in §2.6.1, molecules and dust can form at temperatures ($\log T \lesssim 3.3$), indicating $\Gamma = 5/3$ may not be appropriate. This may affect the calculated temperatures of the outermost particles in the ejecta, which fall out of range of the TEOS after $t = 12.375$ days, hence affecting $T_{\rm eff}$ and the resultant light curve.

As seen in Fig. 6.12, many outer particles experience an increase in optical depth τ_i by ~ 4 orders of magnitude some time after $10.42 \lesssim t \lesssim 12.5$ days. Using the equation of local spatial resolution in Eq. 3.26, τ_i from Eq. 5.16 can be written as,

$$\tau_i = \frac{4\pi}{3} \frac{m_i}{N_{nb,i}} n_i h_i \bar{\kappa}_i C_\tau. \tag{7.1}$$

As described in §3.1, neighbor numbers change in `StarSmasher` as a function of local mass resolution. In the assumption that the local mass resolution is approximately constant in the ejecta, the number of neighbors $N_{nb,i}$ and mass m_i of each particle is approximately constant. Thus, $\tau_i \propto h_i \bar{\kappa}_i$, so both $h_i \bar{\kappa}_i$ must increase by ~ 4 orders of magnitude from $t \approx 10.42$ days to $t \approx 12.5$ days.

As seen in Fig. 6.12, smoothing lengths appear to increase marginally, by less than a factor of 2, so the increase in τ_i is a result of an increasing $\bar{\kappa}_i$. As seen in Fig. 6.17, these particles have temperatures low enough to form molecules and dust, as discussed in §2.6.1. Thus, the rapid increase in optical depth may be a result of dust and molecule formation.

7.2.3 Effective Temperatures and LTE

As discussed in §5.7, `FluxCal` calculates the effective temperature T_{eff} at each driving grid vertex using the Planck blackbody function as the source function for the optically thin regime. This assumes strict LTE, as described in §2.3, which does not necessarily apply to the optically thin regime. The ejecta in the dynamical simulation first becomes optically thin at $t \gtrsim 2.08$ days until $t \gtrsim 14.25$ days, as seen in Fig. 6.11 and Fig. 6.12, and thus may not be in the LTE regime. To account for possible non–LTE situations such as this, the detailed physics of emission and absorption for the full range of transitional states of the gas components must be accounted for and is the possible subject of future work.

At $t \gtrsim 10.42$ days, it is clear that the effective temperatures T_{eff} calculated by `FluxCal` are very similar between the Runge–Kutta integrator only case (Fig. 6.16) and the Runge–Kutta integrator plus envelope fitting method case (Fig. 6.14). This implies the optically thick particles for which the envelope fitting method is used are located at large optical depths, so their contributions to T_{eff} in Eq. 5.43 are small compared to that from the optically thin particles in front of them.

7.3 Light Curve Features

Prior to peak luminosity, there exists an "S" shape in the light curve that is most pronounced for the ($\xi = 0°$) viewing angle and is less pronounced with

increasing ξ. Ejected material generally has a lower temperature than the hot, inner material, as seen in Fig. 6.13 and Fig. 6.14. From viewing angle ($\xi = 0°$) at $t \lesssim 6.25$ days, more of the hot inner material is visible than at any other viewing angle and the projected area of the photosurface is greater than that from any other viewing angle. Thus, the luminosity measured from the ($\xi = 0$) viewing angle is greater than that from any other viewing angle until $t \approx 6$ days.

At $t \gtrsim 6$ days, outflow starts to obstruct the hot inner material visible from small ξ and the overall shape of the ejecta becomes approximately spherical. These combined effects result in similar luminosity values measured from all viewing angles at $t \approx 6$ days, as seen in Fig. 6.18.

At $t \gtrsim 6$ days, the projected photosurface area from ($\xi = 0°$) is smaller than that from any other viewing angle and very little of the inner hot material is visible. As $T_{\rm eff}$ is greater in the equatorial region than the polar regions for these times and projected photosurface areas are larger, as seen in Fig. 6.14, the measured luminosity is the lowest from the ($\xi = 0°$) viewing angle.

The rate at which energy is lost by the radiative cooling prescription discussed in §4.2.1, shown as a black curve in Fig. 6.18, does not scale with the luminosity calculated by `FluxCal`. This is unexpected, as both quantities describe the energy lost from the system by radiation. The reason for their differences is uncertain.

