University of Southampton

Faculty of Social Sciences
School of Mathematical Sciences

Numerical Evolution of Binary Neutron Star Mergers

by

Peter Christopher Hammond
MPhys
ORCiD: 0000-0002-9447-1043

A thesis for the degree of
Doctor of Philosophy

August 2022

University of Southampton

Abstract

Faculty of Social Sciences
School of Mathematical Sciences

Doctor of Philosophy

Numerical Evolution of Binary Neutron Star Mergers

by Peter Christopher Hammond

Binary neutron star mergers feature some of the most extreme physical conditions known throughout the universe: densities exceeding that of atomic nuclei, temperatures in excess of 10^{11} K, and speeds approaching 0.5 c. Due to these conditions, they provide an excellent test bed for theoretical physics at the limit of our current understanding, and beyond. In order to learn about these phenomena through observation we need to first obtain an idea of what sorts of signals we should be looking for, and then we need to determine what we can learn from analysing this data. Numerical simulations of neutron star mergers can give us answers to these questions. However, binary neutron star mergers pose a significant computational challenge: they involve physics on greatly differing length scales, and systems of highly non-linear coupled partial differential equations. In this thesis we present state-of-the-art simulations of binary neutron star mergers. We cover: the theoretical background of these simulations, the software we use to perform simulations, analysis and visualisation techniques used for studying simulations, our work on the extension of the simulations to include new physics, and results from such simulations.

Through these simulations we explore concepts central to the extraction of information from multi-messenger observation of binary neutron star mergers, namely the reactions between protons, neutrons, and electrons in neutron star fluid, the equilibrium they work to establish, and the bulk viscosity resulting from out-of-equilibrium conditions. We show these reactions affect not only electromagnetic and neutrino radiation observed in merger simulations, they need to be implemented to ensure accurate gravitational waveform templates are produced for third generation detectors. We go on to show that the notion of equilibrium between species in the fluid is complex, depending on both local conditions (particularly the temperature of the fluid), and also on the behaviour of surrounding matter. Finally, we discuss the difficulty in implementing reactions in simulations, with a view of modelling bulk viscosity. We show that the timescales involved vary over several orders of magnitude, with the slower behaviour being easily resolvable in current simulations, and the fast behaviour being far out of reach. An approach that includes the effects of both resolvable and unresolvable reactions, with a consistent treatment of the crossover, is therefore needed to properly model bulk viscosity.

Contents

List of Figures

List of Tables

Declaration of Authorship

I declare that this thesis and the work presented in it is my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. Parts of this work have been published as:

   P. Hammond, I. Hawke, and N. Andersson. Thermal aspects of neutron star mergers. *Physical Review D*, 104(10), Nov. 2021. doi:10.1103/physrevd.104.103006
   T. Celora, I. Hawke, P. C. Hammond, N. Andersson, and G. L. Comer. Formulating bulk viscosity for neutron star simulations. *Physical Review D*, 105(10), May 2022. doi:10.1103/physrevd.105.103016
   P. Hammond, I. Hawke, and N. Andersson. Detecting the impact of nuclear reactions on neutron star mergers through gravitational waves, 2022. URL https://arxiv.org/abs/2205.11377. Submitted to *Physical Review Letters*

Signed:..... Date:.....

Acknowledgements

Almost nine years have passed since I started at Southampton. In that time I have had the privilege of meeting a great many wonderful people, all of whom have made an impact on me. Obviously I cannot name you all, but I would like to take the time here to highlight those whose companionship and support has had the greatest impact.

I know it is cliché, but the first thanks must go to my family. For their unwavering support, especially when the "C" word forced me to move back in with them, I am indebted to my parents. I don't, and cannot, say it enough: thank you for putting up with me, without you, this work would not have been possible.

Next, I would like to thank (and probably apologise to) all those with whom I have lived over the past four years. Will, Aaron and Elliot, and Guy and Sam, while our time together was in some cases cut short, you were each a pleasure to live with, and I deeply enjoyed being part of all three households that I lived in as a PGR.

It would, of course, be remiss of me not to mention the welcoming and inclusive community that each and every PGR in maths, past and present, has helped nurture. I would like to thank, in particular, all those of you with whom I have shared an office for the conversation, both scientific and otherwise, that made working with you all so pleasant. In a similar vein, I very much enjoyed the time spent completing puzzles and crosswords at lunch, and beating you all at ultimate noughts and crosses with PeakyStop.

It is music that has led me to some of the best and deepest friendships I have made while in Southampton. Bex, Ed, Joe, Paddy, Sophie, Stella, and Will, you are, all of you, the best humans. You know, as well as I do, that I do not have the words to express how much your friendship means to me. The weekly chats, the wine and cheese nights, the weekends away, all the time we have spent together, it has always been immense pleasure to share that time with you.

Last, but by no means least, I must thank Nils and Ian. Dipping again into the world of clichés, while it is my name at the top of this thesis, the contributions made by my supervisory dynamic duo cannot be understated. It has been a wild ride, and while we have spent a large part of the last four years sequestered apart, I have always felt the presence of your support and guidance. While we may not have achieved all that we wanted to with this project thanks to the "C" word, I am very proud of the work we have done, and I hope my moving on does not spell the end of our collaboration.

On the Shoulders of Giants

Chapter 1

Introduction

1.1 Motivation

Binary neutron star mergers provide some of the most impressive fireworks in the cosmos. Reaching densities inaccessible to terrestrial laboratories and temperatures of up to several 10s of MeVs they provide a window into physics at the very edge of our understanding, and indeed beyond. GW170817[1;2;4] was a significant milestone in multi-messenger astronomy: for the first time we observed an event not only with gravitational radiation, we also captured the electromagnetic counterpart, in particular the short gamma-ray burst associated with the merger, and the kilonova associated with r-process nucleosynthesis in outflowing matter[43].

The merger of two compact objects is a much studied problem, especially in the case that the two objects are black holes, but current analytical models break down as the orbital separation of the two objects approaches a few times their radii[1]. This means that in order to study these systems from the late inspiral and beyond, we must model the problem using numerical methods. The modelling of neutron star mergers still poses a significant computational challenge[22;111]: as we develop increasingly complex models that, we hope, more closely reflect reality in order to keep up with evolving detector technology, we need increasingly sophisticated methods and more computational power to perform the simulations. Using the results of simulations and observations from future high-precision gravitational wave observatories like the Einstein Telescope[73;89;106;134], and the Cosmic Explorer[56;114;133] we hope to understand the behaviour of matter under the extreme conditions experienced in neutron star mergers.

This thesis is an exploration of the effects of nuclear reactions and the associated equilibrium on binary neutron star mergers, in particular how these might affect any

[1]See Dietrich et al.[50] for a recent review of approximants to the inspiral gravitational wave signal.

detected gravitation wave signals, through the development and performance of state-of-the-art simulations. This problem lies at the forefront of the field, having implications for gravitational wave, electromagnetic, and neutrino observations of mergers, and is of vital importance for our understanding of the nuclear physics probed these events.

1.2 Outline

This thesis is divided into two parts: in part I we discuss the tools necessary to perform cutting edge simulations of binary neutron star mergers, and our contribution to them, and in part II we present the results of said simulations, and three publications related thereto: Hammond et al.[68], Celora et al.[37], and Hammond et al.[69].

The simulation of binary neutron star mergers effectively consists of solving two problems: the dynamics of the fluid that makes up the stars, and those of the spacetime it occupies. These two problems are strongly coupled: the fluid influences the curvature of the spacetime, and this curvature influences the movement of the fluid. In chapters 2 and 3 we will cover the theoretical basis of the dynamics of the spacetime and fluid respectively. Chapter 4 discusses the method by which initial data is generated, and the method by which the main piece of software we use, the Einstein Toolkit, evolves the spacetime and fluid components of simulations. In chapter 5 we discuss the methods by which the data resulting from a simulation is analysed and visualised, and what we can expect to learn from this data, and in chapter 6 we discuss the work we are doing to extend the modelling of the equation of state of the neutron star fluid to more realistic cases motivated by nuclear and particle physics, and what these more realistic equations of state can tell us.

In chapter 7 we present the results from simulations based upon the above, giving a breakdown of the phases of a binary neutron star merger, and discuss effects related to temperature and composition. In the final chapters we go into detail on how nuclear reactions can affect the dynamics in a merger. We discuss equilibrium at zero-temperature in chapter 8, and the effects of deviation therefrom. In chapter 9 we consider how the picture changes as finite temperature effects are included, and finally in chapter 10 we give a discussion of bulk viscosity.

1.3 Conventions

In this thesis we follow the sign conventions of Misner et al.[94]. We use Greek letters for spacetime indicies on tensors (and as we always work with cartesian-like grids we use the letters (t, x, y, z) to make explicit reference to each of the dimensions), and Latin letters (except t, x, y, z) for spacial indices (which take the values (x, y, z)). In general

Quantity	Expression	Value in SI
Length	$\frac{M_\odot G}{c^2}$	$1.476 \times 10^3\,\mathrm{m}$
Mass	$M_\odot$	$1.988 \times 10^{30}\,\mathrm{kg}$
Time	$\frac{M_\odot G}{c^3}$	$4.924 \times 10^{-6}\,\mathrm{s}$

TABLE 1.1: Table of basic units in the $c = G = 1$ system. Unit conversions are performed using $c = 299792458\,\mathrm{ms}^{-2}$, $G = 6.674 \times 10^{-11}\,\mathrm{m}^3\mathrm{kg}^{-1}\mathrm{s}^{-2}$, and $M_\odot = 1.988 \times 10^{30}\,\mathrm{kg}$.

we denote vectors using bold and italic font (e.g. $\boldsymbol{v}$). The units used in simulations are chosen such that $c = G = k_B = 1$, where values are quoted in multiples of the solar mass $M_\odot$ (see table 1.1), and the unit of temperature is MeV. In the majority of this work we will also use these units. It is common, however, in the field of astrophysics to use the centimetre-gram-second (cgs) system of units along with kilometers where appropriate, so when we reach chapter 5 we will follow this convention, continuing to use MeV as the unit of temperature, which will also affect quantities whose units involve temperature, such as entropy S. The conversion from MeV to K is $T\,[\mathrm{K}] \approx 1.160 \times 10^9\,T\,[\mathrm{MeV}]$.

Part I

Background

Chapter 2

Numerical Relativity

The Einstein field equations, which describe how matter and energy curve spacetime, can be written as

$$G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu},$$

where $G_{\mu\nu}$ is the Einstein tensor—which we associate with the curvature of spacetime—and $T_{\mu\nu}$ is the stress energy tensor—which we associate with the matter contained within said spacetime–assuming the cosmological constant Λ is 0. This equation when written out fully yields 10 coupled non-linear hyperbolic-elliptic partial differential equations. In this chapter we discuss how the equations must be reformulated in order to evolve binary neutron star spacetimes using numerical methods. The standard method for doing this is using the 3+1 formalism [6].

2.1 3+1 Formalism

The 3+1 formalism is used to split 4-dimensional spacetime into 3 dimensions of space and 1 of time. This allows us to set our simulation up as an initial value problem, and evolve the spacial solutions in time. For a more complete discussion of the 3+1 formalism see Alcubierre [6].

2.1.1 Splitting the Metric

We start by considering a general globally hyperbolic spacetime[1] with metric $g_{\mu\nu}$. We can slice this spacetime into a set of spacelike hypersurfaces Σ_t which we label with a *universal time parameter* t_k (note that t_k does not necessarily coincide with the proper

[1] There are several definitions of globally hyperbolic, but the important point here is that there are no closed timelike curves [6].

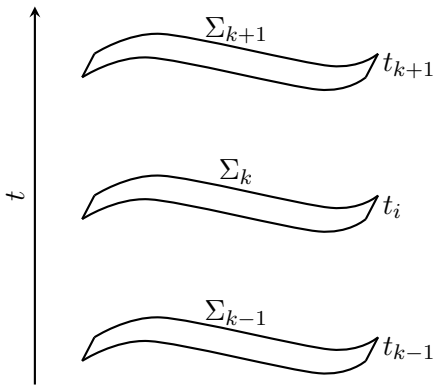

FIGURE 2.1: Division of spacetime into 3-dimensional spacelike hypersurfaces.

time as measured by an observer) as demonstrated in fig. 2.1. This set of hypersurfaces is commonly called a *foliation*.

Suppose we take two adjacent hypersurfaces from a specific foliation Σ_t and Σ_{t+dt}. We can determine the geometry between the two hypersurfaces through three equations (see fig. 2.2 for a graphical representation):

– We define proper distances within each hypersurface using

$$dl^2 = \gamma_{ij} dx^i dx^j, \tag{2.1}$$

where γ_{ij} is the 3-dimensional spacial metric of each hypersurface.

– We define how proper time τ changes between the two hypersurfaces as measured by observers moving along the normal to the hypersurfaces (these are known as *Eulerian* observers):

$$d\tau = \alpha\left(t, x^i\right) dt, \tag{2.2}$$

where α is called the *lapse function*.

– Finally, we define the *shift vector* β^i as the relative velocity between Eulerian observers and lines of constant coordinate position:

$$x^i_{t+dt} = x^i_t - \beta^i\left(t, x^i\right) dt. \tag{2.3}$$

From eqs. (2.1) to (2.3) we can read off the metric in terms of the new variables, giving us

$$ds^2 = \left(-\alpha^2 + \beta_i \beta^i\right) dt^2 + 2\beta_i dt dx^i + \gamma_{ij} dx^i dx^j, \tag{2.4}$$

where we have lowered indices using γ_{ij}. Equation (2.4) is known as the 3+1 split of the metric. Note that it is possible to define the quantities α and β^i in a more coordinate independent way (see [6]), but this method provides a better intuition as to their meaning.

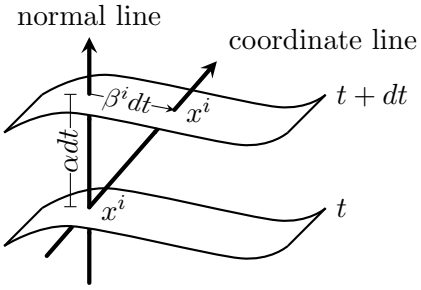

FIGURE 2.2: Adjacent spacelike hypersurfaces with definitions of shift vector β^i and lapse function α.

Writing the metric out more explicitly as

$$g_{\mu\nu} = \begin{pmatrix} -\alpha^2 + \beta_i\beta^i & \beta_i \\ \beta_i & \gamma_{ij} \end{pmatrix} \tag{2.5}$$

$$g^{\mu\nu} = \frac{1}{\alpha^2}\begin{pmatrix} -1 & \beta^i \\ \beta^i & \alpha^2\gamma^{ij} - \beta^i\beta^j \end{pmatrix}, \tag{2.6}$$

we can see that the 4-dimensional volume element becomes

$$\sqrt{-g} = \alpha\sqrt{\gamma}, \tag{2.7}$$

where g and γ are the determinants of $g_{\mu\nu}$ and γ_{ij} respectively.

We will also require the normal vector n_μ to the hypersurface Σ_t which is given (in co- and contra-variant form) by

$$n_\mu = (-\alpha, 0) \tag{2.8}$$

$$n^\mu = \left(1/\alpha, -\beta^i/\alpha\right). \tag{2.9}$$

2.1.2 Measuring curvature

Given a set of hypersurfaces that make up a spacetime foliation, each will possess two forms of curvature:

– *Intrinsic* curvature coming from the internal geometry of each hypersurface,

– *Extrinsic* curvature associated with the embedding of each 3-dimensional hypersurface into 4-dimensional spacetime.

The intrinsic curvature can be measured using the 3-dimensional Riemann tensor as given by the spacial metric of a hypersurface γ_{ij}, whereas the extrinsic curvature

can be measured using parallel transport of the unit normal vector $\boldsymbol{n}$ within a given hypersurface.

We start by defining the projection operator onto a hypersurface P^{α}_{β} as

$$P^{\alpha}_{\beta} = \delta^{\alpha}_{\beta} + n^{\alpha} n_{\beta}, \tag{2.10}$$

which allows us to then define the extrinsic curvature $K_{\mu\nu}$ as

$$\begin{aligned} K_{\mu\nu} &= -P^{\alpha}_{\mu} \nabla_{\alpha} n_{\nu} \\ &= -\frac{1}{2} \mathcal{L}_{\boldsymbol{n}} \gamma_{\mu\nu}, \end{aligned} \tag{2.11}$$

where we have also given the definition in terms of the Lie derivative of the 4-metric of a hypersurface $\gamma_{\mu\nu}$, which is defined as

$$\gamma_{\mu\nu} = g_{\mu\nu} + n_{\mu} n_{\nu}. \tag{2.12}$$

The Lie derivative $\mathcal{L}_{\boldsymbol{V}}$ along a vector field V^{α} gives a measure of how a tensor field changes when transported along said vector field when compared to an infinitesimal coordinate change generated by V^{α}. For a general scalar field ϕ this goes like

$$\mathcal{L}_{\boldsymbol{V}} \phi = V^{\alpha} \partial_{\alpha} \phi, \tag{2.13}$$

for a general vector v^{μ} and covector v_{μ} we have

$$\mathcal{L}_{\boldsymbol{V}} v^{\mu} = V^{\alpha} \partial_{\alpha} v^{\mu} - v^{\alpha} \partial_{\alpha} V^{\mu}, \tag{2.14}$$

$$\mathcal{L}_{\boldsymbol{V}} v_{\mu} = V^{\alpha} \partial_{\alpha} v_{\mu} + v_{\alpha} \partial_{\mu} V^{\alpha}. \tag{2.15}$$

For a mixed second rank tensor T^{μ}_{ν} we have

$$\mathcal{L}_{\boldsymbol{V}} T^{\mu}_{\nu} = V^{\alpha} \partial_{\alpha} T^{\mu}_{\nu} - T^{\alpha}_{\nu} \partial_{\alpha} V^{\mu} + T^{\mu}_{\alpha} \partial_{\nu} V^{\alpha}, \tag{2.16}$$

from which the generalisation to any tensor field is evident. It is also worth noting that we can also construct the projection operator above using $\gamma_{\mu\nu}$ through

$$\begin{aligned} \gamma^{\alpha}_{\beta} &= g^{\alpha\mu} \gamma_{\beta\mu} \\ &= g^{\alpha}_{\beta} + n^{\alpha} n_{\beta} \\ &= \delta^{\alpha}_{\beta} + n^{\alpha} n_{\beta} \\ &= P^{\alpha}_{\beta}. \end{aligned} \tag{2.17}$$

As $\boldsymbol{n}$ is always normal to the hypersurface, we have for any scalar function ϕ the relation

$$\mathcal{L}_{\boldsymbol{n}} \gamma_{\mu\nu} = \frac{1}{\phi} \mathcal{L}_{\phi \boldsymbol{n}} \gamma_{\mu\nu}. \tag{2.18}$$

By setting $\phi = \alpha$, we can manipulate eq. (2.11) to give

$$\begin{aligned} \mathcal{L}_{\alpha \boldsymbol{n}} \gamma_{\mu\nu} &= -2\alpha K_{\mu\nu} \\ &= (\mathcal{L}_{\boldsymbol{t}} - \mathcal{L}_{\boldsymbol{\beta}})\, \gamma_{\mu\nu}, \end{aligned} \tag{2.19}$$

where we have also used the relation $\alpha \boldsymbol{n} = \boldsymbol{t} - \boldsymbol{\beta}$.

Focusing on the spacial components of the two right hand sides of eq. (2.19) and using the relation $\mathcal{L}_{\boldsymbol{t}} = \partial_t$ (in 3+1 language) we can write

$$\partial_t \gamma_{ij} = -2\alpha K_{ij} + D_i \beta_j + D_j \beta_i, \tag{2.20}$$

where we have rewritten $\mathcal{L}_{\boldsymbol{\beta}}$ in terms of the spacial components of the projection of the covariant derivative onto the hypersurface

$$D_\mu = P^\alpha_\mu \nabla_\alpha. \tag{2.21}$$

Equation (2.20) represents the beginnings of a method for solving the Einstein field equations as an initial value problem given a spacetime represented by $g_{\mu\nu}$: we have an equation for the evolution of γ_{ij}, but we also need to determine the evolution of K_{ij}. To help us with this we obtain some constraints from the contraction and projection of Einstein field equations with n_μ and P^α_β respectively.

2.1.3 Constraining the Einstein field equations

We start by expressing the full 4-dimensional Riemann curvature tensor $R^\alpha_{\beta\mu\nu}$ in terms of the intrinsic curvature of the hypersurface as given by the 3-dimensional Riemann curvature tensor ${}^{(3)}R^\alpha_{\beta\mu\nu}$ and the extrinsic curvature of the hypersurface as given by $K_{\mu\nu}$. This brings us to the *Gauss-Codazzi* equations [2], which are given by

$$P^\delta_\alpha P^\kappa_\beta P^\lambda_\mu P^\sigma_\nu R_{\delta\kappa\lambda\sigma} = {}^{(3)}R_{\alpha\beta\mu\nu} + K_{\alpha\mu} K_{\beta\nu} - K_{\alpha\nu} K_{\beta\mu}. \tag{2.22}$$

We also require the projection on the hypersurface of $R^\alpha_{\beta\mu\nu}$ contracted with the normal vector, which yields the *Codazzi-Mainardi* equations

$$P^\delta_\alpha P^\kappa_\beta P^\lambda_\mu n^\nu R_{\delta\kappa\lambda\nu} = D_\beta K_{\alpha\mu} - D_\alpha K_{\beta\mu}. \tag{2.23}$$

Next, we note that by twice contracting the 4-dimensional Riemann tensor with the projection operator we obtain

$$P^{\alpha\mu} P^{\beta\nu} R_{\alpha\beta\mu\nu} = 2 n^\mu n^\nu G_{\mu\nu}, \tag{2.24}$$

[2] For a full derivation, see Wald [139].

whereas eq. (2.22) implies

$$P^{\alpha\mu}P^{\beta\nu}R_{\alpha\beta\mu\nu} = {}^{(3)}R + K^2 - K_{\mu\nu}K^{\mu\nu}. \tag{2.25}$$

Equations (2.24) and (2.25) together with the Einstein field equations and the definition of the local energy density[3] as measured by Eulerian observers $E = n^\mu n^\nu T_{\mu\nu}$ gives us

$${}^{(3)}R + K^2 - K_{\mu\nu}K^{\mu\nu} = 16\pi E, \tag{2.26}$$

which is known as the *energy* (or *Hamiltonian*) constraint. Note that there are no time derivatives in eq. (2.26), it is a constraint on the local energy density that must be satisfied at all times.

If we instead take the mixed contraction of the Einstein tensor with the hypersurface projection and the normal vector, we obtain

$$P^{\alpha\mu}n^\nu G_{\mu\nu} = P^{\alpha\mu}n^\nu R_{\mu\nu}, \tag{2.27}$$

which together with eq. (2.23) implies

$$\gamma^{\alpha\mu}n^\nu G_{\mu\nu} = D^\alpha K - D_\mu K^{\alpha\mu}. \tag{2.28}$$

Utilising eq. (2.28) along with the Einstein field equations we obtain

$$D_\mu \left(K^{\alpha\mu} - \gamma^{\alpha\mu}K\right) = 8\pi S^\alpha, \tag{2.29}$$

where we have defined $S^\alpha = -P^{\alpha\mu}n^\nu T_{\mu\nu}$ which corresponds to the momentum density as measured by Eulerian observers[4]. These 3 equations (the index α is free to take any of the 4 values $\alpha \in (t, x, y, z)$, but the t component is trivial), which again contain no time derivatives, are known as the *momentum* constraints.

By moving into a coordinate system adapted to our foliation of spacetime, we obtain the final form of our 4 constraints:

$${}^{(3)}R + K^2 - K_{ij}K^{ij} = 16\pi E, \tag{2.30}$$

$$D_j \left(K^{ij} - \gamma^{ij}K\right) = 8\pi S^i, \tag{2.31}$$

where the definition of E does not change, and $S^i = -P^{i\mu}n^\nu T_{\mu\nu}$. It is also important to notice that while these constraints contain no time derivatives, they also have no dependence on the relationship between hypersurfaces (neither α nor β^i appear in either), so they must be satisfied on each hypersurface.

[3] Not to be confused with either the rest mass density ρ or the total energy density e in the fluid frame.

[4] Not to be confused with the rest mass density current J^α.

We also have a choice as to whether we impose these constraints on the simulation, or calculate the degree to which they are violated as the simulation progresses. This is the question of *constrained* versus *free evolution*. However, our choices are rather limited: it is difficult to obtain a system of equations for constrained evolution without taking advantage of high degrees of spacial symmetry (e.g. spherical symmetry, which will not be generally possible for neutron star mergers) [6;115], the mathematical properties of such a system are difficult to analyse (e.g. well-posedness) [6;115], and as the constraints will involve the solution of elliptical equations [6;18;115] they will be computationally expensive. Hence, we focus on freely-evolved systems.

The final note we make in this section is that given the 12 independent quantities contained in γ_{ij} and K_{ij}, the constraints imply that these cannot all be specified freely, hence the initial data must also satisfy these constraints in order for the evolution to be valid.

2.1.4 The ADM equations

From eqs. (2.30) and (2.31) we have 4 of the 10 independent equations contained within the Einstein field equations, the other 6 are contained in the time-evolution of K_{ij}. The next relation we need is the Riemann curvature tensor twice contracted with the normal vector and projected onto the hypersurfaces, which is given by

$$P^{\delta}_{\mu} P^{\kappa}_{\nu} n^{\lambda} n^{\sigma} R_{\delta\kappa\lambda\sigma} = \mathcal{L}_{\boldsymbol{n}} K_{\mu\nu} + K_{\mu\lambda} K^{\lambda}_{\nu} + \frac{1}{\alpha} D_{\mu} D_{\nu} \alpha, \tag{2.32}$$

which, together with eq. (2.22) and the Einstein field equations, can be used to show

$$\begin{aligned} \mathcal{L}_{\boldsymbol{t}} K_{\mu\nu} - \mathcal{L}_{\boldsymbol{\beta}} K_{\mu\nu} = -D_{\mu} D_{\nu} \alpha + \alpha \left({}^{(3)}R_{\mu\nu} + K K_{\mu\nu} - 2 K_{\mu\lambda} K^{\lambda}_{\nu} \right) \\ + 4\pi\alpha \left(\gamma_{\mu\nu} \left(S - E \right) - 2 S_{\mu\nu} \right), \end{aligned} \tag{2.33}$$

where we have defined $S_{\mu\nu} = P^{\alpha}_{\mu} P^{\beta}_{\nu} T_{\alpha\beta}$ (and $S = S^{\mu}_{\mu}$) which corresponds to the stress-energy tensor as measured by an Eulerian observer. By concentrating on spacial indices and identifying $\mathcal{L}_{\boldsymbol{t}}$ with ∂_t, we arrive at a set of evolution equations for K_{ij}, given by

$$\begin{aligned} \partial_t K_{ij} = \mathcal{L}_{\boldsymbol{\beta}} K_{ij} - D_i D_j \alpha + \alpha \left({}^{(3)}R_{ij} + K K_{ij} - 2 K_{ik} K^{k}_{j} \right) \\ + 4\pi\alpha \left(\gamma_{ij} \left(S - E \right) - 2 S_{ij} \right), \end{aligned} \tag{2.34}$$

which are known as the *Arnowitt-Deser-Misner* (ADM) equations. The ADM equations give us the final 6 independent equations in the Einstein field equations, allowing us to pose the Einstein field equations as an initial value problem.

There is, however, a problem: using the ADM equations directly to evolve a spacetime is numerically unstable [27]. There are a number of solutions to this problem (we discuss

another formulation, CCZ4, in section 4.4, see also discussions in Alcubierre[6] and Rezzolla and Zanotti[115]) so we will focus here on one in particular: the BSSNOK (or BSSN) formulation, based on the work of Nakamura, Oohara, and Kojima[98] and built upon by Shibata and Nakamura[125], and Baumgarte and Shapiro[27], the principle idea being to introduce a conformal factor and evolve this and the trace of the extrinsic curvature K separately.

2.2 BSSN

To derive the BSSN formulation, we start by introducing a conformal factor ψ by rescaling the spacial metric of the hypersurfaces

$$\tilde{\gamma}_{ij} = \psi^{-4}\gamma_{ij}, \tag{2.35}$$

where $\tilde{\gamma}_{ij}$ is the conformal metric, and the value of ψ is free to evolve in time. There are a number of different ways of choosing a value for ψ, but a particularly helpful choice is one such that $\tilde{\gamma}_{ij}$ has unitary determinant, giving

$$\psi^4 = \gamma^{1/3} \rightarrow \psi = \gamma^{1/12}, \tag{2.36}$$

where γ is the determinant of the non-conformal metric. Using eq. (2.20) and the substitution $\psi = \exp(\phi)$, we can write down an evolution equation for ϕ

$$\partial_t \phi = -\frac{1}{6}\left(\alpha K - \partial_i \beta^i\right) + \beta^i \partial_i \phi. \tag{2.37}$$

Another ingredient in the BSSN formulation is the separation of the extrinsic curvature into a trace-free part A_{ij} to be evolved separately from the trace K. This trace free part is given by

$$A_{ij} = K_{ij} - \frac{1}{3}\gamma_{ij}K, \tag{2.38}$$

which we conformally rescale to give

$$\tilde{A}_{ij} = e^{-4\phi}A_{ij}. \tag{2.39}$$

Finally, we introduce the *conformal connection functions* $\tilde{\Gamma}^i$ which are defined as the contraction of the lower two indices of the Christoffel symbols associated with the conformally rescaled metric

$$\begin{aligned} \tilde{\Gamma}^i &= \tilde{\gamma}_{jk}\tilde{\Gamma}^i_{jk} \\ &= -\partial_j \tilde{\gamma}^{ij}, \end{aligned} \tag{2.40}$$

where the second line comes from the definition of the Christoffel symbols of a metric with determinant 1 (which is how we have constructed $\tilde{\gamma}_{ij}$).

Now we have defined all of the new variables we require, we can start to re-write the ADM equations in terms of these new variables. We already have an evolution equation for ϕ in eq. (2.37), and we can use eq. (2.34) and the definition $d/dt := \partial_t - \mathcal{L}_{\boldsymbol{\beta}}$ to write evolution equations for $\tilde{\gamma}_{ij}$, $\tilde{A}_{ij}$, and K:

$$\frac{d}{dt}\tilde{\gamma}_{ij} = -2\alpha\tilde{A}_{ij}, \tag{2.41}$$

$$\begin{aligned}\frac{d}{dt}\tilde{A}_{ij} = &\left\{-D_iD_j\alpha + \alpha R_{ij} + 4\pi\alpha\left[\gamma_{ij}\left(S-E\right) - 2S_{ij}\right]\right\}^{\mathrm{TF}} \\ &+\alpha\left(K\tilde{A}_{ij} - 2\tilde{A}_{ik}\tilde{A}^k_j\right),\end{aligned} \tag{2.42}$$

$$\frac{d}{dt}K = -D_iD^i\alpha + \alpha\left(\tilde{A}_{ij}\tilde{A}^{ij} + \frac{1}{3}K^2\right) + 4\pi\alpha\left(E+S\right), \tag{2.43}$$

where $\{...\}^{\mathrm{TF}}$ denotes the trace-free part of the enclosed expression. We can obtain the evolution of the conformal connection functions from eqs. (2.20), (2.31) and (2.40). This gives us

$$\begin{aligned}\frac{d}{dt}\tilde{\Gamma}^i = &\tilde{\gamma}^{jk}\partial_j\partial_k\beta^i + \frac{1}{3}\tilde{\gamma}^{ij}\partial_j\partial_k\beta^k - 2\tilde{A}^{ij}\partial_j\alpha \\ &+ 2\alpha\left(\tilde{\Gamma}^i_{jk}\tilde{A}^{jk} + 6\tilde{A}^{ij}\partial_j\phi - \frac{2}{3}\tilde{\gamma}^{ij}\partial_j K - 8\pi\tilde{S}^i\right),\end{aligned} \tag{2.44}$$

where we have defined $\tilde{S}^i = e^{4\phi}S^i$. Thus our final system of equations for spacetime evolution is given by eqs. (2.37) and (2.41) to (2.44).

The difference in stability between BSSN and ADM can be explained by studying the well-posedness of the two formulations: the ADM formulation is only weakly hyperbolic, hence it is ill-posed, whereas the BSSN formulation is strongly hyperbolic, and thus well-posed [28]. A complete discussion on the differing notions of hyperbolicity in this context can be found in Baumgarte and Shapiro [28], however the key result here is that while both strongly- and weakly-hyperbolic systems can be cast as initial value problems, only for strongly-hyperbolic systems can it be guaranteed that the solutions grow less rapidly than exponentially, meaning the solution will depend smoothly on the initial data, hence well-posed.

2.3 Gauge Freedom

With the system of equations above, we still have the freedom to decide on a set of *gauge conditions*. Namely, on each hypersurface we need to specify a *slicing condition* through the lapse function α, and a *spacial shift condition* through the shift vector β^i.

The choice of gauge conditions has no impact on observable quantities, but can, if done poorly, lead to disastrous numerical results[6;35;128].

Here we will cover the commonly used "1+log" slicing condition, and the "Gamma-driver" spacial slicing condition, following Rezzolla and Zanotti[115]. For a detailed discussion of gauge choices see Alcubierre[6] and Baumgarte and Shapiro[27].

A set of optimal gauge conditions for a simulation should be formulated such that:

– singularities present in the spacetime are avoided,

– coordinate distortions caused by large spacial curvatures are reduced,

– the gauge is prescribed in the least computationally expensive manner (preferably through an algebraic expression or as the solution to a set of simple evolution equations).

2.3.1 Slicing condition

The first point, often referred to as the *singularity-avoiding slicing condition*, is satisfied by ensuring that α, and hence the speed of proper time evolution, is small in the vicinity of any present singularity. This condition can be imposed by requiring that the coordinate volume elements associated with Eulerian observers are both maximal, and do not change with time, i.e. $K = \partial_t K = 0$, which is known as the *maximal slicing condition*. Using the ADM equations and the Hamiltonian constraint (eq. (2.30)) it can be shown that the maximal slicing condition can be recast as

$$D^2\alpha = \alpha \left(K_{ij}K^{ij} + 4\pi \left(E + S \right) \right), \tag{2.45}$$

where $D^2 = \gamma^{ij}\partial_i\partial_j$ is the Laplacian associated with the 3-metric γ_{ij}. This elliptical equation must be solved on every spacial hypersurface Σ_t, which is very computationally costly for $3+1$-dimensional numerical spacetimes. Many authors (see Alcubierre[6] for a historical account) have attempted to imitate the properties of maximal slicing condition with computationally cheaper methods, with the hyperbolic *Bona-Massó* family of slicing conditions[31] becoming the most prevalent. This family of conditions can be written generally as

$$\partial_t\alpha - \beta^k\partial_k\alpha = -f(\alpha)\alpha^2 \left(K - K_0 \right), \tag{2.46}$$

where $f(\alpha) > 0$ is a general function that distinguishes between the members of the family, and $K_0 = K(t = 0)$ is the initial trace of the extrinsic curvature. The *"1+log" slicing condition* emerges by setting $f(\alpha) = q/\alpha$ for a positive integer q, which leads to the solution

$$\alpha = g(x^i) + \ln(\gamma^{q/2}), \tag{2.47}$$

where $g(x^i)$ is a positive, time-independent, but otherwise arbitrary function of spacial position. This slicing condition mimics the behaviour of the maximal slicing condition

well[19], and has been shown to be robust in both isolated and binary neutron star simulations[24;142].

2.3.2 Shift condition

In order to satisfy the second point, we must lay out the spacial coordinates in such a way that changes in size and shape are minimised when moving from one spacial hypersurface to the next. We can measure these distortions through the *metric strain tensor* along the time direction $\boldsymbol{t}$, Θ_{ij}, defined as[128]

$$\begin{aligned}\Theta_{ij} &= \frac{1}{2}\mathcal{L}_{\boldsymbol{t}}\gamma_{ij} \\ &= \frac{1}{2}\left(\alpha\mathcal{L}_{\boldsymbol{n}} + \mathcal{L}_{\boldsymbol{\beta}}\right)\gamma_{ij} \\ &= -\alpha K_{ij} + \frac{1}{2}\mathcal{L}_{\boldsymbol{\beta}}\gamma_{ij}.\end{aligned} \tag{2.48}$$

Integrating $\Theta_{ij}\Theta^{ij}$ over a spacial slice and minimising the variation by setting $D_j\Theta^{ij} = 0$ leads to the *minimal-strain shift condition*:

$$D^2\beta^i + D^iD_j\beta^j + R^i_j\beta^j = 2D_j\left(\alpha K^{ij}\right). \tag{2.49}$$

Equation (2.49) represents three elliptical equations which must be solved on every spacial hypersurface Σ_t, which, as with the maximal slicing condition, is too computationally expensive. We could also try minimising the variation of the *metric distortion tensor*, Σ_{ij} (not to be confused with the spacial hypersurfaces Σ_t of a foliation), which measures the shape of the coordinate volume elements independently from their size, and is defined as

$$\begin{aligned}\Sigma_{ij} &= \Theta_{ij} - \frac{1}{3}\gamma_{ij}\Theta_{kl}\gamma^{kl} \\ &= \frac{1}{2}\gamma^{1/3}\mathcal{L}_{\boldsymbol{t}}\tilde{\gamma}_{ij},\end{aligned} \tag{2.50}$$

where $\tilde{\gamma}_{ij} = \gamma^{-1/3}\gamma_{ij}$ is the conformal metric. Integrating $\Sigma_{ij}\Sigma^{ij}$ over a given hypersurface and setting $D_j\Sigma^{ij} = 0$ gives the *minimal-distortion shift condition*:

$$D^2\beta^i + \frac{1}{3}D^iD_j\beta^j + R^i_j\beta^j = 2D_j\left(\alpha A^{ij}\right), \tag{2.51}$$

which, similar to eq. (2.49), represents three elliptical equations to be solved over every spacial hypersurface, and is thus unfeasible to implement.

As was the case for the slicing condition, hyperbolic expressions that mimic the desirable behaviour of these elliptical equations have become the norm, principally the *Gamma-driver shift condition* proposed by Alcubierre et al.[7]. The "Gamma-driver" name comes from that fact that the conformal connection functions $\tilde{\Gamma}^i$ are driven such that they

are approximately constant in time, approximating the *Gamma-freezing shift condition* $\partial_t \tilde{\Gamma}^i = 0$, which can be shown to be closely related to the minimal-distortion shift condition (eq. (2.51))[115]. The evolution of β^i is typically given by the evolution equations (due to Alcubierre et al.[8])

$$\partial_t \beta^i - \beta^j \partial_j \beta^i = \frac{3}{4} B^i, \tag{2.52}$$

$$\partial_t B^i - \beta^j \partial_j B^i = \partial_t \tilde{\Gamma}^i - \beta^j \partial_j \tilde{\Gamma}^i - \eta B^i, \tag{2.53}$$

where B^i are auxiliary variables, and η is a damping coefficient used to suppress strong oscillations in β^i where there is significant and rapid variation in the gauges and is on the order of $1/M$ for a binary system with total mass M[116].

Hence, through eqs. (2.47), (2.52) and (2.53), we have a set of gauge conditions that satisfy the three requirements laid out above.

Chapter 3

Numerical Relativistic Hydrodynamics

In chapter 2 we discussed the system of equations that will be used to evolve the spacetime in our simulation. However this discussion took place under the assumption that the stress-energy tensor $T_{\mu\nu}$ was known. This tensor pertains to the fluid part of our simulation, and in this section we will discuss the formulation used to describe said fluid. A full discussion of the ideas presented in this section can be found in Rezzolla and Zanotti[115].

3.1 Perfect Fluid

Neutron stars are formed in supernovae by the core collapse of large stars ($M \geq 8\,M_\odot$) at the end of their lifetime, and were first theorised to exist by Baade and Zwicky[21] in 1934, but it was not until 1967 that their existence was confirmed by observation[72]. They are compact objects with mass $\sim 1 - 2\,M_\odot$ and radius $\sim 10 - 15\,\mathrm{km}$[49;93;117]. To model matter under such conditions we use a *relativistic perfect fluid*. As an idealised model, a perfect fluid neglects a number of physical phenomena, principally: anisotropic pressure, shear stresses, viscosity, and heat conduction, although it is possible to implement some of these effects through other means (see for example section 3.4, chapters 8 to 10). The advantage of using such a fluid is that they can be completely described through a rest frame density and (isotropic) pressure.

In order to build a general stress-energy tensor for a perfect fluid, we begin by looking at the construction of $T_{\mu\nu}$ and the *rest mass density current* J^μ from fluid variables in contravariant form. We start by considering these tensors in the fluid rest frame (using a hat, $\hat{}$, on indices to make clear this explicit choice), and from there we can generalise to any frame.

We first look at $J^{\hat{\mu}}$. Since the frame moves along with the fluid, the $\hat{\imath}$ components should all be 0, and the $\hat{t}$ component should be the rest mass density ρ, giving us

$$J^{\hat{\mu}} = (\rho, 0, 0, 0)\,. \tag{3.1}$$

Similarly, in this choice of frame, the components of $T^{\hat{\mu}\hat{\nu}}$ are defined as the flux of the $\hat{\mu}$-momentum in the $\hat{\nu}$-direction, taking the form

$$\begin{aligned} &T^{\hat{t}\hat{t}} = e, \\ &T^{\hat{t}\hat{\imath}} = T^{\hat{\imath}\hat{t}} = T^{\hat{\imath}\hat{\jmath}} = 0, && \text{for } \hat{\imath} \neq \hat{\jmath}, \\ &T^{\hat{\imath}\hat{\jmath}} = p, && \text{for } \hat{\imath} = \hat{\jmath}, \end{aligned} \tag{3.2}$$

where p is the isotropic fluid pressure, and e is the total energy density of the fluid, and is itself defined as

$$e = \rho\,(1 + \epsilon)\,, \tag{3.3}$$

where ϵ is the specific internal energy.

The equation of state of a fluid gives the relation between the microphysical behaviour of the fluid and the macrophysical hydrodynamic variables. In simulations it is used to remove the need to compute quantities such as p from a microphysical viewpoint and instead use quantities such as ρ and ϵ. The equation of state of neutron star matter is currently unknown, and as such there are a number of forms of equations of state used in simulations, which we will discuss in more detail in chapter 6.

We now rewrite the tensor components given in eqs. (3.1) and (3.2) in terms of the fluid 4-velocity in the comoving frame $u^{\hat{\mu}} = (1, 0, 0, 0)$ as

$$J^{\hat{\mu}} = \rho u^{\hat{\mu}}, \tag{3.4}$$

$$T^{\hat{\mu}\hat{\nu}} = e u^{\hat{\mu}} u^{\hat{\nu}} + p\left(\eta^{\hat{\mu}\hat{\nu}} + u^{\hat{\mu}} u^{\hat{\nu}}\right), \tag{3.5}$$

which generalise by the principle of covariance to give us

$$J^{\mu} = \rho u^{\mu}, \tag{3.6}$$

$$\begin{aligned} T^{\mu\nu} &= e u^{\mu} u^{\nu} + p\,(g^{\mu\nu} + u^{\mu} u^{\nu}) \\ &= (e + p) u^{\mu} u^{\nu} + p g^{\mu\nu}. \end{aligned} \tag{3.7}$$

A schematic representation of the information contained in $T^{\mu\nu}$ can be found in fig. 3.1.

$$T^{\mu\nu} = \begin{pmatrix} T^{tt} & T^{tx} & T^{ty} & T^{tz} \\ T^{xt} & T^{xx} & T^{xy} & T^{xz} \\ T^{yt} & T^{yx} & T^{yy} & T^{yz} \\ T^{zt} & T^{zx} & T^{zy} & T^{zz} \end{pmatrix}$$

FIGURE 3.1: Schematic representation of the components of the fluid stress energy tensor $T^{\mu\nu}$ as measured by a co-moving observer their local rest frame. Regions are highlighted as follows: T^{tt} is the energy density of the fluid, T^{ti} is the energy-flux in the i-th direction, T^{it} is the i-th momentum density, T^{ij} with $i = j$ is the isotropic pressure, and T^{ij} with $i \neq j$ is the flux of the i-th momentum in the j-th direction.

3.2 Relativistic Euler Equations

In order to develop a set of evolution equations for our fluid we start by looking at the conservation laws for rest mass and energy-momentum. The conservation of energy-momentum is given by

$$\nabla_\mu T^{\mu\nu} = 0, \tag{3.8}$$

and the conservation of rest mass[1] is given by

$$\nabla_\mu (\rho u^\mu) = 0. \tag{3.9}$$

We can split the conservation of energy-momentum equation into separate energy and momentum conservation equations by taking the projection of eq. (3.8) along u^μ and orthogonal to it. We first introduce the projection operator[2] orthogonal to the fluid flow $\perp^\mu_\nu$ as

$$\perp^\mu_\nu = \delta^\mu_\nu + u^\mu u_\nu. \tag{3.10}$$

Taking $\perp^\mu_\nu$ and operating on eq. (3.8) we obtain

$$\perp^\nu_\alpha \nabla_\mu T^{\mu\alpha} = u^\mu \nabla_\mu u_\nu + \frac{1}{\rho h} \perp^\mu_\nu \nabla_\mu p = 0, \tag{3.11}$$

where we have introduced the specific enthalpy $h = 1 + \epsilon + p/\rho$.

These equations represent the *Euler equations* in relativity. We can identify the first term on the right hand side of eq. (3.11) with the acceleration of the fluid $a_\nu = u^\mu \nabla_\mu u_\nu$ and recast the relation as

$$\rho h a^\mu = - \perp^{\mu\nu} \nabla_\nu p, \tag{3.12}$$

[1] Assuming the fluid is composed of baryons with rest mass m_b and number density $n_b = \rho/m_b$, eq. (3.9) equivalently gives us conservation of baryon number.

[2] Not to be confused with the projection on to each hypersurface $P^\mu_\nu = \gamma^\mu_\nu = \delta^\mu_\nu + n^\mu n_\nu$.

which can be interpreted in a similar way to Newton's second law of motion, where the force is given by the pressure gradient, and the inertial mass is identified with the enthalpy density ρh.

By taking the projection of eq. (3.8) along u^μ and using eq. (3.9), we obtain the expression

$$u^\mu \nabla_\mu e - h u^\mu \nabla_\mu \rho = 0, \tag{3.13}$$

which is a relativistic expression for the conservation of energy in the fluid.

3.3 Valencia Formulation

Now that we have a relativistic analogue of the conservation equations for a perfect fluid, we can begin to formulate these in a manner adapted to both numerical calculations, and our 3+1 description of spacetime. Early work on numerical simulations of relativistic hydrodynamic fluids in curved spacetimes was carried out by Wilson in the 1970s (see Wilson and Mathews[140] for a full account), but it was not until the work of Martí et al.[91] in the early 1990s that a conservative form of the equations above was derived, with the so-called *Valencia formulation* arriving in 1997[26]. Being in a conservative form allows a number of highly developed methods and codes written for Newtonian fluids to be easily adapted to the relativistic versions of the equations.