Chapter 8

Conclusions

We calculate the spectral properties of optically thick SPH particles using a new code we developed called `FluxCal` with an accuracy improvement of ~ 2 orders of magnitude over a traditional Runge–Kutta adaptive step size ray tracing approach. We create stellar models at three resolutions $N = 1 \times 10^5$, $N = 2 \times 10^5$, and $N = 3 \times 10^5$ using `StarSmasher` based on our initial $M = 1.52\ M_\odot$, $R = 3.715\ R_\odot$ `MESA` model, which is a candidate for the V1309 Sco progenitor primary (masses determined by Stępień 2011). We find the `nnopt` values that best match the total energy profiles in the top $\sim 8\%$ of the envelope by mass for all three models.

We use our $N = 2 \times 10^5$ primary model and an $N = 1 \times 10^4$, $M = 0.16\ M_\odot$, $R = 0.203\ R_\odot$ brown dwarf secondary (Stępień, 2011) in a `StarSmasher` merger simulation to model the presumed V1309 Sco red nova (Tylenda et al., 2011). We calculate a grid of effective temperatures every 10 hours using `FluxCal` and produce light curves from multiple viewing angles.

`FluxCal` operates over a grid constructed individually for each provided input file. `FluxCal` calculates particle optical depths to detect optically thick particles, whose individual effective temperatures are calculated using an analytic envelope fitting technique. Effective temperatures are calculated at each vertex on the grid by integrating the outgoing intensity from the surface inwards to a

sufficiently large optical depth ($\tau_{\rm thick} = 10$) such that additional contributions to the flux are negligible ($\sim 10^{-4}$ ergs s^{-1} cm^{-2}). `FluxCal` presents the first ever solution to the problem of calculating outgoing radiation from a poorly spatially resolved optically thick region in SPH simulations.

We observe `StarSmasher` stellar models artificially expanding outwards as a result of poor initial conditions. This may affect our dynamical simulation by allowing a common envelope to form at larger separations, altering the merger dynamics. This expansion may also affect the physical properties of surface particles, altering the results from `FluxCal` when calculating effective temperatures and visible luminosities.

Our dynamical simulation, which ran for $\approx$ 4 months wall time with 16 NVIDIA P100 Pascal GPUs and 96 Intel E5-2683 v4 "Broadwell" CPUs on the ComputeCanada `Cedar` supercluster has not yet evolved to long enough times to properly compare the resultant light curve to V1309 Sco observations. We require a TEOS with a broader range of density and specific internal energies to correctly calculate temperatures in the low density ejecta at late times. `FluxCal` lacks the non–LTE physics of absorption and emission of transitional states, which must be used to correctly calculate effective temperatures outside of the LTE regime.

An accurate dynamical simulation of V1309 Sco as a contact binary merger would allow for an excellent understanding of contact binary merger events. If successful, primary and secondary masses could be explored to understand how a merger light curve changes, bringing an end to over a century of conjecture. With these new tools, V1309 Sco is a dragon that may soon be slain.

Appendix A

Appendix

A.1 TEOS MESA Inlist

```
&eos_table

        !file that contains mass fractions of metals C thru Ni
        mass_list = 'mass_frac.txt'

        !MESA/eos table version number
        table_version = 48

        !FreeEOS option suite, see FreeEOS README for details
        eos_version = 4

        !option to do only eosDT or eosPT tables
        do_eosPT = .true.
        do_eosDT = .true.

        !set log10T range
        log10Tmin = 3d0
        log10Tmax = 8.2d0
        dlog10T = 0.02d0        !default 0.02

        !for eosDT: set log10Q range ( log10Q = log10Rho
        !                                    - 2*log10T + 12 )
        log10Qmin = -8.0d0
        log10Qmax =  4.5d0
```

```
        dlog10Q = 0.03d0         !default 0.03

        !for eosPT: set log10W range ( logW = logPgas - 4*logT )
        log10Wmin = -14.0d0
        log10Wmax = -2.8d0
        dlog10W = 0.1d0

/
```

A.2 V1309 Sco Primary Star Inlist

```
&star_job

  ! choose starting model
  !    create_pre_main_sequence_model = .true.

  ! plot or not
      pgstar_flag = .true.

  ! control the output
      history_columns_file = 'history_columns.list'
      profile_columns_file = 'profile_columns.list'

  ! save a model at the end of the run
    save_model_when_terminate = .true.
    save_model_filename = 'v1309_last'

    write_profile_when_terminate=.true.
    filename_for_profile_when_terminate='profile_last.data'

    change_initial_net = .true. ! switch nuclear reaction
                                ! network
    new_net_name = 'pp_and_cno_extras.net'

    set_rate_c12ag = 'Kunz'
    set_rate_n14pg = 'jina reaclib'

    kappa_file_prefix = 'OP_gs98'
    kappa_lowT_prefix = 'lowT_fa05_gs98' ! for lower
                                         ! temperatures
```

```
30
31
32  / !end of star_job
33
34  &controls
35
36    ! starting specifications: calibrated values
37        initial_mass = 1.52d0 ! in Msun units
38        ! there is rather disagreement on what is true solar
39        ! metallicity
40        initial_Z = 0.02d0 ! one of the calibrated solar
41                           ! metalicity values
42
43        use_Type2_opacities = .true.
44        Zbase = 0.02 ! this is set to Z of the star in
45                     ! run_star_extras.f
46
47        which_atm_option = 'photosphere_tables'
48
49        use_Ledoux_criterion = .true.
50        alpha_semiconvection = 0.1
51        mixing_length_alpha = 1.8
52
53    ! calibrated overshooting parameters for the Sun
54        overshoot_f_above_nonburn_core = 0.0174
55        overshoot_f_above_burn_h_core = 0.0174
56        overshoot_f0_above_nonburn_core = 0.008
57        overshoot_f0_above_burn_h_core = 0.008
58
59        max_years_for_timestep = 1.e7
60
61  ! This is where you are to set stops
62
63
64
65    max_age=2.54e9
66
67
68    ! make output more frequent and make sure that last one is
69    ! saved
70        history_interval = 1
71        profile_interval = 10
72
```

```
73
74 / !end of controls
```

Bibliography

Gabriel Altay and Tom Theuns. URCHIN: a reverse ray tracer for astrophysical applications. *MNRAS*, 434:748–764, September 2013. doi: 10.1093/mnras/stt1067.

Gabriel Altay, Rupert A. C. Croft, and Inti Pelupessy. SPHRAY: a smoothed particle hydrodynamics ray tracer for radiative transfer. *MNRAS*, 386:1931–1946, June 2008. doi: 10.1111/j.1365-2966.2008.13212.x.

D. S. Balsara. von Neumann stability analysis of smooth particle hydrodynamics–suggestions for optimal algorithms. *Journal of Computational Physics*, 121:357–372, January 1995. doi: 10.1016/S0021-9991(95)90221-X.

Richard Barvainis. Hot Dust and the Near-Infrared Bump in the Continuum Spectra of Quasars and Active Galactic Nuclei. *ApJ*, 320:537, September 1987. doi: 10.1086/165571.

Subrahmanyan Chandrasekhar. *An introduction to the study of stellar structure.* The University of Chicago press, 1939.

M. de Kool, E. P. J. van den Heuvel, and E. Pylyser. An evolutionary scenario for the black hole binary A0620-00. *A&A*, 183:47–52, September 1987.

W. Dehnen and H. Aly. Improving convergence in smoothed particle hydrodynamics simulations without pairing instability. *MNRAS*, 425:1068–1082, September 2012. doi: 10.1111/j.1365-2966.2012.21439.x.

Gaspard Duchêne and Adam Kraus. Stellar Multiplicity. *Annual Review of Astronomy and Astrophysics*, 51:269–310, August 2013. doi: 10.1146/annurev-astro-081710-102602.

P. P. Eggleton. Aproximations to the radii of Roche lobes. *ApJ*, 268:368–369, May 1983. doi: 10.1086/160960.

Jason W. Ferguson, David R. Alexander, France Allard, Travis Barman, Julia G. Bodnarik, Peter H. Hauschildt, Amanda Heffner-Wong, and Akemi Tamanai. Low-Temperature Opacities. *ApJ*, 623:585–596, April 2005. doi: 10.1086/428642.