We start by expanding eq. (3.9) using the identity for the divergence of a generic 4-vector w^μ

$$\nabla_\mu w^\mu = \frac{1}{\sqrt{-g}} \left[\partial_t \left(\sqrt{-g} w^t \right) + \partial_i \left(\sqrt{-g} w^i \right) \right], \tag{3.14}$$

to give us

$$\frac{1}{\sqrt{-g}} \left[\partial_t \left(\sqrt{-g} \rho u^t \right) + \partial_i \left(\sqrt{-g} \rho u^i \right) \right] = 0. \tag{3.15}$$

We also define the *Eulerian 3-velocity* of the fluid v^i as the velocity of the fluid as measured by an observer moving orthogonal to spacial hypersurfaces with 4-velocity n^μ, which is given by

$$\begin{aligned} v^i &= \frac{1}{W} u^i + \frac{1}{\alpha} \beta^i, \\ v^t &= 0, \end{aligned} \tag{3.16}$$

where we have used W to denote the *Lorentz factor* of the fluid given by

$$\begin{aligned} W &= \alpha u^t \\ &= \frac{1}{\sqrt{1 - v_i v^i}}. \end{aligned} \tag{3.17}$$

Returning to eq. (3.15), we can define the conserved mass density D (not to be confused with any derivative notation) as

$$D = \rho\alpha u^t = \rho W, \tag{3.18}$$

which we can then substitute back into eq. (3.15) and make use of eq. (3.16). This gives us

$$\partial_t \left(\sqrt{\gamma} D\right) + \partial_i \left[\sqrt{\gamma} D \left(\alpha v^i - \beta^i\right)\right] = 0, \tag{3.19}$$

which expresses the conservation of rest mass in terms of fluid variables and 3+1 quantities only.

Before we expand eq. (3.8), we will first decompose $T^{\mu\nu}$ into components as measured by Eulerian observers moving along n^μ (cf. section 2.1), giving us three parts:

– The full projection along n^μ:

$$n_\alpha n_\beta T^{\alpha\beta} = E, \tag{3.20}$$

– The parallel-transverse component using projections along both n^μ and $\gamma_{\mu\nu}$:

$$\gamma^\mu_\alpha n_\beta T^{\alpha\beta} = -S^\mu, \tag{3.21}$$

– The full projection orthogonal to n^μ using $\gamma_{\mu\nu}$:

$$\gamma^\mu_\alpha \gamma^\nu_\beta T^{\alpha\beta} = S^{\mu\nu}. \tag{3.22}$$

We can then use eqs. (3.20) to (3.22) to re-write eq. (3.7) as

$$T^{\mu\nu} = E n^\mu n^\nu + S^\mu n^\nu + S^\nu n^\mu + S^{\mu\nu}. \tag{3.23}$$

Note also that we can re-write u^μ and $g^{\mu\nu}$ as

$$u^\mu = W\left(n^\mu + v^\mu\right), \tag{3.24}$$

$$g^{\mu\nu} = \gamma^{\mu\nu} - n^\mu n^\nu, \tag{3.25}$$

and that we can also re-write eq. (3.7) as

$$T^{\mu\nu} = \rho h W^2 \left(n^\mu + v^\mu\right)\left(n^\nu + v^\nu\right) + p\left(\gamma^{\mu\nu} - n^\mu n^\nu\right). \tag{3.26}$$

We can then compare terms between eq. (3.23) and eq. (3.26) and write the following expressions:

$$E = \rho h W^2 - p, \tag{3.27}$$

$$S^\mu = \rho h W^2 v^\mu, \tag{3.28}$$

$$S^{\mu\nu} = \rho h W^2 v^\mu v^\nu + p\gamma^{\mu\nu}. \tag{3.29}$$

Using the identity

$$\nabla_\mu C^{\mu\nu} = g^{\nu\lambda}\left[\frac{1}{\sqrt{-g}}\partial_\mu\left(\sqrt{-g}C^\mu_\lambda\right) - \frac{1}{2}C^{\alpha\beta}\partial_\lambda g_{\alpha\beta}\right], \tag{3.30}$$

where $C^{\mu\nu}$ is a generic symmetric rank-2 tensor, we can expand eq. (3.8) to give

$$\frac{1}{\sqrt{-g}}\partial_\mu\left(\sqrt{-g}T^\mu_\nu\right) = \frac{1}{2}T^{\mu\lambda}\partial_\nu g_{\nu\lambda}, \tag{3.31}$$

into which we can substitute the decomposed expression for $T^{\mu\nu}$ as given in eq. (3.23) to yield

$$\partial_t\left(\sqrt{\gamma}S_j\right) + \partial_i\left[\sqrt{\gamma}\left(\alpha S^i_j - \beta^i S_j\right)\right] = \frac{\sqrt{-g}}{2}T^{\mu\nu}\partial_j g_{\mu\nu}, \tag{3.32}$$

where we have restricted our free index to spacial components only.

Finally we obtain an expression for E by taking the contraction of eq. (3.8) with n_ν which gives us

$$\nabla_\mu\left(T^{\mu\nu}n_\nu\right) - T^{\mu\nu}\nabla_\mu n_\nu = 0, \tag{3.33}$$

and substituting this into eq. (3.23) to give

$$\partial_t\left(\sqrt{\gamma}E\right) + \partial_i\left[\sqrt{\gamma}\left(\alpha S^i - \beta^i E\right)\right] = -\sqrt{-g}T^{\mu\nu}\nabla_\mu n_\nu. \tag{3.34}$$

The terms on the right hand sides of eqs. (3.32) and (3.34) are referred to as *source terms* where it is notable that they do not contain differential operators acting on fluid variables. We can re-write these source terms to more explicitly highlight their dependence on 3+1 variables as

$$\frac{\sqrt{-g}}{2}T^{\mu\nu}\partial_j g_{\mu\nu} = \sqrt{\gamma}\left(\frac{1}{2}\alpha S^{ik}\partial_j\gamma_{ik} + S_i\partial_j\beta^i - E\partial_j\alpha\right), \tag{3.35}$$

$$-\sqrt{-g}T^{\mu\nu}\nabla_\mu n_\nu = \sqrt{\gamma}\left(\alpha S^{ij}K_{ij} - S^j\partial_j\alpha\right). \tag{3.36}$$

Before we write down the full system of equations, it is useful to note that since linear combinations of the conserved variables are still solutions to the equations in conserved form, it is common to use $\tau = E - D$ instead of E itself in simulations as τ has more desirable numerical characteristics[115], as for small v ($W \to 1$) and T ($e \to \rho$) the total

energy is dominated by the self energy, leading to $E \simeq D$. The corresponding flux term (the part operated on by ∂_i in eq. (3.34)) is given by

$$\alpha \left(S^i - v^i D \right) - \beta^i \tau, \tag{3.37}$$

and the source term remains the same.

We can tidy up these expressions and cast them in a more compact form as

$$\partial_t \left(\sqrt{\gamma} \mathbf{U} \right) + \partial_i \left(\sqrt{\gamma} \mathbf{F}^i \right) = \sqrt{\gamma} \mathbf{S}, \tag{3.38}$$

where we have defined the vector of conserved variables $\mathbf{U}$, the vector of fluxes $\mathbf{F}^i$ and the vector of sources $\mathbf{S}$. Using eqs. (3.19), (3.27) to (3.29), (3.32) and (3.34) to (3.37) we obtain expressions for these three vectors as follows

$$\mathbf{U} = \begin{pmatrix} D \\ S_j \\ \tau \end{pmatrix} = \begin{pmatrix} \rho W \\ \rho h W^2 v_j \\ \rho h W^2 - p - D \end{pmatrix}, \tag{3.39}$$

$$\mathbf{F}^i = \begin{pmatrix} \alpha v^i D - \beta^i D \\ \alpha S^i_j - \beta^i S_j \\ \alpha \left(S^i - v^i D \right) - \beta^i \tau \end{pmatrix}, \tag{3.40}$$

$$\mathbf{S} = \begin{pmatrix} 0 \\ \frac{1}{2} \alpha S^{ik} \partial_j \gamma_{ik} + S_i \partial_j \beta^i - E \partial_j \alpha \\ \alpha S^{ij} K_{ij} - S^j \partial_j \alpha \end{pmatrix}. \tag{3.41}$$

The five components of $\mathbf{U}$ can be interpreted as:

– the conserved mass density D,

– the conserved 3-momentum density S_j,

– and the conserved internal energy density τ.

Thus we have a set of balance laws (conservative equations with source terms) for the dynamics of the fluid (assuming we also have an equation of state) written in the language of 3+1. There is, however, still more to the puzzle:

– We have written down the system of equations to be solved, but a method is needed to do this in an intelligent manner: such methods will be discussed in chapter 4.

– While it is trivial to write the conserved variables $\mathbf{U}$ in terms of the primitive fluid variables, the opposite is not true: a root-finding procedure is needed, which we discuss in sections 4.5.3 and 6.2.

3.4 Beyond the Perfect Fluid

As we mentioned at the beginning of this chapter, perfect fluids neglect a number of physical phenomena, some of which may be of interest to mergers. It is, however, possible to extend the physics of a perfect fluid to include some of these behaviours, for example in chapter 10 we discuss some aspects of bulk viscosity. Here we will briefly mention another aspect, composition, as the evolution of the composition (in the absence of reactions, see chapters 8 to 10) can be framed as a simple extension to the Valencia formulation.

As we move towards more realistic equation of state, we will want some measure to keep track of the composition of the fluid. When considering a locally charge neutral fluid composed of protons, neutron, and electrons, we should have

$$n_b = n_\mathrm{n} + n_\mathrm{p}, \tag{3.42}$$

$$n_\mathrm{p} = n_\mathrm{e}, \tag{3.43}$$

where n_x is the number density of species x, and b, n, p, and e refer to baryons, protons, neutrons and electrons respectively. We can then define the electron fraction Y_e as

$$Y_\mathrm{e} = \frac{n_\mathrm{e}}{n_b} = \frac{n_\mathrm{p}}{n_b}. \tag{3.44}$$

In the absence of reactions this fraction will be advected along with the fluid flow (see section 7.3), which can be achieved by introducing a conserved electron fraction DY_e, which is simply the product of the conserved mass density D and the electron fraction Y_e. Therefore we can expand the system of balance laws above (eqs. (3.39) to (3.41)) to include an additional term related to the composition (which we label here with a subscript C), which for each of $\mathbf{U}$, $\mathbf{F}^i$, and $\mathbf{S}$ will be

$$U_C = DY_\mathrm{e} = \rho W Y_\mathrm{e} \tag{3.45}$$

$$F^i_C = \alpha v^i DY_\mathrm{e} - \beta^i DY_\mathrm{e} \tag{3.46}$$

$$S_C = 0 \tag{3.47}$$

Note that Y_e can be trivially calculated from DY_e through $Y_\mathrm{e} = DY_\mathrm{e}/D$, so the primitive recovery schemes we discuss later in sections 4.5.3 and 6.2 are not affected by this extension to the system.

If in the future we decide to also include other leptons, such as muons, it will be useful to instead use the charge (referring only to charged baryons) or proton fractions, Y_q or Y_p respectively, which follow similar definitions to Y_e.

Chapter 4

The Einstein Toolkit

This chapter is dedicated to a discussion of various aspects of the software suite we use to perform the binary neutron star mergers we present in part II, the Einstein toolkit [54;87], while highlighting some of the issues that we needed to address to make those simulations possible, and others that may impact similar research in the future. The toolkit was designed with the purpose of being a portable, modular suite of code for evolving large scale numerical hydrodynamics/spacetime simulations. The Einstein toolkit is built on the *Cactus* environment [61], which consists of a *flesh* that handles the interface between the *thorns* that implement the evolution. As part of this modularity, there are a number of thorns of implementing the many parts of each simulation, however we will discuss only those that are of physical interest to our research. These are namely:

- `Carpet`: The main *driver* thorn,
- `MoL`: Implements the Method of Lines for time integration,
- `ML_BSSN`: McLachlan implementation of BSSN based on code generated by Kranc,
- `GRHydro`: Handles evolution of hydrodynamic variables,
- `EOS_Omni`: Equation of state interface.

Also worthy of note are `ADMBase`, `TmunuBase`, and `HydroBase`, which provide interfaces to the spacetime (γ_{ij}, K_{ij}, α, β^i), stress-energy tensor (T_{tt}, T_{ti}, T_{ij}), and fluid (ρ,p,ϵ,etc.) variables respectively.

Before we cover the details of the thorns in more detail, we will first discuss the creation of initial data.

4.1 Initial Data

The Einstein toolkit features a number of different methods for the generation of initial data, and also allows initial data that has been created outside the toolkit to be loaded in. The simplest form of initial data for binary neutron stars is based on the *Tolman-Oppenheimer-Volkoff* (TOV) equation.

4.1.1 TOV solution

The TOV equation[100] describes a spherically symmetric isotropic fluid body in gravitational equilibrium, and can be written as

$$\frac{dp}{dr} = -\frac{(e+p)\left(m(r)+4\pi r^3 p\right)}{r\left(r-2m(r)\right)}, \tag{4.1}$$

where $m(r)$ is the (total) mass of fluid enclosed within the radius r calculated through

$$m(r) = \int_0^r 4\pi r'^2 e(r')dr', \tag{4.2}$$

$e(r)$ is the total mass density of the fluid at a given radius, $p(r)$ is the isotropic pressure at a given radius, and the equation of state $f(e,p)=0$ is known.

A star described by this equation will induce a metric in the surrounding spacetime given by

$$ds^2 = -\exp(\nu(r))dt^2 + \frac{r}{r-2m(r)}dr^2 + r^2\left(d\theta^2 + \sin^2\theta\, d\phi^2\right), \tag{4.3}$$

where $\nu(r)$ is calculated from

$$\frac{d\nu}{dr} = -\left(\frac{2}{p+e}\right)\frac{dp}{dr}, \tag{4.4}$$

and the boundary constraints are given by

$$\begin{aligned} m(0) &= 0, \\ p(R) &= 0, \\ \exp(\nu(R)) &= 1 - \frac{2M}{R}, \end{aligned} \tag{4.5}$$

where R is the total radius of the star, and M is the total mass as given by $M = m(R)$ using eq. (4.2).

To create initial data for a binary neutron star merger, it is possible to construct the two stars separately using the above equations, then place them in the computational domain at the desired separation, and set the fluid velocity to give an approximately

circular orbit (for example using Newtonian mechanics), and possibly add axial spin to each star.

This method will give initial data that is close to pseudo-equilibrium and will (assuming the stars are placed far enough apart/tidal interactions are small) not introduce fatal instabilities. Due to the fact the system is out of equilibrium to begin with, data obtained at the start of the simulation (especially gravitational wave signals) will not be physical, and a period of "settling down" is required.

As the stellar configurations obtained using a TOV solver do not take into account interactions between the two stars, the solution can be considered as a perturbation of the "true" pseudo-equilibrium solution, with both the fluid and spacetime affected. Once the simulation is started, and the interaction between the stars begins, these un-physical perturbations will evolve, spoiling the data in the early part of the simulation, until the stars have reached a settled state. The so-called "junk" radiation in the spacetime is propagated away, and the numerical viscosity of the fluid causes those perturbations to die down. A better method is to relax the out-of-pseudo-equilibrium data before the simulation is started. Therefore we use the LORENE code, which was developed for exactly this purpose of allowing an out-of-pseudo-equilibrium set of initial data to be relaxed closer to pseudo-equilibrium thus reducing the "spoiling" of the early parts of the simulation.

4.1.2 LORENE

LORENE[63] (standing for Langage Objet pour la RElativité NumériquE) was developed at the Observatoire de Paris from an earlier FORTRAN code for the purpose of using spectral methods to create initial data for numerical relativistic simulations[63]. LORENE uses a multi-domain approach: for binary neutron star data, there are three "origins", one at the centre of each star, and one at the barycentre. Each stellar origin has two domains, one that extends from the origin out to the stellar surface, and a second that extends from the surface out to infinity, and the barycentre origin features a single domain extending from the origin to infinity.

LORENE works by creating an initial solution in a similar fashion to that described in section 4.1.1 and then relaxing that solution using full spacetime calculations of both stars (as opposed to treating each star separately) to create a set of initial data much closer to the quasi-circular orbits we want. Another useful feature of LORENE is a well developed equation of state interface, making it easy to produce initial data for many forms of equation of state. There are codes in LORENE for producing a number of configurations of compact objects, and also for producing series of similar data without the need to re-run the entire relaxation routine from scratch.

There are some features missing from LORENE that may be desirable in future initial data routines. First, LORENE does not allow arbitrary spins to be imposed on binary neutron star configurations, nor can differential rotation profiles be applied, only rigidly rotating stars that are static in either the sidereal or synodic frames are possible, (referred to as irrotational and co-rotational respectively). Additionally, LORENE requires a 1-dimensional equation of state (see section 4.6), therefore realistic temperature profiles (e.g. constant redshifted temperature) cannot be used, however it is possible to construct a 1-dimensional tabulated equation of state from a 3-dimensional table by, for example, assuming a constant temperature and beta equilibrium.

4.2 Carpet

`Carpet` is one of two driver thorns included in the Einstein toolkit, the other being `PUGH`. Primarily, it provides the *Adaptive Mesh Refinement* (AMR), parallelisation, and some of the IO capabilities of the toolkit.

4.2.1 AMR

The type of adaptive mesh refinement provided by `Carpet` was first presented by Berger and Oliger in 1984[29]. The computational domain consists of several nested grids, each of which has spacing half that of its parent (see fig. 4.1). Interpolation of the data in the coarser spaced grids provides data for the edges of the finer grids as they are evolved, and the finer grids then pass their data back up through the hierarchy onto the coarser grids they overlap.

The *Courant-Friedrichs-Lewy* (CFL) condition states that in order to properly evolve a wave moving across a discrete spacial grid, the time taken for the wave to move between grid points must be greater than the time between successive iterations. In order to preserve the CFL condition, each finer grid is evolved twice as often as the level above, thus the ratio of $\Delta x/\Delta t$ is a constant for all grids.

Additionally, the position and extent of each grid is not fixed: the finer grids can be made to track the neutron stars though their evolution, and the size of the grids/number of refined layers can be changed dynamically to adapt to the changing needs to the simulation. This is what differentiates AMR from *Fixed Mesh Refinement*. The two principal advantages to AMR are reduced memory usage, and reduced computation time without significant loss of accuracy. To demonstrate these, we will consider the length scales we expect to see in our simulations.

The typical radius of a neutron star is $\sim 8\,M_\odot$, and in order to resolve the features of neutron stars through merger, we want a grid resolution on the order of $10 - 100$ times

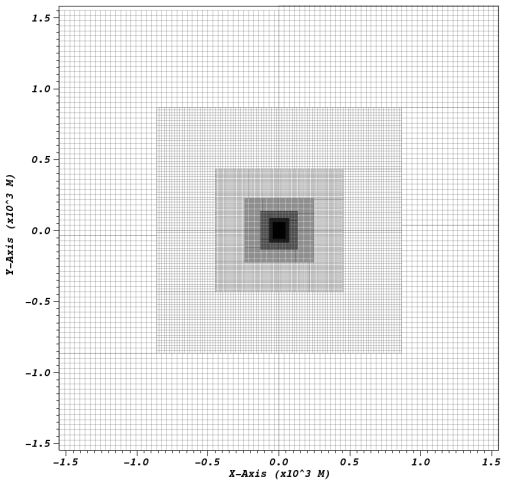

FIGURE 4.1: Nested grid structure of `Carpet`. Figure shows the x-y plane at $z = 0M_\odot$. Scale is in units of solar mass $M_\odot \approx 1.48$km

smaller than this. We also want to resolve gravitational waves in the simulation, and a typical lower bound for the gravitational wave frequency is $\sim 150\,\mathrm{Hz}$ corresponding to $\lambda \approx 1500\,M_\odot$. A cubic grid with extent $\pm 1500\,M_\odot$ in each direction and with a spacing of $0.15\,M_\odot$ would contain $\sim 10^{13}$ grid points, and would require RAM on the scale of multiple 10s of terabytes to store the required simulation variables.

If, on the other hand, we use a grid structure such that each level has the same number of grid points, but half the length extent, and goes down 8 levels of refinement (9 in total), using the same total extent as above, and setting the spacing on the smallest grid to that above, we obtain a much more manageable $\sim 10^7$ grid points.

Also, due to the fact we can double the timestep each level we go up the grid hierarchy, the total number of grid point evolutions drops by a factor of $\sim 2 \times 10^7$ compared to the naïve grid above. Instead of the computation being dominated by the extent of the domain and the sample spacing there, in the AMR regime the computation is dominated by the extent and grid spacing of the finest grid. It should also be noted that there are computational overheads in the driving of a grid hierarchy such as this, but they are insignificant compared to the sorts of speedups AMR can provide when compared to a uniform grid.

This is only an example of the power of AMR, and real simulations may fare better or worse than this, but it does give a rough idea of typical speedups, and demonstrates how AMR make these simulations not only faster, but actually feasible (in terms of both memory consumption and CPU time) in the first place.

4.2.2 Parallelisation

`Carpet` provides the ability to subdivide the AMR grid such that the simulation can be run across a number of MPI processes. A *ghost zone* is placed around each sub-grid, the contents of which are either copied from adjacent sub-grids on the same level, or interpolated from the level above as described above. These are large enough to allow the process responsible for the sub-grid to perform central spacial differences of the desired accuracy using only data local to that process.

This way of dividing the work up reduces the inter-process communication, and allows simulations to be run efficiently across multiple nodes on a supercomputer. Typically, each physical CPU (containing a number of cores) runs one MPI process, and OMP is then used by each thorn performing a calculation to make the most of the available hardware.

While `Carpet` allows the toolkit to scale well up to $\gtrsim 10^5$ CPU cores[87], the future of large scale computing lies in GPUs[1]. The Einstein toolkit is not currently able to take advantage of GPUs for acceleration, although the in-development `CarpetX`[123] (which uses AMReX[143]) driver thorn is intended to replace the current AMR driver in the future, allowing GPU acceleration.

There are some new and in-development codes that use spectral methods which are, in principle, more adept to being run on GPUs (due, in part, to a reduction in the communication required), for example SpEC[79], which is based on the generalized harmonic formalism[85] using pseudo-spectral methods with adaptive mesh refinement[131], and SpECTRE[47], which uses discontinuous Galerkin methods (see Deppe et al.[46] and references therein). Also in development is `GR-Athena++`[44], which uses the task-based parallelism paradigm of `Athena++`[130] that has been successfully used to simulate non-relativistic MHD fluid[64]. Some special-relativistic MHD fluid simulations have been performed using GPU acceleration (see Wright and Hawke[141]), but, to our knowledge, no full-scale GR simulations of neutron star mergers have been performed with GPU-based acceleration.

4.3 Method of Lines

The `MoL` thorn provides the time integration for the Einstein toolkit through the *Method of Lines*. The method of lines is a way of turning PDEs such as those derived in chapters 2 and 3 into ODEs, which are then evaluated using standard ODE integration algorithms. The thorn itself provides an interface to other thorns which can register variables to be evolved/constrained using the method of lines.

[1] 4 of the top 5 performing supercomputers worldwide (at the time of writing, see TOP500[136]) derive at least some of their performance from GPU-like coprocessor cards.

The idea behind the method of lines is to discretise the spacial derivatives appearing in the PDEs, leaving the time derivatives as a set of coupled continuum ODEs[135]. By doing this, the PDEs are converted into ODEs along the worldline of each grid point. By decoupling the spacial and temporal derivatives, the method of lines allows a much greater degree of modularity than would be possible by building an integrator to work directly on the PDEs.

As an example of the method of lines, let us consider the evolution equation for the natural logarithm of the conformal scalar, $\phi = \ln \psi$, given in eq. (2.37). Starting with

$$\partial_t \phi = -\frac{1}{6}\left(\alpha K - \partial_i \beta^i\right) + \beta^i \partial_i \phi, \tag{4.6}$$

we must first discretise the spacial coordinates. We can do this by assuming our variables are sampled on an equally spaced grid (with sample spacing Δx, and ignoring boundaries) indexed by integers l, m, n for the three dimensions respectively, so we can re-write the above as

$$\partial_t \left(\phi\right)_{l,m,n} = -\frac{1}{6}\left[\left(\alpha K\right)_{l,m,n} - \partial_i \left(\beta^i\right)_{l,m,n}\right] + \left(\beta^i\right)_{l,m,n} \partial_i \left(\phi\right)_{l,m,n}, \tag{4.7}$$

where the subscripts (not to be confused with tensor indices) on the brackets are a shorthand for

$$\left(f\right)_{l,m,n} = \left(f\left(t, x_l, y_m, z_n\right)\right). \tag{4.8}$$

Next, we must replace the spacial derivatives with their discretised counterparts. Choice of discretisation can vary from one variable to another to adapt to the particular characteristics of each, however, for simplicity, here we use a second order accurate central finite difference

$$\frac{\partial f(t,x,y,z)}{\partial x} \approx \frac{-f(t, x-\Delta x, y, z) + f(t, x+\Delta x, y, z)}{2\Delta x}, \tag{4.9}$$

or to use the subscript notation above

$$\frac{\partial}{\partial x}\left(f\right)_{l,m,n} \approx \frac{1}{2\Delta x}\left[-\left(f\right)_{l-1,m,n} + \left(f\right)_{l+1,m,n}\right]. \tag{4.10}$$

Using eq. (4.10) we can then write eq. (4.7) as

$$\begin{aligned}\partial_t (\phi)_{l,m,n} = -\frac{1}{6}\Bigg[(\alpha K)_{l,m,n} - \frac{1}{2\Delta x}\Big[&-(\beta^x)_{l-1,m,n} + (\beta^x)_{l+1,m,n} \\ &- (\beta^y)_{l,m-1,n} + (\beta^y)_{l,m+1,n} \\ &- (\beta^z)_{l,m,n-1} + (\beta^z)_{l,m,n+1}\Big]\Bigg] \\ &+ \frac{1}{2\Delta x}\Big[(\beta^x)_{l,m,n}\Big[-(\phi)_{l-1,m,n} + (\phi)_{l+1,m,n}\Big] \\ &+ (\beta^y)_{l,m,n}\Big[-(\phi)_{l,m-1,n} + (\phi)_{l,m+1,n}\Big] \\ &+ (\beta^z)_{l,m,n}\Big[-(\phi)_{l,m,n-1} + (\phi)_{l,m,n+1}\Big]\Big]. \quad (4.11)\end{aligned}$$

By removing the spacial partial derivatives, this procedure has left us with a set (l, m, n can take as many values as there are grid points) of coupled ODEs in t, which are then evolved by `MoL`.

Each thorn that relies on `MoL` for time integration is responsible for the spacial discretisation of its associated variables, and the calculation of the right hand side of those variables at a given t. This is important, as, for example, one might want to discretise the spacetime variables using high-order central differencing, and the fluid variables using shock capturing methods (see section 4.5), then evolve all variables together using a high-order explicit Runge-Kutta method. The modularity of `MoL` makes this far simpler and more adaptable than writing a PDE solver of the desired form for the coupled spacetime and fluid equations.

4.4 McLachlan

`McLachlan` is a collection of thorns providing spacetime evolution using code generated by Kranc[34;74]. Kranc takes the tensor form of a set of PDEs (in our case, those derived in section 2.2) and turns them into a `Cactus` thorn. Using Mathematica to manipulate the PDEs, Kranc then converts the Mathematica statements into code in either C or Fortran 90. Kranc also handles the declaration of grid functions, registering of evolved variables with `MoL`, and scheduling of function calls in `Cactus`.

Two different spacetime evolution methods are included in the version of `McLachlan` that ships with the Einstein toolkit: BSSN and CCZ4[16], which are implemented through the thorns `ML_BSSN` and `ML_CCZ4` respectively. As we discussed in section 2.2, BSSN is freely evolved, as is CCZ4, however CCZ4 uses a technique known as *constraint dampening*. CCZ4 is obtained in a similar fashion to BSSN, but through the addition of auxiliary variables and mixing with the constraint equations, a system is obtained such that any constraint violations that appear propagate out of the system[30]. The CCZ4 equations reduce to the Einstein field equations when these auxiliary variables

vanish, so by damping them throughout the evolution any constraint violations can be suppressed[32;66]. While CCZ4 can provide a reduction in the size of constraint violations of over an order of magnitude[17], there are downsides: A poor choice of damping parameters can significantly impact the simulation, sometimes leading to over-damped effects, or even instabilities[16;119].

4.5 GRHydro

`GRHydro`[97] is one of two main *General Relativistic Magnetohydrodynamics* (GRMHD) codes shipped with the Einstein toolkit, the other being `IllinoisGRMHD`. Based on the `Whisky` code[23], `GRHydro` uses *High Resolution Shock Capturing* (HRSC) methods[57;81;90;137] to evolve a fluid described by GRMHD equations, in particular the Valencia formulation[26] (see also section 3.3 for the system of equations used, without magnetic fields). The principal advantages of using modern HRSC methods are that: the non-linear discontinuities that form in hydrodynamics (the *shocks*) are treated in such a way that the numerical oscillations (not physical) at shocks known as *Gibbs' phenomena* are avoided, a high degree of accuracy (at least second order) is maintained over most of the computational domain, shocks are sharply resolved, and the solution converges to the "true" solution as the grid resolution is increased[115].

Modern HRSC methods are based on *Godunov methods*[60] and still use the same basic steps to control the evolution of the fluid:

1. Reconstruct variables at cell boundaries
2. Solve Riemann problems
3. Update conserved variables

The time-integration ("update" term) is controlled by `MoL` (see section 4.3), so here we describe the reconstruction and Riemann problem steps. In addition, as we require the *primitive variables* (ρ, T, v^i) to calculate the evolution terms, we must recover them from the conserved variables ($\mathbf{U}$ in section 3.3), a process we describe in section 4.5.3.

4.5.1 Reconstruction

`GRHydro` breaks the fluid down into a series of cells, each centered on a grid point. In order to calculate the fluxes between two adjacent cells, it is first necessary to *reconstruct* the value of the variables at the cell boundaries. It is important to note that as it is the reconstructed variables that are used to calculate the fluxes, the method used for reconstruction must not violate the conservation laws (for this reason, reconstruction

is typically performed on primitive variables as physicality of the solution is easier to check), and it should not introduce oscillations.

There are a number of reconstruction methods implemented in the Einstein toolkit, but we choose to use the *enhanced Piecewise Parabolic Method*[40] (ePPM) (a variant of PPM[41]) which compares favourably in terms of performance to the other reconstruction methods[97], and has a more mature implementation in the toolkit.

The principal concept behind PPM methods is to use a fourth order polynomial interpolator of the fluid variables, with limiting steps used to constrain the interpolant such that no spurious extrema are introduced. Both PPM and ePPM reduce to first order accuracy at shocks, but while PPM also reduces to first order accuracy at smooth extrema, ePPM does not.

4.5.2 Riemann problems

Once we have values for the fluid variables at the cell boundaries, the next step is to calculate the fluxes between adjacent cells. This means solving local *Riemann problems*. The Einstein toolkit currently features 3 *Riemann solvers*: Roe, HLLE, and HLLC, and two methods that solve directly for the fluxes without addressing the Riemann problem explicitly: Marquina and LLF. Not all of these are compatible with all choices of simulation setups (e.g. reconstruction method), so we choose to use HLLE which shows good stability and accuracy (see below), and has good compatibility with other parts of the toolkit.

Riemann solvers find solutions to flux problems in terms of *waves* which differentiate a number of constant *states*. These waves can represent shocks (all fluid variables change discontinuously at a boundary), rarefactions (all fluid variables change continuously), or contact discontinuities (some fluid variables change continuously, others discontinuously).

The *Harten-Lax-van Leer-Einfeldt* (HLLE) solver[52;70] uses a two characteristic wave approximation to calculate the fluxes across cell boundaries. The two waves are represented by ξ_+ and ξ_- where the $+$ and $-$ denote the largest and smallest wave speed eigenvalues present on each side of the cell boundary. In order to calculate $\xi_\pm$ quickly, the approximation outlined in Gammie et al.[59] is used. This method introduces extra diffusion into the simulation, but is shown to overestimate the wavespeeds significantly only in certain fluid configurations, and even then only by no more than a factor of $\sqrt{2}$[97], but this does not affect convergence to a solution in the limit of $\Delta x \to 0$. A comparison of HLLE and the more accurate Marquina is given in the Appendix of Baiotti et al.[24], where it is shown that, as long as high-order reconstruction is used (there PPM), the choice of Riemann problem solver does not significantly impact the simulation.

4.5.3 Primitive recovery

Once the conservative fluid variables have been updated, it is then necessary to convert these back to the primitive variables, a process variously called primitive recovery, conservative-to-primitive, Con2Pri, C2P, et cetera. This is achieved by inverting eq. (3.39). This is not generally possible analytically, so `GRHydro` uses a Newton-Raphson root finding scheme.

The function for which we wish to find the root is

$$f(p_i) = p_i - p(\rho_i, \epsilon_i), \tag{4.12}$$

where the subscripts denote the root finding iteration from which the values are taken, and the function $p(\rho, \epsilon)$ comes from the EoS.

With a guess for the pressure p_i, ρ_i and ϵ_i are calculated from the conserved variables, and the gradient of $f(p_i)$ is given by

$$f' = 1 - \frac{\partial p}{\partial \rho}\frac{\partial \rho}{\partial p} - \frac{\partial p}{\partial \epsilon}\frac{\partial \epsilon}{\partial p}, \tag{4.13}$$

where $\partial p/\partial \rho$ and $\partial p/\partial \epsilon$ come from the EoS, and $\partial \rho/\partial p$ and $\partial \epsilon/\partial p$ are calculated from the conserved variables. This gradient is then used to give a new guess for the pressure p_{i+1} through

$$p_{i+1} = p_i - \frac{f(p_i)}{f'(p_i)}. \tag{4.14}$$

This iteration procedure is repeated until the residual from the root function is below some desired accuracy threshold.

Due to the presence of the partial derivative with respect to ϵ in eq. (4.13) this method is not well suited to tabulated equations of state in the form (ρ, T, Y_e). In section 6.2 we discuss the implementation of other primitive recover schemes that are better suited to such tables, which we have implemented as an extension to `GRHydro`. Recently the RePrimAnd[78] recovery scheme has been introduced, and has been used in relativistic hydrodynamic simulations[75], including neutron star mergers[102]. This scheme is also designed for tabulated equations of state, and is claimed to always converge to a unique solution (assuming one exists)[78].

4.6 Equations of State

The final implementation detail we will discuss is the Equation of State interface, which is provided by `EOS_Omni`. This thorn provides a number of different equations of state, both functional and tabulated.

The basic equations of state in `EOS_Omni` are Polytropic, Γ-Law, and Hybrid (a combination of polytropic and Γ-law) equations of state[54]. These equations of state are expressible in functional form, with a number of parameters used to control their behaviour. `EOS_Omni` also provides some realistic equations of state, which we briefly discuss in section 4.6.2, however the default interface is insufficient for our purposes, so we developed an extension that we discuss in chapter 6.

4.6.1 Functional equations of state

The polytropic equation of state gives p and ϵ as functions of ρ, with two constants, the adiabatic index Γ, and the polytropic constant K, defining the specific polytrope. The relationships are

$$p(\rho) = K\rho^{\Gamma}, \tag{4.15}$$

$$\epsilon(\rho) = \frac{K}{\Gamma - 1}\rho^{\Gamma-1}. \tag{4.16}$$

As the polytropic equation of state is purely functional, it is fast to evaluate, and the conservative-to-primitive routine can be simplified (although this is not done in the Einstein toolkit). The drawback of this equation of state is that, as p and ϵ are functions only of ρ, no thermal effects (e.g. shock heating) are possible in simulations using polytropic equations of state.

The Gamma-Law (or Γ-law) equation of state corresponds to an *ideal gas*, with adiabatic index Γ. For this equation of state, p is a function of both ρ and ϵ (which must then be evolved as a separate variable), specifically

$$p(\rho, \epsilon) = (\Gamma - 1)\rho\epsilon. \tag{4.17}$$

As ϵ is evolved separately to ρ, Γ-law equations of state support thermal effects, and using the ideal gas equations, we can recover T from ϵ through

$$T = (\Gamma - 1)m_b\epsilon, \tag{4.18}$$

where m_b is the baryon mass, i.e. $\rho = m_b n_b$ where n_b is the baryon number density. Initial data for ϵ is typically obtained by using a polytrope, then allowing ϵ to evolve freely in the simulation.

The Hybrid equation of state consists of N (cold) polytropic phases, with a thermal ideal gas component. Let $(K_0, K_1, \ldots, K_N, \Gamma_0, \Gamma_1, \ldots, \Gamma_N)$ characterise the polytropic pieces, where (K_i, Γ_i) is used below the density ρ_i, and Γ_{th} be the adiabatic index of the thermal component. As we require a differentiable matching from polytrope i to $i+1$ at ρ_i, K_{i+1} for $i > 0$ is determined by $K_0, \Gamma_0, \Gamma_1, \ldots, \Gamma_i + 1$. Letting p_{cold} and ϵ_{cold} be

the pressure and specific internal energy of the cold component, and ρ be in the interval $[\rho_{i-1}, \rho_i]$, we have

$$p_{\text{cold}}(\rho) = K_i \rho^{\Gamma_i}, \tag{4.19}$$

$$\epsilon_{\text{cold}}(\rho) = \frac{K_i}{\Gamma_i - 1} \rho^{\Gamma_i - 1} + \epsilon_i, \tag{4.20}$$

where ϵ_i is used to ensure continuity of $\epsilon_{\text{cold}}(\rho)$. The thermal component of p, p_{th}, is given by

$$p_{\text{th}}(\rho, \epsilon) = \rho(\Gamma_{\text{th}} - 1)(\epsilon - \epsilon_{\text{cold}}(\rho)), \tag{4.21}$$

and the final pressure $p(\rho, \epsilon)$ is given by

$$p(\rho, \epsilon) = p_{\text{cold}}(\rho) + p_{\text{th}}(\rho, \epsilon). \tag{4.22}$$

In general, simulations using functional equations of state run faster than those using the tabulated versions described below, but while they are phenomenologically similar to equations of state derived from microphysics, they necessarily lose some of the more complex behaviour we expect to see in neutron star matter, and can introduce artefacts of their own. For more realistic simulations, precalculated tables are used.

4.6.2 Realistic equations of state

Tabluated equations of state are described by a table of values for desired variables, and are generally between 1- and 3-dimensional. `EOS_Omni` implements both Cold-Nuclear+Γ-Law, and Finite-Temperature equations of state. The first of these uses a 1-dimensional table in ρ and a thermal Γ-law, whereas the second uses a 2-dimensional table, given in (ρ, T) or (ρ, ϵ) form, to describe the equation of state.

The Cold-Nuclear+Γ-Law equation of state works in a similar fashion to the Hybrid equations of state described above, with the cold components p_{cold} and ϵ_{cold} coming from a precalculated table in ρ, and a thermal Γ-law component added to simulate thermal effects:

$$p(\rho, \epsilon) = p_{\text{cold}}(\rho) + \rho(\Gamma_{\text{th}} - 1)(\epsilon - \epsilon_{\text{cold}}(\rho)), \tag{4.23}$$

where Γ_{th} is the adiabatic index of the thermal component.

Given a table describing an equation of state, it is necessary to interpolate within that table to obtain values of the dependent variables that do not line up with the grid defined by the table. Care must be taken to ensure that this interpolation is sufficiently smooth such that spurious shocks do not arise in the simulation owing to inaccurate equation of state calls.

As we wish to study composition and temperature dependent effects (see chapters 8 to 10), we require 3-dimensional tables in (ρ, T, Y_e), such as those available on CompOSE[42]. In chapter 6 we present modifications to the toolkit, specifically to `EOS_Omni` and `GRHydro`, that allow us to perform simulations using such tables.

Chapter 5

Data Processing

In this chapter we discuss the process of analysing simulation data, and creating visualisations of results. As example data, we use the APR and Γ-law simulations described in further detail in chapter 7 for fluid and gravitational wave data respectively. It is natural to divide this section into a discussion focusing on the fluid, and one on the spacetime, but we will first talk briefly about how we move from the complex grid structure discussed in section 4.2.1 into a simpler structure more suitable for visualisation. We note that there are some freely available visualisation packages designed specifically for the Einstein Toolkit, notably `kuibit`[33] and `PyCactus`[76], however for historical reasons (`kuibit` was not public when this project began, and `PyCactus` is both lacking desired features and written in an outdated language) we chose to build our own tools.

5.1 Grid Reconstruction

While the AMR grids used by `Carpet` are good for computational workload, when it comes to performing analysis and creating visualisations it can be more helpful to have a uniform grid that covers only a small region of interest. In fig. 5.1 we can see a snapshot of the grid structure for a binary neutron star merger as produced by the Einstein toolkit. The domain contains a number of refinement levels, some of which are centered on the neutron stars, and some at the barycentre of the system (which coincides with the coordinate origin). Each of these refinement levels is then divided up into a sub-grid for each CPU used in the simulation (in the case of fig. 5.1, 4 CPUs were used).

We have two methods for taking data from the AMR grids and placing it on a uniform mesh. The first method is useful for simple 2- or 3- dimensional cartesian slices, and uses the finest AMR grid available in the region of interest as the basis (origin and grid spacing) for a *receiving* grid that spans the entire region of interest with a uniform grid

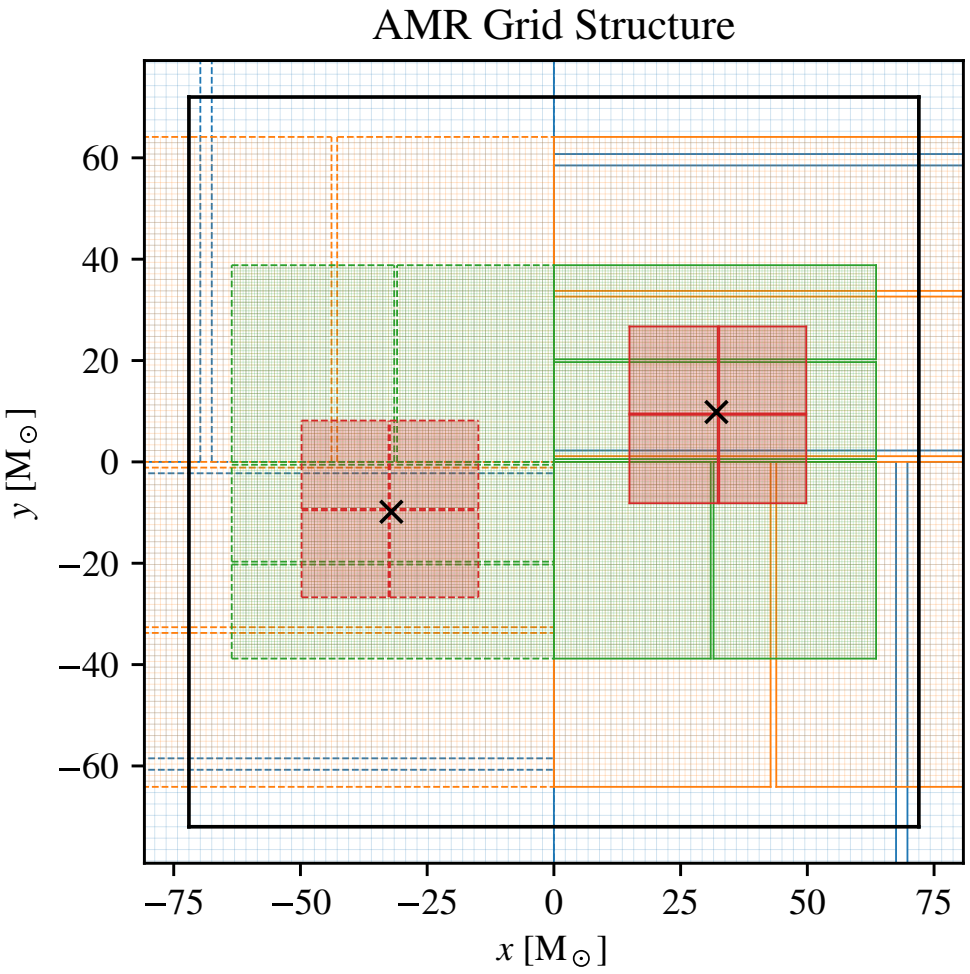

FIGURE 5.1: `Carpet` AMR grid near the neutron stars, with the black square denoting the domain of interest. Colours are used on a per refinement level basis, and the extent of each sub-grid (with ghost zones removed) is shown. This simulation used a rotational symmetry of π about the z-axis, with only the positive-x region evolved. The repeated part of the domain is represented by grids surrounded by dashed lines. The locations of the neutron stars are marked by $\times$s at their centres, where we define the centre of the star as the point where ρ is at its maximum.

spacing. We then determine which AMR grids are needed to cover the entire receiving grid (giving priority to finer grids), and use linear interpolation on all grids other than the finest to increase the resolution such that it matches that of the receiving grid (using ghost zones between AMR grids on the same refinement level to bridge the gap between them). Each point in the receiving grid then takes the data from the finest AMR grid that covers that location.

For more complicated receiving grids (e.g. a polar slice), once the domain has been defined and the AMR grids which completely cover it have been chosen, starting from the coarsest AMR grid and working towards the finest, we build a 3-D polynomial

interpolator for the data on each AMR grid (including ghost zones), and evaluate the interpolator at all receiving grid points inside that particular AMR grid. Typically we only use degree 1 interpolation, as higher orders tend to introduce spurious oscillations in regions where the gradient of the variable changes quickly (e.g. the density at the stellar surface).

These methods are defined only for scalar data, so for vector data we perform the interpolation separately on each component.

5.2 Fluid Variables

Many of the fluid variables evolved in the simulation are of interest for post-processing, analysis, and visualisation. We will briefly discuss a few of these quantities and how they are calculated and visualised. In order to detail these processes, we will split the discussion into sections on scalar and vector quantities. Throughout this section we take example data from the APR simulation described in chapter 7.

5.2.1 Scalars

Much of the data we plot in later chapters is scalar. In addition to the variables evolved in the simulation, such as density and temperature, we also require variables that are not directly evolved by the simulation, but instead can be calculated from the equation of state. In fig. 5.2 we plot the three equation of state parameters ρ, T, Y_{e} (see chapter 6 for more detail), and a variable that is not evolved directly but calculable in post-processing using the equation of state, the neutron chemical potential μ_{n}. Additionally, the rest mass density is show on a log scale in units of the nuclear saturation density ρ_{nuc}.

Another interesting quantity is the enthalpy weighted vorticity $\Omega_{\mu\nu}$ given by

$$\Omega_{\mu\nu} = \partial_\mu(hu_\nu) - \partial_\nu(hu_\mu). \tag{5.1}$$

This quantity is of interest as, loosely, a non-zero $\Omega_{\mu\nu}$ implies the presence of a vortex in the μ-ν plane. In particular, we are interested in the component of $\Omega_{\mu\nu}$ aligned with the rotational axis of the neutron star binary, Ω_{xy}, which is a scalar. In terms of value, we are interested in a range of a few orders of magnitude on both sides of 0, but not values around 0. Therefore, we use a *log-lin-log* plot. A log-lin-log plot contains a central linear region that covers either side of 0, with logarithmic plots extending above and below (for the lower portion, the sign of the data is swapped before taking the logarithm) allowing a range of orders of magnitude to be seen with both positive and negative sign. Figure 5.3 contains a plot of Ω_{xy} as a binary neutron star merges. We can see that there

is strong positive vorticity (right handed vortex with respect to the positive z-axis) at the interface of the two stars, and also regions of negative vorticity.

There are, of course, other scalar simulation variables of interest, but the plots here demonstrate the presentation of scalar fluid data.