E. Gaburov, J. C. Lombardi, Jr., and S. Portegies Zwart. On the onset of runaway stellar collisions in dense star clusters - II. Hydrodynamics of three-body interactions. *MNRAS*, 402:105–126, February 2010. doi: 10.1111/j.1365-2966.2009.15900.x.

Peter Goldreich and John Kwan. Molecular Clouds. *ApJ*, 189:441–454, May 1974. doi: 10.1086/152821.

Aldana Grichener, Efrat Sabach, and Noam Soker. The limited role of recombination energy in common envelope removal. *MNRAS*, 478:1818–1824, August 2018. doi: 10.1093/mnras/sty1178.

Z. Han, P. Podsiadlowski, and P. P. Eggleton. A possible criterion for envelope ejection in asymptotic giant branch or first giant branch stars. *MNRAS*, 270:121–130, Sep 1994. doi: 10.1093/mnras/270.1.121.

Lars Hernquist and Neal Katz. TREESPH: A Unification of SPH with the Hierarchical Tree Method. *The Astrophysical Journal Supplement Series*, 70:419, June 1989. doi: 10.1086/191344.

Natsuki Hosono, Takayuki R. Saitoh, and Junichiro Makino. A Comparison of SPH Artificial Viscosities and Their Impact on the Keplerian Disk. *The*

Astrophysical Journal Supplement Series, 224:32, June 2016. doi: 10.3847/0067-0049/224/2/32.

D. A. Hubber, B. Ercolano, and J. Dale. Observing gas and dust in simulations of star formation with Monte Carlo radiation transport on Voronoi meshes. *MNRAS*, 456:756–766, February 2016. doi: 10.1093/mnras/stv2676.

J. Hwang, J. C. Lombardi, Jr., F. A. Rasio, and V. Kalogera. Stability and Coalescence of Massive Twin Binaries. *ApJ*, 806:135, June 2015. doi: 10.1088/0004-637X/806/1/135.

N. Ivanova and S. Chaichenets. Common Envelope: Enthalpy Consideration. *ApJ*, 731:L36, Apr 2011. doi: 10.1088/2041-8205/731/2/L36.

N. Ivanova, S. Justham, X. Chen, O. De Marco, C. L. Fryer, E. Gaburov, H. Ge, E. Glebbeek, Z. Han, X. D. Li, G. Lu, T. Marsh, P. Podsiadlowski, A. Potter, N. Soker, R. Taam, T. M. Tauris, E. P. J. van den Heuvel, and R. F. Webbink. Common envelope evolution: where we stand and how we can move forward. *Astronomy and Astrophysics Review*, 21:59, February 2013. doi: 10.1007/s00159-013-0059-2.

N. Ivanova, S. Justham, and Ph. Podsiadlowski. On the role of recombination in common-envelope ejections. *MNRAS*, 447:2181–2197, March 2015. doi: 10.1093/mnras/stu2582.

Natalia Ivanova. On the Use of Hydrogen Recombination Energy during Common Envelope Events. *ApJ*, 858:L24, May 2018. doi: 10.3847/2041-8213/aac101.

T. Kamiński, E. Mason, R. Tylenda, and M. R. Schmidt. Post-outburst spectra of a stellar-merger remnant of V1309 Scorpii: from a twin of V838 Monocerotis to a clone of V4332 Sagittarii. *A&A*, 580:A34, August 2015. doi: 10.1051/0004-6361/201526212.

T. Kamiński, W. Steffen, R. Tylenda, K. H. Young, N. A. Patel, and K. M. Menten. Submillimeter-wave emission of three Galactic red novae: cool molecular outflows produced by stellar mergers. *A&A*, 617:A129, October 2018. doi: 10.1051/0004-6361/201833165.

Daniel Kasen and S. E. Woosley. Type II Supernovae: Model Light Curves and Standard Candle Relationships. *ApJ*, 703:2205–2216, October 2009. doi: 10.1088/0004-637X/703/2/2205.

R. Kippenhahn and A. Weigert. *Stellar Structure and Evolution*. Springer-Verlag Berlin Heidelberg New York, 1994. pp. 73.

Mario Livio and Noam Soker. The Common Envelope Phase in the Evolution of Binary Stars. *ApJ*, 329:764, June 1988. doi: 10.1086/166419.

O. Lomax and A. P. Whitworth. SPAMCART: a code for smoothed particle Monte Carlo radiative transfer. *MNRAS*, 461:3542–3551, October 2016. doi: 10.1093/mnras/stw1568.

James C. Lombardi, William G. McInally, and Joshua A. Faber. An efficient radiative cooling approximation for use in hydrodynamic simulations. *MNRAS*, 447:25–35, February 2015. doi: 10.1093/mnras/stu2432.

Jr. Lombardi, J. C., Z. F. Proulx, K. L. Dooley, E. M. Theriault, N. Ivanova, and F. A. Rasio. Stellar Collisions and Ultracompact X-Ray Binary Formation. *ApJ*, 640:441–458, March 2006. doi: 10.1086/499938.

E. Mason, M. Diaz, R. E. Williams, G. Preston, and T. Bensby. The peculiar nova V1309 Scorpii/nova Scorpii 2008. A candidate twin of V838 Monocerotis. *A&A*, 516:A108, Jun 2010. doi: 10.1051/0004-6361/200913610.

Christopher F. McKee and Eve C. Ostriker. Theory of Star Formation. *Annual Review of Astronomy and Astrophysics*, 45:565–687, September 2007. doi: 10.1146/annurev.astro.45.051806.110602.

J. J. Monaghan. Smoothed particle hydrodynamics. *ARA&A*, 30:543–574, 1992. doi: 10.1146/annurev.aa.30.090192.002551.

J. J. Monaghan and R. A. Gingold. Shock Simulation by the Particle Method SPH. *Journal of Computational Physics*, 52:374–389, Nov 1983. doi: 10.1016/0021-9991(83)90036-0.

J. J. Monaghan and J. C. Lattanzio. A refined particle method for astrophysical problems. *A&A*, 149:135–143, August 1985.

S. Nakano, K. Nishiyama, F. Kabashima, Y. Sakurai, C. Jacques, E. Pimentel, D. Chekhovich, S. Korotkiy, T. Kryachko, and N. N. Samus. V1309 Scorpii = Nova Scorpii 2008. *International Astronomical Union Circular*, 8972:1, September 2008.

J. L. A. Nandez, N. Ivanova, and J. C. Jr Lombardi. Recombination energy in double white dwarf formation. *MNRAS*, 450:L39–L43, June 2015. doi: 10.1093/mnrasl/slv043.

Giovanni Natale, Cristina C. Popescu, Richard. J. Tuffs, Victor P. Debattista, Jörg Fischera, and Meiert W. Grootes. Predicting the stellar and non-equilibrium dust emission spectra of high-resolution simulated galaxies with DART-RAY. *MNRAS*, 449:243–267, May 2015. doi: 10.1093/mnras/stv286.

B. Paxton, L. Bildsten, A. Dotter, F. Herwig, P. Lesaffre, and F. Timmes. Modules for Experiments in Stellar Astrophysics (MESA). *ApJS*, 192:3, January 2011. doi: 10.1088/0067-0049/192/1/3.

W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes in Fortran 77. The art of scientific computing. Second Edition.* Cambridge: University Press, 1992. ISBN 0-521-43064-X. pp. 704-715.

Daniel J. Price. SPLASH: An Interactive Visualization Tool for Smoothed Particle Hydrodynamics Simulations, Mar 2011.

Deepak Raghavan, Harold A. McAlister, Todd J. Henry, David W. Latham, Geoffrey W. Marcy, Brian D. Mason, Douglas R. Gies, Russel J. White, and Theo A. ten Brummelaar. A Survey of Stellar Families: Multiplicity of Solar-type Stars. *The Astrophysical Journal Supplement Series*, 190:1–42, September 2010. doi: 10.1088/0067-0049/190/1/1.

F. J. Rogers and A. Nayfonov. Updated and Expanded OPAL Equation-of-State Tables: Implications for Helioseismology. *ApJ*, 576:1064–1074, September 2002. doi: 10.1086/341894.

S. Rosseland. Note on the absorption of radiation within a star. *MNRAS*, 84: 525–528, May 1924. doi: 10.1093/mnras/84.7.525.

S. Rosswog. Boosting the accuracy of SPH techniques: Newtonian and special-relativistic tests. *MNRAS*, 448:3628–3664, April 2015. doi: 10.1093/mnras/stv225.