5.2.2 Vectors

Next, we will cover the two principal means by which we convey information about vector fluid variables. Given that we are not studying magnetic fields in our simulations, the most interesting of these is the fluid velocity.

The most common means of displaying fluid velocity is through quiver diagrams, where a small number of vector arrow are added to the plot corresponding to the magnitude and direction of the fluid flow at their base. While it is possible to plot the raw velocity through a quiver plot, the fast moving fluid at the edge of the stars and in the atmosphere tend to dominate, but as these regions are several orders of magnitude lower in density than the cores of the stars, their effect on the overall dynamics of the merger is much lower. To combat this, instead of plotting just the raw velocity data, we can use a mask based on the density of the fluid to suppress the velocity in the regions that are not of interest. Figure 5.4 shows a selection of such plots with densities $\rho < 10^{11}\mathrm{g/cm}^3$ ignored.

Another method we have developed is to map variables such as ρ, $|\boldsymbol{v}|$, and direction to a colour-space. In fig. 5.5 we use the Hue-Saturation-Value (HSV) colour-space. The direction in the xy-plane θ of the velocity vector $\boldsymbol{v}$ maps to the hue of the pixel, the speed $|\boldsymbol{v}|$ maps to saturation, and ρ maps to value. While these plots do not easily allow reading of quantitative values, they do provide a good qualitative idea of how the fluid in different areas is moving, and features such as vorticies and differential rotation are much easier to pick out than with a traditional quiver diagram.

5.3 Spacetime Variables

When looking at the dynamics of the spacetime, the field variables can be plotted in a similar fashion to the fluid variables we have looked at above. The main feature of interest that cannot be analysed in such a way is gravitational wave emission. As it is the longest simulation we have performed (and therefore has the most gravitational wave data available), in this section we use data from the Γ-law simulation described in chapter 7.

5.3.1 Gravitational waves

It is possible to extract the gravitational wave signal from a signal using a retarded-time integral over the entire domain, for each direction we want to know the signal, often enough to get a clean time series. This obviously poses an extreme computational challenge, so we use a method based on the *Weyl-scalars* $\Psi_{0,\ldots,4}$, in particular Ψ_4.

The Weyl-scalars come from the *Newman-Penrose* formalism of general relativity. A full derivation of Ψ_4 can be found in[124], but here we quote the result (which differs from[124] by a sign due to a convention in the thorn that measures Ψ_4 in the Einstein toolkit):

$$\Psi_4 = -\ddot{h}_+ + i\ddot{h}_\times, \tag{5.2}$$

where the $+, \times$ denote the two polarisations of gravitational waves. The Einstein toolkit allows the calculation of the -2 *spin-weighted spherical harmonics* ${}_{-2}Y_{lm}(\theta,\phi)$ of Ψ_4 over spherical surfaces at chosen distances from the origin. This allows us to pick any direction from which to measure the gravitational wave signal.

In order to obtain a gravitational wave signal that can be compared to detector capabilities from the decomposed Ψ_4 we must:

1. Integrate Ψ_4 twice in time
2. Evaluate ${}_{-2}Y_{lm}(\theta,\phi)$ for our chosen direction and sum the contributions
3. Extrapolate the amplitude out to a realistic distance

The last two of these are done using this equation for ${}_{-2}Y_{lm}(\theta,\phi)$:

$$\begin{aligned} {}_{-2}Y_{lm}(\theta,\phi) = (-1)^m \sqrt{\frac{(l+m)!(l-m)!(2l+1)}{4\pi(l+2)!(l-2)!}} \sin^{2l}\left(\frac{\theta}{2}\right) \\ \times \sum_{r=0}^{l+2} \left[\left(\frac{l+2}{r}\right) \left(\frac{l-2}{r-m-2}\right) (-1)^{l-r+2} e^{im\phi} \cot^{2r-m-2}\left(\frac{\theta}{2}\right) \right], \end{aligned} \tag{5.3}$$

and summing the separate components, then using the fact that $h_{+,\times}$ will fall off as $1/r$ to get a realistic amplitude (a commonly chosen distance is 100 Mpc). The double time integration poses more of an issue.

The problem with numerically integrating Ψ_4 twice is that even small amounts of noise (on the order of numerical roundoff) can spoil the signal due to "wandering"[1]. Instead, we use the technique of *Fixed-Frequency-Integration* (FFI) (see[113] for a comparison of methods for performing this integration).

[1] The first integral of a 1-D Gaussian white noise process is Brownian motion, so this wandering is not unexpected after a second integration.

The main idea behind FFI is surpressing the noise in low frequencies while performing the integrations in the frequency domain. The first step is to take the *Discrete Fourier Transform* (DFT) $\mathcal{F}$ of the times series of $\Psi_4(t)$, then using the relation

$$\iint \Psi_4(t)dt = \mathcal{F}^{-1}\left(\frac{\mathcal{F}\left(\Psi_4(t)\right)}{-\omega^2}\right) \tag{5.4}$$

to perform the integration. The difference with FFI is that instead of the raw frequency ω, we use a function of the frequency $\omega^*(\omega)$ in the denominator of eq. (5.4) defined by

$$\omega^*(\omega) = \begin{cases} \omega & \omega > \omega_0 \\ \omega_0 & \omega \leq \omega_0 \end{cases}, \tag{5.5}$$

where the cutoff frequency ω_0 should be lower than any physical gravitational wave frequency we expect to see in the simulation.

Once we have obtained the time series of the gravitational wave signal in a given direction, we can study it in both time and frequency domains. Figure 5.6 contains a number of plots detailing the simulated gravitational wave signal from a pair of $1.2M_\odot$ neutron stars. The figure shows the signal expected for a detector positioned 100 Mpc away from the merger directly above the orbital plane. Plotted are the two polarisations of the strain data $h_{+,\times}$, the amplitude envelope (calculated as $|h_+ + ih_\times|$), and the instantaneous frequency, along with a spectrogram of the signal.

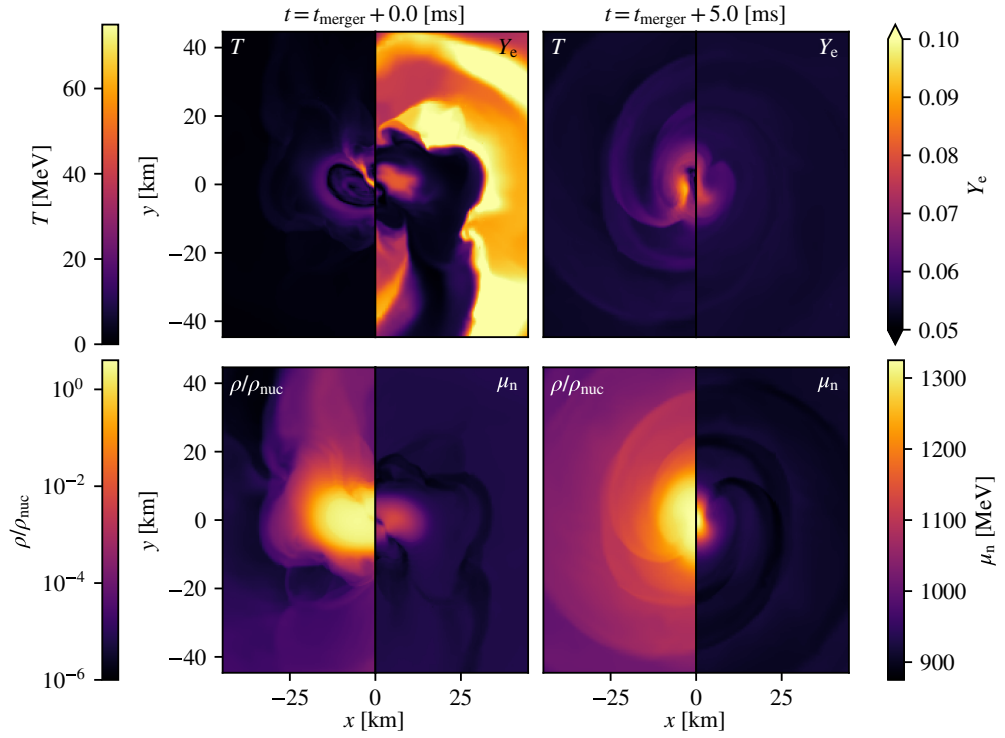

FIGURE 5.2: Plots of several example fluid variables demonstrating scalar plots. Data was taken from the simulation of two $1.4\,M_\odot$ neutron stars, using the APR[5;122] equation of state. The left column is from the point the stars begin to merge, and the right plot is from 5 ms later. Plotted are the fluid temperature T, electron fraction Y_e, rest mass density ρ divided by nuclear saturation density ρ_{nuc} (on a log scale), and neutron chemical potential μ_n.

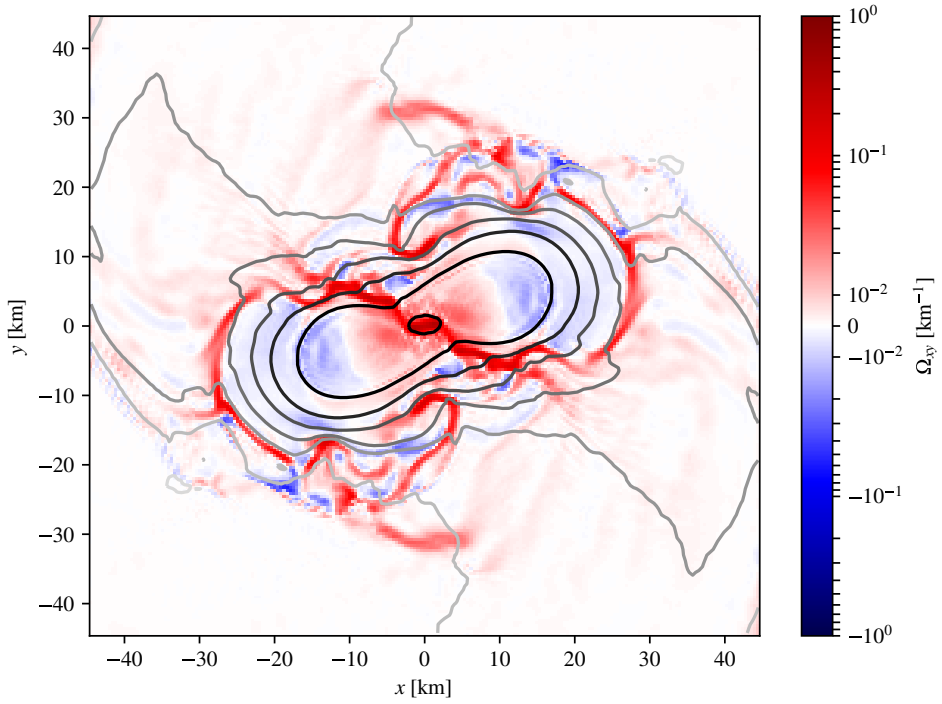

FIGURE 5.3: Log-lin-log plot of the xy-vorticity Ω_{xy}. The logarithmic portions of the plot extend over the intervals $\Omega_{xy} = \pm[1.0\times10^{-2}, 1.0\times10^{0}]\,\mathrm{km}^{-1}$, and the plot is linear for values falling in the range $\Omega_{xy} = [-0.01, 0.01]\,\mathrm{km}^{-1}$. Both stars had initial mass of $1.4\,M_{\odot}$, using the APR[5;122] equation of state. The plot shows the stars around 0.5 ms after they first touch. Contours of ρ/ρ_{nuc} are shown in shades of grey at unit intervals in order of magnitude from 1.0×10^{-6} to 1.0×10^{0}.

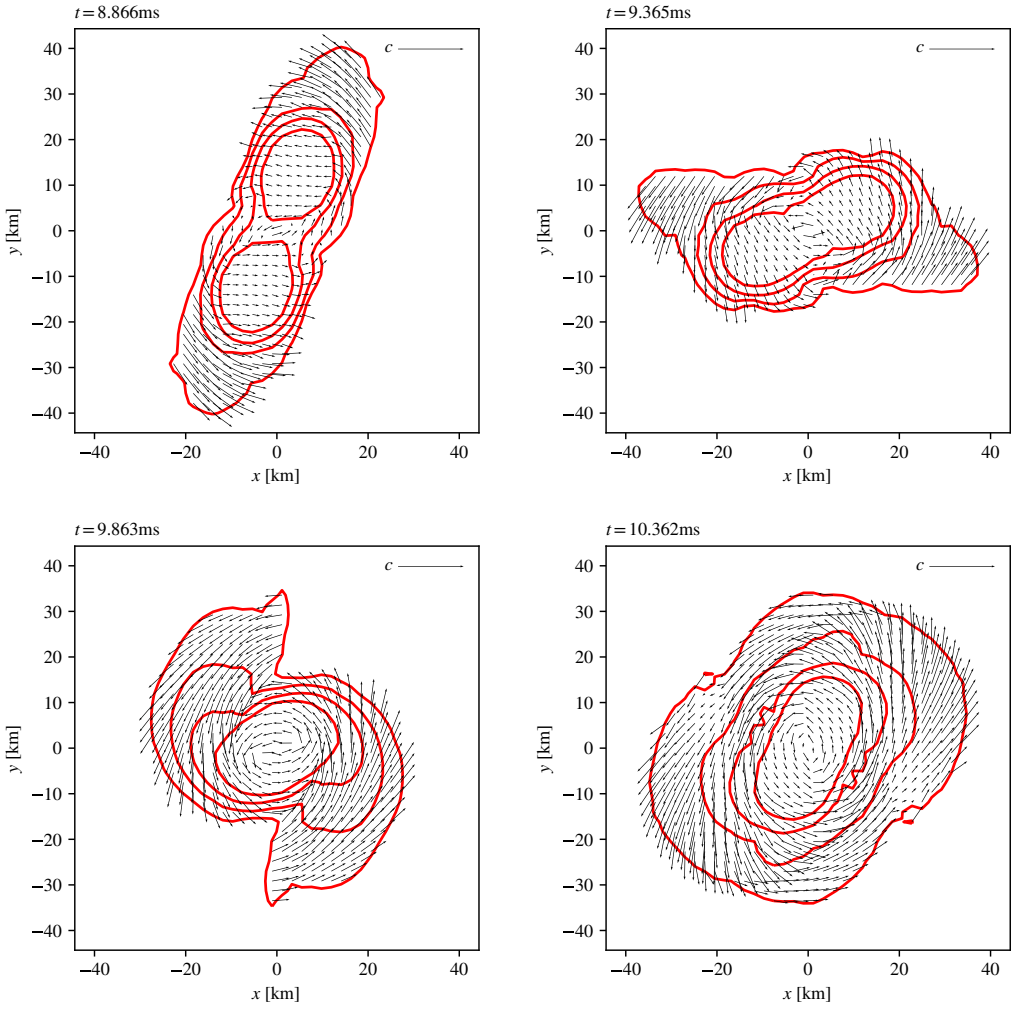

FIGURE 5.4: Quiver plots of the fluid velocity in the orbital plane at 4 times in a simulation: the first is at the time the stars begin to merge, and the others follow at 0.5 ms intervals. The quivers are normalised to c as displayed in the top right-hand corner of each plot. Also shown are rest mass density ρ contours at unit intervals in order of magnitude in red from 1.0×10^{11} to $1.0 \times 10^{14}\,\mathrm{g/cm^3}$. Values of velocity where the density is $\rho < 1.0 \times 10^{11}\,\mathrm{gcm^{-3}}$ are ignored.

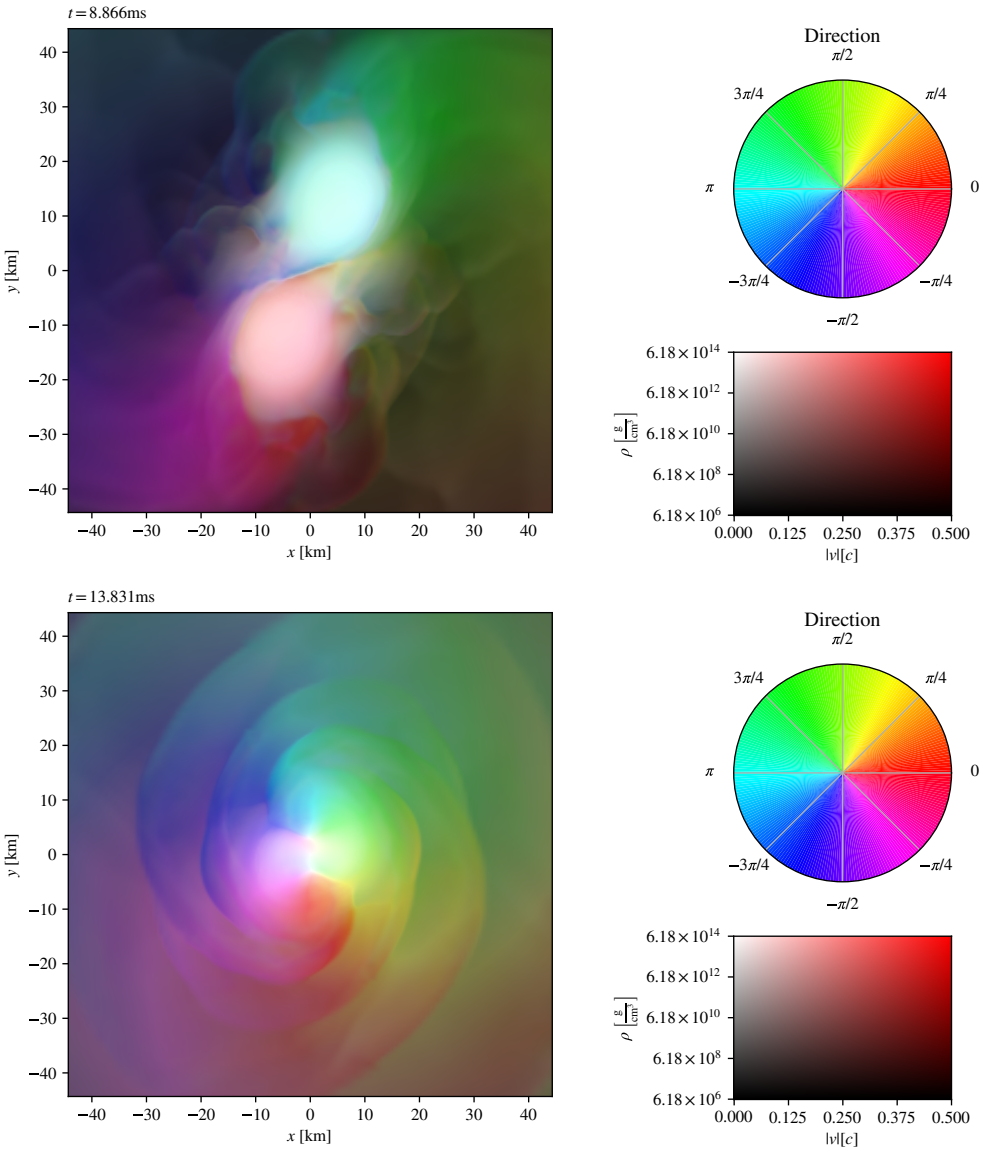

FIGURE 5.5: Two plots illustrating the velocity and density of the fluid in a simulation of a pair of 1.4 $M_\odot$ neutron stars with the APR [5;122] equation of state. Using the HSV colour-space, the hue of a pixel denotes the direction in which the fluid is moving at that point, the saturation denotes the speed of the fluid at that point, and the value denotes the density. In the top panel the stars are shown as they begin to merge, and in the bottom panel they are shown after the remnant has been formed but before it relaxes to a steady state, approximately 5 ms later.

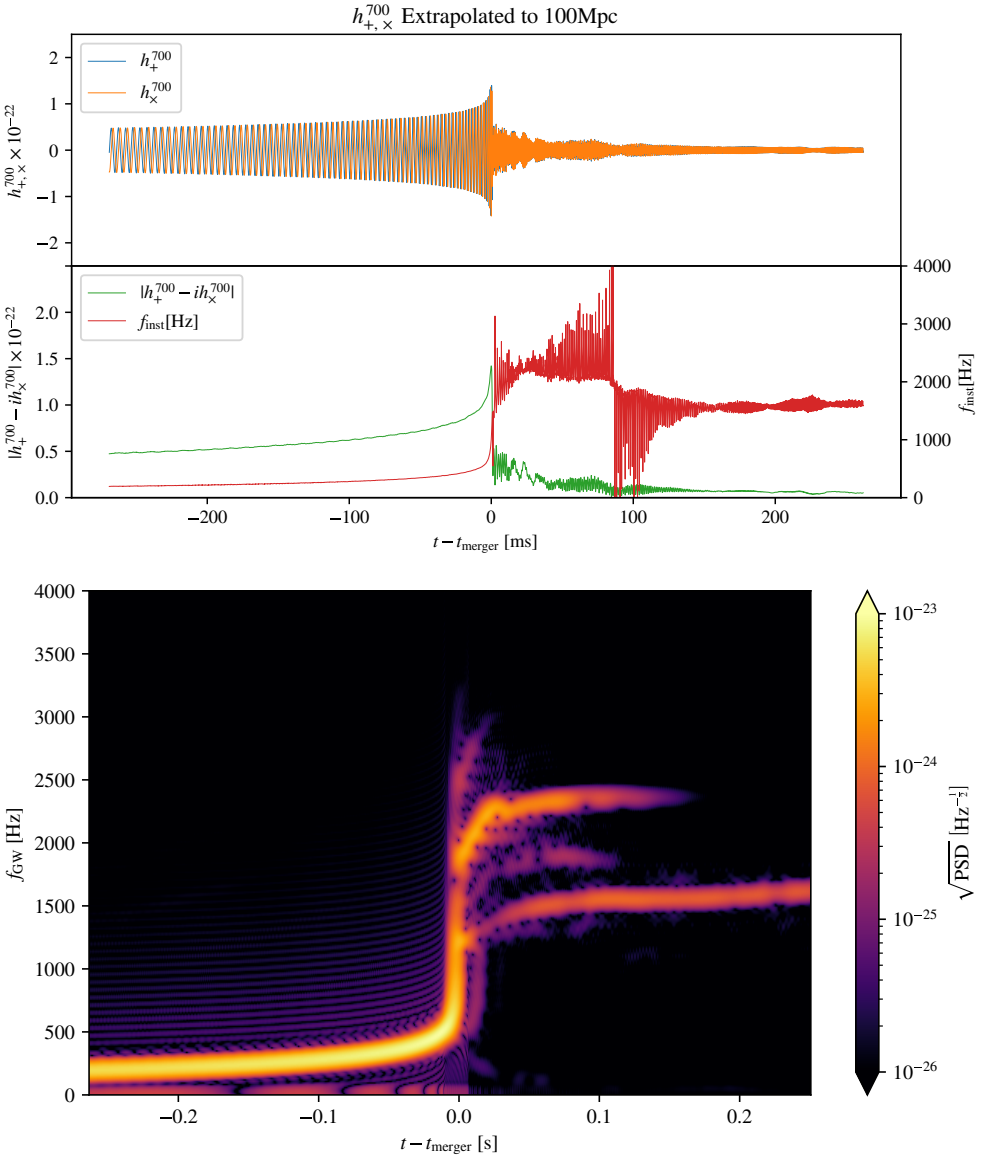

FIGURE 5.6: Two plots detailing the gravitational wave emission of a pair of $1.2M_\odot$ neutron stars as they inspiral, merge, and ring-down. The signal was calculated from measurements on a sphere of radius $\sim 1000\,\mathrm{km}$ ($700\,M_\odot$ in simulation units) centered on the barycenter of the system. The top panel shows the raw signal in the top half, and the envelope and instantaneous frequency in the bottom. The bottom panel shows a spectrogram of the signal using a short-time-fourier-transform to show the evolution of the signal.

Chapter 6

Extending Equations of State

The equation of state interface in the Einstein toolkit, `EOS_Omni` (see section 4.6), features support for 1- and 2- parameter tabulated equations of state. 1-parameter equations of state are tabulated only in ρ, and assume the fluid is cold[1] and in β-equilibrium, also known as *nuclear statistical equilibrium* (see section 8.1 for a discussion of β-equilibrium as typically defined in the context of equations of state). 2-parameter equations of state are tabulated by ρ, and typically either T or ϵ, and allow the inclusion of thermal effects while maintaining the assumption of β-equilibrium. In order to study both thermal effects, and effects that arise due to violation of β-equilibrium, we wish to remove this assumption. Therefore, we need to evolve the fluid electron fraction Y_e, and utilise an equation of state that depends on it. `GRHydro` has the ability to advect Y_e, but we must extend `EOS_Omni` so that tabulated 3-parameter equations of state can be used (a library of tabulated equations of state is available online through CompOSE [42]). We then need to interpolate within the table to obtain the variables required by `GRHydro`. In this chapter we present modifications made to `EOS_Omni` and `GRHydro` to enable this.

6.1 Tabulated Equations of State

In order to use tabulated equations of state within `EOS_Omni`, we have developed a means of generating the code for an n-th order polynomial interpolator in d-dimensions (where n is odd). We chose to create a general interpolation routine as the order of interpolation may need to be increased to ensure the results are sufficiently smooth, and in the future we may need to use a higher number of parameters in the equation of state (to account for other physics such as magnetic fields, neutrinos, strangeness, etc.).

[1] It is possible to have a 1-dimensional equation of state with $T = T(\rho)$, however as they are commonly used with a thermal Γ-law component (see section 4.6.2), they typically assume $T = 0$.

6.1.1 Interpolation

To create the general n-th order polynomial interpolator, we begin by looking at the tabulated values of the variable y, which we denote $\boldsymbol{T}_y$. We assume these values are given at the vertices of a d-dimensional grid of unit d-cubes, and given a coordinate vector $\boldsymbol{X}$ of length d with components X_i, the value in the table at that location is labelled $\boldsymbol{T}_y(\boldsymbol{X})$. We then look at each d-cube individually to construct the interpolator (each d-cube has a different set of values for the interpolation coefficients, but they are constructed in the same way).

Assuming we have chosen a d-cube with coordinates $\boldsymbol{X}$, our interpolator $f_y(\boldsymbol{x})$ will take the form

$$f_y(\boldsymbol{x}) = \sum_{p_1,\ldots,p_d=0}^{n} a_{p_1,\ldots,p_d} \prod_{i=1}^{d} x_i^{p_i}, \tag{6.1}$$

where $\boldsymbol{x}$ is a vector of length d whose components are bounded by $0 \leq x_i < 1$ that gives the coordinates at which we are evaluating the interpolator inside the given d-cube, and $a_{p_1,\ldots,p_d}$ are the coefficients of the polynomial for that d-cube. It is these coefficients $a_{p_1,\ldots,p_d}$ that we must solve for.

For a given d,n, there will be $(n+1)^d$ coefficients to solve for. For each d-cube, we have the value of y given at each corner, of which there are 2^d. We can also calculate finite differences on the table values to obtain derivatives of the variable y with respect to each direction in the table. Each derivative will give us another 2^d values, so (including the variable itself as the 0^{th} order derivative) the number of derivatives we need to calculate N at each vertex is

$$N = \left(\frac{n+1}{2}\right)^d, \tag{6.2}$$

in order to solve for the coefficients $a_{p_1,\ldots,p_d}$. So as not to over-constrain the system, these derivatives cannot go higher than order $(n-1)/2$ in any given direction (Lekien[84] proves this for 2^{nd} order derivatives for a 3^{rd} order interpolator, but the result generalises to higher n). The remaining derivative terms are then given by

$$m_{l_1,\ldots,l_d} = \frac{\partial^{\left(\sum_{i=1}^{d} l_i\right)} \boldsymbol{T}_y(\boldsymbol{X})}{\partial x_1^{l_1} \ldots \partial x_d^{l_d}}, \tag{6.3}$$

where $0 \leq l_i \leq (n-1)/2$. There will therefore be $(n+1)/2$ values for each of the derivative orders l_i, and as we are working in d dimensions the total number of derivative terms will be $((n+1)/2)^d$, exactly the number we need to keep the system properly determined.

Now we have the correct number of values to solve for $a_{p_1,\ldots,p_d}$, we first need to find the matrix $\mathbf{B}$ that satisfies the equation

$$\boldsymbol{m} = \mathbf{B}\boldsymbol{a}, \tag{6.4}$$

where $\boldsymbol{m}$ is a vector containing the derivative terms $m_{l_1,\ldots,l_d}$ evaluated at each vertex of the d-cube which we denote as $m^k_{l_1,\ldots,l_d}$ where $1 \leq k \leq 2^d$ labels each of the vertices, and $\boldsymbol{a}$ contains the coefficients $a_{p_1,\ldots,p_d}$. To determine the elements of $\mathbf{B}$, we apply the derivative terms in eq. (6.3) to our ansatz of the interpolator in eq. (6.1) instead of $\boldsymbol{T}_y(\boldsymbol{X})$ and read off the numerical coefficients that appear in front of $a_{p_1,\ldots,p_d}$ for each $m^k_{l_1,\ldots,l_d}$. We then invert the matrix to give $\mathbf{B}^{-1}$, so the coefficients can be written as

$$\boldsymbol{a} = \mathbf{B}^{-1}\boldsymbol{m}. \tag{6.5}$$

The elements in $\mathbf{B}^{-1}$ are general to each d-cube, so the interpolators for each d-cube differ only in the values used for $\boldsymbol{m}$. For some equation of state parameters (namely ρ and T) it makes sense to tabulate by the logarithm of the variable, as they vary over a significant number of orders of magnitude. For such variables we use the logarithms as the independent variable in the construction of the interpolator. We also use similar measures for dependent variables that vary over several orders of magnitude:

- We use $\log_{10}(p)$ as p is strictly positive and grows like ρ to some power,
- We use $\operatorname{asinh}(\epsilon)$ as ϵ can be negative, but also grows significantly for large ρ, T.

By making these modifications to the tabulated data, we also reduce oscillation in the output of the finite order interpolators when the data is exponential-like.

6.1.2 Evaluation

Now that we can interpolate inside the table, when the simulation requires the equation of state for parameter values $\boldsymbol{Y}$, we first determine which d-cube in the table contains the location $\boldsymbol{Y}$, rescale $\boldsymbol{Y}$ to its location in the unit d-cube we have selected to give $\boldsymbol{y}$, and use this in the interpolator for that cube. Once we have evaluated the interpolating polynomial, we must then take the inverse of any function that was used when tabulating the data ($\log_{10}$ for p and asinh for ϵ) to obtain the the value of the equation of state at the given point.

In addition to the raw table values, we also require the derivatives of some of the table values with respect to the table parameters. As previously, we first determine the correct d-cube for our given $\boldsymbol{Y}$ and calculate $\boldsymbol{y}$. We then exploit the polynomial nature of the interpolating functions to calculate the desired derivative. Looking back at our interpolator ansatz in eq. (6.1), we can evaluate the sum and product terms for all parameters except the one we are taking a derivative with respect to, say the j-direction, which leaves us with

$$f(\boldsymbol{y}) = \sum_{p_j=0}^{n} b_{p_j} y_j^{p_j}, \tag{6.6}$$

where the b_{p_j} terms collect the evaluation over the other table parameters $y_{i\neq j}$ for each power of y_j. We can then take the derivative of this polynomial in y_j with respect to y_j to give

$$\frac{\partial f(\boldsymbol{y})}{\partial y_j} = \sum_{p_j=1}^{n} p_j b_{p_j} y_j^{p_j-1}. \tag{6.7}$$

We then use

$$\frac{\partial f(\boldsymbol{y})}{\partial Y_j} = \frac{\partial y_j}{\partial Y_j}\frac{\partial f(\boldsymbol{y})}{\partial y_j} = \frac{1}{\Delta Y_j}\frac{\partial f(\boldsymbol{y})}{\partial y_j}, \tag{6.8}$$

where ΔY_j is the size of the table cell in the Y_j direction.

Where we have used $Y_j = \log_{10}(X_j)$ as a table parameter instead of X_j itself, we recover the derivative with respect to X_j by using

$$\begin{aligned}\frac{\partial}{\partial X_j} &= \frac{\partial \log_{10}(X_j)}{\partial X_j}\frac{\partial}{\partial \log_{10}(X_j)} \\ &= \frac{1}{X_j \log(10)}\frac{\partial}{\partial \log_{10}(X_j)}.\end{aligned} \tag{6.9}$$

Where we have used $g(\boldsymbol{X}) = \log_{10}(f(\boldsymbol{X}))$ as a dependent variable, we recover the derivative of $f(\boldsymbol{X})$ with respect to X_j from the derivative of $g(\boldsymbol{X})$ through

$$\begin{aligned}\frac{\partial f(\boldsymbol{X})}{\partial X_j} &= \frac{\partial f(\boldsymbol{X})}{\partial g(\boldsymbol{X})}\frac{\partial g(\boldsymbol{X})}{\partial X_j} \\ &= \left(\frac{\partial g(\boldsymbol{X})}{\partial f(\boldsymbol{X})}\right)^{-1}\frac{\partial g(\boldsymbol{X})}{\partial X_j} \\ &= f(\boldsymbol{X})\log(10)\frac{\partial g(\boldsymbol{X})}{\partial X_j}.\end{aligned} \tag{6.10}$$

Similarly, when the dependent variable $h(\boldsymbol{X}) = \operatorname{asinh}(f(\boldsymbol{X}))$ is used, we obtain the derivative of $f(\boldsymbol{X})$ through

$$\begin{aligned}\frac{\partial f(\boldsymbol{X})}{\partial X_j} &= \left(\frac{\partial h(\boldsymbol{X})}{\partial f(\boldsymbol{X})}\right)^{-1}\frac{\partial h(\boldsymbol{X})}{\partial X_j} \\ &= \sqrt{f(\boldsymbol{X})^2+1}\frac{\partial h(\boldsymbol{X})}{\partial X_j}.\end{aligned} \tag{6.11}$$

6.1.3 Inversion

For some conservative-to-primitive methods, it is necessary to determine T from a given ϵ (and other known table parameters). Unlike the previous work in this section, for this step we have developed a method specific to a 3-D table with parameters ρ, T, and Y_e, using 3$^{\mathrm{rd}}$ order interpolation.

Given an initial guess for the temperature T_{guess} and a target ϵ, we locate the table cell containing $(\rho, T_{\mathrm{guess}}, Y_\mathrm{e})$, then evaluate eq. (6.1) in the ρ and Y_e directions to obtain a

cubic polynomial in the T direction

$$f(t) = at^3 + bt^2 + ct + d - \epsilon, \tag{6.12}$$

where $0 \leq t < 1$ is the local T coordinate for that cell. We then solve $f(t) = 0$ using a standard cubic equation method, discarding any roots outside the interval $0 \leq t_{\rm root} < 1$. If $f(t)$ gives multiple roots inside the interval, the one minimising $\Delta T = |T(t_{\rm root}) - T_{\rm guess}|$ is used. If either no root is found in the interval, or it is possible that a root closer to $T_{\rm guess}$ exists in an adjacent cell, we then check for roots in the cells above and below the cell containing $T_{\rm guess}$.

This process of moving up and down the table in the T direction continues until a root has been found in the correct interval for a cell, and it is not possible for there to be a root with smaller ΔT. This signifies that the inversion of ϵ is complete, and $T = T(t_{\rm root})$ is returned. If no such root is found in the entire temperature range of the equation of state used, the method returns unsuccessfully, and either the conservative-to-primitive solver moves onto a backup method, or the simulation terminates.

6.2 Modifying Con2Prim

The conservative-to-primitive (Con2Prim) method used by `GRHydro` relies on the availability of the derivatives $\partial p/\partial \rho$ and $\partial p/\partial \epsilon$ from the equation of state, however, most 3-parameter equations of state are tabulated by ρ, T, and $Y_{\rm e}$, with ϵ a function of these parameters. This means that $\partial p/\partial \epsilon$ must be calculated through

$$\begin{aligned} \frac{\partial p}{\partial \epsilon} &= \frac{\partial p}{\partial T}\frac{\partial T}{\partial \epsilon} + \frac{\partial p}{\partial \rho}\frac{\partial \rho}{\partial \epsilon} \\ &= \frac{\partial p}{\partial T}\left(\frac{\partial \epsilon}{\partial T}\right)^{-1} + \frac{\partial p}{\partial \rho}\left(\frac{\partial \epsilon}{\partial \rho}\right)^{-1}, \end{aligned} \tag{6.13}$$

which, when any of the $\partial \epsilon/\partial X$ terms approaches 0, can explode to infinity, leading to instability in the Newton-Raphson iteration.

To remedy this, we have implemented two different Con2Prim methods taken from Siegel et al.[127] which are more suited to equations of state in the form described above, and a backup method for when these both fail. These methods (which we have named for the dimensionality and type of solver used, and the chosen variables in which the solver searches) are:

- 2-D Newton-Raphson in (W, T),
- 1-D bracketed in $x = hW$,
- 1-D bracketed in W with fixed T,

where

$$W = \frac{1}{\sqrt{1 - v_i v^i}} \tag{6.14}$$

is the local Lorentz factor of the fluid, T is the fluid temperature, and $h = 1 + \epsilon + p/\rho$ is the specific enthalpy of the fluid.

These methods in particular were chosen to give a balance between performance and stability[127]. The 2-dimensional Newton-Rapshon method is fast, in terms of both number and type of equation of state call, the 1-dimensional bracketed method in hW is guaranteed to converge (assuming a robust temperature inversion procedure, and a set of conservative variables that correspond to primitives within the range of the equation of state)[127], and the final method is designed to for use when the conservative variables correspond to a primitive solution that is out of reach of the table.

6.2.1 2-D Newton-Raphson

This Con2Prim method is based on the 2-D scheme presented by Anton et al.[20], reformulated in terms more suited to tables parameterised by (ρ, T, Y_e)[127].

Looking back at the definitions of the conserved variables D, S_i, τ, and Y_e in section 3.3 we have

$$D = \rho W \tag{6.15}$$
$$S_i = z v_i \tag{6.16}$$
$$\tau = z - p - D \tag{6.17}$$
$$DY_\mathrm{e} = D \cdot Y_\mathrm{e} \tag{6.18}$$

where $z \equiv \rho h W^2$, and we also note that

$$\begin{aligned} S^2 &= S_i S^i \\ &= z^2 v_i v^i \\ &= z^2 \frac{W^2 - 1}{W^2}. \end{aligned} \tag{6.19}$$

Using eqs. (6.17) and (6.19) we can define the two functions

$$f_1(W, T) = \left[\tau + D - z(W, T) + p(W, T)\right] W^2 = 0, \tag{6.20}$$
$$f_2(W, T) = \left[z(W, T)^2 - S^2\right] W^2 - z(W, T)^2 = 0, \tag{6.21}$$

where $z(W,T)$ and $p(W,T)$ are calculated from the equation of state table parameterised by (ρ, T, Y_e) using

$$\rho(W) = \frac{D}{W}, \tag{6.22}$$

$$Y_\mathrm{e} = \frac{DY_\mathrm{e}}{D}, \tag{6.23}$$

and

$$p(W,T) = p(\rho(W), T, Y_\mathrm{e}), \tag{6.24}$$

$$\epsilon(W,T) = \epsilon(\rho(W), T, Y_\mathrm{e}), \tag{6.25}$$

$$z(W,T) = \rho(W)\left[1 + \epsilon(W,T) + \frac{p(W,T)}{\rho(W)}\right]W^2. \tag{6.26}$$

We can then use a 2-D Newton-Raphson method in the unknowns (W,T) to perform the conservative-to-primitive inversion.

With initial guesses for $(W,T) = (W_0, T_0)$, the root-finding iteration will be

$$\begin{pmatrix} W_{n+1} \\ T_{n+1} \end{pmatrix} = \begin{pmatrix} W_n \\ T_n \end{pmatrix} - J_{\boldsymbol{F}}(W_n, T_n)^{-1}\boldsymbol{F}(W_n, T_n), \tag{6.27}$$

where $\boldsymbol{F}(W,T)$ is given by

$$\boldsymbol{F}(W,T) = \begin{pmatrix} f_1(W,T) \\ f_2(W,T) \end{pmatrix}. \tag{6.28}$$

The Jacobian is given by

$$J_{\boldsymbol{F}}(W,T) = \begin{pmatrix} \dfrac{\partial f_1}{\partial W} & \dfrac{\partial f_1}{\partial T} \\ \dfrac{\partial f_2}{\partial W} & \dfrac{\partial f_2}{\partial T} \end{pmatrix}, \tag{6.29}$$

so the four derivatives we need are:

$$\frac{\partial f_1}{\partial W} = 2W\left[\tau + D - z(W,T) + p(W,T)\right] - D\frac{\partial p}{\partial \rho} - W^2\frac{\partial z}{\partial W}, \tag{6.30}$$

$$\frac{\partial f_1}{\partial T} = W^2\left[\frac{\partial p}{\partial T} - \frac{\partial z}{\partial T}\right], \tag{6.31}$$

$$\frac{\partial f_2}{\partial W} = 2W\left[z(W,T)^2 - S^2\right] + 2z(W,T)\left[W^2 - 1\right]\frac{\partial z}{\partial W}, \tag{6.32}$$

$$\frac{\partial f_2}{\partial T} = 2z(W,T)\left[W^2 - 1\right]\frac{\partial z}{\partial W}, \tag{6.33}$$

where the derivatives of z with respect to the equation of state table parameters are

$$\frac{\partial z}{\partial W} = D\left[1 + \epsilon(W,T) - \rho(W)\frac{\partial \epsilon}{\partial \rho}\right] + 2Wp(W,T) - D\frac{\partial p}{\partial \rho}, \tag{6.34}$$

$$\frac{\partial z}{\partial T} = DW\frac{\partial \epsilon}{\partial T} + W^2\frac{\partial p}{\partial T}. \tag{6.35}$$

Note that as we discussed above, all of these derivatives are with respect to table parameters.

The iteration is continued until both $|f_1(W_n, T_n)|$ and $|f_2(W_n, T_n)|$ fall below an accuracy threshold, or n exceeds the maximum allowed number of iterations $n_{\max}$. If the process fails to converge before $n_{\max}$ is reached, the first backup Con2Prim method, a bracketed root find in $x = hW$, is used.

6.2.2 1-D bracketed in $x = hW$

This method by Palenzuela et al.[101] solves for $x = hW$ in the bracketed interval

$$1 + q \leq x \leq 2(1 + q), \tag{6.36}$$

where $q = \tau/D$, and we also define $r = S^2/D^2$. The first inequality can be reformulated as $0 \leq p$, whereas the second inequality can be written as

$$\left(\frac{2}{W^2} - 1\right) p \leq \rho\,(1 + \epsilon)\,. \tag{6.37}$$

We have $W \geq 1$ by definition, which gives

$$\left(\frac{2}{W^2} - 1\right) \leq 1, \tag{6.38}$$

and the dominant energy condition[55], which can be written as

$$p^2 \leq \rho^2\,(1 + \epsilon)^2\,. \tag{6.39}$$

Together, eqs. (6.38) and (6.39) imply eq. (6.37).

The function of x for which we wish to find the root is

$$\begin{aligned} f(x) &= x - h(x)W(x) \\ &= x - \left[1 + \epsilon(x) + \frac{p(x)}{\rho(x)}\right] W(x). \end{aligned} \tag{6.40}$$

From the definitions of r and S^2, we can obtain

$$W(x) = \sqrt{1 - \frac{r}{x^2}}, \tag{6.41}$$

which also gives us

$$\rho(x) = \frac{D}{W(x)}. \tag{6.42}$$

We can then use $\epsilon = h - 1 - p/\rho$ to obtain

$$\epsilon(x) = -1 + \frac{x}{W(x)}\left[1 - W(x)^2\right] + W(x)\left[1 + q\right]. \tag{6.43}$$

To calculate $p(x)$, we need to determine T. We do this by inverting the equation of state function $\epsilon = \epsilon(\rho, T, Y_\mathrm{e})$. Using eq. (6.23) and eq. (6.42) to obtain ρ and Y_e respectively, we then use the method prescribed in section 6.1.3 to invert $\epsilon(x)$ for $T(x)$, which we then use in the equation of state to obtain $p(x) = p(\rho(x), T(x), Y_\mathrm{e})$.

To solve for $f(x) = 0$, we use Dekker's method[45], which combines the bisection and secant methods. Given two initial values a_0 and b_0 (in our case $1+q$ and $2(1+q)$), such that

$$f(a_0) \cdot f(b_0) \leq 0 \text{ and } |f(a_0)| \geq |f(b_0)|, \tag{6.44}$$

the intermediate value theorem guarantees that there exists a root between a_0 and b_0 if f is continuous on $[a_0, b_0]$. Each iteration needs three points:

- the current guess b_n,
- the contrapoint a_n, such that $f(a_n) \cdot f(b_n) \leq 0$ and $|f(a_n)| \geq |f(b_n)|$,
- the previous guess b_{n-1} (b_{-1} is set to a_0).

From these three points, two values for the next guess are calculated, m by the midpoint method, and s by the secant method:

$$m = \frac{a_n + b_n}{2}, \tag{6.45}$$

$$s = b_n - \frac{b_n - b_{n-1}}{f(b_n) - f(b_{n-1})} f(b_n). \tag{6.46}$$

If s lies between a_n and m, or $f(b_n) = f(b_{n-1})$, the next guess $b_{n+1} = m$, otherwise $b_{n+1} = s$. For the contrapoint at the next iteration, we have

$$a_{n+1} = \begin{cases} a_n, & \text{if } f(a_n) \cdot f(b_{n+1}) \leq 0 \\ b_n, & \text{otherwise} \end{cases} \tag{6.47}$$

guaranteeing that $f(a_{n+1}) \cdot f(b_{n+1}) \leq 0$. Finally, if a_{n+1} is a better guess than b_{n+1}, i.e. $|f(a_{n+1})| \leq |f(b_{n+1})|$, then we swap the values. This process is iterated until either

$|f(b_n)|$ falls below a desired accuracy threshold, or n exceeds the maximum allowed number of iterations $n_{\max}$.

It is possible that the T inversion step in $f(b_{n+1})$ fails to find a solution. If this occurs, we use a search method between a_n and b_{n+1}, and between b_{n+1} and b_n to find a point c such that $f(c)$ exists, and either $f(c) \cdot f(a_n) \leq 0$ or $f(c) \cdot f(b_n) \leq 0$. Assuming $a_n < b_{n+1} < b_n$ (with a similar method for the opposite case), and looking at only the $[a_n, b_{n+1}]$ interval (with a similar method occurring in parallel for the $[b_{n+1}, b_n]$ interval), we set

$$c_k = \frac{(B_k - A_k)a_n + A_k b_n}{B_k}, \tag{6.48}$$

with initial values

$$A_0 = 1, \tag{6.49}$$

$$B_0 = 2, \tag{6.50}$$

and attempt to find $f(c_k)$. If $f(c_k)$ exists and $f(c_k) \cdot f(a_n) \leq 0$, then we set $b_{n+1} = c_k$ and continue with the Dekker root finding. If $f(c_k)$ exists but $f(c_k) \cdot f(a_n) \geq 0$, then we set

$$A_{k+1} = 2A_k + 1, \tag{6.51}$$

$$B_{k+1} = 2B_k, \tag{6.52}$$

and try c_{k+1}. If $f(c_k)$ does not exist, we set

$$A_{k+1} = 2A_k - 1, \tag{6.53}$$

$$B_{k+1} = 2B_k, \tag{6.54}$$

and try c_{k+1}. This process is repeated until either a valid c_k is found for b_{n+1}, or k exceeds a chosen $k_{\max}$, and the Con2Prim method fails. This method of searching is also used if either $f(a_0)$ or $f(b_0)$ fail, so a different a_0 or b_0 can be found, or if both fail, we search the interval $[a_0, b_0]$ for any point that succeeds, then use the above method to find the other bracket.

If the Dekker root find fails either due to $n > n_{\max}$ or $k > k_{\max}$, then we move onto our final backup Con2Prim method, a 1-D bracketed method in W, with a fixed T.

6.2.3 1-D bracketed in W

This method is used only when the Dekker method above fails to converge, which should only be the case when no physical solution for the Con2Prim problem exists. This typically occurs when the effective temperature of the fluid falls below the lowest temperature available in the table $T_{\min}$. In this case, we assume $T = T_{\min}$ and solve

for W. Due to this mismatch in temperatures, we know that the primitives we obtain will not correspond exactly to the conservatives with which we started, and so we must decide where we want this error to lie.

We experimented with trying to minimise the total error across the conservatives, but found this introduced spurious kinetic energy into the simulation. Instead, we do our best to conserve the 3−momentum density $p^i = \rho W v^i = D v^i$. As the value of W will not impact on the direction of p^i, we can instead choose to attempt to minimise the error in $p^2 = p_i p^i = D^2 v^2$, and because D does not depend on the equation of state we can safely ignore it. Therefore, we must solve to minimise error in v^2.

An expression in only the conserved variables which reduces to v^2 for $p \ll z$ is

$$\begin{aligned} \frac{S^2}{(\tau + D)^2} &= \frac{z^2 v^2}{(z - p)^2} \\ &= v^2 \frac{1}{\left(1 - \frac{p}{z}\right)^2}, \end{aligned} \tag{6.55}$$

We can also calculate this quantity from the primitive variables using

$$v^2 \frac{1}{\left(1 - \frac{p}{z}\right)^2} = \frac{W^2 - 1}{W^2 \left(1 - \frac{p(W)}{z(W)}\right)^2}, \tag{6.56}$$

where $p(W)$ and $z(W)$ are calculated from the equation of state. Combining eqs. (6.55) and (6.56) we obtain

$$f(W) = \log\left(\frac{W^2 - 1}{W^2 \left(1 - \frac{p(W)}{z(W)}\right)^2}\right) - \log\left(\frac{S^2}{(\tau + D)^2}\right), \tag{6.57}$$

where we have taken the log of each equation in order to improve behaviour far from $f(W) = 0$. We also obtain better behaviour where $v^2 \ll 1 \Rightarrow W \approx 1$ by using $w = \log(W - 1)$, which yields

$$f(w) = \log\left(\frac{e^{2w} + 2e^w}{(e^{2w} + 2e^w + 1)\left(1 - \frac{p(w)}{z(w)}\right)^2}\right) - \log\left(\frac{S^2}{(\tau + D)^2}\right). \tag{6.58}$$

W is physically constrained by the lower bound $1 \leq W$, which corresponds to $-\infty \leq w$. However the limits of floating point number representation limit meaningful values of $W > 1$ to

$$W = 1 + \epsilon, \tag{6.59}$$

where ϵ is the machine epsilon (in the case of double precision floating point numbers, $\epsilon = 2^{-52} \approx 2.22 \times 10^{-16}$), thus giving a lower bound for w of

$$w_{\min} = \log \epsilon. \tag{6.60}$$

We can obtain an upper bound for w by considering the fact that if ρ drops below an atmosphere density ρ_{atmo}, the simulation will consider that point to be part of the atmosphere and reset all of the fluid variables to pre-defined defaults. This means that for the point in question w must be such that

$$\frac{D}{(1+e^w)} \geq \rho_{\text{atmo}}, \tag{6.61}$$

to avoid resetting, which we can rearrange to

$$w_{\max} = \log\left(\frac{D}{\rho_{\text{atmo}}} - 1\right). \tag{6.62}$$

Using the two bounds for w and the function $f(w)$, we again use Dekker's method[45] to solve for w such that $f(w) = 0$. If we find there is no solution for $f(w) = 0$ on the interval $w_{\min} \leq w \leq w_{\max}$, we use a modified objective function $g(w)$, defined by

$$g(w) = \left(\frac{e^{2w} + 2e^w}{(e^{2w} + 2e^w + 1)\left(1 - \frac{p(w)}{z(w)}\right)^2}\right) - \left(\frac{S^2}{(\tau + D)^2}\right), \tag{6.63}$$

to determine whether there is a solution on the interval $-\infty \leq w \leq w_{\min}$ by observing that

$$\lim_{w\to-\infty} g(w) = \left(\frac{0}{0+1}\frac{1}{\left(1 - \frac{p(w)}{z(w)}\right)^2}\right) - \frac{S^2}{(\tau + D)^2} \tag{6.64}$$

$$= -\frac{S^2}{(\tau + D)^2} \tag{6.65}$$

If we find that $g(w_{\min}) \times g(-\infty) \leq 0$, then we accept $W = 1$ as the solution. We can also test for a solution on the interval $w_{\max} \leq w \leq \infty$ by observing that

$$\lim_{w\to\infty} g(\infty) = \lim_{w\to\infty}\left(\frac{e^{2w} + 2e^w}{(e^{2w} + 2e^w + 1)} \times \left(1 - \frac{1}{W^2}\frac{1}{\frac{\rho(1+\epsilon)}{p} + 1}\right)^{-2}\right) - \left(\frac{S^2}{(\tau + D)^2}\right), \tag{6.66}$$

where we can immediately see that

$$\lim_{w\to\infty}\left(\frac{e^{2w} + 2e^w}{(e^{2w} + 2e^w + 1)}\right) = 1, \tag{6.67}$$

and we can use the dominant energy condition (see eq. (6.39)) to show that

$$\frac{1}{\frac{\rho(1+\epsilon)}{p}+1} \leq \frac{1}{2}. \tag{6.68}$$

This leaves us with

$$\lim_{w\to\infty} g(w) = \left(1 \times \left(1 - \lim_{w\to\infty} \frac{1}{2\left(1+e^{w}\right)^{2}}\right)^{-2}\right) - \left(\frac{S^2}{(\tau+D)^2}\right) \tag{6.69}$$

$$= 1 - \left(\frac{S^2}{(\tau+D)^2}\right). \tag{6.70}$$

and so if $g(w_{\max}) \times g(\infty) \leq 0$ then the solution is sufficiently large that the point will be set to atmosphere, and this is accepted as a solution. If both of these tests fail, then Con2Prim fails, and the simulation is aborted.

Once we have a solution for W, we can use this and $T_{\min}$ to calculate the values of the primitives in the same way as with the 2-D Newton-Raphson method above (see eqs. (6.22) to (6.26)), and as we know the primitive variables obtained will be inconsistent with the conservative variables with which we started, we calculate the values of the conserved variables using eqs. (6.15) to (6.18) and use these to replace the old values.

Part II

Results

Chapter 7

Simulation Overview

In part I we covered the background and tools needed to perform simulations of binary neutron star mergers. Now, in this second part, we discuss results from such simulations. In later chapters we investigate the meaning of equilibrium in neutron star mergers, and how out-of-equilibrium effects and reactions might impact on the dynamics of a merger. As we focused our investigation on these effects, particularly in the post-merger phase, we used simple equal mass initial data configurations, with total mass small enough to produce a long lived neutron star remnant, and we ignored magnetic fields. First, however, we give a general overview of what a neutron star merger looks like, and the information we can extract from simulations using realistic equations of state.

In this chapter we discuss the results of two simulations run using the modifications to `EOS_Omni` and `GRHydro` detailed in chapter 6, and compare them to a simulation run using a $\Gamma = 2$ Γ-law equation of state.

The first simulation we discuss began with initial data created using LORENE[63] describing two $1.2\,M_\odot$ neutron stars with an initial separation of 100 km, with a $\Gamma = 2, K = 123.6$ polytropic equation of state. The initial rotational profile of the stars was set such that they appear static in a co-rotating frame. The simulation was evolved using a $\Gamma = 2$ Γ-law equation of state.

A second simulation was run using the modifications discussed in chapter 6 using the *Steiner-Fischer-Hempel-x* (SFHx)[129] equation of state from the CompOSE[42] database, and covered the merger of two $1.2\,M_\odot$ neutron stars with an initial separation of 35 km. Initial data was calculated using LORENE with the matter in β-equilibrium at a temperature of $T = 0.5\,\mathrm{MeV}$ everywhere and, as with the Γ-law simulation, used a co-rotating rotational profile. This simulation was run with the intention of testing how robust the modifications are, and to provide a comparison with our own Γ-law simulation, but was only run for a total of around ~ 15 ms (with the merger occurring slightly over halfway through), not long enough to provide an interesting gravitational wave signal. As such, we will focus our discussion on the fluid variables.

The third simulation used the *Schneider-Roberts-Ott* (SRO) implementation of the *Akmal-Pandharipande-Ravenhall* (APR)[122] equation of state (again from CompOSE), with initial data calculated using LORENE that described two $1.4\,M_\odot$ neutron stars in β-equilibrium at a temperature of $T = 0.02\,\mathrm{MeV}$, with 40 km initial separation. Co-rotating initial data was used in the first two simulations, however, to better represent close neutron star binaries, we use irrotational initial data for this simulation.

The discussion of these three simulations takes place in three sections: we begin in section 7.1 with a walk-through of the basic features we expect to see in a neutron star merger using the Γ-law simulation mentioned above as an example, then in section 7.2 we give a comparison between the matter and temperature distribution in the Γ-law and SFHx simulations, and lastly in section 7.3 we discuss effects stemming from the fluid composition in the APR simulation. We then close the chapter with a presentation of some known issues with current simulations[1] in section 7.4.

7.1 The Anatomy of a Merger

A neutron star merger consists of three main phases, each displaying unique features depending on different physics within the simulation, namely: the *Inspiral*, the *Merger*, and the *Post-merger* phase, typically called the *Ringdown* when the remnant is a black hole.

7.1.1 Inspiral

The inspiral phase of a neutron star merger is characterised by the loss of orbital energy due to gravitational wave emission. As energy is radiated away, the orbital radius of the two neutron stars decreases, and the orbital and gravitational wave frequencies increase, as can be seen in figs. 7.1 and 7.2.

As the two stars approach each other, they become distorted by tidal forces into elongated shapes, and begin to shed matter as they lose cohesion. We can see these effects in fig. 7.3. As the orbital separation r gets smaller, the lower density regions in the stars become more ellipsoidal in appearance, an effect which appears in higher and higher density contours as the inspiral progresses. We can also see the formation of spiral arms of matter being ejected behind the stars, which increase in density the closer the stars become.

[1] Published as part of Hammond et al.[68].

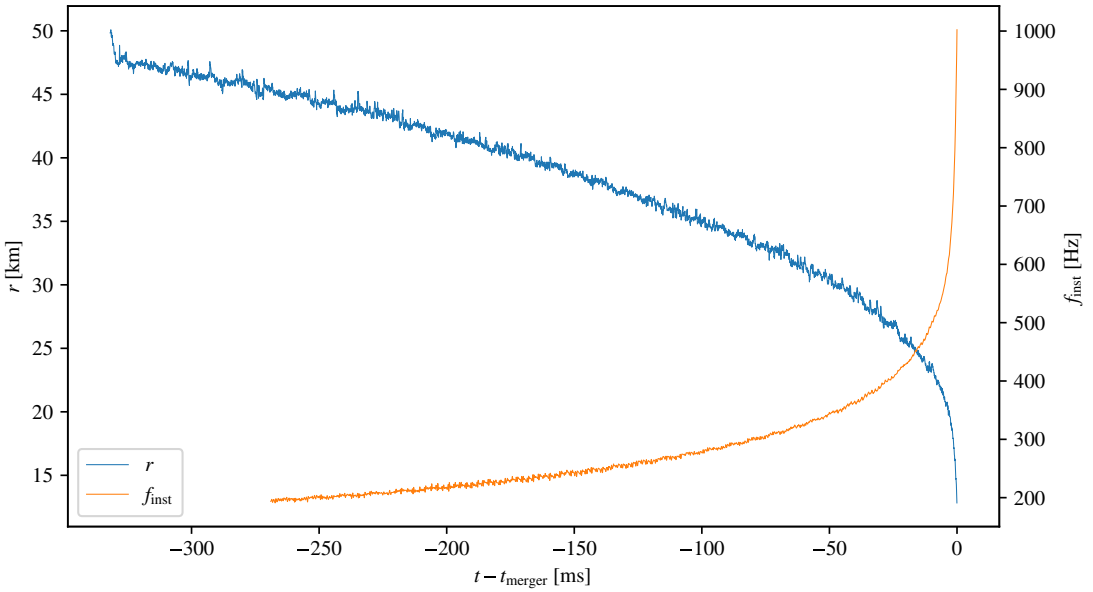

FIGURE 7.1: Plot of the orbital radius r and instantaneous gravitational wave frequency f_{inst} for the merger of two 1.2 $M_\odot$ neutron stars, with initial orbital radius of $r = 50\,\text{km}$. The radius curve is not smooth as the centre of the star is measured as the coordinate point with highest rest mass density which, as it jumps from one discrete point to the next, along with oscillations of the stellar matter, causes noise in the calculation of r.

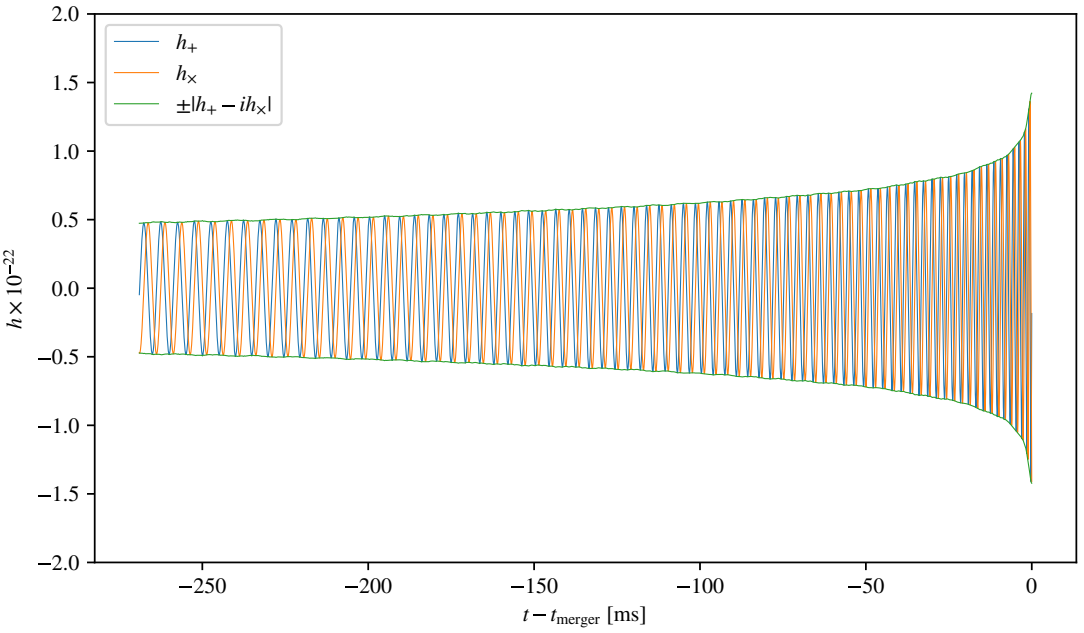

FIGURE 7.2: Plot of the two gravitational wave strain polarisations h_+ and $h_\times$, and the strain envelope $\pm|h_+ - ih_\times|$, as would be measured at a distance of 100 Mpc along the orbital axis from the merger of two 1.2 $M_\odot$ neutron stars, with initial orbital radius of $r = 50\,\text{km}$.

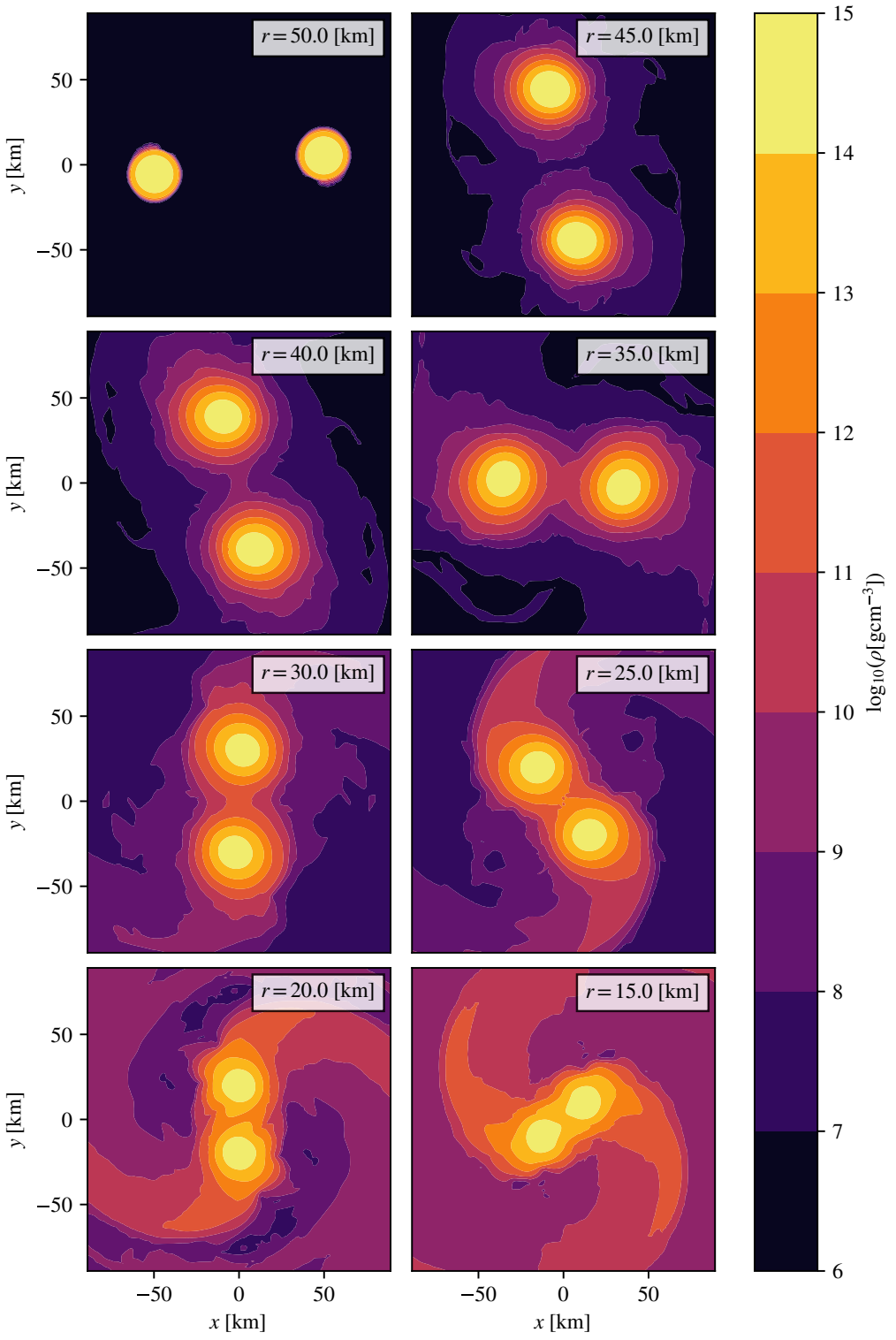

FIGURE 7.3: Contour plots of the logarithm of rest mass density $\log_{10}(\rho)$ for the merger of two $1.2\,M_\odot$ neutron stars, with initial orbital radius of $r = 50\,\text{km}$ plotted at $5\,\text{km}$ intervals in r.

7.1.2 Merger

This phase of the merger is dominated by the coming together of the two stars, and the formation of the post-merger remnant, either a neutron star (as is the case with this simulation) or a black hole. In fig. 7.4 we can see how the matter transitions from the two merging stars into the dumbbell-like configuration of the post merger remnant surrounded by a high density envelope.

This collision also has the effect of heating the matter on the interface of the two merging stars to significant temperatures, with the peak temperature reached being very dependent on the equation of state used, in this case reaching $\gtrsim 100\,\mathrm{MeV}$. In fig. 7.5 we can see the creation of this high temperature region in the low density interface between the two stars. As the denser matter moves inwards the high temperature fluid is squeezed towards the outside of the star, where it expands and cools. In this figure we can also see the heating that takes place on the leading edge of the spiral arms.

7.1.3 Post-merger

The post-merger remnant is created in a highly perturbed state, oscillating significantly, leading to the emission of gravitational radiation. As these perturbations are damped (by the emission of gravitational waves, but possibly by other effects e.g. viscosity), the amplitude of the radiation falls away over time, leading to the name *ringdown* (this is most obvious when a black hole is produced, as the radiation falls away very quickly, whereas a stable neutron star can continue to radiate gravitational waves for some time, as we see in fig. 7.6). Figure 7.6 show the gravitational wave strain plotted from the time it reaches its maximum, t_{merger}, onwards. The chaotic period of the merger itself can be seen at the beginning, followed by quasi-periodic oscillations.

The variation in the amplitude envelope of the post-merger gravitational wave signal is caused by the presence of multiple modes in the oscillation of the fluid, as can be seen in fig. 7.7, which interfere with each other. In this spectrogram we can see the growing frequency and amplitude during the inspiral phase, the sharp transition from the inspiral to the post-merger phase, and the presence of several distinct peaks at different frequencies in the Power Spectral Density (PSD) during the post-merger. As expected (see, for example, Takami et al.[132], Dietrich et al.[48]) we see multiple strong peaks shortly after merger, which are related to the parameters of the system[132].

In fig. 7.8 we can see that from the point of view of rest mass density and temperature, the remnant in our simulation reaches a steady state $100 - 150\,\mathrm{ms}$ after the merger occurs.

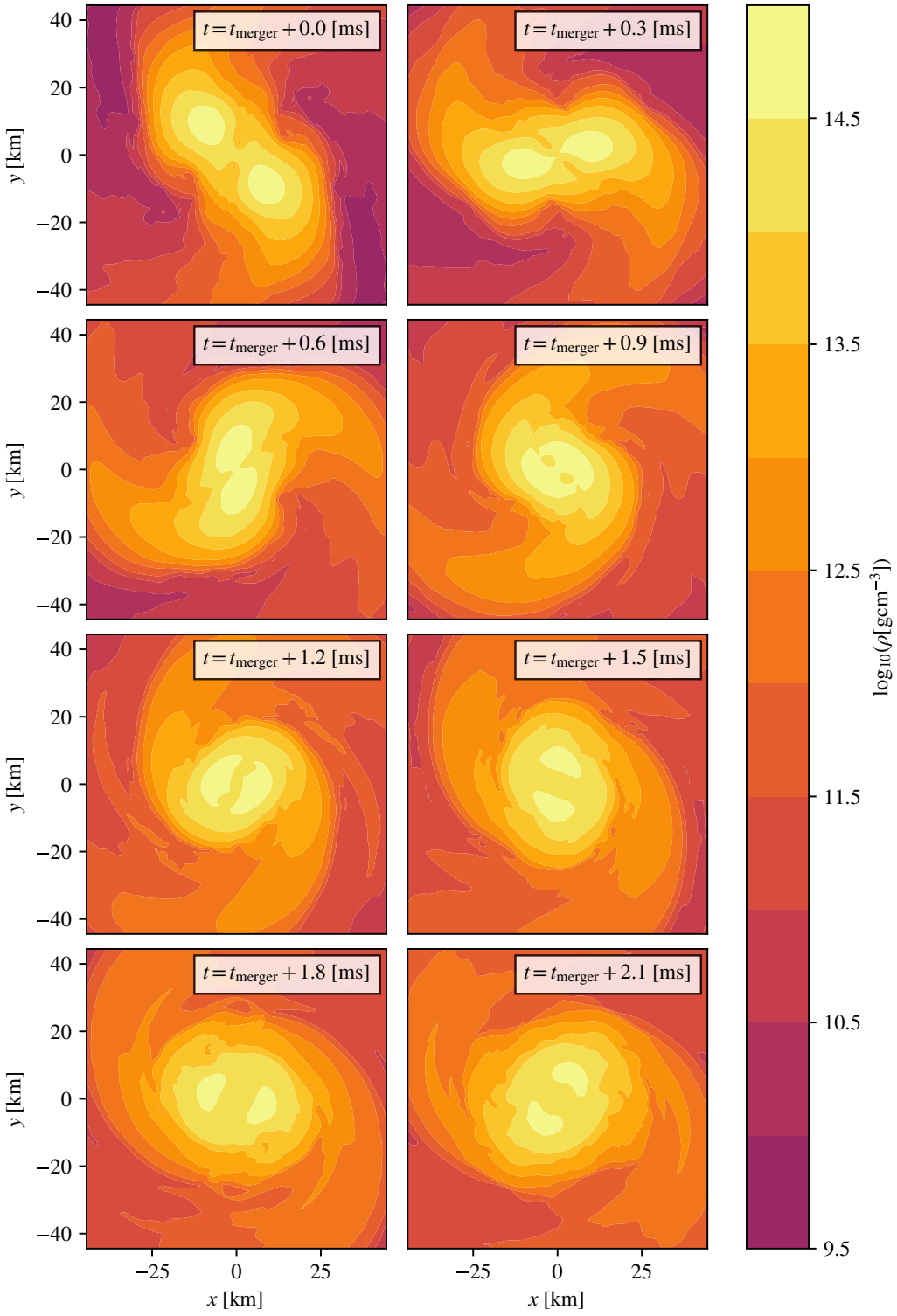

FIGURE 7.4: Contour plots of the logarithm of rest mass density $\log_{10}(\rho)$ for the merger of two $1.2\,M_{\odot}$ neutron stars, plotted in 0.3 ms intervals from t_{merger}.

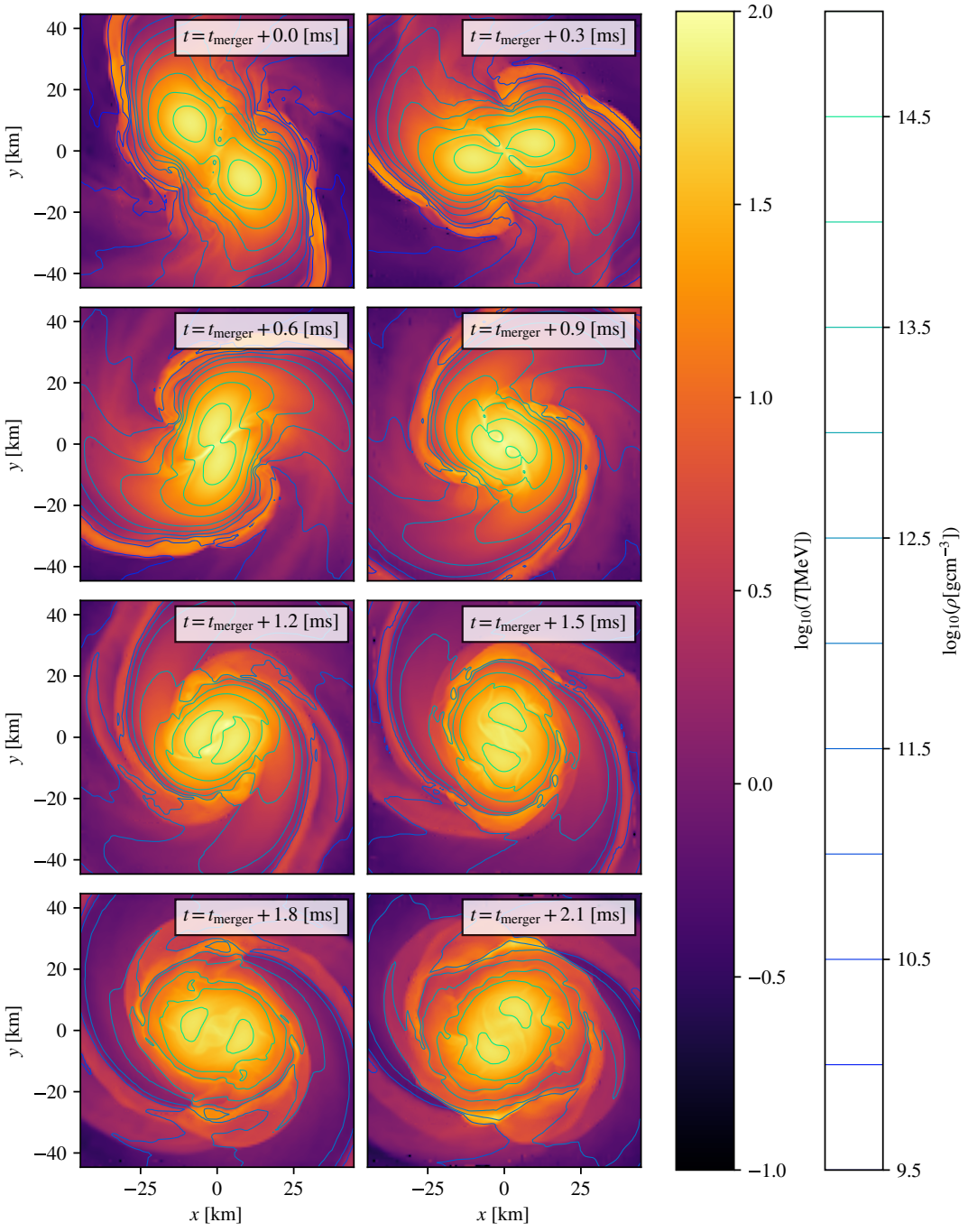

FIGURE 7.5: Plots of the logarithm of temperature $\log_{10}(T)$ with contours in the logarithm of rest mass density $\log_{10}(\rho)$ for the merger of two $1.2\,M_{\odot}$ neutron stars, plotted in 0.3 ms intervals from t_{merger}.

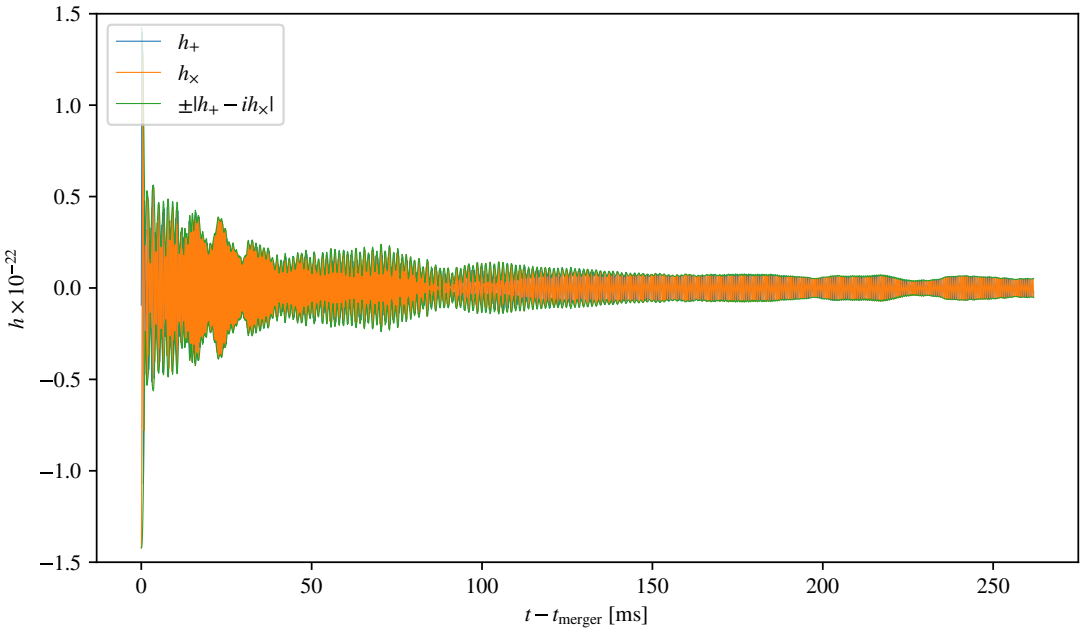

FIGURE 7.6: Plot of the two gravitational wave strain polarisations h_+ and $h_\times$, and the strain envelope $\pm|h_+ - ih_\times|$, as would be measured at a distance of 100 Mpc along the orbital axis from the merger of two 1.2 $M_\odot$ neutron stars, measured using retarded time t from the merger time t_{merger}.

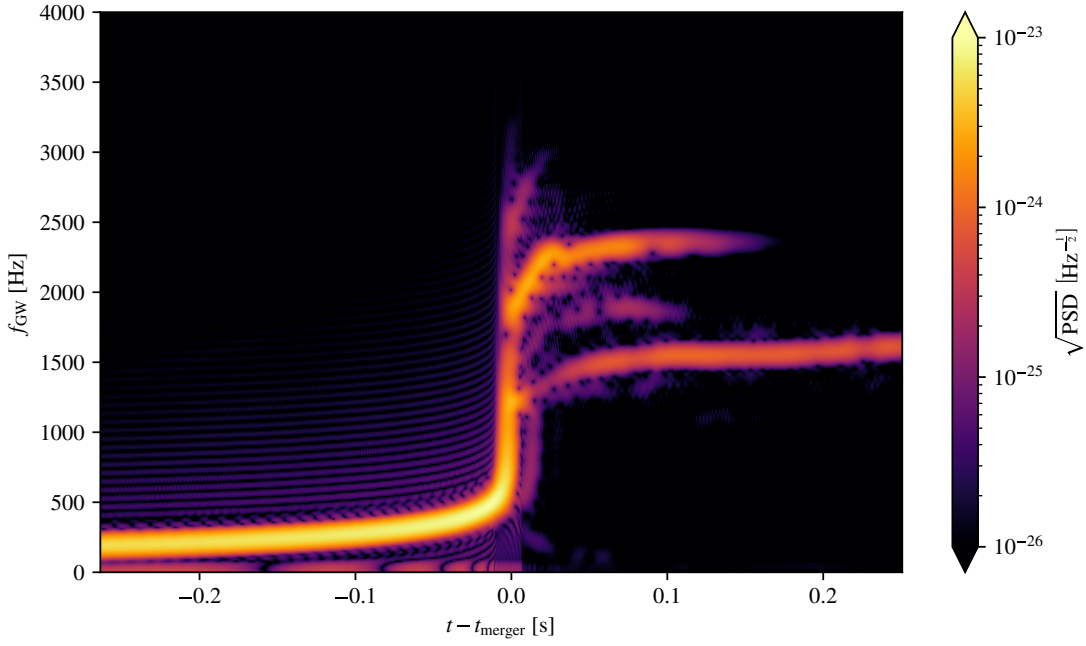

FIGURE 7.7: Spectrogram of the logarithm of square root Power Spectral Density $\log_{10}(\sqrt{\mathrm{PSD}})$ as would be measured at a distance of 100 Mpc along the orbital axis from the merger of two 1.2 $M_\odot$ neutron stars, measured using retarded time t offset by the merger time t_{merger}. This is a copy of the lower panel of fig. 5.6.

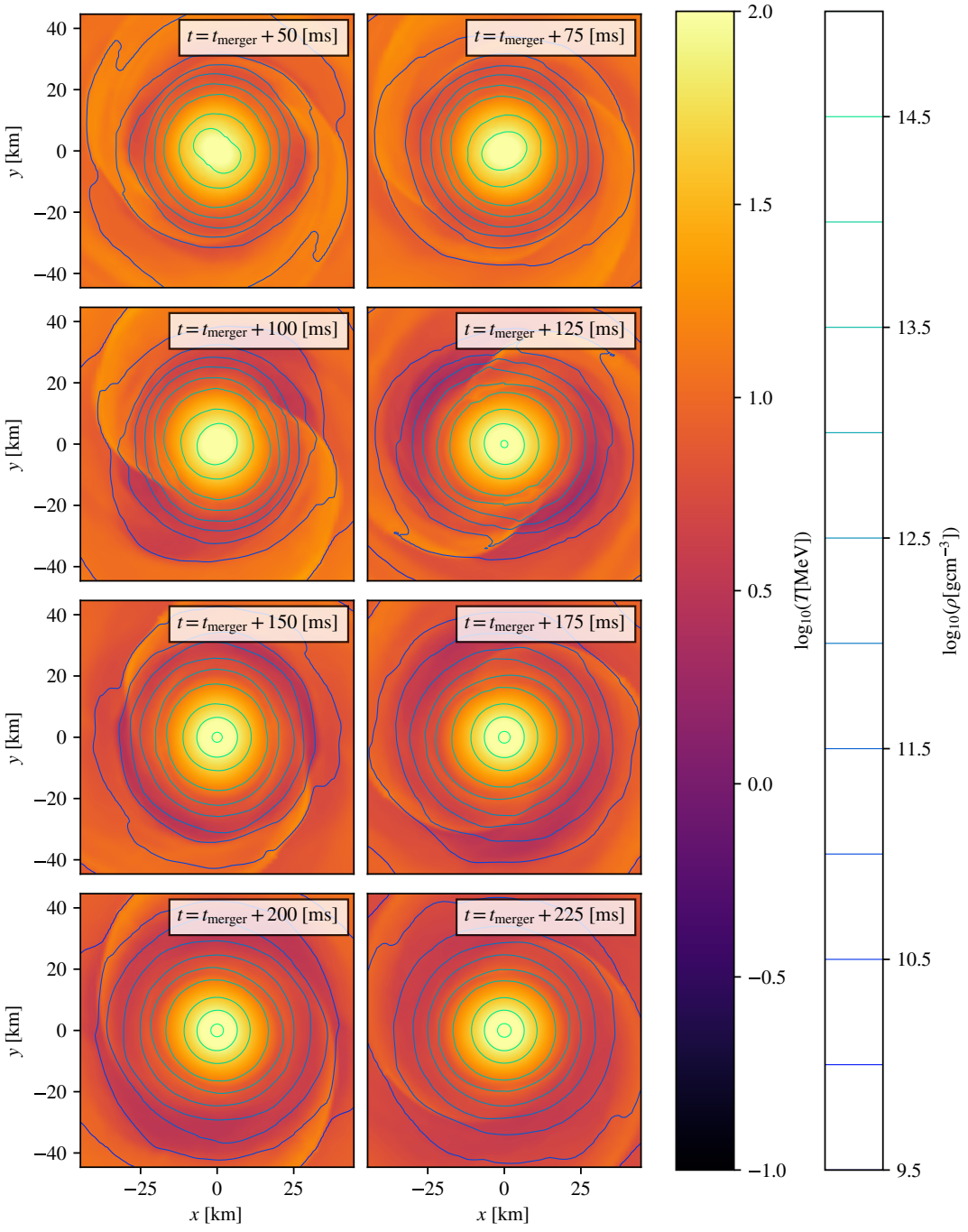

FIGURE 7.8: Plots of the logarithm of temperature $\log_{10}(T)$ with contours in the logarithm of rest mass density $\log_{10}(\rho)$ for the merger of two $1.2\,M_{\odot}$ neutron stars, plotted in 25 ms intervals from $t = t_{\mathrm{merger}} + 50\,\mathrm{ms}$.

7.2 Density and Temperature

7.2.1 Matter distribution

Figure 7.9 plots the rest mass density of the fluid ρ as a multiple of nuclear saturation density[2] $\rho_{\text{nuc}} = m_b \times 1.6 \times 10^{44}\text{m}^{-3} \approx 2.4 \times 10^{14}$ gcm^{-3}, where m_b is the baryon mass, and fig. 7.10 plots $\log_{10}\rho$. From both of these plots, it is clear that the SFHx equation of state produces a more compact neutron star for a given mass, with higher central density. From fig. 7.9 we can also see that the SFHx neutron stars reach a smaller separation before merging.

Comparing these plots with simulations performed by other groups using realistic equations of state (for example, Shibata et al.[126] using various cold SLy equations of state with a thermal Γ-law temperature component, Most et al.[95] using the finite temperature TNTYST equation of state) we are confident that the modifications we have implemented (see chapter 6) are working correctly.

7.2.2 Temperature

In fig. 7.11 we see a comparison between the fluid temperature T of the SFHx and Γ-law neutron star simulations. We can see that the centres of the SFHx neutron stars stay cold compared to the rest of the matter, and only show significant heating as the fluid begins to mix in the post-merger phase, whereas the Γ-law neutron stars have hotter cores, due to the polytropic equation of state used in the creation of initial data for ϵ which effectively gives a density dependent temperature that increases with density (which can be seen by combining eqs. (4.16) and (4.18)).

Both simulations show shock heating along the boundary between the two stars as they collide, with the SFHx merger reaching a peak of $\sim 50\,\text{MeV}$, and the Γ-law merger peaking around $\sim 100\,\text{MeV}$. The higher peak temperatures in the Γ-law equation of state simulation are likely due to the softer equation of state allowing a greater degree of compression heating to occur as the stars merge.

In the SFHx simulation we can see that, as the remnant begins to form, the shock heated matter is squeezed to the outside, forming two hotspots in the outer regions of the remnant neutron star, a feature predicted by Kastaun et al.[77], while the denser core remains at a low temperature. This feature is likely the result of a lack of mixing between low and high density regions.

Figure 7.11 also shows the matter being ejected from the merger is at a high temperature, around 20 MeV, which is important for *r-process nucleosynthesis*, both predicted[51;82] and observed[104] to take place in binary neutron star mergers.

[2] The notations ρ_{nuc} and ρ_{sat} are used interchangeably throughout.

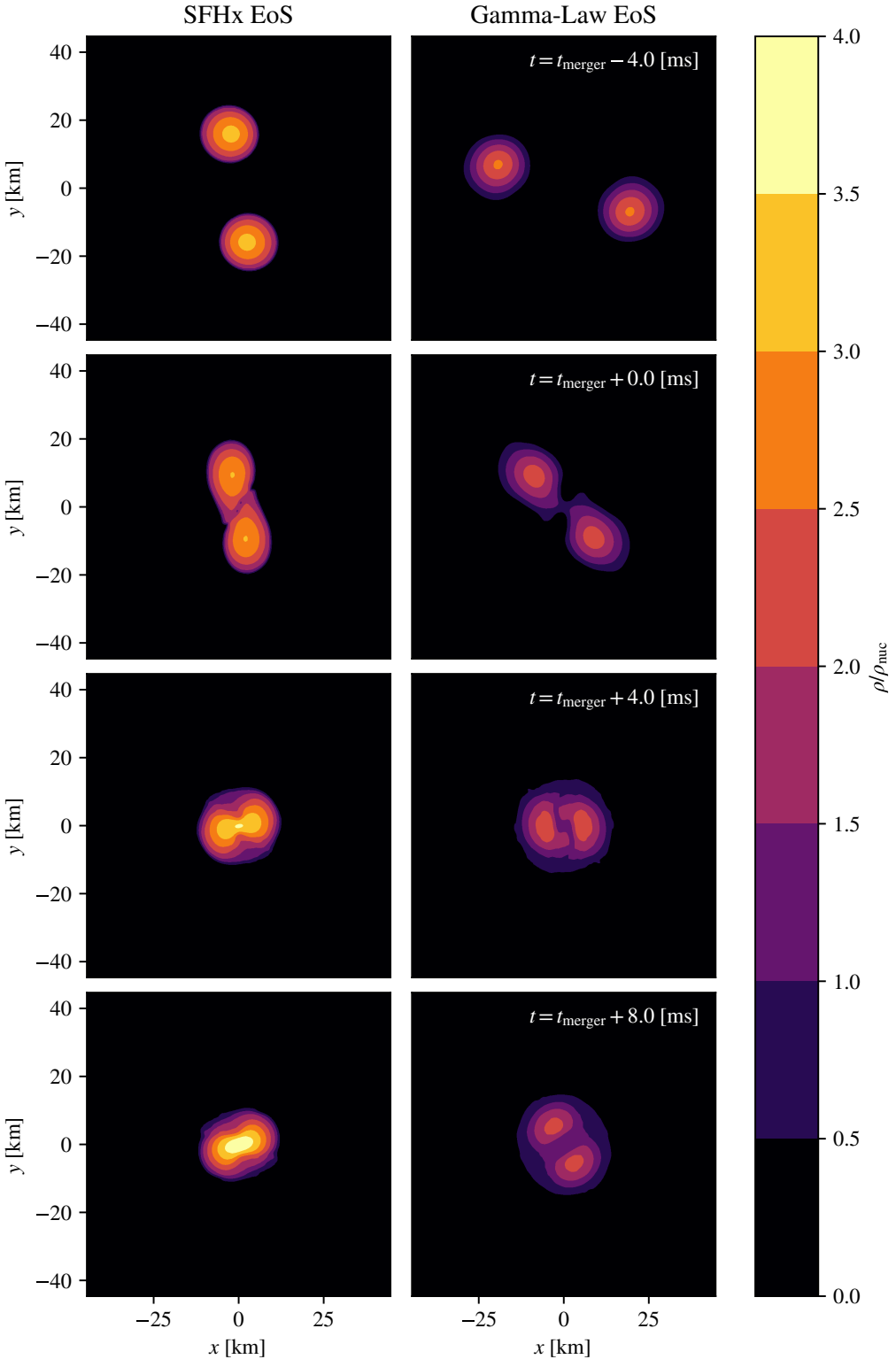

FIGURE 7.9: Plots of the rest mass density ρ as a multiple of nuclear saturation density $\rho_{\text{nuc}} \approx 2.04 \times 10^{14}\,\text{gcm}^{-3}$ for two $1.2\,M_\odot + 1.2\,M_\odot$ neutron star merger simulations, one using the SFHx[129] equation of state, and the other using a $\Gamma = 2$ Gamma-Law equation of state. Frames are at 4 ms intervals, with the second frame coinciding with the merger time for each simulation.