R. J. Rudy, D. K. Lynch, R. W. Russell, B. Kaneshiro, M. Sitko, and H. Hammel. V1309 Scorpii. *International Astronomical Union Circular*, 8976:1, September 2008a.

R. J. Rudy, D. K. Lynch, R. W. Russell, M. Sitko, C. E. Woodward, and C. Aspin. V1309 Scorpii. *International Astronomical Union Circular*, 8997:2, October 2008b.

D. Saumon, G. Chabrier, and H. M. van Horn. An Equation of State for Low-Mass Stars and Giant Planets. *The Astrophysical Journal Supplement Series*, 99:713, August 1995. doi: 10.1086/192204.

D. Semenov, Th. Henning, Ch. Helling, M. Ilgner, and E. Sedlmayr. Rosseland and Planck mean opacities for protoplanetary discs. *A&A*, 410:611–621, November 2003. doi: 10.1051/0004-6361:20031279.

Noam Soker, Aldana Grichener, and Efrat Sabach. Radiating the Hydrogen Recombination Energy during Common Envelope Evolution. *ApJ*, 863:L14, August 2018. doi: 10.3847/2041-8213/aad736.

D. Stamatellos, A. P. Whitworth, T. Bisbas, and S. Goodwin. Radiative transfer and the energy equation in SPH simulations of star formation. *A&A*, 475: 37–49, November 2007. doi: 10.1051/0004-6361:20077373.

K. Stępień. Evolution of the progenitor binary of V1309 Scorpii before merger. *A&A*, 531:A18, July 2011. doi: 10.1051/0004-6361/201116689.

Matthias Steinmetz. GRAPESPH: cosmological smoothed particle hydrodynamics simulations with the special-purpose hardware GRAPE. *MNRAS*, 278:1005–1017, February 1996. doi: 10.1093/mnras/278.4.1005.

J. Stillwell. *Mathematics and Its History*. Springer, New York, 1989. doi: 10.1007/978-1-4899-0007-4.

R. Tylenda. Evolution of V838 Monocerotis during and after the 2002 eruption. *A&A*, 436:1009–1020, Jun 2005. doi: 10.1051/0004-6361:20052800.

R. Tylenda and N. Soker. Eruptions of the V838 Mon type: stellar merger versus nuclear outburst models. *A&A*, 451:223–236, May 2006. doi: 10.1051/0004-6361:20054201.

R. Tylenda, M. Hajduk, T. Kamiński, A. Udalski, I. Soszyński, M. K. Szymański, M. Kubiak, G. Pietrzyński, R. Poleski, L. Wyrzykowski, and K. Ulaczyk. V1309 Scorpii: merger of a contact binary. *A&A*, 528:A114, April 2011. doi: 10.1051/0004-6361/201016221.

A. Udalski. The Optical Gravitational Lensing Experiment. Real Time Data Analysis Systems in the OGLE-III Survey. *AcA*, 53:291–305, December 2003.

A. Udalski, M. K. Szymanski, I. Soszynski, and R. Poleski. The Optical Gravitational Lensing Experiment. Final Reductions of the OGLE-III Data. *AcA*, 58:69–87, June 2008.

M. V. van der Sluijs. *Formation and Evolution of Compact Binaries*. Utrecht University, 2006. ISBN 978-90-393-4234-3. pp. 144.

Chen Wang, Kun Jia, and Xiang-Dong Li. The binding energy parameter for common envelope evolution. *Research in Astronomy and Astrophysics*, 16: 126, August 2016. doi: 10.1088/1674-4527/16/8/126.

R. F. Webbink. Double white dwarfs as progenitors of R Coronae Borealis stars and type I supernovae. *ApJ*, 277:355–360, February 1984. doi: 10.1086/161701.

Holger Wendland. Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Advances in Computational Mathematics*, 4(1):389–396, Dec 1995. ISSN 1572-9044. doi: 10.1007/BF02123482. URL `https://doi.org/10.1007/BF02123482`.

Q. Zhu, L. Hernquist, and Y. Li. Numerical Convergence In Smoothed Particle Hydrodynamics. *ApJ*, 800:6, February 2015. doi: 10.1088/0004-637X/800/1/6.