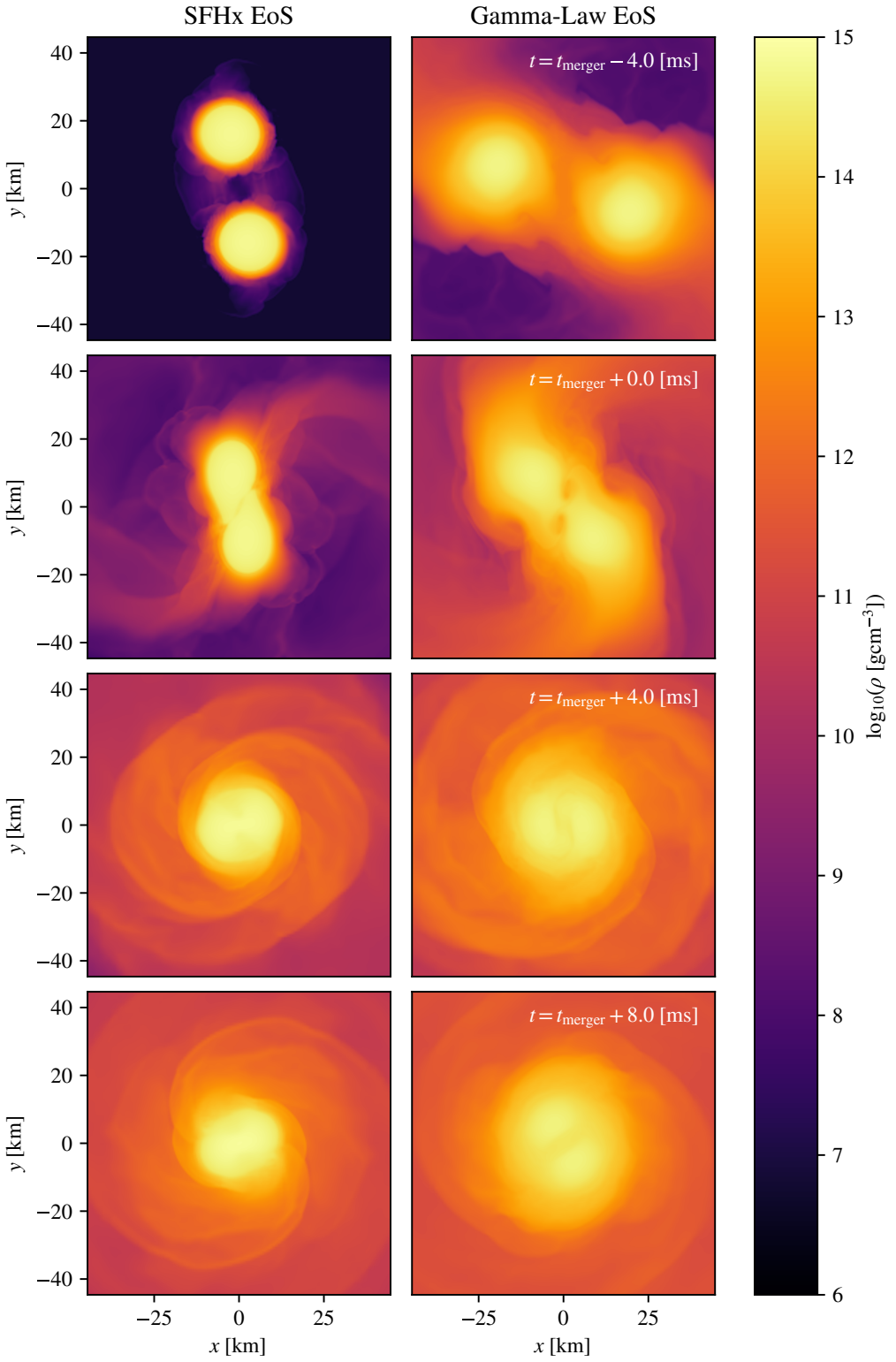

FIGURE 7.10: Plots of the logarithm of the rest mass density $\log_{10}\rho$ for two $1.2\,M_\odot + 1.2\,M_\odot$ neutron star merger simulations, one using the SFHx[129] equation of state, and the other using a $\Gamma = 2$ Gamma-Law equation of state. Frames are at 4 ms intervals, with the second frame coinciding with the merger time for each simulation.

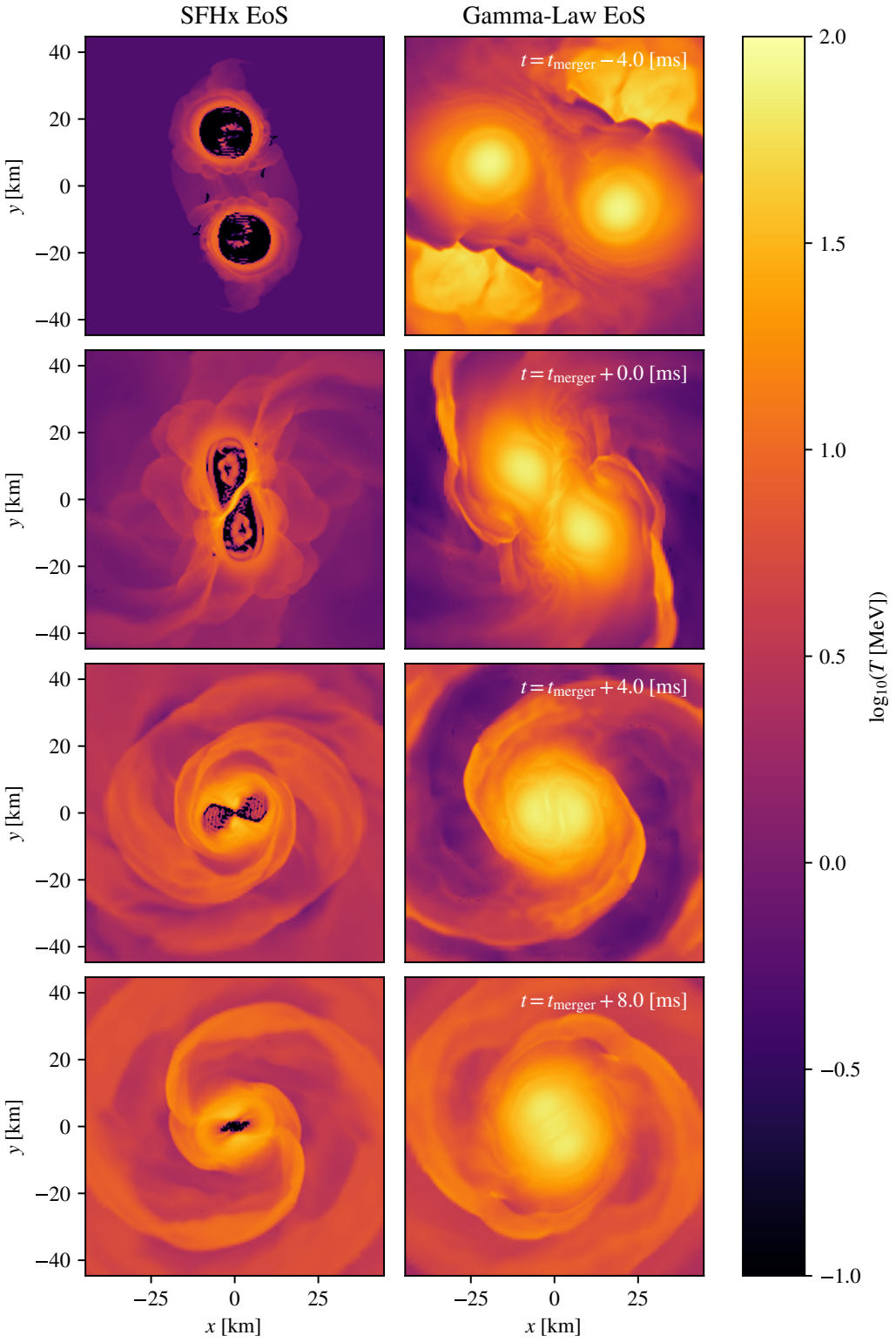

FIGURE 7.11: Plots of the logarithm of the fluid temperature $\log_{10} T$ for two $1.2\,M_\odot + 1.2\,M_\odot$ neutron star merger simulations, one using the SFHx[129] equation of state, and the other using a $\Gamma = 2$ Gamma-Law equation of state. Frames are at 4 ms intervals, with the second frame coinciding with the merger time for each simulation.

7.3 Composition

In this section, we will discuss composition dependent effects we can explore using the data from our APR simulation. The fluid is assumed to consist of a mixture of neutrons, protons, and electrons, each with their own number density, $n_{\rm n}$, $n_{\rm p}$, and $n_{\rm e}$ respectively. The baryon number density n_b will simply be the sum of the neutron and proton number densities. Imposing local charge neutrality we require that $n_{\rm p} = n_{\rm e}$, so we can define the electron fraction $Y_{\rm e}$ as

$$Y_{\rm e} = \frac{n_{\rm e}}{n_b}, \tag{7.1}$$

which completely determines the composition of the fluid. $Y_{\rm e}$ should then be evolved through

$$u^a \nabla_a Y_{\rm e} = \frac{\Gamma_{\rm e}}{n_b}, \tag{7.2}$$

where $\Gamma_{\rm e}$ (not to be confused with the adiabatic index) is the net production rate of electrons. However, there are issues with the naïve numerical implementation of this. We will discuss these issues later in chapter 10. For now, we assume that $\Gamma_{\rm e} = 0$.

Figure 7.12 shows the electron fraction (which has been plotted with a clipped range of $0.0 \leq Y_{\rm e} \leq 0.15$). $Y_{\rm e}$ reaches ~ 0.5 (approximately symmetric matter) in the atmosphere of the simulation, but, as can be seen in the plots, the neutron rich core matter ejected before and during merger causes $Y_{\rm e}$ in the entire domain close to the merger to drop to below $Y_{\rm e} = 0.1$.

7.3.1 Fermi energies

Using the composition information, we can consider the degeneracy of the proton, neutron, and electron components of the fluid. This information will be useful as we try to extend the equation of state treatment towards $T = 0$, as simpler expressions which ignore either quantum or thermal effects can be used if the components are either non-degenerate or completely degenerate respectively. To measure this degeneracy, we look at T/E_{F_x}, where E_{F_x} is the Fermi energy of species x.

To calculate the Fermi energy of the species, we use the non-relativistic limit for baryons, and the relativistic limit for electrons. Under the assumption of an ideal gas, the average thermal kinetic energy $E_{K_{\rm ave}}$ is given by

$$E_{K_{\rm ave}} = \frac{3}{2} T, \tag{7.3}$$

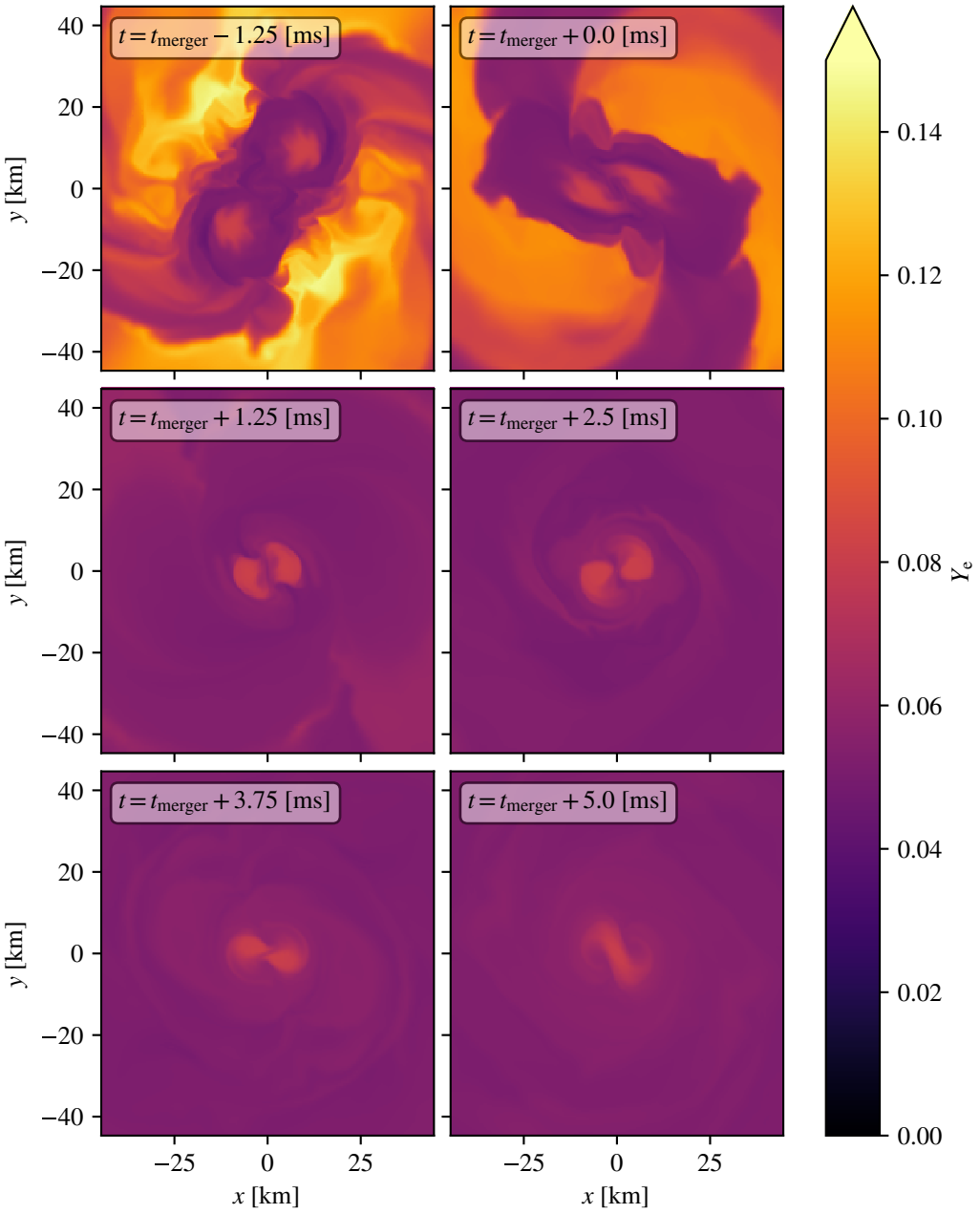

FIGURE 7.12: Plots of the electron fraction Y_e for a simulation of the merger of two $1.4\,M_\odot$ neutron stars using the APR[122] equation of state, clipped to the range $0.0 \leq Y_e \leq 0.15$. Frames are at 1.25 ms intervals, with the second frame coinciding with the merger time of the binary.

where for baryons we have $E_{K_{\text{ave}}} < m_b$ across the entire domain, and for electrons we have $E_{K_{\text{ave}}} > m_e$. Thus for baryons, the Fermi energy is given by

$$E_{F_x} = \frac{\hbar^2 \left(3\pi^2 n_x\right)^{2/3}}{2m_x^*}, \tag{7.4}$$

where x is either p for protons or n for neutrons, n_x is the number density of species x, and m_x^* is the effective mass for a particle of species x, and for electrons we use

$$E_{F_\text{e}} = \hbar \left(3\pi^2 n_\text{e}\right)^{1/3}. \tag{7.5}$$

Figures 7.13, 7.14, and 7.15 show T/E_{F_x} for $x = \text{p}, \text{n}, \text{e}$ respectively. From these plots we can see that while the two baryon species are completely non-degenerate (i.e. $T \gg E_{F_x}$) in the low density atmosphere regions, all species show mixed-degeneracy in the interesting regions, thus neither the degenerate nor non-degenerate simplifying cases can be used.

7.3.2 Entropy

Figure 7.16 shows another useful measure of degeneracy: the entropy per baryon $\mathcal{S}$, given through the equation of state by

$$\mathcal{S} = -\frac{1}{n_b} \left.\frac{\partial f}{\partial T}\right|_{n_b, Y_\text{e}}, \tag{7.6}$$

where f is the Helmholtz free energy density. The value of $\mathcal{S}$ is physically constrained to be $\mathcal{S} \geq 0$, which our simulation does not violate, with smaller values $\mathcal{S} \ll 1$ indicating degenerate matter, and large values $\mathcal{S} \gg 1$ indicating a non-degenerate state. As we can see, the indication of degeneracy in fig. 7.16 agrees well with figs. 7.13 to 7.15, again showing a completely non-degenerate state in the atmosphere region, and mixed degeneracy in the physically relevant regions.

Finally, in fig. 7.17 we plot the regions of the μ_n-T plane reached by the fluid. From this plot, we can deduce the qualitative behaviour of the matter in the μ_n-T phase space. We can see that as time progresses, the matter becomes less concentrated in phase space as mixing between the different regions in the remnant occurs, and the remnant reaches increasing μ_n as the maximum density increases.

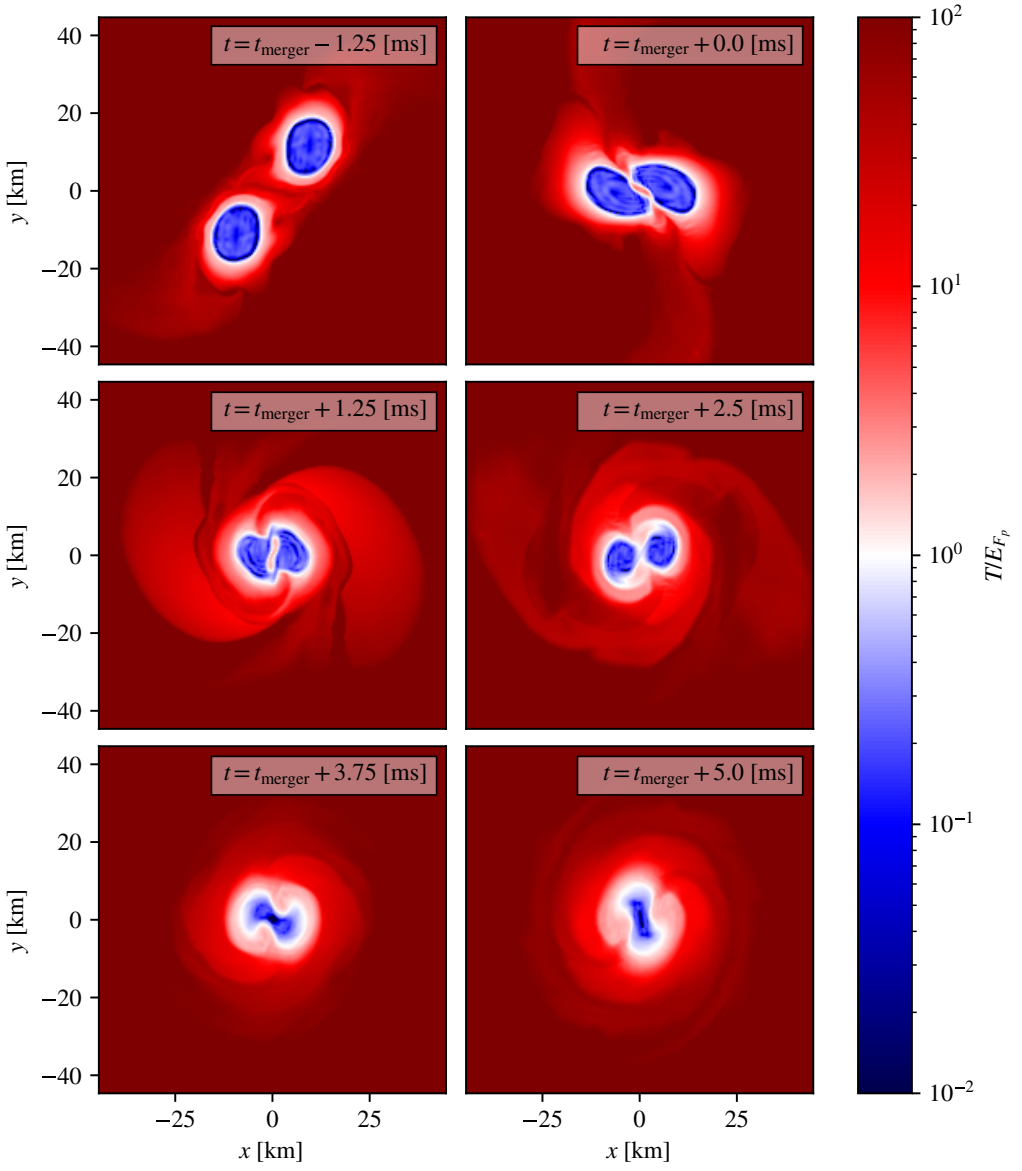

FIGURE 7.13: Plots of the ratio of fluid temperature T to proton Fermi energy E_{F_p}, T/E_{F_p} for a simulation of the merger of two $1.4\,M_\odot$ neutron stars using the APR[122] equation of state. Frames are at 1.25 ms intervals, with the second frame coinciding with the merger time of the binary.

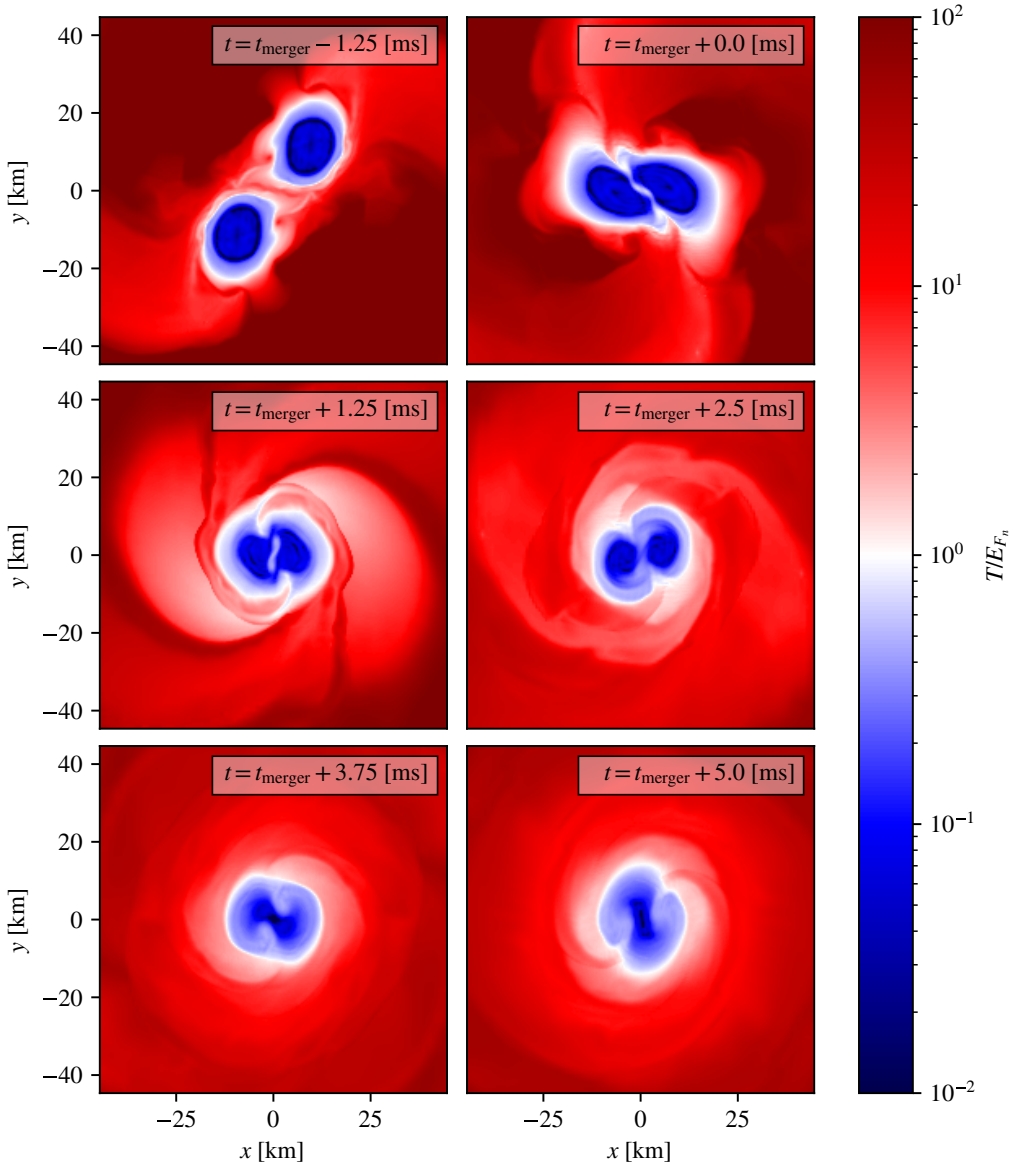

FIGURE 7.14: Plots of the ratio of fluid temperature T to neutron Fermi energy E_{F_n}, T/E_{F_n} for a simulation of the merger of two $1.4\,M_\odot$ neutron stars using the APR[122] equation of state. Frames are at 1.25 ms intervals, with the second frame coinciding with the merger time of the binary.

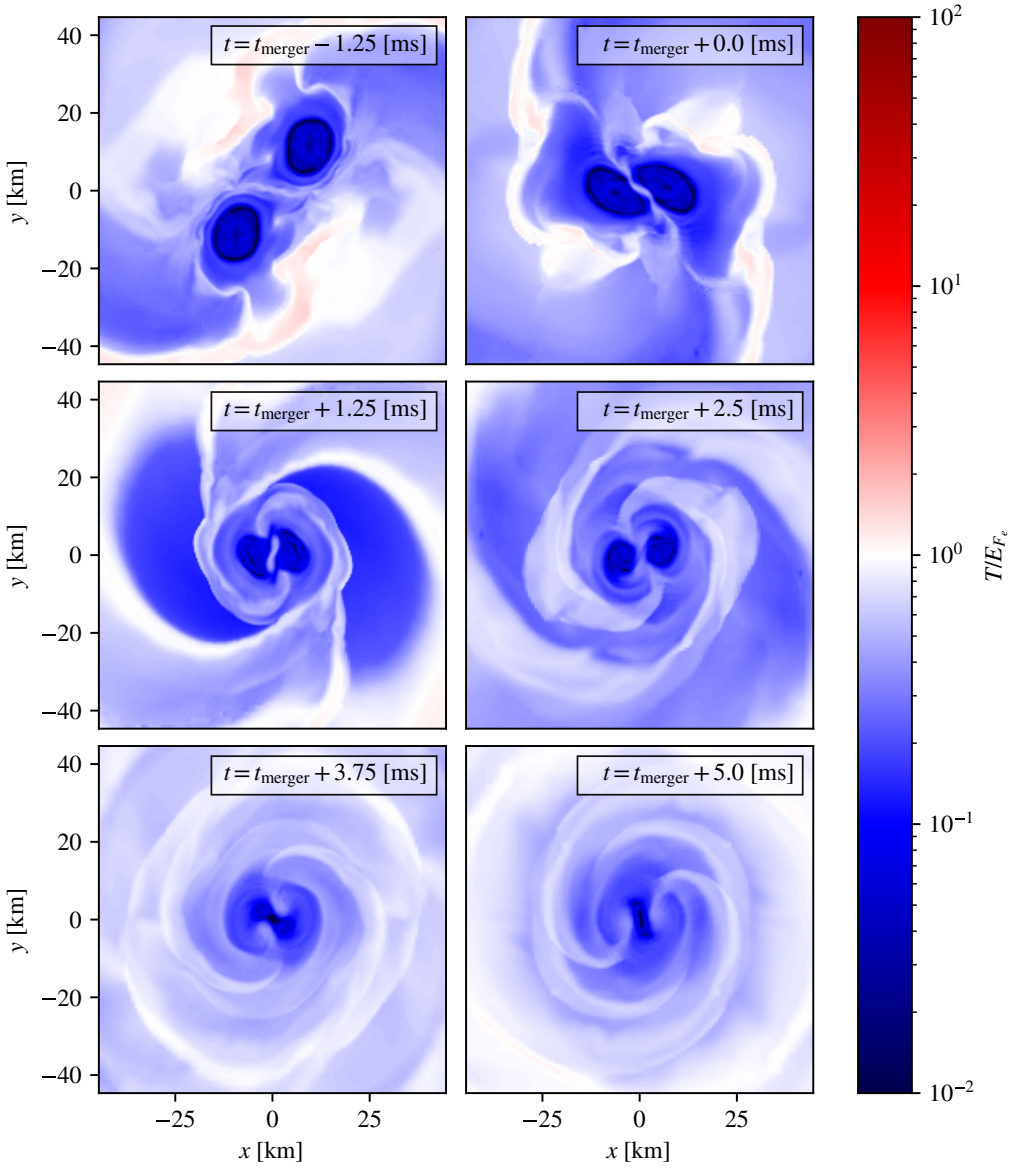

FIGURE 7.15: Plots of the ratio of fluid temperature T to electron Fermi energy E_{F_e}, T/E_{F_e} for a simulation of the merger of two $1.4\,M_\odot$ neutron stars using the APR[122] equation of state. Frames are at 1.25 ms intervals, with the second frame coinciding with the merger time of the binary.

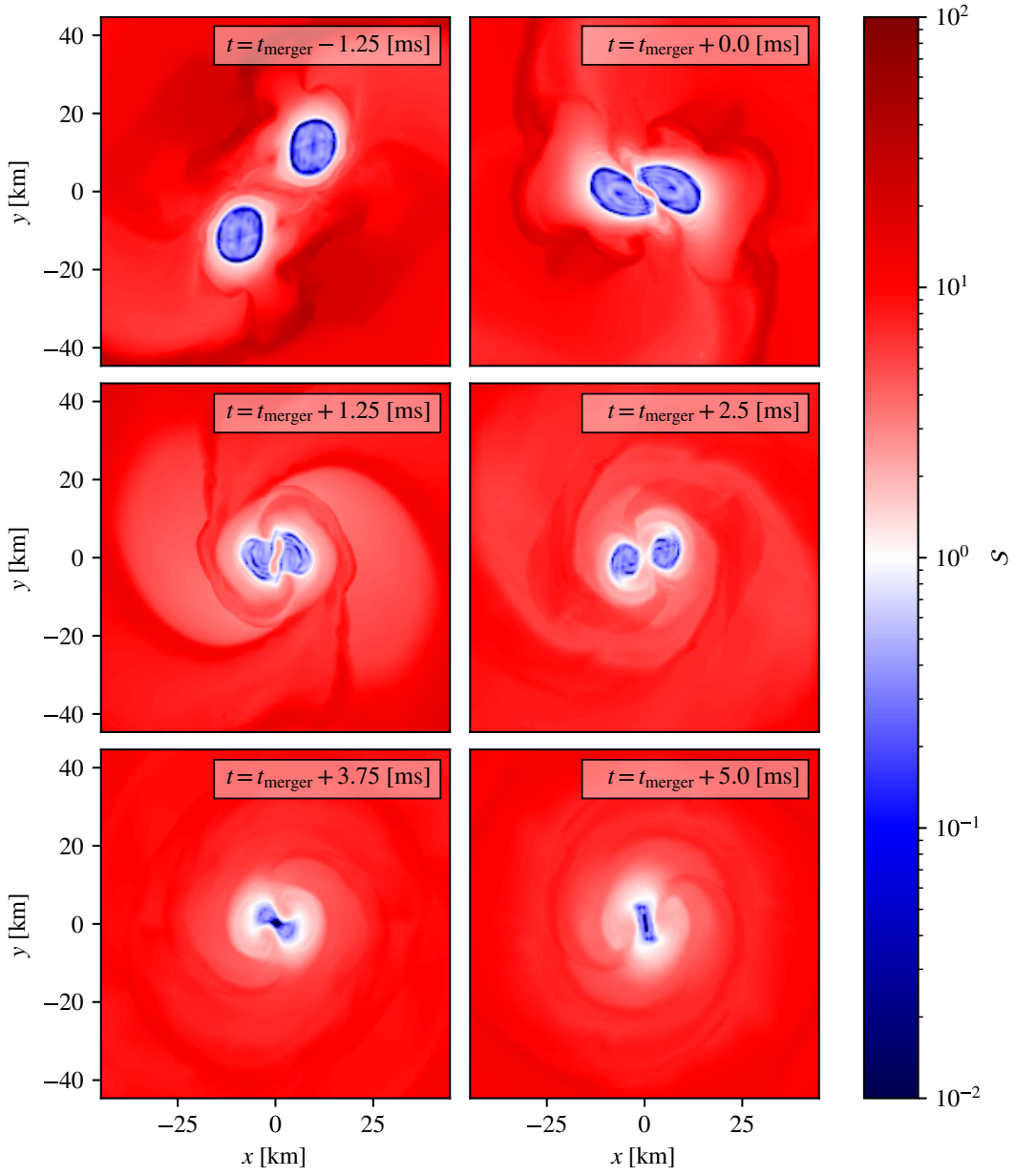

FIGURE 7.16: Plots of the entropy per baryon $\mathcal{S}$ for a simulation of the merger of two $1.4\,M_\odot$ neutron stars using the APR [122] equation of state. Frames are at 1.25 ms intervals, with the second frame coinciding with the merger time of the binary.

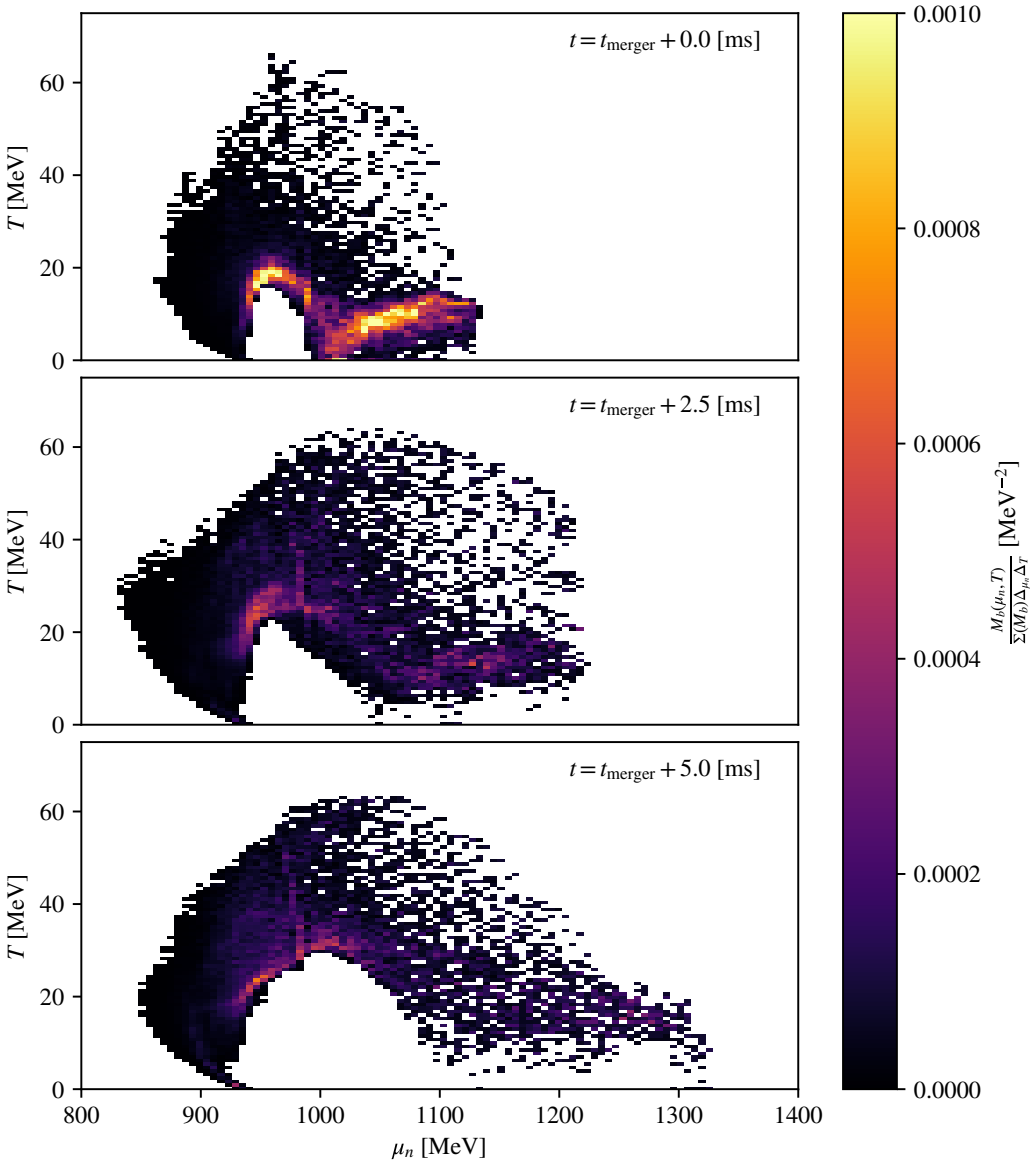

FIGURE 7.17: Histogram of the proportion of baryon mass at a given μ_{n} and T, $M_b(\mu_{\text{n}}, T)/\Sigma(M_b)$, divided by the bin widths, $\Delta\mu_{\text{n}}$ and ΔT, for a simulation of the merger of two $1.4\,M_\odot$ neutron stars using the APR[122] equation of state. Frames are at 2.5 ms intervals, with the first frame coinciding with the merger time of the binary. Matter at atmosphere density ($\rho_{\text{atmo}} = 1 \times 10^{-11}\,M_\odot^{-2}$) is ignored.

7.4 Known Issues

Studying the results of the three simulations detailed above, there are some issues, particularly involving the temperature, that are brought to light[3].

The entropy results bring out the expectations—the high density core remains cold—but they highlight an artefact of the simulations (apparent in, for example, the results of[53;103;105]). The region close to the neutron star surface is artificially hot already from the early stages of the simulation. This is also apparent from the temperature result in fig. 7.11. We see that, away from the shock region (where the two stars first touch) the low density matter near each star's surface has reached a temperature well above 10 MeV. This feature is clearly "unfortunate" as it means that we cannot discuss the fine print of the thermal physics (for example the point at which the neutron star crust melts).

The artificially high surface temperature may, perhaps intuitively, be explained in terms of a numerical shock associated with the transition from the high-density neutron star interior to the low-density (artificial) atmosphere. Consider any perturbation in the interior that leads to a wave propagating outwards. As it travels "down" the density gradient, the speed of sound (and hence the characteristic speed) decreases, meaning that the perturbation steepens and will eventually lead to a shock. There is an assumption that this shock takes place before hitting the atmosphere (a relevant toy problem demonstrating the effect is discussed in[65]), which seems to match the numerical results, but the subsequent propagation through the atmosphere is certainly artificial. Such a feature may (to some extent) be unavoidable. One may argue that this is not a major problem, as the artificial heating is overwhelmed by the entropy generated in the merger and therefore does not feature prominently in the post-merger dynamics. Pragmatically, this may be true on average and hence sufficient for gravitational wave emission (which is determined by the bulk matter dynamics), but may not be true for determining correct local matter properties (which is needed for e.g. electromagnetic and neutrino emission). One would also like to be able to distinguish the (fairly large) modelling error due to this artificial heating from the robust merger physics.

The surface artefact warrants closer scrutiny since the feature may cause problems for more than the detailed modelling of the neutrino aspects and the associated weak interactions. In particular, the temperature is required to decide if neutrinos are trapped (a point we return to in section 9.2). The artificial surface feature may "confuse" any automated neutrino treatment and hence it would seem important to understand if this makes a—qualitative or quantitative—difference.

It is also worth noting that there may be a link to the initial data prescription. Our initial model assumes a uniform temperature, which would not be a realistic representation of

[3]This section is based on a discussion in Hammond et al.[68].

a mature neutron star (more likely represented by a uniform redshifted temperature). The choice is due to the initial data construction within LORENE, which requires a barotropic model. The immediate alternative to our choice would be to assume a uniform entropy distribution (see Cipolletta et al.[39] where they compare simulations constant T and constant $\mathcal{S}$ initial conditions), but this does not represent the expected physics either. Recent work[36] has considered more realistic entropy profiles, but these have not yet been used in merger simulations.

It is also worth commenting on the temperature of the relatively cold high-density region. As is evident from the snapshots in figs. 7.11, 7.16 and 7.17, the temperature in the core of the merger remnant (in the simulations using tabulated equations of state) is at the level of $5 - 10\,\mathrm{MeV}$. This may be a reasonable reflection of the physics but the result also has to be considered with caveats. As the evolved internal energy τ only depends weakly on the temperatures, the numerical inversion to extract the temperature may be associated with significant uncertainties.

We can approximately quantify this by a *condition number* $\mathcal{K}_{A\to B}$ which gives, to first order, the relative error in B given a relative error in A. For the temperature, we assume an error $\Delta\tau$ in the energy τ, which induces an error in T as

$$\begin{aligned}\frac{\Delta T}{T} &= \frac{1}{T}\left(\frac{\partial\tau}{\partial T}\right)^{-1}\Delta\tau \\ &= \left[\frac{\tau}{T}\left(\frac{\partial\tau}{\partial T}\right)^{-1}\right]\frac{\Delta\tau}{\tau} \\ &= \mathcal{K}_{\tau\to T}\frac{\Delta\tau}{\tau}. \end{aligned} \tag{7.7}$$

We then compute $\mathcal{K}_{\tau\to T}$ for a range of densities, velocities, and temperatures (using the APR equation of state), leading to the results shown in fig. 7.18.

We see that the condition number, indicating the growth in the inevitable numerical error, is largely insensitive to the density and velocity. It is, however, extremely sensitive to the temperature and grows rapidly as the temperature decreases. Whilst condition numbers only give a qualitative, rather than quantitative indication, this suggests that working directly with the temperature will be prone to large numerical errors. This is not a feature of all numerical operations converting between conserved and primitive variables, as the analogous condition number for the densities $\mathcal{K}_{D\to\rho} = 1$.

If we want to do (significantly) better, we may have to work with a different set of variables (e.g. the entropy). This kind of development may be required if we want to be able to explore astrophysically motivated questions associated with the neutron star crust and high-density condensates, which come into play at temperatures below about $1\,\mathrm{MeV}$. However, as it would either involve a different equation of state parameterisation,

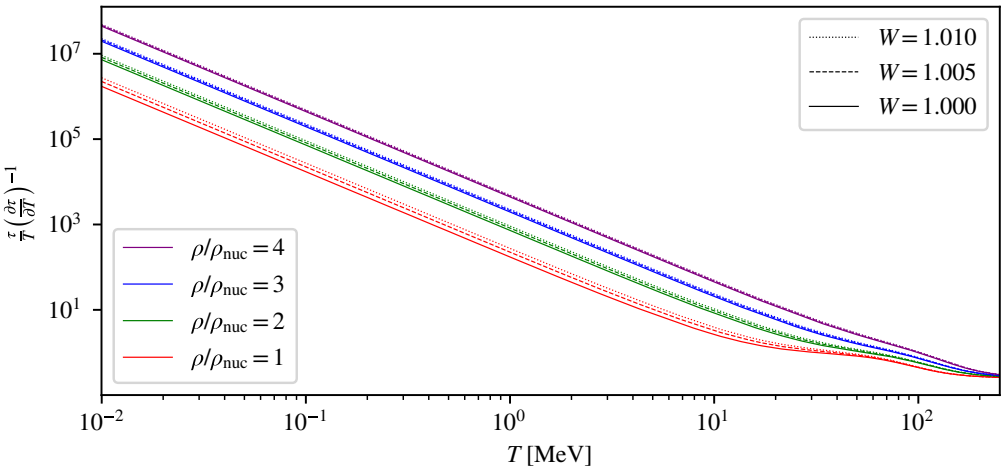

FIGURE 7.18: The condition number from τ to T, $\mathcal{K}_{\tau\to T} = \frac{\tau}{T}\left(\frac{\partial\tau}{\partial T}\right)^{-1}$, for the APR equation of state[122] for a range of temperatures, densities, and velocities (as encoded in the Lorentz factor W).

or dealing with non-conserved fluxes, it is unlikely to be a preferred immediate direction of travel (see discussion in section 11.2).

Chapter 8

Cold Beta-Equilibrium

The simplest picture of the composition of neutron star matter is a fluid of neutrons, protons, and electrons. As the individual pre-merger stars are typically assumed to be old, we expect that matter to have reached some equilibrium configuration. As this equilibrium involves the exchange of energetic electrons it is referred to as β-equilibrium. We will go on to discuss the effects of finite temperature on β-equilibrium in chapter 9, however we start here by discussing equilibrium in cold neutron star matter. After covering the definition of this equilibrium, we present, using data from the APR simulation described in chapter 7, a study of how deviation from it might be detectable using future gravitational wave observatories[1].

8.1 Defining Cold Equilibrium

To study the effects of reactions in simulations, we need to define the equilibrium state of the matter, and some measure of deviation from that state. So called β-equilibrium (also referred to as Nuclear Statistical Equilibrium) in cold neutron star matter is primarily driven by the direct Urca processes [83]

$$\mathrm{n} \rightarrow \mathrm{p} + \mathrm{e}^- + \overline{\nu}_\mathrm{e}, \tag{8.1}$$

$$\mathrm{p} + \mathrm{e}^- \rightarrow \mathrm{n} + \nu_\mathrm{e}, \tag{8.2}$$

and the modified Urca processes

$$\mathrm{n} + \mathrm{n} \rightarrow \mathrm{p} + \mathrm{n} + \mathrm{e}^- + \overline{\nu}_\mathrm{e}, \tag{8.3}$$

$$\mathrm{n} + \mathrm{p} \rightarrow \mathrm{p} + \mathrm{p} + \mathrm{e}^- + \overline{\nu}_\mathrm{e}, \tag{8.4}$$

$$\mathrm{p} + \mathrm{n} + \mathrm{e}^- \rightarrow \mathrm{n} + \mathrm{n} + \nu_\mathrm{e}, \tag{8.5}$$

$$\mathrm{p} + \mathrm{p} + \mathrm{e}^- \rightarrow \mathrm{n} + \mathrm{p} + \nu_\mathrm{e}. \tag{8.6}$$

[1] Published in Hammond et al. [69].

There are some important things to note here: direct Urca has a significantly higher rate than modified Urca (by a factor of 10^4 in cold, core density matter using the APR equation of state[12]) but is kinematically forbidden below some equation of state dependent threshold density[12], and as we are dealing with cold matter it is transparent to neutrinos so they only appear as products in the above reactions. In addition, there are equivalent versions of all the above reactions involving muons in place of electrons which we ignore here for simplicity. Finally, as the electrons are relativistic across the regions of interest (we have $\mu_\mathrm{e} \gg T$), positron occupation will be suppressed by a factor of $\exp\left(-\mu_\mathrm{e}/T\right)$, so we can safely ignore reactions involving them as well[12] (although they are important for other related phenomena e.g. setting outflow composition[110]).

In order to obtain an equilibrium relation in the chemical potentials of the form

$$N_{\mathrm{i}_1}\mu_{\mathrm{i}_1} + N_{\mathrm{i}_2}\mu_{\mathrm{i}_2} + \ldots = N_{\mathrm{j}_1}\mu_{\mathrm{j}_1} + N_{\mathrm{j}_2}\mu_{\mathrm{j}_2} + \ldots, \tag{8.7}$$

the principle of detailed balance dictates that we need a relation of the form

$$N_{\mathrm{i}_1}P_{\mathrm{i}_1} + N_{\mathrm{i}_2}P_{\mathrm{i}_2} + \ldots \leftrightarrow N_{\mathrm{j}_1}P_{\mathrm{j}_1} + N_{\mathrm{j}_2}P_{\mathrm{j}_2} + \ldots, \tag{8.8}$$

where P_k are the particle species, μ_k is the chemical potential of species P_k, and N_k are stoichiometric coefficients representing the balance between products and reactants in a given reaction.

Unfortunately none of the 6 reactions above (eqs. (8.1) to (8.6)) are the exact inverse of any other (as stated above they must all include a neutrino as a product) so we cannot yet write something in the form of eq. (8.8). However, we can adopt the Fermi surface approximation[12] and assume that the neutrinos are kinematically negligible (their energy will be on the order of a few times the temperature[67]), and thus ignore them in the reactions. Therefore we can pair respective electron capture and neutron decay reactions together to give

$$\mathrm{n} \leftrightarrow \mathrm{p} + \mathrm{e}^-, \tag{8.9}$$

$$\mathrm{n} + \mathrm{n} \leftrightarrow \mathrm{n} + \mathrm{p} + \mathrm{e}^-, \tag{8.10}$$

$$\mathrm{n} + \mathrm{p} \leftrightarrow \mathrm{p} + \mathrm{p} + \mathrm{e}^-. \tag{8.11}$$

By eq. (8.7) these all have the equilibrium condition

$$\mu_\mathrm{n} = \mu_\mathrm{p} + \mu_\mathrm{e}, \tag{8.12}$$

which dictates the equilibrium matter composition from $T = 0$ up to temperatures of around a few hundred keV[12;68]. Finally, we can then measure the deviation from

chemical equilibrium by defining

$$\mu_\Delta = \mu_\mathrm{n} - \mu_\mathrm{p} + \mu_\mathrm{e}. \tag{8.13}$$

To connect this relation with chemical potentials commonly used in microphysics, such as the baryon, charge, and lepton electron chemical potentials, μ_b, μ_q, and μ_{le} respectively, it is possible to break the chemical potential of each species down into parts related to the quantum numbers of said species. The chemical potential of a species i is defined by

$$\mu_i = \left.\frac{\partial f}{\partial n_i}\right|_{T,n_j \text{ for } j\neq i}, \tag{8.14}$$

where f is the Helmholtz free energy density, and n_i is the number density of species i. We can break the partial derivative into parts (in this example using only quantum numbers relevant to the reactions in eqs. (8.9) to (8.11)), giving

$$\mu_i = \frac{dn_b}{dn_i}\frac{\partial f}{\partial n_b} + \frac{dn_q}{dn_i}\frac{\partial f}{\partial n_q} + \frac{dn_{le}}{dn_i}\frac{\partial f}{\partial n_{le}}, \tag{8.15}$$

where we have omitted the variables held constant on each partial derivative for brevity. The total derivatives in this equation are equal to the baryon number B_i, electric charge number Q_i, and the electron lepton number L_i^e respectively, and the partial derivatives are equal to their respective conjugate chemical potentials. Thus, we can write μ_i as

$$\mu_i = B_i\mu_b + Q_i\mu_q + L_i^\mathrm{e}\mu_{le}. \tag{8.16}$$

For equations of state or interactions that involve other quantum numbers, this expression can be expanded as required to include those terms.

Using eq. (8.16) we can write the neutron, proton, and electron chemical potentials as

$$\mu_\mathrm{n} = \mu_b, \tag{8.17}$$

$$\mu_\mathrm{p} = \mu_b + \mu_q, \tag{8.18}$$

$$\mu_\mathrm{e} = \mu_{le} - \mu_q. \tag{8.19}$$

By substituting these into eq. (8.13), we can see that

$$\mu_\Delta = -\mu_{le}. \tag{8.20}$$

8.2 Deviation from Equilibrium

Before we look at the effects of deviation from equilibrium, it is useful to get a handle on how far from equilibrium the fluid is pushed in the “best case” (no reactions). In

fig. 8.1 we plot snapshots from the APR simulations discussed in chapter 7. We see significant deviation from β-equilibrium (μ_Δ in the region of 150 MeV) in the interface between the two stars as they collide, and also in the post-merger remnant, following the hotspots described in section 7.2.2. In addition, we also see that the cores of the stars have moved to around $\mu_\Delta \approx 20 - 30\,\mathrm{MeV}$. However, an absence of reactions is not the only limiting case we can study without implementing the reactions in full.

Above, we assumed that the reaction rate Γ_e in eq. (7.2) was small compared to the timescale of the dynamics (effectively freezing the composition). If we assume the opposite limit, i.e. the reactions are much faster than the dynamics, then each fluid element will always be in equilibrium. By solving the equation of state for $\mu_\Delta(\rho, T, Y_\mathrm{e}) = 0$ at each point in (ρ, T) we obtain a 2-dimensional table that describes a fluid always in equilibrium. The remainder of this section is dedicated to a comparison of two simulations, the "out-of-equilibrium" APR simulation above, and a second using an equilibriated APR equation of state, calculated using this prescription. We start by comparing the gravitational wave spectra of the two simulations, then discuss how the deviation from equilibrium affects the physical properties of the matter, and finally how this affects the structure of the neutron star. This work has been published in Hammond et al.[69].

8.2.1 Comparing gravitational wave spectra

In the left panel of fig. 8.2 we compare the spectra of post-merger gravitational wave output of simulations using reactions in these two limits (fast: $\mu_\Delta(Y_\mathrm{e}) = 0$ and slow: $\Gamma_\mathrm{e} = 0$) alongside the speculative noise curve for the ET-D design of the Einstein Telescope[73]. Although similar in overall shape, the peak frequencies for the two simulations are visibly different: $2992 \pm 8\,\mathrm{Hz}$ in the slow-limit simulation, and $3050 \pm 10\,\mathrm{Hz}$ in the fast-limit simulation, giving a difference of $\Delta f = 58 \pm 13\,\mathrm{Hz}$. Peak frequencies are calculated using the method of MacLeod[88], and checked using the method of Quinn[108] (in all cases the estimate from the second method is within the error bounds of the first).

In addition to the peak frequencies we can also compute the mismatch $\mathcal{M}$, a commonly used[25;80;86;92] quantity when comparing gravitational wave spectra. To define the mismatch we start with the waveform overlap, $\langle h_1 | h_2 \rangle$, defined by

$$\langle h_1 | h_2 \rangle = 4 \int_{-\infty}^{\infty} \frac{\tilde{h}_1(f)\,\tilde{h}_2^*(f)}{S_n(f)} \mathrm{d}f, \tag{8.21}$$

where $\hat{h}(f)$ is the (continuous) Fourier transform of h and $*$ denotes its complex conjugate, while S_n is the strain sensitivity of a detector as power spectral density.

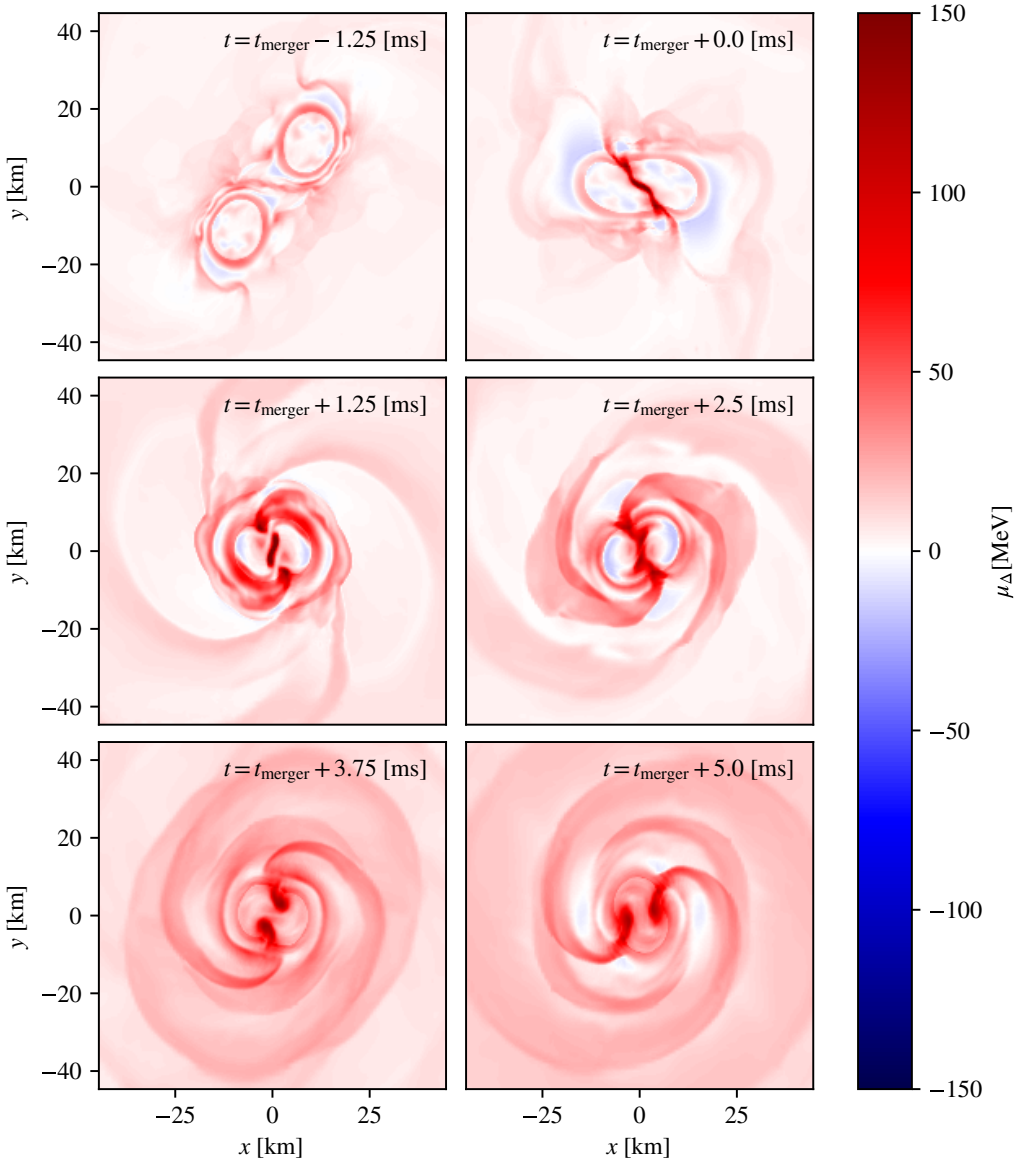

FIGURE 8.1: Plots of the deviation from β-equlibrium $\mu_\Delta = \mu_n - (\mu_p + \mu_e)$ for a simulation of the merger of two $1.4\,M_\odot$ neutron stars using the APR[122] equation of state. Frames are at 1.25 ms intervals, with the second frame coinciding with the merger time of the binary.

As the simulation outputs a discretely sampled signal, we discretise eq. (8.21) to obtain

$$\langle h_1|h_2\rangle = 4\Delta f \sum_{f=f_{\min}}^{f_{\max}} \left(\frac{\tilde{h}_1(f)\,\tilde{h}_2^*(f)}{S_n(f)} \right), \tag{8.22}$$

where the tilde denotes the discrete Fourier transform (taking care to correctly normalise for consistency with the continuous version), $f_{\max}$ is the maximum resolvable frequency in the data (typically the Nyquist frequency), $f_{\min}$ is the minimum resolvable frequency (which for us is either the ω_* used in the fixed frequency integration, or the filter frequency), and Δf is the frequency resolution of the discrete Fourier transform.

The match M (sometimes referred to as the "maximised overlap") between the two signals is given by

$$\mathrm{M} = \frac{\max\left(\langle h_1|h_2\rangle\right)}{\sqrt{\langle h_1|h_1\rangle \langle h_2|h_2\rangle}}, \tag{8.23}$$

where the overlap between h_1 and h_2 should be maximised by time and phase shifting h_2 by (t_c, ϕ_c). The mismatch $\mathcal{M}$ is then given by one minus the real part of the (maximised) match, so we obtain

$$\mathcal{M} = 1 - \Re\left(\frac{\max\left(\left\langle h_1|h_2 e^{i(\phi_c - 2\pi f t_c)}\right\rangle\right)}{\sqrt{\langle h_1|h_1\rangle \langle h_2|h_2\rangle}} \right). \tag{8.24}$$

The mismatch between two spectra is related to the signal-to-noise ratio of the difference between the two signals ρ_{diff} through[92]

$$\rho_{\mathrm{diff}}^2 = 2\rho_{\mathrm{sig}}^2 \mathcal{M}, \tag{8.25}$$

where ρ_{sig} is the signal-to-noise ratio of a detected signal (not to be confused with the matter density). A widely used criterion for the two signals to be distinguishable is $\rho_{\mathrm{diff}} \geq 1$[86;107]. It should be noted that this does not relate to the ability to observe a signal amongst some noise, nor to using a given signal to estimate the specific model differences. Instead it speaks to whether or not one is able distinguish between one model and another, knowing that there is a signal present in the data.

For our purposes, we want to know what signal-to-noise ratio is required for us to be able to distinguish between the two reaction limits, so we rearrange eq. (8.25) to provide the condition

$$\rho_{\mathrm{sig}} \geq \frac{1}{\sqrt{2\mathcal{M}}}. \tag{8.26}$$

To reiterate, if this condition is satisfied for a given detected signal we should be able to differentiate between fast- and slow-reaction limit behaviour. We define $\rho_{\rm req} = 1/\sqrt{2\mathcal{M}}$ as the smallest signal-to-noise ratio required to satisfy this condition.

For the two spectra in the left panel of fig. 8.2 we find that $\rho_{\rm req} = 1.2$. As this is lower than any reasonable threshold to claim a detection, if the post-merger signal is detected then the effect of the reactions must be taken into account to avoid the introduction of an unknown systematic error into the parameters extracted from the waveform [69].

As a sanity check, we also performed a simulation with 20% coarser resolution but otherwise identical to the slow limit simulation, the output of which is plotted (alongside the original slow-limit results) in the right panel of fig. 8.2. Comparing these results with the original slow limit simulation we find a mismatch on the order of 1%, and the lower resolution simulation has a measured peak frequency of $2992 \pm 6\,\mathrm{Hz}$. The mismatch in this case is much lower than the mismatch between the fast- and slow-limit simulations, and the peak frequency extracted from the low resolution simulation is within the error bounds of the high-resolution simulation, so we are confident that the numerical error due to the finite resolution of the simulations is much less than the difference driven by the contrasting physics models in the two cases. There are, of course, other numerical uncertainties beyond resolution-dependent errors that affect the signals obtained, for example the (unknown) equation of state of neutron star matter, the issues related to temperature mentioned in previous chapters, and various modelling errors in both the spacetime and fluid evolution methods. However, the purpose of these simulations is to determine whether the *difference* between the fast- and slow-reaction limits, both of which are un-physical in their own right, might be significant enough to introduce systematic errors in waveforms intended for use with future detectors (especially in the context of parameter extraction by template matching).

8.2.2 Softening of the equation of state

Having established that there is a measurable difference between the gravitational wave signals in the two reaction limits, it is natural to question what drives this difference. We start by using a simple model motivated by the results of the simulations to explore what effect being out of equilibrium has on an isolated star. By constructing two 1-D equations of state from the full 3-D table, one in- and one out-of-equilibrium, we can see what effect being out of equilibrium has on the structure of the stars. For the out-of-equilibrium equation of state we solve the composition for a constant $\mu_\Delta = \mu_{\rm n} - (\mu_{\rm p} + \mu_{\rm e})$ throughout the table. This value of the offset need not be wholly representative of the full simulations, but for the sake of comparison we use the simulation data to inform our choice. In fig. 8.1 we see the deviation from chemical equilibrium in the two stars around the time of merger. At 5 ms post merger most of the core matter is in the $\mu_\Delta = 20 - 30\,\mathrm{MeV}$ range (excluding the hotspots), hence we choose $\mu_\Delta = 20\,\mathrm{MeV}$ as

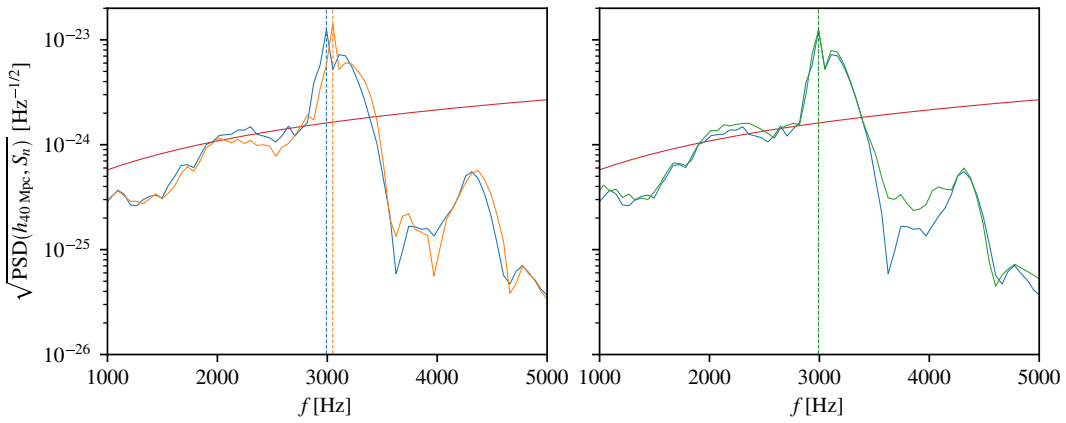

FIGURE 8.2: Left panel: Square-root power spectral density plots of the recovered waveforms from slow- and fast-reaction-limit simulations (blue and orange curves, respectively). Right panel: The same comparison for the high- and low-resolution simulations (blue and green curves, respectively), both in the slow reaction limit. Both panels: Peak frequencies for each curve calculated using the MacLeod method [88] are shown with dashed lines. The ET-D design sensitivity curve for the Einstein Telescope [73] is shown in red. The waveforms are normalised to a distance of 40 Mpc and assume the source and detector are perfectly aligned.

the deviation from equilibrium for our out-of-equilibrium equation of state. Finally, we obtain a 1-D equation of state by choosing a temperature of $T = 5\,\mathrm{MeV}$.

Using LORENE [63] we first construct a rotating star using the equilibrium equation of state. We choose the mass of the star such that the central baryon number density is similar to that found in our simulation ($3.5\,n_{\mathrm{sat}}$), giving a total baryon mass for the star of $1.7\,M_{\odot}$, and an arbitrary rotation frequency of 500 Hz (significant, but well below the Keplerian frequency). This gives us a total angular momentum. We then construct a rotating star using the out-of-equilibrium equation of state conserving the total baryon mass and angular momentum from the equilibrium star, giving us a different rotation frequency.

We find that to conserve angular momentum, the rotation frequency must be reduced to 496 Hz (a fractional difference of $\sim 1\%$). We repeated this procedure for the DD2 [71;138], SFHx [129], and SLy4 [38;121] equations of state, and obtained similar results, with the change in rotation frequency required being of order 1%, and always in the same direction: out-of-equilibrium stars need a lower frequency to match the angular momentum of the respective in-equilibrium stars due to an increase in moment of inertia. This increase is caused by a softening of the equation of state at core densities as the matter is taken out of equilibrium, creating a flatter density profile, and thus more of the mass of the star is located further from the axis of rotation.

Having observed similar effects in several equations of state, one might ask whether this effect is truly general. We can try to answer this through the adiabatic index

$$\Gamma = \left.\frac{\partial \ln p}{\partial \ln n_b}\right|_{\mathcal{S},Y_\mathrm{e}} = \frac{n_b}{p}\left.\frac{\partial p}{\partial n_b}\right|_{\mathcal{S},Y_\mathrm{e}}, \tag{8.27}$$

where $\mathcal{S}$ is the specific entropy per baryon. Taylor expanding around equilibrium and keeping only the first term we obtain

$$\Gamma\left(\mu_\Delta\right) = \Gamma\left(0\right) + \mu_\Delta \left.\frac{\partial \Gamma\left(0\right)}{\partial \mu_\Delta}\right|_{\mathcal{S},n_b} + \mathcal{O}\left(\mu_\Delta^2\right), \tag{8.28}$$

so the change in stiffness when moving out of equilibrium will depend on the sign of the second term. Focusing first on the derivative, and shifting from μ_Δ to Y_e we get

$$\left.\frac{\partial \Gamma\left(0\right)}{\partial \mu_\Delta}\right|_{\mathcal{S},n_b} = \left(\left.\frac{\partial \mu_\Delta}{\partial Y_\mathrm{e}}\right|_{\mathcal{S},n_b}\right)^{-1} \left.\frac{\partial \Gamma}{\partial Y_\mathrm{e}}\right|_{\mathcal{S},n_b} \tag{8.29}$$

where we have dropped the evaluation at $\mu_\Delta = 0$ for brevity. It is intuitive that in the vicinity of equilibrium we should have $\partial\mu_\Delta/\partial Y_\mathrm{e} < 0$ (increasing Y_e away from equilibrium means moving to matter with more protons and electrons than the equilibrium fraction, so μ_p and μ_e increase, resulting in a decrease in μ_Δ), so again we focus on the derivative of Γ. Substituting in eq. (8.27) we obtain

$$\left.\frac{\partial \Gamma}{\partial Y_\mathrm{e}}\right|_{\mathcal{S},n_b} = \left.\frac{\partial}{\partial Y_\mathrm{e}}\right|_{\mathcal{S},n_b}\left(\frac{n_b}{p}\right)\left.\frac{\partial p}{\partial n_b}\right|_{\mathcal{S},Y_\mathrm{e}} + \frac{n_b}{p}\left.\frac{\partial}{\partial Y_\mathrm{e}}\right|_{\mathcal{S},n_b}\left(\left.\frac{\partial p}{\partial n_b}\right|_{\mathcal{S},Y_\mathrm{e}}\right) \tag{8.30}$$

$$= \frac{-n_b}{p^2}\left.\frac{\partial p}{\partial Y_\mathrm{e}}\right|_{\mathcal{S},n_b}\left.\frac{\partial p}{\partial n_b}\right|_{\mathcal{S},Y_\mathrm{e}} + \frac{n_b}{p}\left.\frac{\partial}{\partial n_b}\right|_{\mathcal{S},Y_\mathrm{e}}\left(\left.\frac{\partial p}{\partial Y_\mathrm{e}}\right|_{\mathcal{S},n_b}\right) \tag{8.31}$$

$$= \frac{-\Gamma}{p}\left.\frac{\partial p}{\partial Y_\mathrm{e}}\right|_{\mathcal{S},n_b} + \frac{n_b}{p}\left.\frac{\partial}{\partial n_b}\right|_{\mathcal{S},Y_\mathrm{e}}\left(\left.\frac{\partial p}{\partial Y_\mathrm{e}}\right|_{\mathcal{S},n_b}\right). \tag{8.32}$$

We can expand the $\partial p/\partial Y_\mathrm{e}$ term to get

$$\left.\frac{\partial p}{\partial Y_\mathrm{e}}\right|_{\mathcal{S},n_b} = \left.\frac{\partial p}{\partial Y_\mathrm{e}}\right|_{T,n_b} - \left.\frac{\partial p}{\partial T}\right|_{Y_\mathrm{e},n_b}\left(\left.\frac{\partial \mathcal{S}}{\partial T}\right|_{Y_\mathrm{e},n_b}\right)^{-1}\left.\frac{\partial \mathcal{S}}{\partial Y_\mathrm{e}}\right|_{T,n_b}, \tag{8.33}$$

the first part of which becomes

$$\left.\frac{\partial p}{\partial Y_{\rm e}}\right|_{T,n_b} = \left.\frac{\partial}{\partial Y_{\rm e}}\right|_{T,n_b}\left(n_b \left.\frac{\partial f}{\partial n_b}\right|_{T,Y_{\rm e}} - f\right) \tag{8.34}$$

$$= n_b \left.\frac{\partial}{\partial n_b}\right|_{T,Y_{\rm e}}\left(\left.\frac{\partial f}{\partial Y_{\rm e}}\right|_{T,n_b}\right) - \left.\frac{\partial f}{\partial Y_{\rm e}}\right|_{T,n_b} \tag{8.35}$$

$$= n_b \left.\frac{\partial}{\partial n_b}\right|_{T,Y_{\rm e}}(-n_b\mu_\Delta) + n_b\mu_\Delta \tag{8.36}$$

$$= -n_b^2 \left.\frac{\partial \mu_\Delta}{\partial n_b}\right|_{T,Y_{\rm e}}, \tag{8.37}$$

where f is the Helmholtz free energy density.

The second part of eq. (8.33) can be manipulated to give

$$\left.\frac{\partial p}{\partial T}\right|_{Y_{\rm e},n_b} = \left.\frac{\partial}{\partial T}\right|_{Y_{\rm e},n_b}\left(n_b \left.\frac{\partial f}{\partial n_b}\right|_{T,Y_{\rm e}} - f\right) \tag{8.38}$$

$$= n_b \left.\frac{\partial}{\partial n_b}\right|_{T,Y_{\rm e}}\left(\left.\frac{\partial f}{\partial T}\right|_{Y_{\rm e},n_b}\right) - \left.\frac{\partial f}{\partial T}\right|_{Y_{\rm e},n_b} \tag{8.39}$$

$$= n_b \left.\frac{\partial}{\partial n_b}\right|_{T,Y_{\rm e}}(-n_b\mathcal{S}) + n_b\mathcal{S} \tag{8.40}$$

$$= -n_b^2 \left.\frac{\partial \mathcal{S}}{\partial n_b}\right|_{T,Y_{\rm e}}. \tag{8.41}$$

Finally the fourth term in eq. (8.33) gives

$$\left.\frac{\partial \mathcal{S}}{\partial Y_{\rm e}}\right|_{T,n_b} = \left.\frac{\partial}{\partial Y_{\rm e}}\right|_{T,n_b}\left(\frac{-1}{n_b}\left.\frac{\partial f}{\partial T}\right|_{Y_{\rm e},n_b}\right) \tag{8.42}$$

$$= \left.\frac{\partial}{\partial T}\right|_{Y_{\rm e},n_b}\left(\frac{-1}{n_b}\left.\frac{\partial f}{\partial Y_{\rm e}}\right|_{T,n_b}\right) \tag{8.43}$$

$$= \left.\frac{\partial \mu_\Delta}{\partial T}\right|_{Y_{\rm e},n_b}. \tag{8.44}$$

Bringing these together we obtain

$$\left.\frac{\partial p}{\partial Y_{\rm e}}\right|_{\mathcal{S},n_b} = -n_b^2 \left.\frac{\partial \mu_\Delta}{\partial n_b}\right|_{T,Y_{\rm e}} + n_b^2 \left.\frac{\partial \mathcal{S}}{\partial n_b}\right|_{T,Y_{\rm e}}\left(\left.\frac{\partial \mathcal{S}}{\partial T}\right|_{Y_{\rm e},n_b}\right)^{-1}\left.\frac{\partial \mu_\Delta}{\partial T}\right|_{Y_{\rm e},n_b} \tag{8.45}$$

$$= -n_b^2 \left.\frac{\partial \mu_\Delta}{\partial n_b}\right|_{\mathcal{S},Y_{\rm e}}. \tag{8.46}$$

Substituting back into eq. (8.32) we get

$$\left.\frac{\partial\Gamma}{\partial Y_{\mathrm{e}}}\right|_{\mathcal{S},n_b} = \frac{\Gamma n_b^2}{p}\left.\frac{\partial\mu_\Delta}{\partial n_b}\right|_{\mathcal{S},Y_{\mathrm{e}}} - \frac{n_b}{p}\left.\frac{\partial}{\partial n_b}\right|_{\mathcal{S},Y_{\mathrm{e}}}\left(n_b^2\left.\frac{\partial\mu_\Delta}{\partial n_b}\right|_{\mathcal{S},Y_{\mathrm{e}}}\right) \tag{8.47}$$

$$= \frac{\Gamma n_b^2}{p}\left.\frac{\partial\mu_\Delta}{\partial n_b}\right|_{\mathcal{S},Y_{\mathrm{e}}} - \frac{2n_b^2}{p}\left.\frac{\partial\mu_\Delta}{\partial n_b}\right|_{\mathcal{S},Y_{\mathrm{e}}} - \frac{n_b^3}{p}\left.\frac{\partial^2\mu_\Delta}{\partial n_b^2}\right|_{\mathcal{S},Y_{\mathrm{e}}} \tag{8.48}$$

$$= \frac{n_b^2}{p}\left(\Gamma-2\right)\left.\frac{\partial\mu_\Delta}{\partial n_b}\right|_{\mathcal{S},Y_{\mathrm{e}}} - \frac{n_b^3}{p}\left.\frac{\partial^2\mu_\Delta}{\partial n_b^2}\right|_{\mathcal{S},Y_{\mathrm{e}}}. \tag{8.49}$$

In a neutron star merger we can simplistically think of the shift away from equilibrium as being driven by an adiabatic compression of the matter. We already have the entropy held constant, and we can think of the compression as a fractional change in n_b, so we want to switch out the derivatives with respect to n_b for derivatives with respect to $\ln n_b$. For some general function $f(x)$ this looks like

$$\frac{\partial f}{\partial x} = \frac{\partial\ln x}{\partial x}\frac{\partial f}{\partial\ln x} \tag{8.50}$$

$$= \frac{\partial}{\partial x}\left(\ln x\right)\frac{\partial f}{\partial\ln x} \tag{8.51}$$

$$= \frac{1}{x}\frac{\partial f}{\partial\ln x} \tag{8.52}$$

$$\frac{\partial^2 f}{\partial x^2} = \frac{\partial}{\partial x}\left(\frac{\partial f}{\partial x}\right) \tag{8.53}$$

$$= \frac{\partial}{\partial x}\left(\frac{1}{x}\frac{\partial f}{\partial\ln x}\right) \tag{8.54}$$

$$= \frac{1}{x}\frac{\partial}{\partial x}\left(\frac{\partial f}{\partial\ln x}\right) + \frac{-1}{x^2}\frac{\partial f}{\partial\ln x} \tag{8.55}$$

$$= \frac{1}{x^2}\left(\frac{\partial^2 f}{\partial\ln x^2} - \frac{\partial f}{\partial\ln x}\right). \tag{8.56}$$

Using this to convert eq. (8.49) we get

$$\left.\frac{\partial\Gamma}{\partial Y_{\mathrm{e}}}\right|_{\mathcal{S},n_b} = \frac{n_b^2}{p}\left(\Gamma-2\right)\frac{1}{n_b}\left.\frac{\partial\mu_\Delta}{\partial\ln n_b}\right|_{\mathcal{S},Y_{\mathrm{e}}} - \frac{n_b^3}{p}\frac{1}{n_b^2}\left(\left.\frac{\partial^2\mu_\Delta}{\partial\ln n_b^2}\right|_{\mathcal{S},Y_{\mathrm{e}}} - \left.\frac{\partial\mu_\Delta}{\partial\ln n_b}\right|_{\mathcal{S},Y_{\mathrm{e}}}\right) \tag{8.57}$$

$$= \frac{n_b}{p}\left(\left(\Gamma-1\right)\left.\frac{\partial\mu_\Delta}{\partial\ln n_b}\right|_{\mathcal{S},Y_{\mathrm{e}}} - \left.\frac{\partial^2\mu_\Delta}{\partial\ln n_b^2}\right|_{\mathcal{S},Y_{\mathrm{e}}}\right). \tag{8.58}$$

Substituting this into eq. (8.28) we obtain

$$\Gamma\left(\mu_\Delta\right) = \Gamma\left(0\right) + \mu_\Delta\left(\left.\frac{\partial\mu_\Delta}{\partial Y_{\mathrm{e}}}\right|_{\mathcal{S},n_b}\right)^{-1}\frac{n_b}{p}\left(\left(\Gamma-1\right)\left.\frac{\partial\mu_\Delta}{\partial\ln n_b}\right|_{\mathcal{S},Y_{\mathrm{e}}} - \left.\frac{\partial^2\mu_\Delta}{\partial\ln n_b^2}\right|_{\mathcal{S},Y_{\mathrm{e}}}\right). \tag{8.59}$$

Expanding this we have

$$\Gamma(\mu_\Delta) = \Gamma(0) + \mu_\Delta \left(\frac{\partial \mu_\Delta}{\partial Y_\mathrm{e}}\bigg|_{\mathcal{S},n_b} \right)^{-1} \frac{n_b}{p} (\Gamma - 1) \frac{\partial \mu_\Delta}{\partial \ln n_b}\bigg|_{\mathcal{S},Y_\mathrm{e}} - \mu_\Delta \left(\frac{\partial \mu_\Delta}{\partial Y_\mathrm{e}}\bigg|_{\mathcal{S},n_b} \right)^{-1} \frac{n_b}{p} \frac{\partial^2 \mu_\Delta}{\partial \ln n_b^2}\bigg|_{\mathcal{S},Y_\mathrm{e}}, \quad (8.60)$$

so we define

$$\Gamma(\mu_\Delta) = \Gamma(0) + \Delta\Gamma_1 + \Delta\Gamma_2, \quad (8.61)$$

where

$$\Delta\Gamma_1 = \mu_\Delta \left(\frac{\partial \mu_\Delta}{\partial Y_\mathrm{e}}\bigg|_{\mathcal{S},n_b} \right)^{-1} \frac{n_b}{p} (\Gamma - 1) \frac{\partial \mu_\Delta}{\partial \ln n_b}\bigg|_{\mathcal{S},Y_\mathrm{e}}, \quad (8.62)$$

$$\Delta\Gamma_2 = -\mu_\Delta \left(\frac{\partial \mu_\Delta}{\partial Y_\mathrm{e}}\bigg|_{\mathcal{S},n_b} \right)^{-1} \frac{n_b}{p} \frac{\partial^2 \mu_\Delta}{\partial \ln n_b^2}\bigg|_{\mathcal{S},Y_\mathrm{e}}. \quad (8.63)$$

Assuming that μ_Δ is driven by an adiabatic compression with constant composition, it will pick up the same sign as

$$\mathrm{sgn}(\mu_\Delta) = \mathrm{sgn}\left(\frac{\partial \mu_\Delta}{\partial n_b}\bigg|_{\mathcal{S},Y_\mathrm{e}} \right) = \mathrm{sgn}\left(\frac{\partial \mu_\Delta}{\partial \ln n_b}\bigg|_{\mathcal{S},Y_\mathrm{e}} \right), \quad (8.64)$$

so, given $\Gamma > 1$ and $\partial\mu_\Delta/\partial Y_\mathrm{e} < 0$, $\Delta\Gamma_1$ will always work to decrease Γ, softening the equation of state. However, without knowledge of the equation of state, we cannot say in which direction $\Delta\Gamma_2$ goes, nor can we know which term is dominant.

In order to make progress here, we assume a very simple equation of state: a zero-temperature Fermi gas of non-interacting, non-relativistic neutrons and protons, and relativistic electrons. Under this assumption, the chemical potentials of the three species will take the form

$$\mu_n \approx m_n + E_{F_n} = m_n + \frac{(\hbar c)^2}{2m_n} \left(3\pi^2(1 - Y_\mathrm{e})\right)^{2/3} e^{\frac{2}{3}\ln n_b}, \quad (8.65)$$

$$\mu_p \approx m_p + E_{F_p} = m_p + \frac{(\hbar c)^2}{2m_p} \left(3\pi^2 Y_\mathrm{e}\right)^{2/3} e^{\frac{2}{3}\ln n_b}, \quad (8.66)$$

$$\mu_e \approx E_{F_e} = \hbar c \left(3\pi^2 Y_\mathrm{e}\right)^{1/3} e^{\frac{1}{3}\ln n_b}, \quad (8.67)$$

so we will have

$$\frac{\partial \mu_\Delta}{\partial \ln n_b} \approx \frac{2}{3}\left(E_{F_n} - E_{F_p}\right) - \frac{1}{3}E_{F_e}, \tag{8.68}$$

$$\frac{\partial^2 \mu_\Delta}{\partial \ln n_b^2} \approx \frac{4}{9}\left(E_{F_n} - E_{F_p}\right) - \frac{1}{9}E_{F_e}. \tag{8.69}$$

Comparing eqs. (8.62) and (8.63) and eqs. (8.68) and (8.69), we see that if the baryonic Fermi energies dominate then $|\Delta\Gamma_1/\Delta\Gamma_2| \approx |3(\Gamma-1)/2|$, and if the electron Fermi energy dominates then $|\Delta\Gamma_1/\Delta\Gamma_2| \approx |3(\Gamma-1)|$. In the relevant regions (i.e. at densities of order n_{sat}) the equation of state will be stiff, with $\Gamma \geq 3$ typical, so the $\Delta\Gamma_1$ term dominates in both cases, and the fluid will soften when moved out of equilibrium. Using the more realistic BSk density functionals[62] instead of a Fermi gas model we also obtain a negative sign at the densities of interest.

Having examined some simpler systems to determine a possible source for the frequency difference presented above, we can also look to see whether the softening effect is visible in the full merger simulation. The naïve approach would be to measure the moment of inertia of the whole system in each case. However, after merger the moment of inertia is dominated by low density matter far from the remnant, so we need some local measure that we can apply only to the remnant.

8.2.3 Comparing matter distribution

Starting from the Newtonian definition of the moment of inertia for a mass distribution along the z-axis

$$I_z(\rho) = \iiint_V \rho(x,y,z)(x^2+y^2)\mathrm{d}V \tag{8.70}$$

we substitute in the proper mass density ρW, and correct the volume element to obtain

$$I^z = \iiint_V \rho W(x^1x^1 + x^2x^2)\sqrt{\gamma}\mathrm{d}^3x. \tag{8.71}$$

To better see the differences between the remnants we will use a density cutoff to mask the effects of low density matter at large radius, and to determine a relevant density for the cutoff we will look at how the contribution to the moment of inertia varies in the x-y plane of the two stars. To do this we calculate the infinitesimal contribution to the total moment of inertia of the loop at radius r, $I^{\text{loop}}(r)$, then integrate radially outwards (to obtain $I^{\text{disc}}(r)$) to find the density at which the difference between the two simulations settles down.

In fig. 8.3 we see $I^{\text{loop}}(r)$ for the in- and out-of-equilibrium simulations 5 ms after merger, the angle averaged density, and the difference between $I^{\text{disc}}(r)$ for the two simulations.

From this plot we see that the difference between the two moments of inertia has settled down by $\rho \sim 10^{-2} - 10^{-3}\rho_{\rm sat}$. Ignoring densities below this cutoff, we can then apply eq. (8.71) to the whole domain, the results of which we plot in fig. 8.4. This shows that when considering only the densities above $\rho_{\rm cutoff} \sim 10^{-2} - 10^{-3}\rho_{\rm sat}$, the moment of inertia of the remnant is a few percent higher in the out-of-equilibrium simulation, which would intuitively lead to a few percent decrease in the frequency at which the remnant rotates, which is consistent with what we observe in the simulations, as seen in fig. 8.2.

The simulations here show that the phase of the gravitational wave signal is sufficiently sensitive to models of weak reactions in next generation gravitational wave observatories such as the Einstein Telescope. The general impact of reactions on the stiffness of the effective equation of state and hence their impact on the moment of inertia strongly suggests the results are broadly independent of the details of the bulk properties. High accuracy parameter estimation of, for example, the equation of state, via gravitational waves will therefore require more accuracy in the models of weak reactions, in addition to the bulk equation of state.

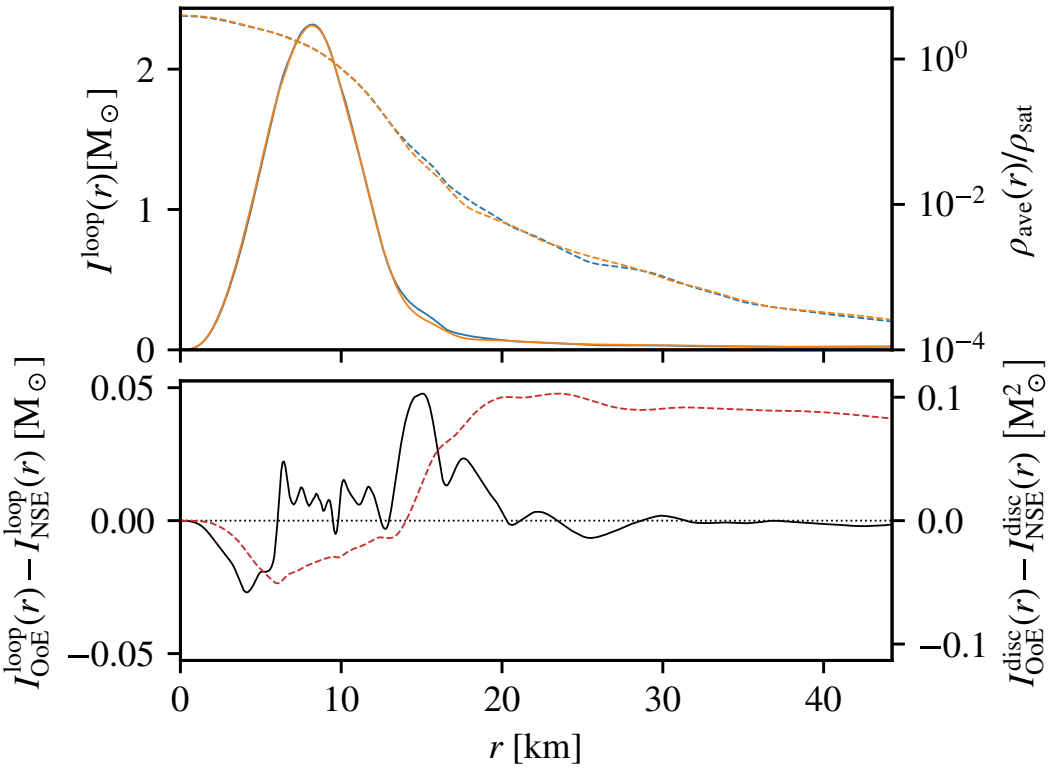

FIGURE 8.3: Upper panel: infinitesimal contribution to the total moment of inertia along the loop at radius r in the x-y plane, $I^{\rm loop}(r)$, (solid lines, left scale), and angle averaged density at radius r in the x-y plane, $\rho_{\rm ave}$, with respect to nuclear saturation density, $\rho_{\rm sat}$, (dashed lines, right scale) from slow- and fast-reaction-limit simulations (blue and orange curves respectively) taken 5 ms after merger. Lower panel: difference between $I^{\rm loop}(r)$ for slow- and fast-limit simulations (labelled OoE and NSE respectively) at each at radius r (solid black line, left scale) and difference between $I^{\rm disc}(r)$ for each simulation (dashed red line, right scale).

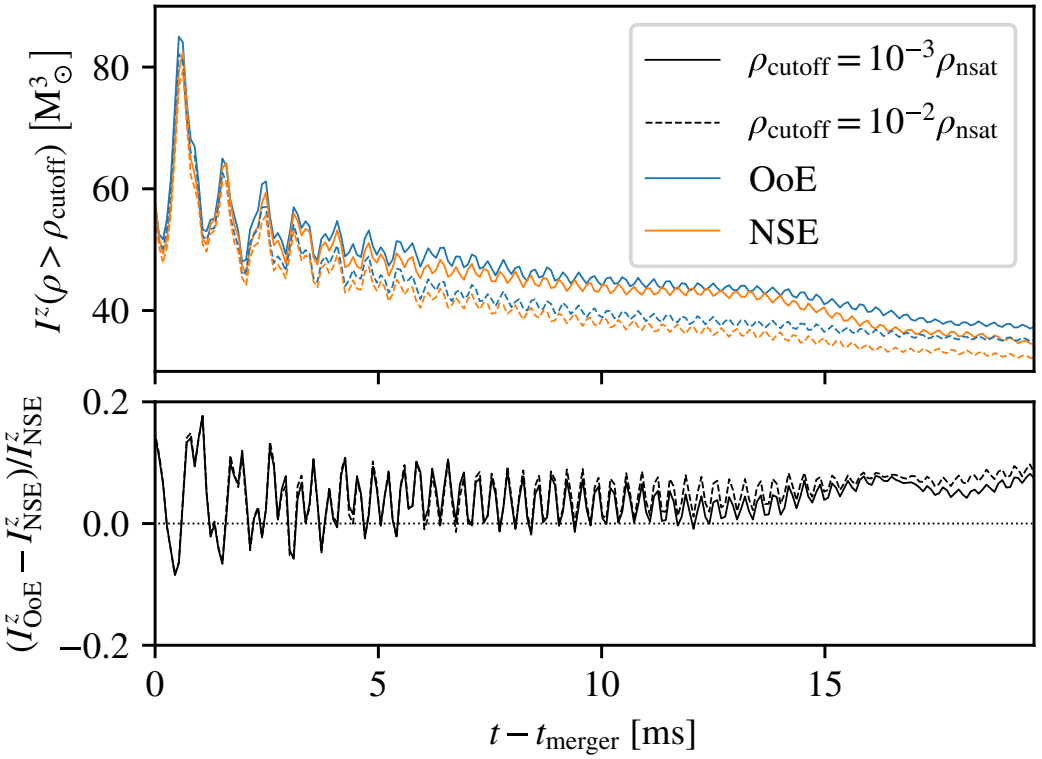

FIGURE 8.4: Upper panel: Evolution of the total moment of inertia (see eq. (8.71)) ignoring matter with rest mass density below $\rho_{\rm cutoff}$ for slow- and fast-reaction limit simulations (labelled OoE and NSE, solid and dotted lines, respectively). Lower panel: Relative difference between results for slow- and fast-limit simulations.

Chapter 9

Equilibrium at Finite Temperature

In the previous chapter we discussed equilibrium at zero temperature. Here, we explore the effects of relaxing that assumption. We start by summarising the work of [12] discussing how temperature affects the reactions in neutrino transparent matter. Next, we move on to conditions where neutrino effects become significant, then we briefly discuss the effects of reactions beyond the standard Urca processes. Finally, we present some unpublished work detailing the use of ray tracing to determine regions where neutrinos might be trapped[1].

9.1 Relaxing the Fermi Surface Approximation

As we discussed in section 8.1, when at zero temperature the momenta of the particles taking part in the reactions are assumed to be at their respective Fermi surfaces [3]. At low densities the Fermi momenta p_F of the three involved particles (neglecting the neutrinos) satisfy the relation

$$p_{F_\mathrm{n}} > p_{F_\mathrm{p}} + p_{F_\mathrm{e}} \,. \tag{9.1}$$

This blocks the direct Urca processes as there is no way to balance the momenta, requiring the introduction of the spectator nucleon in the modified Urca processes (see section 8.1). However, as the density increases, $p_{F_\mathrm{p}} + p_{F_\mathrm{e}}$ increases faster than p_{F_n}, and some equations of state allow the direct processes to proceed above a threshold density at which $p_{F_\mathrm{n}} = p_{F_\mathrm{p}} + p_{F_\mathrm{e}}$ [12]. For cold matter, this condition typically requires a proton fraction of order 10% (in the case of the APR model we consider here, the direct Urca threshold is reached at around $5\rho_\mathrm{nuc}$, beyond the densities reached in our simulation).

[1] Sections 9.1 to 9.3 have been published as part of Hammond et al. [68].

By relaxing the Fermi surface approximation at finite temperature for the direct processes, one finds that instead of being forbidden, they are Boltzmann suppressed by a factor depending on the single particle free energy γ_i, defined by

$$\gamma_i(p) = E_i(p) - \mu_i = E_i(p) - E_{F_i}, \tag{9.2}$$

where p is the particle momentum (not to be confused with pressure). $\gamma_{\mathrm{i}}(p)$ therefore represents the energy difference between a particle on the Fermi surface and a particle with momentum p. This will result in Boltzmann suppression of the rates of electron capture and neutron decay by a factor of $\exp\left(-\left|\gamma_{\mathrm{i}}\right|/T\right)$ where each reaction will be dominated by a particular γ_{i}. For neutron decay, the dominant factor is finding a hole in the electron Fermi sea below $p_{F_{\mathrm{e}}}$ by an amount in the region of $\gamma_{\mathrm{e}} = 20 - 25\,\mathrm{MeV}$, whereas for electron capture the energy mismatch is lessened by the anti-aligning of the neutrino produced with the resultant neutron, reducing the increase of momentum needed on the initial proton to around $\gamma_{\mathrm{p}} = 10 - 15\,\mathrm{MeV}$[12]. The upshot of this is that the reactions are no longer in balance when eq. (8.12) holds. In fact, as the direct electron capture rate is suppressed to a lesser extent than the neutron decay rate, there will be a net production of neutrons when the cold equilibrium condition is satisfied. To account for this, one may introduce a chemical potential offset μ_δ, such that the fluid reaches its actual equilibrium when[12]

$$\mu_{\mathrm{n}} = \mu_{\mathrm{p}} + \mu_{\mathrm{e}} + \mu_\delta. \tag{9.3}$$

The offset is determined by balancing the rates of the Urca processes, again setting $\Gamma_{\mathrm{ec}} = \Gamma_{\mathrm{nd}}$ but now for a finite temperature. This then allows us to work out the matter composition assuming that the temperature is held fixed.

Alford et al.[12] (see also the recent discussion in[15]) demonstrate that, for temperatures of order 10 MeV the required offset can be as large as $\mu_\delta = 20 - 25\,\mathrm{MeV}$. We get an immediate idea of the impact of this, by comparing to the results for μ_Δ from fig. 8.1. Clearly, there are large regions in the simulation where the suggested effect would be significant, bringing the matter closer to a de facto equilibrium than one might have expected. The effect becomes less important after the merger as the thermal hotspots drive outflows and significant regions are driven much further ($\gtrsim 100\,\mathrm{MeV}$) from the cold beta-equilibrium.

Using the results for μ_δ from Alford et al.[12], we may estimate the degree to which the shift of equilibrium causes the equation of state to soften in equilibriated matter. This provides a useful insight into the level at which the effect may impact on, for example, the gravitational-wave signal. In fig. 9.1 we plot the adiabatic index Γ for the APR equation of state and matter in "cold" equilibrium ($\mu_\Delta = 0$) as well as matter where the suggested temperature effects are taken into account (setting $\mu_\Delta = \mu_\delta$ and using the results from figure 4 in[12]). We see that there is a general trend towards lower

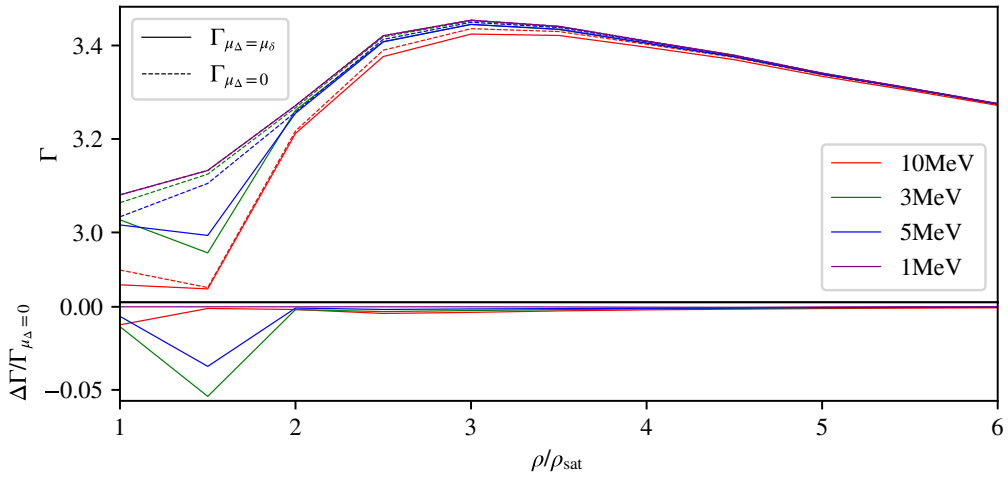

FIGURE 9.1: Illustrating the impact of the warm equilibrium on the stiffness of the equation of state. Dashed lines assume the "cold" β-equilibrium, while solid lines use the "warm" equilibrium prescription with μ_δ take from Figure 4 in [12]. The results show that the effect tends to soften the equation of state by up to 5% at densities up to $2\rho_{\rm nuc}$, but also that the impact is much less pronounced at higher temperatures.

values of Γ as the temperature increases, and the softening effect due to composition changes is most pronounced at intermediate temperatures, peaking at around the 5% level for matter in the $1-2\rho_{\rm nuc}$ density range. While, at this point, we cannot quantify the extent to which this degree of softening impacts on observables, like the frequency of the post-merger oscillations, it is clear that this is a question worth returning to in future work (which would require an equation of state accounting for the temperature dependent μ_δ). Finally, it is worth noting that the softening associated with the warm equilibrium is much less pronounced at a temperature of 10 MeV (and above). This is likely due to the general softening associated with the thermal pressure, an effect that is evident in the top panel of fig. 9.1, dominating at higher temperatures.

9.2 Neutrinos

The equilibrium relations given in eqs. (8.12) and (9.3) both assume that the fluid is transparent to neutrinos, which can then only appear as products in the allowed reactions. However, at sufficiently high temperatures the neutrino mean free path is expected to become short enough that they begin to react with the fluid matter at a meaningful rate, participating in the inverses of the 6 aforementioned Urca reactions. Now that the processes are allowed to proceed in both directions, it follows from eq. (8.7) that all 6 pairs of reactions have the equilibrium condition

$$\mu_{\rm n} + \mu_{\nu_{\rm e}} = \mu_{\rm p} + \mu_{\rm e} \tag{9.4}$$

where $\mu_{\nu_e} = -\mu_{\overline{\nu}_e}$ is the electron-neutrino chemical potential.

Under these conditions, the relaxation timescale is expected to be (see figure 7 in [9]) of order $10^{-8} - 10^{-10}$ s (faster than any current numerical simulation[2] is likely to be able to resolve, but not out of reach of future simulations). This, for example, leads to a weakening of the bulk viscosity (an effect we discuss in chapter 10) by several orders of magnitude [9]. As this has an obvious impact on the matter dynamics—the new equilibrium that the matter will evolve towards is rather different from the cold and warm cases—we clearly need to consider the neutrinos. Unfortunately, this is problematic. As the neutrinos were not included in our simulation (or indeed the equation of state in the first place!), we cannot directly measure the neutrino chemical potential. We have to resort to approximations. For example, we get some idea of the likely magnitude of the neutrino chemical potential by assuming $\mu_\nu \sim \mu_\Delta$ and then considering the results shown in fig. 8.1. Still, if we want to do better then we need to account for the neutrinos in the matter description. Similarly, we cannot reliably quantify the rate at which the neutrinos are involved in reactions to determine the regions of the simulation for which eq. (9.4) is the appropriate statement of β-equilibrium. However, we can estimate these regions by post-processing (as in [53]) the opacity κ (which is related to the mean free path ℓ through $\ell = 1/\kappa$).

Given a path λ of length L, the probability of a neutrino being transmitted along that path is

$$P_{\mathrm{T}}(\lambda) = \exp\left(-\kappa_{\mathrm{ab}} L\right), \tag{9.5}$$

where κ_{ab} is the absorption opacity. Scattering increases the effective length of the path taken through the fluid by the neutrinos, which we can account for by using the effective opacity κ_* given by

$$\kappa_* = \sqrt{\kappa_{\mathrm{ab}}\left(\kappa_{\mathrm{ab}} + \kappa_{\mathrm{sc}}\right)} \tag{9.6}$$

where κ_{sc} is the scattering opacity [118].

Hence, we can estimate the regions where neutrino absorption will be significant by examining the magnitude of $\kappa_* L$. In regions where $\kappa_* L \gg 1$, the probability of transmission is very low, and thus neutrinos are available to partake in reactions, whereas in the opposite limit they are likely to escape freely. The question then is, what should we take to be the relevant length scales? Pragmatically, in merger simulations there are two pertinent length scales: the grid spacing Δx, which for the simulation under discussion is $\Delta x \approx 400\,\mathrm{m}$, and the size of the region containing hot matter r_{hot}, which here is $\sim 30 - 50\,\mathrm{km} \approx 100\Delta x$.

[2] The timestep of a grid with spacing of $\Delta x = 10\,\mathrm{m}$ would be on the order of $\Delta t = 10^{-8}\,\mathrm{s}$

In regions where $\kappa_* \Delta x \gg 1$ the neutrinos are unlikely to escape the computational cell in which they were emitted, so for our purposes they are definitely trapped and available for reactions. Similarly, in regions where $\kappa_* r_{\mathrm{hot}} \ll 1 \implies \kappa_* \Delta x \ll 1/100$ the neutrinos are likely to escape the simulation without reacting, and so these regions should evolve towards the cold/warm equilibrium conditions on some relatively fast timescale. In the intermediate regime, the neutrinos emitted in one place are not likely to be absorbed locally (i.e. within the same computational cell), but they are also unlikely to escape the simulation completely, hence it is difficult to make a definite statement one way or the other—without directly simulating the neutrinos. These arguments show why the neutrino treatment is problematic. The logic that the matter is either neutrino transparent or not is too simplistic to describe the conditions throughout much of the simulation domain. Nevertheless, as one has to start somewhere, let us consider where the estimates we have suggested take us.

The opacity of the fluid to neutrinos depends on the equation of state parameters $(\rho, T, Y_{\mathrm{e}})$, the neutrino energy E_ν, and the neutrino species x. We use NuLib[99] to calculate $\kappa_* \left(\rho, T, Y_{\mathrm{e}}, E_\nu, x\right)$ for the two most relevant species, the electron neutrino/anti-neutrino. In order to obtain an estimate for the appropriate neutrino energies, we assume that there are sufficient neutrinos available to be in chemical equilibrium, and that those neutrinos will be in thermal equilibrium with the fluid. Endrizzi et al.[53] have shown that this assumption is good enough to determine equilibrium surfaces, which depend strongly on the absorption, and thus this approximation should work well for the qualitative analysis here. The neutrinos then follow an isotropic Fermi-Dirac distribution

$$f_\nu \left(E_\nu\right) = \left[\exp\left(\frac{E_\nu - \mu_\nu}{T}\right) + 1\right]^{-1}, \tag{9.7}$$

where $\mu_{\nu_{\mathrm{e}}} = -\mu_{\overline{\nu}_{\mathrm{e}}} = \mu_{\mathrm{p}} + \mu_{\mathrm{e}} - \mu_{\mathrm{n}}$. We then calculate an energy averaged opacity $\tilde{\kappa}_*$ for each species through (as we focus on particle number transport[53])

$$\tilde{\kappa}_* \left(\rho, T, Y_{\mathrm{e}}, x\right) = \frac{\displaystyle\int_0^\infty f_\nu \left(E_\nu\right) \kappa_* \left(\rho, T, Y_{\mathrm{e}}, E_\nu, x\right) E_\nu^2 \mathrm{d}E_\nu}{\displaystyle\int_0^\infty f_\nu \left(E_\nu\right) E_\nu^2 \mathrm{d}E_\nu}. \tag{9.8}$$

The results in fig. 9.2 show that $\tilde{\kappa}_* \Delta x$ is small in the dense core regions up to merger, and although some of the core matter in the remnant has been heated enough that absorption on the scale of grid cells has become significant, there remains a transparent central region. The hotspots mentioned in chapter 7 show large potential for absorption of both species of neutrino, as expected. The outflowing matter visible in the +5 ms panel is mostly in the intermediate regime, however it does highlight the difference between the values of $\tilde{\kappa}_*$ for the two different neutrino species, with the electron neutrinos being subjected to a stronger degree of absorption than their antiparticle. Hence we are likely

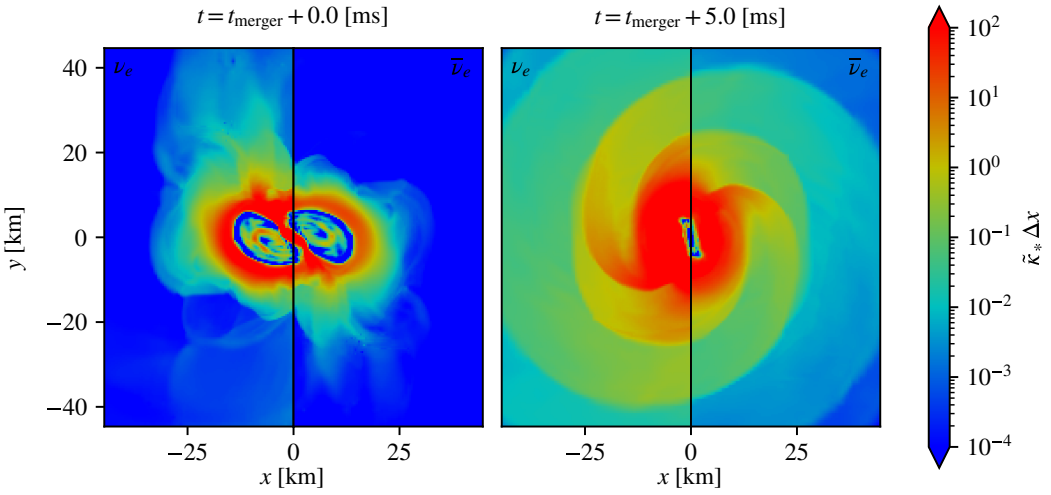

FIGURE 9.2: Energy averaged electron neutrino and anti-neutrino effective opacities $\tilde{\kappa}_*$ multiplied by grid cell spacing $\Delta x = 400\,\mathrm{m}$ for a simulation of the merger of two $1.4\,M_\odot$ neutron stars using the APR equation of state from Schneider et al.[122]. The left panel coincides with the merger time, and the right panel is 5.0 ms post merger. The neutrino opacities were calculated using NuLib[99].

to see regions where the inverses of the Urca processes are suppressed to different degrees, further complicating the calculation of the equilibrium condition.

9.3 Equilibriation via Strong Interactions

So far, we considered the electron to be the only negatively charged particle (muons were ignored for simplicity, but partake in similar reactions to the electrons), meaning that equilibration relied on the weak interaction. However, the situation may change as the temperature ramps up. For example, one may argue that we should account for the presence of thermal pions[58]. A small population of pions could drastically change the story. At high temperatures neutrons may decay to protons and pions on the strong interaction timescale (which at 10^{-23} s is instantaneous for all practical purposes). The system would then reach an equilibrium where

$$\mu_{\rm n} - \mu_{\rm p} = \mu_\pi \tag{9.9}$$

with the pions subsequently equilibrating with the electrons on the weak interaction timescale (through pion decay). In the first estimates of this effect, Fore and Reddy[58] show that pion-equilibriated matter is more proton rich than the weak equilibria we have discussed. As a result, the equation of state may soften by as much at 10-15%, which could have a significant impact on the merger dynamics. In addition, the presence of the thermal pions increases the heat capacity, leading to the matter cooling which may self-regulate the process. These are interesting ideas, but in order to test them

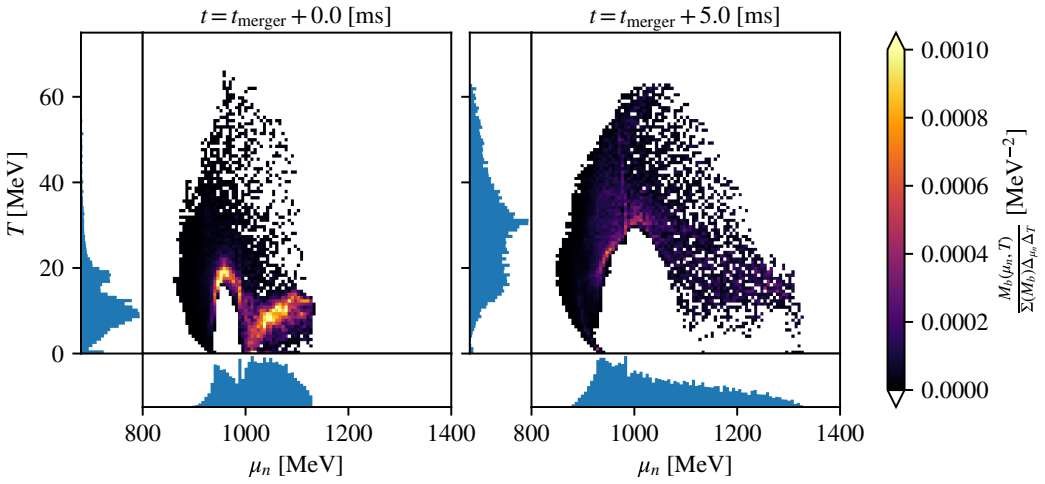

FIGURE 9.3: Distribution of baryon mass $M_{\rm b}$ in the $\mu_{\rm n}$-T plane for a simulation of the merger of two $1.4\,M_\odot$ neutron stars using the APR equation of state from [122]. The left panel coincides with the merger time, and the right panel is 5.0 ms post merger. The results suggest that thermal pions, which may play a role at temperature above $T \gtrsim 25\,\text{MeV}$ [58], may impact on a significant fraction of the simulated matter.

with simulations we would need an equation of state that accounts for the thermal pions. Work in this direction is ongoing, however in the absence of such a model we can estimate if (and where) the thermal pions are likely to come into play.

Drawing on the estimates from Fore and Reddy [58], which suggest that the effect becomes significant when $T \gtrsim 25\,\text{MeV}$, we may consider the results in fig. 9.3, which shows the distribution of matter in the $\mu_{\rm n}$-T phase space, as indicative. We see that after merger the outflows from the core lead to a substantial migration of matter to higher chemical potentials, and the distribution of the matter has shifted to higher temperature, with most of the matter now above $T \gtrsim 15\,\text{MeV}$ – the spacial regions where this become important will be discussed below. Evidently, the temperature reaches the level where the pions may play a role throughout much of the matter. This suggests that we should seriously consider the role of the thermal pions, which inevitably involves moving towards a more complex matter model. Work in this direction is in progress.

As an attempt to summarize the discussion of equilibrium we provide the schematic illustration in fig. 9.4. The figure identifies regions where the different equilibrium conditions discussed in this chapter apply in our simulation (at merger and 5 ms later). We distinguish regions with "cold" matter ($T < 1\,\text{MeV}$) where the condition (8.12) applies, "warm" ($T > 1\,\text{MeV}$ and $\tilde{\kappa}_* r_{\rm hot} < 1$, the transparent regime, for both neutrino species) where we need to account for the imbalance of the nuclear reaction by adding μ_δ as in (9.3), a "warm/hot"region which involves matter with $T > 1\,\text{MeV}$ but with $\Delta x / r_{\rm hot} < \tilde{\kappa}_* \Delta x < 1$ (neither locally trapped, nor transparent) for at least one of the neutrino species, a "hot" region where $\tilde{\kappa}_* \Delta x > 1$ (locally trapped) for both neutrino species, and finally a "strong" region with matter at $T > 25\,\text{MeV}$ where one might

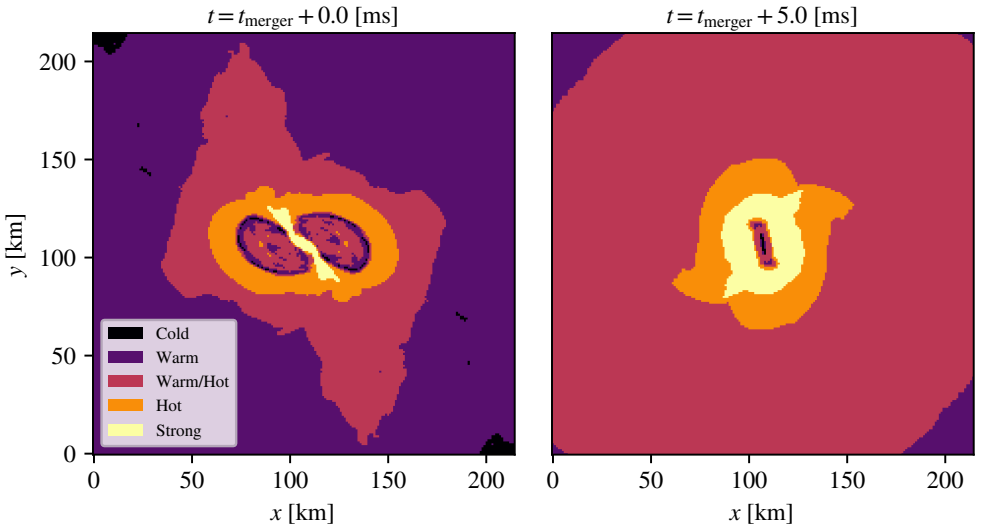

FIGURE 9.4: Schematic of where the different equilibrium conditions discussed in this chapter are expected to occur in a simulation of the merger of two 1.4 $M_\odot$ neutron stars using the APR equation of state from [122]. Left panel coincides with merger time, and right panel is 5.0 ms post-merger. The conditions on the matter are labelled as follows: "Cold" denotes matter with $T < 1$ MeV where the standard cold β equilibrium condition eq. (8.12) is relevant, "Warm" denotes matter with $T > 1$ MeV and $\tilde{\kappa}_* r_{\text{hot}} < 1$ for both neutrino species where the warm β equilibrium condition in eq. (9.3) is relevant, "Hot" denotes matter where $\tilde{\kappa}_* \Delta x > 1$ for both neutrino species, thus the hot β equilibrium condition in eq. (9.4) is relevant, "Warm/Hot" denotes matter with $T > 1$ MeV but with $\Delta x / r_{\text{hot}} < \tilde{\kappa}_* \Delta x < 1$ for at least one of the neutrino species, where we cannot distinguish cleanly between hot and warm equilibrium, and "Strong" denotes matter with $T > 25$ MeV where pions may be relevant.

anticipate thermal pions to play a role. As these regions evolve with time, a live identification of the appropriate local conditions is clearly far from trivial.

9.4 Neutrino Trapping

In section 9.2 we estimated the regions where neutrino interactions would be significant by comparing the mean free path to some relevant length scale (either the simulation grid size or the size of the hot region). We could take this a step further and estimate the regions where neutrinos are trapped, i.e. the region from which any emitted neutrinos are unlikely to escape to infinity.

9.4.1 The optical depth

Trapping regions are calculated through the optical depth[3] τ [53], which is defined for a given opacity κ as

$$\tau_L = \int_L \kappa(x) \sqrt{\gamma_{ij} \mathrm{d}x^i \mathrm{d}x^j}, \tag{9.10}$$

where L denotes the path taken by the neutrino through the fluid, and γ_{ij} is the local spacial metric. The quantity τ_L counts the expected number of interactions associated with κ that a neutrino will experience as it travels along L. Therefore, neutrinos emitted from a point $\boldsymbol{x}$ are in a trapped region if $\tau_L \geq 1$ where L runs from $\boldsymbol{x}$ to any point from which the neutrinos escape freely $\boldsymbol{x}_b$.

As there are many paths a neutrino could take through the fluid in order to escape, there are many values of τ for any given $\boldsymbol{x}$, but in order to define the trapped regions, we need a unique value of τ at each point. Therefore, we consider the fact that, while neutrinos may be produced isotropically, neutrinos that move into more optically dense regions are more likely to scatter (and thus change direction) than neutrinos that are moving into optically thinner regions, hence there will be a preferential movement towards regions with smaller opacities [53]. This means that we can obtain a good estimate for the optical depth at $\boldsymbol{x}$ by calculating τ for a number of paths, and defining $\tau(\boldsymbol{x})$ as the minimum of these.

The next step is to define a procedure for sampling the possible paths the neutrinos could take through the matter, and the boundary from which they can escape. Typically [53] the six cardinal directions ($\pm x$,$\pm y$,$\pm z$), twelve "face" diagonals (towards the four corners of the plane perpendicular to each axis), and eight "space" diagonals (from the centre to the corners of a cube) are used. Symmetries may be available to cut down on the number of directions to sample. While simple to implement, it is possible that using this arrangement may introduce biases based on the coordinate system used, and leaves little room for increasing or decreasing resolution. A better method ("fairer" but significantly slower) may be to produce an (almost) uniform sampling on the sphere with the desired number of points, which can be randomly oriented for each spacial sample point.

Finally, with a value for $\tau(\boldsymbol{x})$ calculable at all points, we can define the surface of last scattering for each neutrino species ν_n as the surface for which

$$\tau_{\mathrm{diff}}(\boldsymbol{x}) = \min_{\{L | L:\boldsymbol{x}\to\boldsymbol{x}_b\}} \int_L \kappa_{\mathrm{diff},\nu_n}(x) \sqrt{\gamma_{ij} \mathrm{d}x^i \mathrm{d}x^j} = 1, \tag{9.11}$$

[3] Not to be confused with the conserved internal energy density from the Valencia formulation.

and the per species trapping region as the region contained within the surface such that

$$\tau_{\mathrm{eq}}(\boldsymbol{x}) = \min_{\{L|L:\boldsymbol{x}\to\boldsymbol{x}_b\}} \int_L \kappa_{\mathrm{eq},\nu_n}(x) \sqrt{\gamma_{ij}\mathrm{d}x^i\mathrm{d}x^j} = 1. \tag{9.12}$$

While this measure is useful when the goal is to determine what the detected neutrino signal might look like from far away, it is less useful for our purposes. Determining if neutrino reactions are likely to be significant at a given point is obviously a non-local problem; neutrinos emitted in one part of the star may be absorbed in another, however this at best only loosely related to the likelihood of neutrinos to escape to infinity from that point. Consider the cold, dense core of a neutron star merger remnant surrounded by a much hotter region (as we saw in the previous section). Neutrinos are unlikely to be absorbed by the cold matter in the core, as the opacity is driven mainly by temperature, and only becomes significant above $\sim 10\,\mathrm{MeV}$. In contrast, the surrounding matter is well above this threshold and the neutrinos will be unlikely to penetrate through. Therefore, by the above measure, neutrinos in the core are considered trapped, but they will not be able to take part in reactions to any significant degree.

9.4.2 Ray tracing

As a step in the direction of being able to identify those regions where neutrinos play a significant role, we have performed some basic ray tracing simulations on data taken from our merger simulations (in particular the APR simulation). We use a single timestep from the simulation to give data for the opacities, and an approximation of the direct and modified Urca rates in equilibrium from Alford et al. 2018 [12]. We chose to use the timeslice that contained the highest temperature out of those available (shortly after the two stars first touch), and we restricted our domain to the sphere approximately two stellar diameters in radius ($\sim 18\,\mathrm{km}$), centered on the origin of the simulation (the barycentre of the system).

At each point within the domain we "emit" a number of "rays" (a tunable resolution parameter), each in its own random direction. Each ray has a number associated with it that starts off proportional to the calculated emission rate at its origin, which we consider as the number of neutrinos associated with that ray. As each ray is propagated (in a straight coordinate line, not a geodesic, and also ignoring any scattering effects) we calculate how many of its associated neutrinos are absorbed or transmitted, using eqs. (9.5) and (9.8)[4], and update both the number of neutrinos still associated with the ray, and how many have been absorbed close to each grid point in the fluid. The ray is allowed to propagate until it leaves the domain.

[4] These include the effective lengthening of the mean free path through scattering, and the assumption that the neutrinos are always in thermal equilibrium with the matter. In addition we look only at electron neutrinos.

In the following figures we used 100 rays for each point in the fluid. As we do not impose any symmetries on the neutrinos, we get a good idea of how uniformly the space has been sampled by how well the symmetries in the fluid data are represented. As this simulation was devised only as a qualitative check on how "good" our previous estimations were, we verify this by eye.

In fig. 9.5 we plot each point in space against the number of neutrinos emitted at that point, and the number of neutrinos absorbed. We then colour each point using the same conditions as for fig. 9.4 (see caption for details). We see that both for the points labelled "Hot" and "Strong" the emission and absorption rates are almost unity, which is indicative of trapping (as would be expected in the hotter regions). We also see that for the points labelled "Cold" the ratio of absorption to emission peaks at around 10^{-2}, so we can say in these regions that the neutrino reactions are not significant. The most interesting features of the plot are the regions where we previously labelled the points "Warm", indicating that we would not expect significant rates of neutrino absorption (i.e. ratios much lower than unity), but in this simulation we see ratios of absorption to emission of over 10^2, indicating that those reactions which include neutrinos on the left hand side (i.e. as an ingredient) are proceeding at a much higher rate than those which include neutrinos as a product.

Looking at fig. 9.6, where we plot the spacial distribution of the ratio of absorptions to emissions in the xy-plane, we see that in the cold cores of the stars there are significantly more emissions than absorptions, as we previously predicted. The hotter regions surrounding the cores have a ratio close to unity, which suggests that the neutrinos are locally trapped in these areas, again as predicted above. There are some red areas (indicating more absorptions than emissions) in the $\pm 10 - 18\,\mathrm{km}$ range which may not be trustworthy as they are near boundary of the domain, however, we do also see a band of red surrounding the cold cores, likely a result of neutrinos emitted in the core being absorbed by the warm matter that forms the boundary between the hot and cold matter.

While the results here do not help us make progress in determining which is the correct equilibrium condition to use at each point live in a merger simulation, they do indicate, again, that the picture is very complex, and dependent on non-local behaviour. Clearly the issue of the impact of neutrinos on not only the composition of the fluid, but on the very definition of equilibrium in a local sense needs further investigation. Correctly representing these effects in a simulation would require a full reaction network coupled to a radiation transport scheme that is valid far from thermal and β-equilibrium, which currently does not exist.

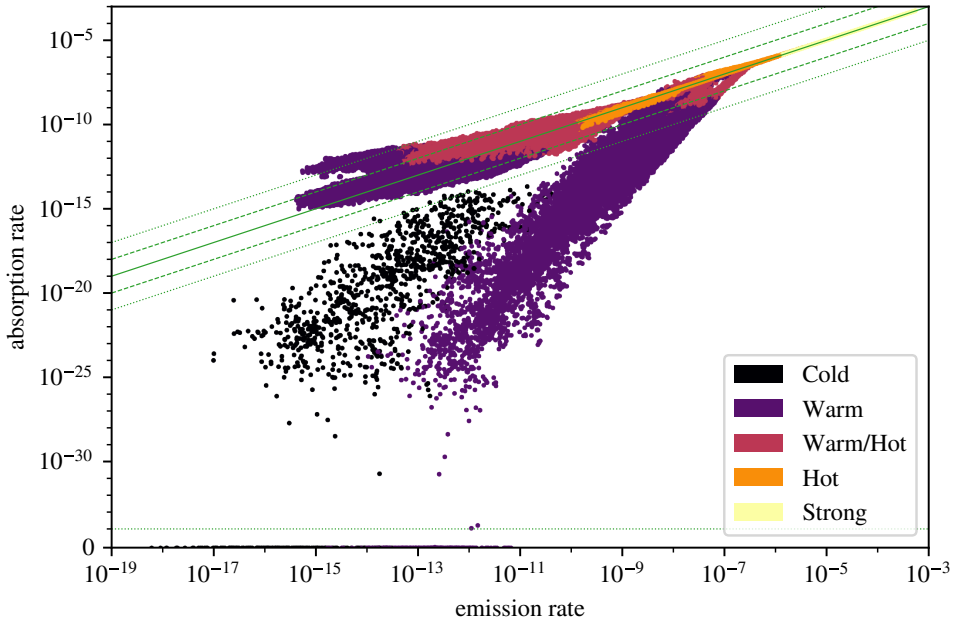

FIGURE 9.5: Absorption and emission rates of all points within the ray tracing simulation. Colours are based on the equilibrium classification in section 9.3. Emission rates are based upon Alford and Harris[12], and absorption opacities were calculated with NuLib[99]. Fluid and spacetime data taken from a simulation of two $1.4\,M_\odot$ at the point where the fluid temperature reaches its maximum. 100 rays were traced from each point, and their directions were randomised.

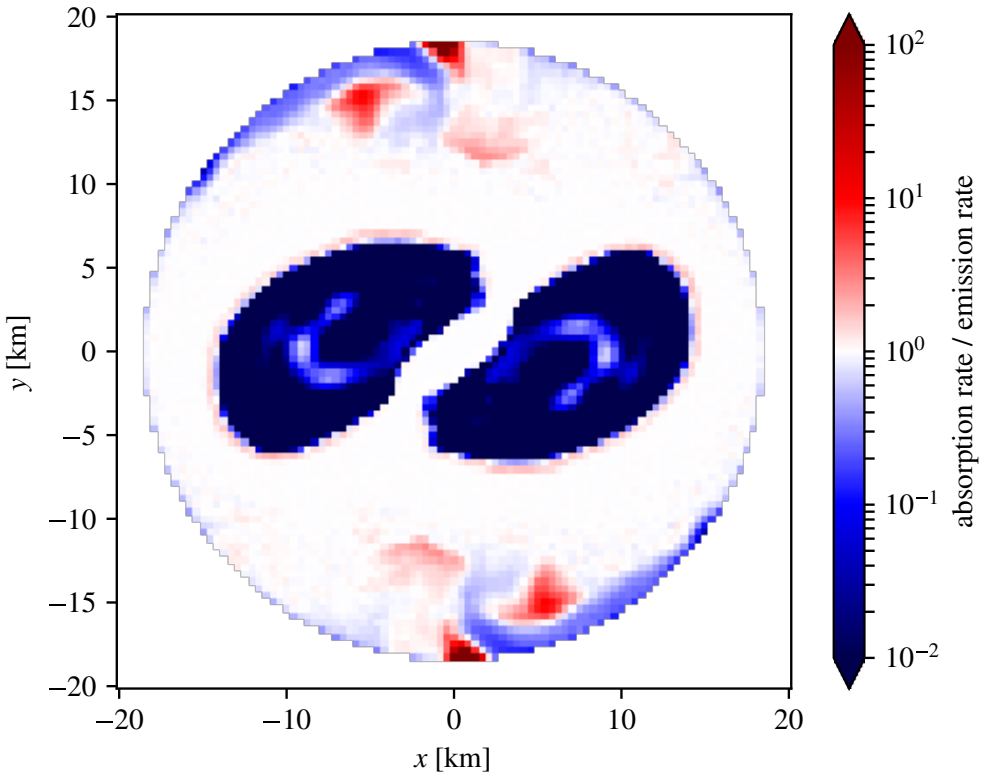

FIGURE 9.6: Ratio of calculated absorption and emission rates resulting from a ray tracing simulation. Red regions have more absorption, and blue regions vice versa. Emission rates are based upon Alford and Harris[12], and absorption opacities were calculated with NuLib[99]. Fluid and spacetime data taken from a simulation of the merger of two 1.4 $M_\odot$ neutron stars at the point where the fluid temperature reaches its maximum. 100 rays were traced from each point, and their directions were randomised.

Chapter 10

Bulk Viscosity

Having discussed the equilibrium state of neutron star matter in chapters 8 and 9, we can begin to discuss effects that emerge from the evolution towards this equilibrium. In particular we will focus on bulk viscosity[10;13;14]. As is indicated by the simulations discussed in section 8.1, in the absence of reactions the composition of the matter is pushed out of equilibrium by the dynamics of the merger. Including reactions would lead to the equilibriation of this matter on some timescale. Bulk viscosity is often thought of as a resonant phenomenon[120], which is particularly important when the dynamical motion of the fluid and the rate of relaxation towards equilibrium occur on comparable timescales. We start by discussing in which regions of the simulation the different "standard" prescriptions of bulk viscosity are valid. We go on to discuss the calculation of the bulk viscous pressure term through coefficients derived for the equation of state, and how this might be implemented in a simulation, finishing with a discussion of the issue of timescales[1].

10.1 Sub or Suprathermal?

Typical calculations of bulk viscosity are perturbative around equilibrium, and are formulated as a periodic solution for the dimensionless parameter[13]

$$\xi = \frac{\mu_\Delta}{T}. \tag{10.1}$$

The calculation of the bulk viscosity from ξ (see Alford et al.[13][2] for a detailed discussion) can be done in three regimes: a general solution and two limiting cases. First of all, the sub-thermal limit, where $\xi \ll 1$, allows an analytic solution, essentially an expansion in small deviations from equilibrium (where it may be sufficient to include

[1] Section 10.1 has been published as part of Hammond et al.[68], and sections 10.2 to 10.4 are modified versions of work published as part of Celora et al.[37].

[2] Note that Alford et al.[13] use $\mathcal{A}$ instead of ξ for μ_Δ/T.

only the linear term). Meanwhile, the general solution and the supra-thermal limit, where $\xi \gg 1$, are more complicated as they both require numerical integration of the periodic solution. Given this understanding, it is interesting to quantify the regions in the merger simulation where each assumption applies.

As an indication of this, we plot in fig. 10.1 the value of $|\mu_\Delta| / T$ (using eq. (8.13)) reached by the matter in our simulation. As far as we are aware, this is the first time this kind of information has been extracted from numerical merger data. The results paint a complex picture. While there are regions in the simulation that remain sub-thermal (where a low-temperature expansion and a perturbative analysis of bulk viscosity would suffice) there are also large regions where the matter is in the suprathermal regime. The upshot is that the analysis of the impact of bulk viscosity in a live simulation would require a careful on-the-fly identification of these regions.

We can also look at this in the μ_Δ-T plane, as shown in fig. 10.2. We see that the majority of the matter reaches the point of merger around $\mu_\Delta = 0$, but in the remnant we see that the modal μ_Δ shifts to around $\mu_\Delta = 40\,\mathrm{MeV}$. There is a strong correlation between T and μ_Δ, which is perhaps unsurprising as much of the temperature increase in the simulation is driven by compression heating, and changes in density are also able to drive changes in μ_Δ. In the left panel we see that before merger most of the matter is in the sub-thermal regime, whereas in the right panel most of the matter has crossed the $T = |\mu_\Delta|$ line to enter the supra-thermal region. While most of this matter is at high temperature, meaning that if reactions were included it would have been quickly driven to a lower μ_Δ, we see that there is some matter (corresponding to the inner core of the remnant) with $T \lesssim 5\,\mathrm{MeV}$ and $\mu_\Delta \sim 40\,\mathrm{MeV}$, meaning that it may well be necessary to consider supra-thermal bulk viscosity.

In principle, we could proceed and use the results to estimate the bulk viscosity. There are, however, a couple of issues that prevent us from taking this step. First, as stated in previous section, the calculation of μ_Δ assumes that the fluid is at $T \to 0$ (we are measuring the deviation from the cold equilibrium). As $T \neq 0$ is required for the calculation of ξ, we need to consider how the problem changes for finite temperatures. The second issue we need to acknowledge relates to the fact that the standard bulk viscosity prescription involves an estimate of the local dynamical timescale. The estimates from [10] suggest that the bulk viscosity resonance may be relevant for neutrino-transparent matter, but the outcome depends on the local conditions and the required information is not easily extracted from a live simulation, at least not in a way that does not significantly ramp up the computational cost. A more practical approach would likely involve implementing the relevant reaction network directly, but this is neither cheap nor easy. While we cannot estimate the observable effect of bulk viscosity directly, we can make some progress towards estimating under what conditions it might be relevant.

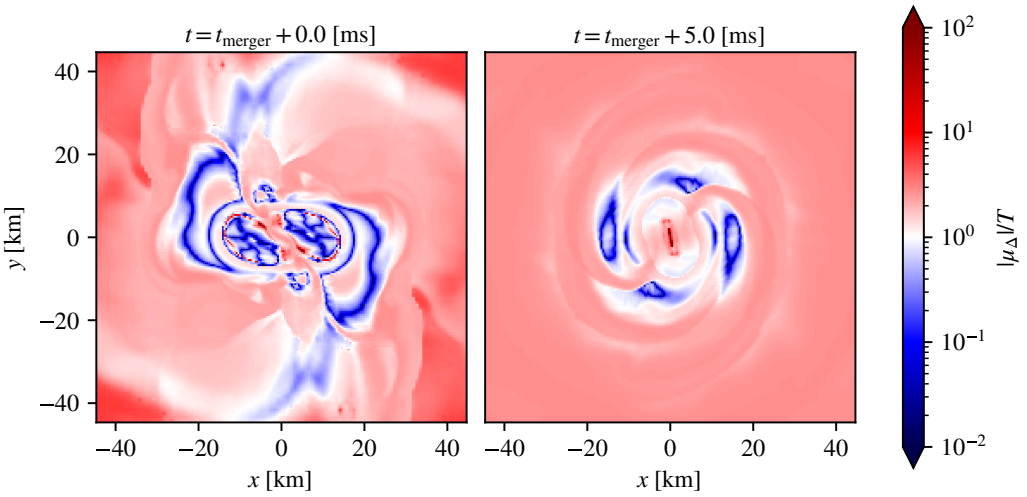

FIGURE 10.1: Bulk viscosity regime parameter $|\mu_\Delta|/T$ for the simulation of the merger of two $1.4\,M_\odot$ neutron stars using the APR equation of state from [122]. The left panel coincides with the merger time, and the right panel is 5.0 ms post merger. We see that there are regions in the simulation that remain sub-thermal (where a low-temperature expansion and a perturbative analysis of bulk viscosity would suffice), but there are also large regions where the matter is in the suprathermal regime.

10.2 Bulk Viscous Coefficients

We can reformulate the evolution equation of $Y_{\rm e}$ (eq. (7.2)) in terms of μ_Δ (noting that the exact definition of μ_Δ in regards to the different equilibrium conditions noted previously is unimportant, only that the local equilibrium condition is $\mu_\Delta = 0$) to obtain[3]

$$u^a \nabla_a \mu_\Delta = \left(\frac{\partial \mu_\Delta}{\partial Y_{\rm e}}\right)_{n_b,e} \frac{\Gamma_{\rm e}}{n_b} - n_b \mathcal{B}\theta, \tag{10.2}$$

where

$$\mathcal{B} = \left(\frac{\partial \mu_\Delta}{\partial n_b}\right)_{e,Y_{\rm e}} + \frac{p+e}{n_b}\left(\frac{\partial \mu_\Delta}{\partial e}\right)_{n_b,Y_{\rm e}}, \tag{10.3}$$

e is the total energy density, and $\theta = \nabla_a u^a$ is the fluid expansion.

Assuming we are working in the sub-thermal regime we can expand the rate around $\mu_\Delta = 0$ to give $\gamma = -\partial\Gamma_{\rm e}/\partial\mu_\Delta$ (not to be confused with the spacial metric, and following the sign convention of Alford et al. [13]). Equation (10.2) then becomes

$$u^a \nabla_a \mu_\Delta = -\mathcal{A}\mu_\Delta - n_b \mathcal{B}\theta, \tag{10.4}$$

[3] This section is based upon the author's contribution to Celora et al. [37], in which one will find a much more complete discussion of the ideas presented here.

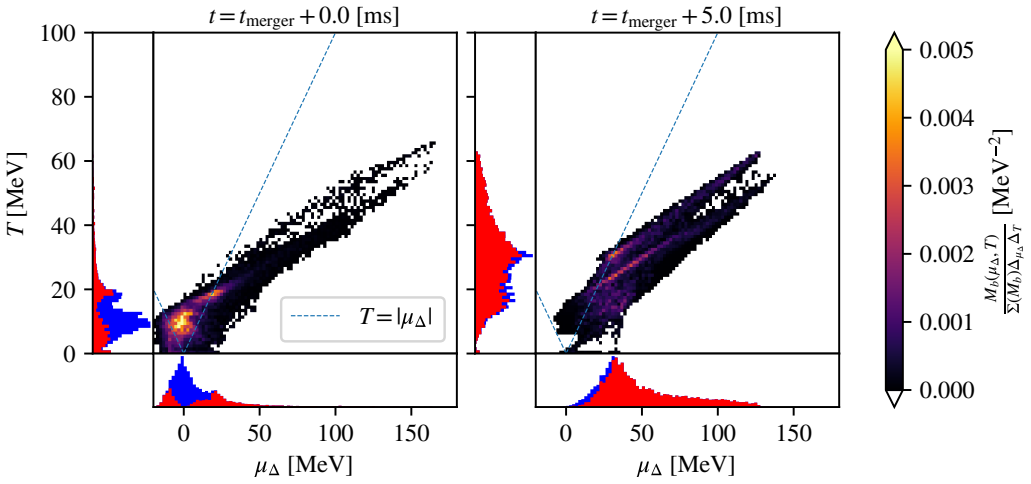

FIGURE 10.2: Distribution of baryon mass M_b in the μ_Δ-T plane for a simulation of the merger of two $1.4\,M_\odot$ neutron stars using the APR equation of state from [122], where $\mu_\Delta = \mu_\text{n} - \mu_\text{p} - \mu_\text{e}$. The left panel coincides with the merger time, and the right panel is 5.0 ms post merger. The dashed line denotes $T = |\mu_\Delta|$. Matter above this line therefore has $\mathcal{A} < 1$ and is in the sub-thermal regime, whereas matter below the line has $\mathcal{A} > 1$ and is in the supra-thermal regime. This distinction is replicated in the histograms to the side of and below each plot, with blue indicating sub-thermal, and red indicating supra-thermal.

where we have introduced the (inverse) bulk viscous timescale

$$\mathcal{A} = \frac{\gamma}{n_b}\left(\frac{\partial \mu_\Delta}{\partial Y_\text{e}}\right)_{n_b,e}. \tag{10.5}$$

We can rewrite the pressure as

$$p = p^\text{eq} + \Pi, \tag{10.6}$$

where the superscript eq denotes that the quantity should be calculated with the equilibrium composition, and the out-of-equilibrium behaviour is moved into Π, which for our purposes will be the bulk viscous pressure (see Celora et al. [37] for a more complete discussion as to why Π is the bulk viscous pressure). Following the above assumptions this term can be expressed as [37]

$$\Pi = -n_b^3\left(\left.\frac{\partial \mu_\Delta}{\partial Y_\text{e}}\right|_{n_b,e}\right)^{-1} \mathcal{B}^2 f(\hat{\omega}), \tag{10.7}$$

where $f(\hat{\omega})$ describes the resonant behaviour of the bulk viscous effect (the "standard" resonance curve, see fig. 10.3, is obtained when $f(\hat{\omega}) = \hat{\omega}/(1+\hat{\omega}^2)$), and $\hat{\omega} = \omega/\mathcal{A}$ is the relative frequency of the density oscillations compared to $\mathcal{A}$. Finally we define the

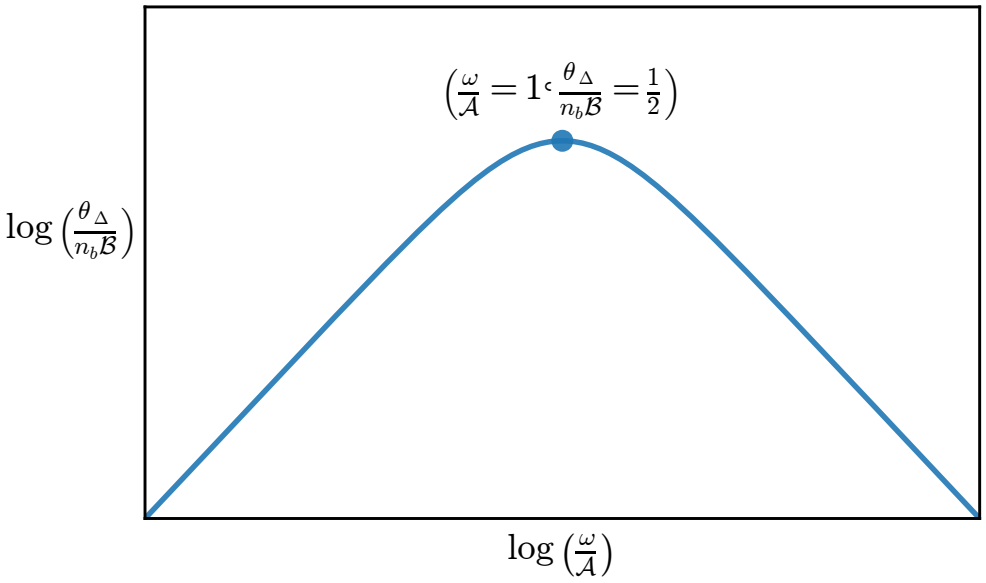

FIGURE 10.3: Resonant behaviour of bulk viscosity. $\omega/\mathcal{A}$ represents the scaled frequency of the driving oscillation, and the resultant bulk viscous pressure goes like $\mu_\Delta / n_b \mathcal{B}$.

maximum relative contribution of bulk viscosity to the pressure

$$\frac{\Pi^{\max}}{p} = -\frac{n_b^3}{2p}\left(\left.\frac{\partial \mu_\Delta}{\partial Y_e}\right|_{n_b,e}\right)^{-1} \mathcal{B}^2. \tag{10.8}$$

Ignoring finite temperature effects[9;11;12;14;15] we can, with an estimate of the rates, then calculate $\mathcal{A}$, $\mathcal{B}$, and $\Pi^{\max}$ from the equation of state, allowing us to make some estimate as to how relevant bulk viscosity is.

10.2.1 Estimating the rates

The final piece of the bulk viscous puzzle is the reaction rate Γ_e. As stated above we are working in the sub-thermal regime, and the fluid in our APR simulation never exceeded the direct Urca threshold, so to calculate Γ_e we use the out-of-equilibrium expansion of the modified Urca rates presented in appendix B of Alford and Harris[12]. Γ_e is the total rate of electron production, so it is the sum of the two neutron decay reactions (eqs. (8.3) and (8.4)), minus the two electron capture reactions (eqs. (8.5) and (8.6)), giving

$$\Gamma_e = \Gamma_{\mathrm{mU,nd(n)}} + \Gamma_{\mathrm{mU,nd(p)}} - \Gamma_{\mathrm{mU,ec(n)}} - \Gamma_{\mathrm{mU,ec(p)}}. \tag{10.9}$$

The out of equilibrium behaviour is encapsulated as a function of $\xi = \mu_\Delta / T$, given by

$$F(\xi) = -\left(\xi^4 + 10\pi^2\xi^2 + 9\pi^4\right)\mathrm{Li}_3\left(-e^\xi\right) + 12\xi\left(\xi^2 + 5\pi^2\right)\mathrm{Li}_4\left(-e^\xi\right) - 24\left(3\xi^2 + 5\pi^2\right)\mathrm{Li}_5\left(-e^\xi\right) + 240\xi\mathrm{Li}_6\left(-e^\xi\right) - 360\mathrm{Li}_7\left(-e^\xi\right), \quad (10.10)$$

where Li_n is the n-th order polylogarithm, defined by

$$\mathrm{Li}_s(z) = \sum_{k=1}^{\infty} \frac{z^k}{k^s}. \quad (10.11)$$

The electron capture rates are equal to the neutron decay rates with ξ swapped for $-\xi$.

As we need the derivative of Γ_e with respect to μ_Δ, the first step is the take the derivative of $F(\xi)$ with respect to ξ. All the polylogarithms take the form $\mathrm{Li}_n\left(-e^\xi\right)$ so we can use the definition of Li_n to write down

$$\mathrm{Li}_n\left(-e^\xi\right) = \sum_{k=1}^{\infty} \frac{\left(-e^\xi\right)^k}{k^n} \quad (10.12)$$

$$= \sum_{k=1}^{\infty} \frac{(-1)^k e^{k\xi}}{k^n}. \quad (10.13)$$

Taking the derivative we see that

$$\frac{\partial}{\partial\xi}\mathrm{Li}_n\left(-e^\xi\right) = \sum_{k=1}^{\infty} \frac{(-1)^k k e^{k\xi}}{k^n} \quad (10.14)$$

$$= \sum_{k=1}^{\infty} \frac{(-1)^k e^{k\xi}}{k^{n-1}} \quad (10.15)$$

$$= \mathrm{Li}_{n-1}\left(-e^\xi\right). \quad (10.16)$$

Next, we know that we are expanding around equilibrium, so we only need to evaluate $\mathrm{Li}_n\left(-e^\xi\right)$ at $\xi = 0$. Using $\xi = 0$ means that the operand is always -1. Going back to the definition of Li_n we can write down

$$\mathrm{Li}_n(-1) = \sum_{k=1}^{\infty} \frac{(-1)^k}{k^n} \quad (10.17)$$

which we can relate to the Dirichlet eta function

$$\eta(n) = \sum_{k=1}^{\infty} \frac{(-1)^{k-1}}{k^n} \quad (10.18)$$

through $\mathrm{Li}_n(-1) = -\eta(n)$. We can then move to the Riemann zeta function

$$\zeta(s) = \sum_{k=1}^{\infty} \frac{1}{k^s} \tag{10.19}$$

by comparing terms:

$$\eta(n) = \frac{1}{1^n} - \frac{1}{2^n} + \frac{1}{3^n} - \frac{1}{4^n} + \frac{1}{5^n} - \frac{1}{6^n} + \ldots, \tag{10.20}$$

$$\zeta(n) = \frac{1}{1^n} + \frac{1}{2^n} + \frac{1}{3^n} + \frac{1}{4^n} + \frac{1}{5^n} + \frac{1}{6^n} + \ldots, \tag{10.21}$$

$$\eta(n) = \zeta(n) - 2\left(\frac{1}{2^n} + \frac{1}{4^n} + \frac{1}{6^n} + \ldots\right) \tag{10.22}$$

$$= \zeta(n) - 2\left(\frac{1}{2^n}\frac{1}{1^n} + \frac{1}{2^n}\frac{1}{2^n} + \frac{1}{2^n}\frac{1}{3^n} + \ldots\right) \tag{10.23}$$

$$= \zeta(n) - 2\frac{1}{2^n}\zeta(n) \tag{10.24}$$

$$= \left(1 - 2^{1-n}\right)\zeta(n). \tag{10.25}$$

So summing all that up, we have

$$\frac{\partial}{\partial\xi}\left(\mathrm{Li}_n\left(-e^{\xi}\right)\right)_{\xi=0} = \mathrm{Li}_{n-1}(-1), \tag{10.26}$$

$$\mathrm{Li}_n(-1) = -\eta(n) \tag{10.27}$$

$$-\eta(n) = -\left(1 - 2^{1-n}\right)\zeta(n) \tag{10.28}$$

where we can look up the values of $\zeta(s)$ that we need (3-7 for F and 2-6 for $\partial F/\partial\xi$):

$$\mathrm{Li}_2(-1) = -\left(1 - 2^{-1}\right)\zeta(2) = -\frac{1}{2}\frac{\pi^2}{6} = -\frac{\pi^2}{12} \approx -0.822467, \tag{10.29}$$

$$\mathrm{Li}_3(-1) = -\left(1 - 2^{-2}\right)\zeta(3) \approx -\frac{3}{4}1.202057 \approx -0.901543, \tag{10.30}$$

$$\mathrm{Li}_4(-1) = -\left(1 - 2^{-3}\right)\zeta(4) = -\frac{7}{8}\frac{\pi^4}{90} = -\frac{7\pi^4}{720} \approx -0.947033, \tag{10.31}$$

$$\mathrm{Li}_5(-1) = -\left(1 - 2^{-4}\right)\zeta(5) \approx -\frac{15}{16}1.036928 \approx -0.972120, \tag{10.32}$$

$$\mathrm{Li}_6(-1) = -\left(1 - 2^{-5}\right)\zeta(6) = -\frac{31}{32}\frac{\pi^6}{945} = -\frac{31\pi^6}{30240} \approx -0.985551, \tag{10.33}$$

$$\mathrm{Li}_7(-1) = -\left(1 - 2^{-6}\right)\zeta(7) \approx -\frac{63}{64}1.008349 \approx -0.992594. \tag{10.34}$$

$$\tag{10.35}$$

Re-writing F slightly gives

$$F(\xi) = -\left(\xi^4 + 10\pi^2\xi^2 + 9\pi^4\right)\mathrm{Li}_3\left(-e^{\xi}\right) + \left(12\xi^3 + 60\pi^2\xi\right)\mathrm{Li}_4\left(-e^{\xi}\right) - \left(72\xi^2 + 120\pi^2\right)\mathrm{Li}_5\left(-e^{\xi}\right) + 240\xi\mathrm{Li}_6\left(-e^{\xi}\right) - 360\mathrm{Li}_7\left(-e^{\xi}\right), \tag{10.36}$$

so the derivative becomes

$$\begin{aligned}\frac{\partial F}{\partial \xi} = &-\left(\xi^4+10\pi^2\xi^2+9\pi^4\right)\mathrm{Li}_2\left(-e^{\xi}\right)-\left(4\xi^3+20\pi^2\xi\right)\mathrm{Li}_3\left(-e^{\xi}\right)\\ &+\left(12\xi^3+60\pi^2\xi\right)\mathrm{Li}_3\left(-e^{\xi}\right)+\left(36\xi^2+60\pi^2\right)\mathrm{Li}_4\left(-e^{\xi}\right)\\ &-\left(72\xi^2+120\pi^2\right)\mathrm{Li}_4\left(-e^{\xi}\right)-(144\xi)\,\mathrm{Li}_5\left(-e^{\xi}\right)\\ &+240\xi\mathrm{Li}_5\left(-e^{\xi}\right)+240\mathrm{Li}_6\left(-e^{\xi}\right)-360\mathrm{Li}_6\left(-e^{\xi}\right). \end{aligned} \tag{10.37}$$

Collecting terms by polylogarithm order we get

$$\begin{aligned}\frac{\partial F}{\partial \xi} = &-\left(\xi^4+10\pi^2\xi^2+9\pi^4\right)\mathrm{Li}_2\left(-e^{\xi}\right)+\left(8\xi^3+40\pi^2\xi\right)\mathrm{Li}_3\left(-e^{\xi}\right)\\ &-\left(36\xi^2+60\pi^2\right)\mathrm{Li}_4\left(-e^{\xi}\right)+96\xi\mathrm{Li}_5\left(-e^{\xi}\right)-120\mathrm{Li}_6\left(-e^{\xi}\right). \end{aligned} \tag{10.38}$$

We can rewrite the rates given in Alford and Harris[12] as

$$\Gamma_{\mathrm{mU,nd(n)}} = G_{\mathrm{mU,nd(n)}}\left(p_{F_n},p_{F_p},p_{F_e}\right)F\left(\xi\right)T^7, \tag{10.39}$$

$$\Gamma_{\mathrm{mU,nd(p)}} = G_{\mathrm{mU,nd(p)}}\left(p_{F_n},p_{F_p},p_{F_e}\right)F\left(\xi\right)T^7, \tag{10.40}$$

remembering we can obtain the electron decay rates by changing the sign on ξ, and that the out of equilibrium behaviour is contained only in the $F(\xi)$ term, so we have collected the rest into the G terms. These G terms are defined as

$$G_{\mathrm{mU,nd(n)}}\left(p_{F_n},p_{F_p},p_{F_e}\right) = \frac{7}{64\pi^9}G^2g_A^2f_{\pi NN}^4\frac{m_n^3m_p}{m_\pi^4}\frac{p_{F_n}^4p_{F_p}}{\left(p_{F_n}^2+m_\pi^2\right)^2}\vartheta_n, \tag{10.41}$$

$$G_{\mathrm{mU,nd(p)}}\left(p_{F_n},p_{F_p},p_{F_e}\right) = \frac{1}{64\pi^9}G^2g_A^2f_{\pi NN}^4\frac{m_nm_p^3}{m_\pi^4}\frac{p_{F_n}\left(p_{F_n}-p_{F_p}\right)^4}{\left(\left(p_{F_n}-p_{F_p}\right)^2+m_\pi^2\right)^2}\vartheta_p, \tag{10.42}$$

where $G^2 = 1.1\times10^{-22}\mathrm{MeV}^{-4}$, $g_A = 1.26$ and $f_{\pi NN}^4\approx 1$. The ϑ terms are given by

$$\vartheta_n = \begin{cases} 1 & \text{if } p_{F_n} > p_{F_p}+p_{F_e} \\ 1-\frac{3}{8}\frac{\left(p_{F_p}+p_{F_e}-p_{F_n}\right)^2}{p_{F_p}p_{F_e}} & \text{if } p_{F_p}+p_{F_e} > p_{F_n} \end{cases}, \tag{10.43}$$

$$\vartheta_p = \begin{cases} 0 & \text{if } p_{F_n} > 3p_{F_p}+p_{F_e} \\ \frac{\left(3p_{F_p}+p_{F_e}-p_{F_n}\right)^2}{p_{F_n}p_{F_e}} & \text{if } 3p_{F_p}+p_{F_e} > p_{F_n} > 3p_{F_p}-p_{F_e} \\ 4\frac{3p_{F_p}-p_{F_n}}{p_{F_n}} & \text{if } 3p_{F_p}-p_{F_e} > p_{F_n} > p_{F_p}+p_{F_e} \\ \left(2+3\frac{2p_{F_p}-p_{F_n}}{p_{F_e}}-3\frac{\left(p_{F_p}-p_{F_e}\right)^2}{p_{F_n}p_{F_e}}\right) & \text{if } p_{F_p}+p_{F_e} > p_{F_n} \end{cases}. \tag{10.44}$$

We can re-write eq. (10.9) as

$$\Gamma_{\mathrm{e}}(\xi) = \left(G_{\mathrm{mU,nd(n)}} + G_{\mathrm{mU,nd(p)}}\right)\left(F(\xi) - F(-\xi)\right)T^7, \tag{10.45}$$

where we have dropped the dependency of the G terms for brevity. The derivative is then

$$\left.\frac{\partial\Gamma}{\partial\mu_\Delta}\right|_{n_b,T,\mu_\Delta=0} = 2\left(G_{\mathrm{mU,nd(n)}} + G_{\mathrm{mU,nd(p)}}\right)T^7 \left.\frac{\partial F(\xi)}{\partial\mu_\Delta}\right|_{n_b,T,\mu_\Delta=0} \tag{10.46}$$

$$= 2\left(G_{\mathrm{mU,nd(n)}} + G_{\mathrm{mU,nd(p)}}\right)T^6 \left.\frac{\partial F(\xi)}{\partial\xi}\right|_{n_b,T,\mu_\Delta=0}, \tag{10.47}$$

where we have used the chain rule, which gives us

$$\left.\frac{\partial F(\xi)}{\partial\mu_\Delta}\right|_{n_b,T,\mu_\Delta=0} = \left.\frac{\partial\xi}{\partial\mu_\Delta}\right|_{n_b,T,\mu_\Delta=0} \left.\frac{\partial F(\xi)}{\partial\xi}\right|_{n_b,T,\mu_\Delta=0}, \tag{10.48}$$

which, remembering $\xi = \mu_\Delta/T$, is

$$\left.\frac{\partial F(\xi)}{\partial\mu_\Delta}\right|_{n_b,T,\mu_\Delta=0} = \frac{1}{T}\left.\frac{\partial F(\xi)}{\partial\xi}\right|_{n_b,T,\mu_\Delta=0}, \tag{10.49}$$

and the result

$$\frac{\partial F(-\xi)}{\partial\mu_\Delta} = -\frac{\partial F(\xi)}{\partial\mu_\Delta} \tag{10.50}$$

where $\xi = 0$. Finally the γ we want is $-\partial\Gamma/\partial\mu_\Delta$ at $\mu_\Delta = 0$, which is

$$\gamma = -2\left(G_{\mathrm{mU,nd(n)}} + G_{\mathrm{mU,nd(p)}}\right)T^6\left(\frac{3\pi^6}{4} + \frac{420\pi^6}{720} + \frac{3720\pi^6}{30240}\right) \tag{10.51}$$

$$= -\frac{367\pi^6}{126}\left(G_{\mathrm{mU,nd(n)}} + G_{\mathrm{mU,nd(p)}}\right)T^6. \tag{10.52}$$

10.2.2 Estimating the relevance of bulk viscosity

In fig. 10.4 we show $\mathcal{A}$ as obtained from the modified Urca rates for the APR equation of state[122] using the value of γ derived above. Alongside $\mathcal{A}$ we also plot contours at a selection of different frequencies/inverse timescales. The two lower contours approximately bracket the frequencies relevant for gravitational waves in the post-merger phase ($1 - 10\,\mathrm{kHz}$). These contours show that at core densities, the bulk viscous resonance peak will occur at similar frequencies to the dynamics when the matter is in the $T = 3 - 5\,\mathrm{MeV}$ range, which agrees well with calculations performed by other groups[10;14]. The dot-dashed contour shows the approximate timestep that represents a "good" resolution in current simulations (equivalent to a grid resolution of around 100 m), and the dotted line represents a speculative resolution we might hope to meet within the next decade (order 1 m). One should keep in mind that our calculations are

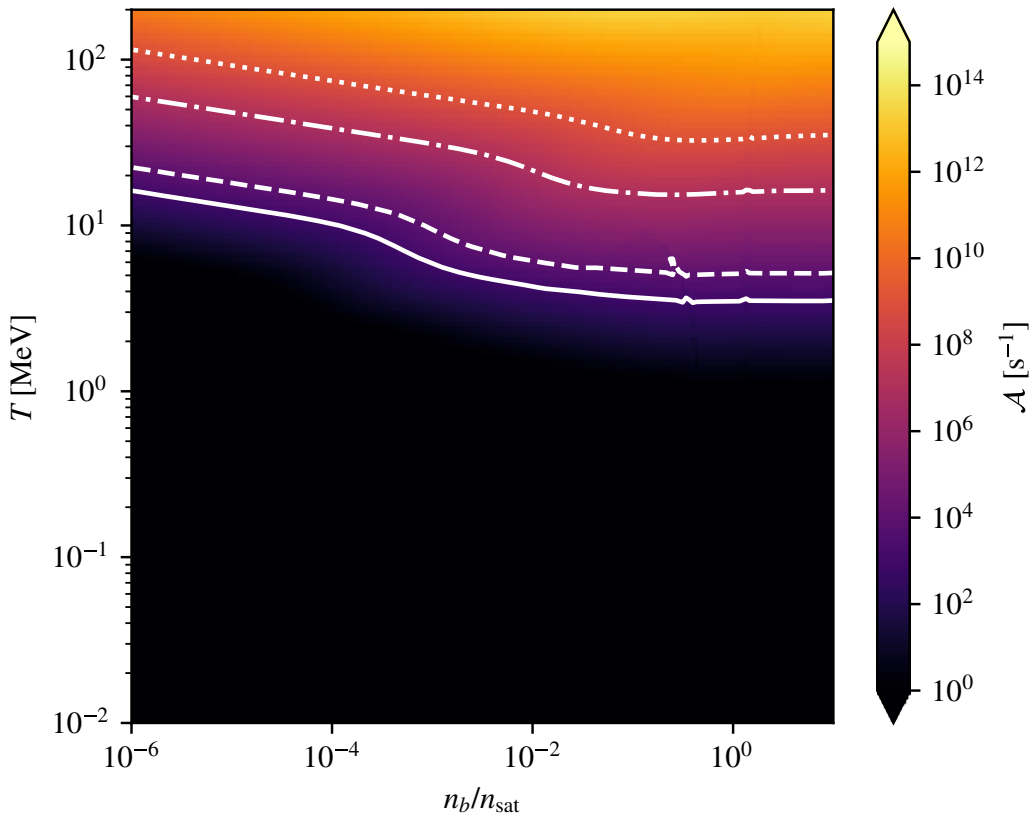

FIGURE 10.4: Plot of $\mathcal{A}$ for the APR equation of state[122]. The restoring term γ is calculated assuming the Fermi surface approximation remains valid using rates from Alford et al. 2018[12]. Contours are at $\mathcal{A} = \{10^3, 10^4, 10^7, 10^9\}\text{s}^{-1}$ (solid, dash, dot-dash, dot).

only valid in the low-T limit, and we should expect shorter timescales to be relevant (at a given density and temperature) if finite temperature effects are included.

In fig. 10.5 we plot the maximum relative contribution of the bulk viscous pressure (eq. (10.8)), keeping the contours from fig. 10.4. We see that the upper right ($T \geq 3\,\text{MeV}$, $n_b \geq 10^{-2} n_{\rm sat}$) region of the plane provides the most promising conditions for the relevance of bulk viscosity to the dynamics of the simulation. There are a number of interesting features in this plot. Vertically on the right hand side, at just above saturation density, we see the APR phase transition. To the left of this there are two features at similar densities. The first, which goes up to around $\sim 10\,\text{MeV}$ and goes off the bottom of the density scale at around $\sim 0.8\,\text{MeV}$ we see what is likely an artefact of the method used to produce the equation of state table (we see similar artefacts on other similarly produced tables) whereby different calculation methods are used in different regions of (ρ, T) space and stitched together. Finally, there is a line that closely follows the previous artefact before going to the top right of the plot where $\Pi^{\rm max}/p^{\rm eq}$ drops to ~ 0. We have not studied this feature in detail, but, speculatively, it may be related to where $\partial\mu_\Delta/\partial Y_{\rm e}$ passes through 0, but, again, we have not explored this fully.

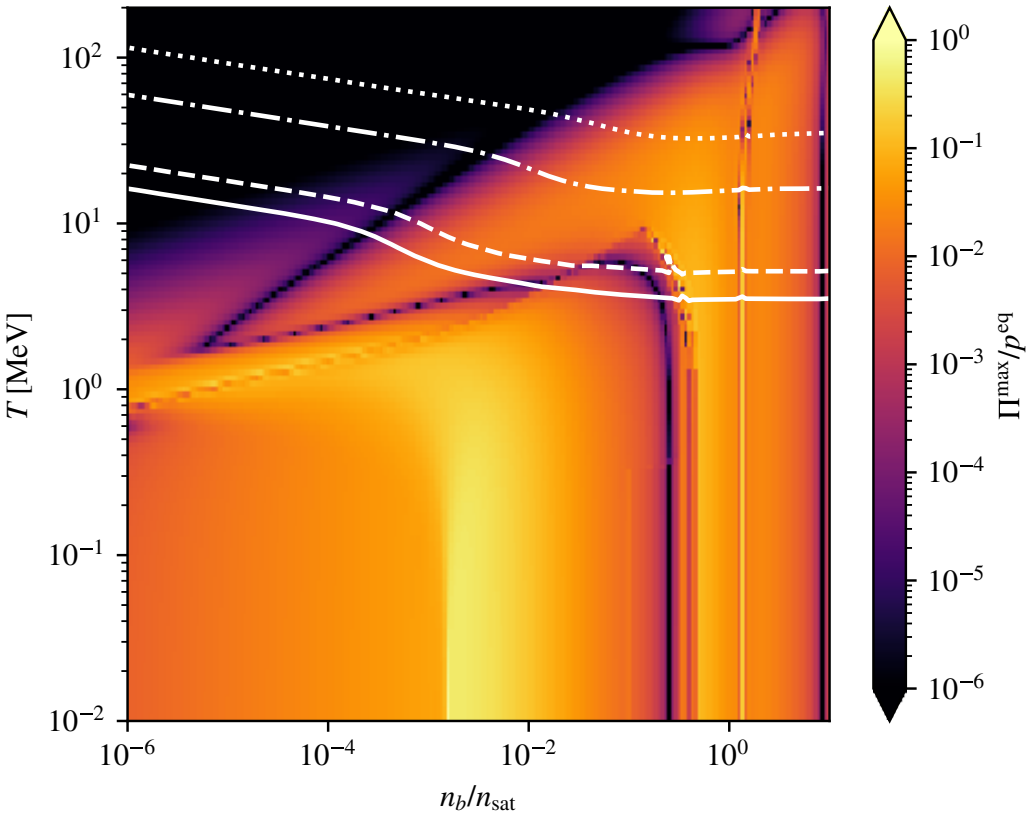

FIGURE 10.5: The maximum (for each point we assume that $\omega = \mathcal{A}$) potential relative contribution of the bulk viscous approximation $\Pi^{\text{max}}/p^{\text{eq}}$ at each point in phase space using the APR equation of state[122]. Contours are copied from fig. 10.4.

Finally, as $\mathcal{A}$ represents the inverse of the timescale of the equilibriation we can use an independent measure to verify our results based on the mean free path of the neutrinos calculated in section 9.2. Equation (9.8) gives the opacity of the fluid to neutrinos, which is related to the mean free path through $\ell = 1/\kappa$. Assuming the neutrinos travel at c the mean free path then sets a timescale for the equilibriation as $\tau_{\text{eq}} = 1/\tilde{\kappa}_*$. In fig. 10.6 we plot this timescale, and (as with fig. 10.5) we also plot the same contours over $\Pi^{\text{max}}/p^{\text{eq}}$ in fig. 10.7.

Comparing these results to figure 10.4 we see that there are some similarities. Both plots agree that the timescales for $T \lesssim 1$ MeV are too long to be relevant for our purposes, and they are also similar for most densities in the 1 MeV $\lesssim T \lesssim$ 10 MeV region. However, at high temperatures there are qualitative differences. We see the contours curving upwards much more noticeably here than for $\mathcal{A}$, and the timescales here are also generally slightly longer. We also note that at very high densities, where $n_b \gtrsim n_{\text{sat}}$, there is an upwards trend to the contours, and they are also much closer together here than for the previous calculation. This suggests that an equivalent bulk viscous approximation will be relevant over less of the phase space when a timescale measure based on neutrino equilibriation is used compared to the Fermi surface approximation. The quantitative

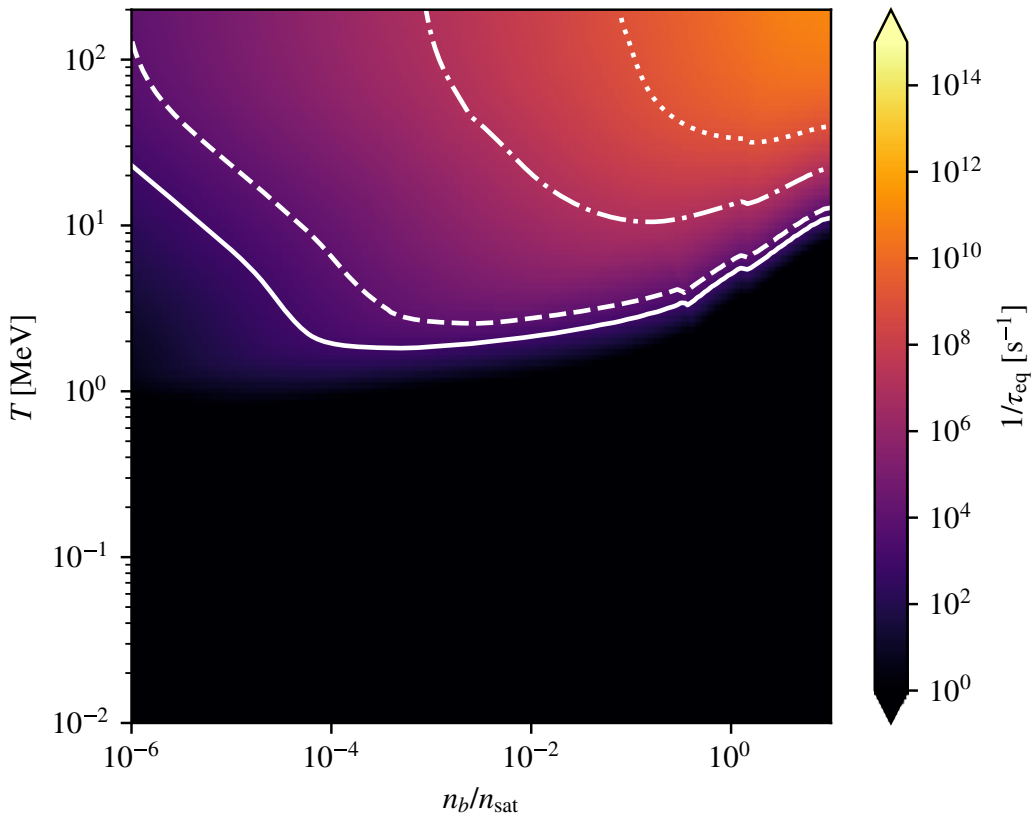

FIGURE 10.6: Plot of the inverse of the neutrino equilibriation timescale $1/\tau_{\rm eq}$. Contours are at $1/\tau_{\rm eq} = \{10^3, 10^4, 10^7, 10^9\}{\rm s}^{-1}$ (solid, dash, dot-dash, dot).

differences between the timescales shown in the two figures illustrates the uncertainties in the precise values due to not including all of the physics.

Despite the qualitative differences between the equilibration timescales obtained through the two different methods, it is encouraging that there are some broad similarities between the two, particularly the regions of interest at relevant temperatures and densities.

10.3 Approximating Bulk Viscosity

Here we briefly outline how one might include an approximation of bulk viscosity in simulations. The low frequency limit of the bulk viscous pressure can be approximated by (the "Navier-Stokes Approximation")[37]

$$\Pi = -\left.\frac{\partial p}{\partial \mu_\Delta}\right|_{n_b,e} \frac{n_b}{\mathcal{A}} \mathcal{B}\theta. \tag{10.53}$$

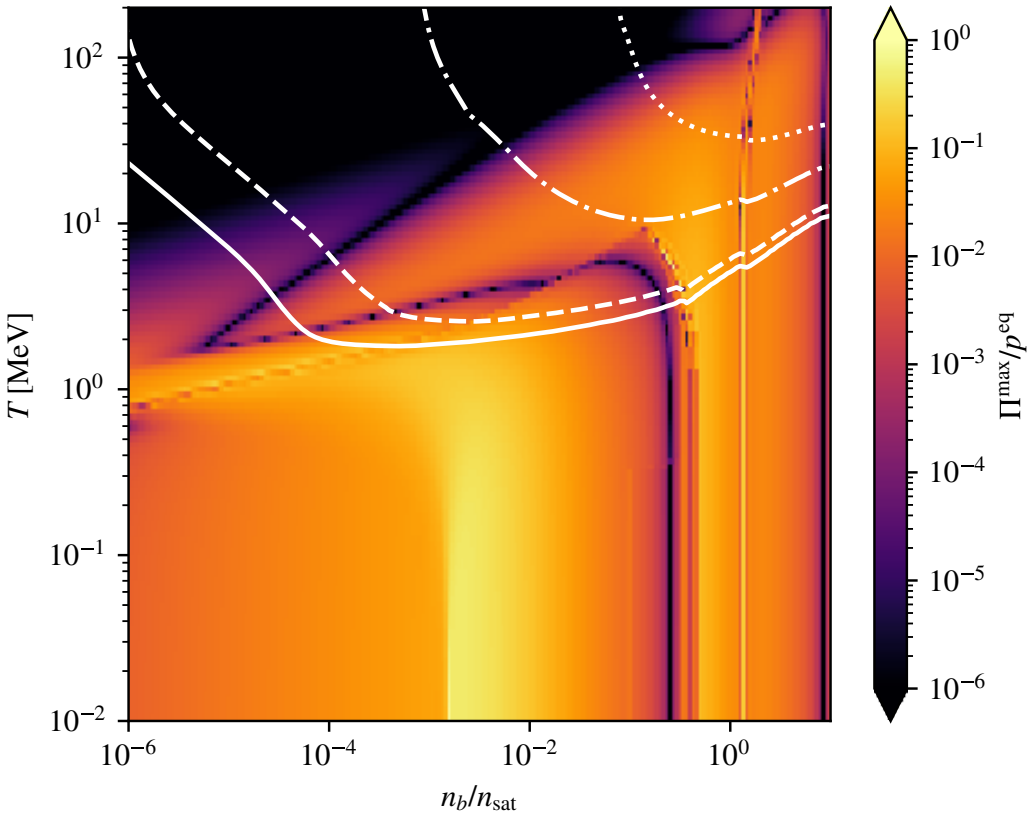

FIGURE 10.7: Copy of figure 10.5 with contours from figure 10.6. Colors give the maximum (for each point we assume that $\omega = \mathcal{A}$) potential relative contribution of the bulk viscous approximation $\Pi^{\rm max}/p^{\rm eq}$ at each point in phase space using the APR equation of state[122].

The total pressure is then

$$p = p^{\rm eq} + \Pi, \tag{10.54}$$

where $p^{\rm eq}$ is the pressure as calculated in equilibrium. As we have shown above, we have the ability to calculate the terms here (except θ) from the equation of state and the reaction rates. The expansion, θ, is given by

$$\theta = \nabla_a u^a. \tag{10.55}$$

Our simulations do not currently measure the time derivatives of variables, but it is possible to modify them to do this. The only place where is the pressure is used in the simulation is when moving between the primitive and conserved variables (sections 3.3 and 6.2). It is therefore be possible to modify the pressure (and enthalpy) terms in eqs. (3.39) to (3.41), and the corresponding terms in the conservative-to-primitive methods in section 6.2.

The main issue with this method is that, as illustrated in fig. 10.8, the error in the bulk

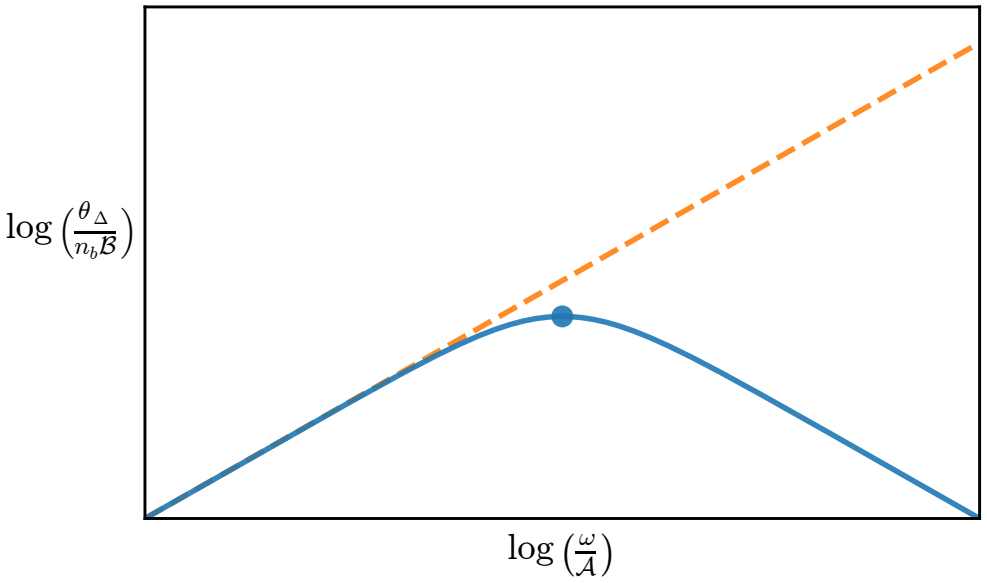

FIGURE 10.8: Navier-Stokes approximation to bulk viscosity (orange, dashed), and resonant behaviour of bulk viscosity (blue, solid). $\omega/\mathcal{A}$ represents the scaled frequency of the driving oscillation, and the resultant bulk viscous pressure goes like $\mu_\Delta/n_b\mathcal{B}$.

viscous pressure term grows rapidly when the oscillation in the fluid is much faster than the frequency at the resonant peak. This is especially problematic in turbulent regions (such as the shearing interface between the two merging stars) where we might expect turbulent mixing to excite oscillations down to the metre scale or below (although in practice this will be limited by the grid resolution) as shown by Radice[109].

10.4 Known Issues

As alluded to multiple times in this chapter, an important issue that we face in including bulk viscosity in simulations is that of timescales. We expect there to be significant regions throughout the simulation where the equilibriation of the fluid takes place on a timescale much faster than the timestep of the simulation. This problem will be exacerbated as we try to include finite temperature effects (such as neutrinos and pions), as we expect these to decrease the equilibriation timescale significantly. Approximations to bulk viscosity, such as those discussed in Celora et al.[37], will not capture the full non-linearity of the effect, so in regions where the reactions are slow enough it is desirable to evolve the reactions directly (using e.g. eq. (7.2)), and only use an approximation where the reactions cannot be captured. Another approach could be to use an LES style method (as is discussed in Celora et al.[37]), however this requires further development.

In addition to such timescale issues, we also expect problems when attempting to include the same set of reactions through different means, which we call "double counting" (see

Hammond et al.[68] and Celora et al.[37]). There are (at least) two cases where double counting may be an issue. First, one can imagine trying to split the reactions into "slow" (i.e. resolvable by the timestep of the simulation) and "fast" (i.e. faster than the numerical timestep) parts (see Celora et al.[37] for a full discussion), implementing the effect of slow reactions through changes to the particle species fractions, and the effect of the fast reactions through an effective bulk viscous pressure. In this case there is a coupling between the fast and slow rates (see appendix D3 of Celora et al.[37]), so one would need to ensure that the effects of the unresolved fast reactions on the resolvable slow reactions are carefully considered. Second, assume we have a "complete" implementation of bulk viscosity that takes the above into account, on top of which we want to include a neutrinos. As the two effects result from the same set of reactions, we would need to be careful how that neutrino treatment interacts with the reaction rates or we may implicitly include "extra" bulk viscosity[68].

Chapter 11

Conclusions

11.1 Summary

In part II of this thesis we presented an exploration of some of the composition- and temperature-dependent physics that impact upon binary neutron star mergers. Chapter 8 covered the "standard" definition of β-equilibrium, and the detectability via gravitational waves (focusing on third generation detectors) of deviation from said equilibrium, the results of which are published in Hammond et al.[69]. We found that while the difference between the spectra using the slow- and fast-limits for reactions is small, it should be detectable by third generation gravitational wave observatories. In addition, we showed that this difference is due to a softening of the equation of state when the equilibrium condition is violated, causing the matter distribution within the star to change, increasing the moment of inertia. We reproduced this effect for isolated rotating stars using several different equations of state (based on different underlying models), and showed that this effect may be general for all (reasonable) equations of state.

In chapter 9 we discussed how the definition of β-equilibrium changes as one starts to account for thermal effects, as published in Hammond et al.[68]. We began by showing how the definition at finite temperature derived in Alford and Harris[12] can cause a softening of the equation of state. We then discussed how the equilibrium will be dominated by neutrinos when they become trapped, and the difficulties in defining where this happens due to the non-locality of the effect. Next we mentioned how at high temperatures ($\sim 25\,\mathrm{MeV}$) the equilibriation between baryonic species might be mediated by thermal pions, which drastically decreases the timescale on which the equilibrium is established. We then applied the conditions for these different equilibria to our simulation data, demonstrating that they might all be simultaneously relevant in different regions of merger, complicating the issue of defining the local equilibrium condition during a merger simulation. Finally we presented some unpublished work

exploring the post processing of simulation data using ray tracing to determine the regions where neutrino reactions are significant, comparing the results to our previous estimate.

Chapter 10 features a discussion of bulk viscosity. In section 10.1 we discuss sub- and supra-thermal deviation from equilibrium as they relate to bulk viscosity, as published in Hammond et al.[68]. In the remainder of the chapter we present the calculation of the bulk-viscous coefficients $\mathcal{A}$ and $\mathcal{B}$ from an equation of state table and a parameterisation of the Urca reaction rates, as published in Celora et al.[37]. Finally, we talk about some of the issues related to the inclusion of bulk viscosity/reactions/neutrinos in simulations, in particular how all three effects are intimately linked, and the problem of having to deal with timescales that are much faster than the timestep of a simulation.

The simulations discussed in part II are built upon over a century of general relativity theory, forty years of numerical relativity and relativistic hydrodynamics simulations, and use a software suite that was first released twelve years ago (although some parts date back nearly thirty years). In part I we gave a broad overview of the theory that underpins the simulation of binary neutron star mergers (chapters 2 and 3), discussed the parts of the Einstein Toolkit (and other codes) that we used in the creation of our simulations (chapter 4), covered some of the techniques used for the analysis and visualisation of the data that comes out of the simulations (chapter 5), and finally in chapter 6 we discussed our own modifications to the code (as published in Hammond et al.[68]) that allowed us to carry out the simulations we have presented.

11.2 Future Work

The work we have presented naturally leads to one obvious idea for future work: perform a simulation that directly implements all relevant reactions and includes a full neutrino treatment. By simulating the reactions and neutrinos using a finite temperature prescription one would automatically account for all of the physics we explored in chapters 9 and 10. However optimistic this may be in terms of the computational power required, there are, unfortunately, other hurdles.

As we mention in section 10.4 there are issues with the timescales at play. We expect that the equilibriation under the neutrino trapped regime would outpace the currently available temporal resolution of simulations, and if the equilibriation via pions is to be taken into account then there is no hope of trying to resolve the reactions in the traditional way. As we discuss in Celora et al.[37], one way around this might be to perform simulations using an "LES" like method, where very short, very high temporal and spacial resolution box simulations are used to inform large scale simulations of the unresolved behaviour.

Further, as we briefly mention in section 7.4, there are issues with the temperature in binary neutron star merger simulations as they currently exist. If we want the initial data to realistically represent old pairs of neutron stars, we need to access temperatures in the $\sim 10^6\,\mathrm{K}$ ($\sim 10^{-5}\,\mathrm{MeV}$) range, something that is not possible with currently available equations of state. Additionally, if we want to study the behaviour of the crust in simulations we need to get a handle on the heating caused by shocks building up at the surface of the star. Finally, we expect the cores to remain relatively cold throughout the merger process. However, as we also show in section 7.4, the conserved quantities that are evolved in numerical relativistic hydrodynamics simulations are not well suited to low temperatures as there are issues of error amplification inherent in the system. Evolving the entropy alongside the current evolved quantities may help, however this becomes an issue when shocks form as the entropy is not conserved.

In section 10.3 we present an approximation of the bulk viscous pressure that is calculable with only minor modifications to current code. Bulk viscosity in mergers is an effect that is being studied by several groups through different methods, with conflicting opinions as to whether the effect is important (compare Most et al.[96] and Radice et al.[112]), motivating further study. There are issues with this prescription that we are currently working on, namely how to deal with the exploding error when the expansion term is in the high-frequency limit, and how to modify the code so that the required quantities are available.

Finally, with the aid of colleagues in the nuclear physics community we are in the early stages of performing a simulation that takes into account the effects of thermal pions (as discussed in section 9.3). By comparing the results of such a simulation to one that does not feature such effects—in a similar manner to the work in section 8.2—we hope to, at least qualitatively, demonstrate whether such pions can have a measurable effect (with respect to future detectors) on the gravitational wave output of a merger.

References

[1] B. P. Abbott et al. GW170817: Observation of gravitational waves from a binary neutron star inspiral. *Physical Review Letters*, 119(16), Oct. 2017. doi:10.1103/physrevlett.119.161101. URL https://doi.org/10.1103/physrevlett.119.161101.

[2] B. P. Abbott et al. Multi-messenger observations of a binary neutron star merger. *The Astrophysical Journal*, 848(2):L12, Oct. 2017. doi:10.3847/2041-8213/aa91c9.

[3] B. P. Abbott et al. GW170817: Measurements of neutron star radii and equation of state. *Physical Review Letters*, 121(16), Oct. 2018. doi:10.1103/physrevlett.121.161101.

[4] B. P. Abbott et al. Properties of the binary neutron star merger GW170817. *Physical Review X*, 9(1), Jan. 2019. doi:10.1103/physrevx.9.011001.

[5] A. Akmal, V. R. Pandharipande, and D. G. Ravenhall. Equation of state of nucleon matter and neutron star structure. *Physical Review C*, 58(3):1804–1828, Sept. 1998. doi:10.1103/physrevc.58.1804.

[6] M. Alcubierre. *Introduction to 3+1 Numerical Relativity*. Oxford University Press, Apr. 2008. doi:10.1093/acprof:oso/9780199205677.001.0001.

[7] M. Alcubierre, B. Brügmann, D. Pollney, E. Seidel, and R. Takahashi. Black hole excision for dynamic black holes. *Physical Review D*, 64(6), Aug. 2001. doi:10.1103/physrevd.64.061501.

[8] M. Alcubierre, B. Brügmann, P. Diener, M. Koppitz, D. Pollney, E. Seidel, and R. Takahashi. Gauge conditions for long-term numerical black hole evolutions without excision. *Physical Review D*, 67(8), Apr. 2003. doi:10.1103/physrevd.67.084023.

[9] M. Alford, A. Harutyunyan, and A. Sedrakian. Bulk viscosity of baryonic matter with trapped neutrinos. *Physical Review D*, 100(10), Nov. 2019. doi:10.1103/physrevd.100.103021.

[10] M. Alford, A. Harutyunyan, and A. Sedrakian. Bulk viscous damping of density oscillations in neutron star mergers. *Particles*, 3(2):500–517, June 2020. doi:10.3390/particles3020034.

[11] M. Alford, A. Harutyunyan, and A. Sedrakian. Bulk viscosity from Urca processes: $npe\mu$ matter in the neutrino-trapped regime. *Physical Review D*, 104(10), Nov. 2021. doi:10.1103/physrevd.104.103027.

[12] M. G. Alford and S. P. Harris. β equilibrium in neutron-star mergers. *Physical Review C*, 98(6), Dec. 2018. doi:10.1103/physrevc.98.065806.

[13] M. G. Alford, S. Mahmoodifar, and K. Schwenzer. Large amplitude behavior of the bulk viscosity of dense matter. *Journal of Physics G: Nuclear and Particle Physics*, 37(12):125202, Nov. 2010. doi:10.1088/0954-3899/37/12/125202.

[14] M. G. Alford, L. Bovard, M. Hanauske, L. Rezzolla, and K. Schwenzer. Viscous dissipation and heat conduction in binary neutron-star mergers. *Physical Review Letters*, 120(4), Jan. 2018. doi:10.1103/physrevlett.120.041101.

[15] M. G. Alford, A. Haber, S. P. Harris, and Z. Zhang. Beta equilibrium under neutron star merger conditions. *Universe*, 7(11):399, Oct. 2021. doi:10.3390/universe7110399.

[16] D. Alic, C. Bona-Casas, C. Bona, L. Rezzolla, and C. Palenzuela. Conformal and covariant formulation of the Z4 system with constraint-violation damping. *Physical Review D*, 85(6), Mar. 2012. doi:10.1103/physrevd.85.064040.

[17] D. Alic, W. Kastaun, and L. Rezzolla. Constraint damping of the conformal and covariant formulation of the Z4 system in simulations of binary neutron stars. *Physical Review D*, 88(6), Sept. 2013. doi:10.1103/physrevd.88.064049.

[18] M. Anderson and R. A. Matzner. Extended lifetime in computational evolution of isolated black holes. *Foundations of Physics*, 35(9):1477–1495, Sept. 2005. doi:10.1007/s10701-005-6477-x.

[19] P. Anninos, J. Massó, E. Seidel, W.-M. Suen, and J. Towns. Three-dimensional numerical relativity: The evolution of black holes. *Physical Review D*, 52(4):2059–2082, Aug. 1995. doi:10.1103/physrevd.52.2059.

[20] L. Anton, O. Zanotti, J. A. Miralles, J. M. Marti, J. M. Ibanez, J. A. Font, and J. A. Pons. Numerical 3+1 general relativistic magnetohydrodynamics: A local characteristic approach. *The Astrophysical Journal*, 637(1):296–312, Jan. 2006. doi:10.1086/498238.

[21] W. Baade and F. Zwicky. On super-novae. *Proceedings of the National Academy of Sciences*, 20(5):254–259, May 1934. doi:10.1073/pnas.20.5.254.

[22] L. Baiotti and L. Rezzolla. Binary neutron star mergers: A review of einstein's richest laboratory. *Reports on Progress in Physics*, 80(9):096901, July 2017. doi:10.1088/1361-6633/aa67bb.

[23] L. Baiotti, I. Hawke, P. J. Montero, and L. Rezzolla. A new three-dimensional general-relativistic hydrodynamics code. *Memorie della Societa Astronomica Italiana Supplementi*, 74:210, Jan. 2003. URL https://ui.adsabs.harvard.edu/abs/2003MSAIS...1..210B.

[24] L. Baiotti, B. Giacomazzo, and L. Rezzolla. Accurate evolutions of inspiralling neutron-star binaries: Prompt and delayed collapse to a black hole. *Physical Review D*, 78(8), Oct. 2008. doi:10.1103/physrevd.78.084033.

[25] E. Baird, S. Fairhurst, M. Hannam, and P. Murphy. Degeneracy between mass and spin in black-hole-binary waveforms. *Physical Review D*, 87(2), Jan. 2013. doi:10.1103/physrevd.87.024035.

[26] F. Banyuls, J. A. Font, J. M. Ibanez, J. M. Marti, and J. A. Miralles. Numerical {3 + 1} general relativistic hydrodynamics: A local characteristic approach. *The Astrophysical Journal*, 476(1):221–231, Feb. 1997. doi:10.1086/303604.

[27] T. W. Baumgarte and S. L. Shapiro. Numerical integration of Einstein's field equations. *Physical Review D*, 59(2), Dec. 1998. doi:10.1103/physrevd.59.024007.

[28] T. W. Baumgarte and S. L. Shapiro. *Numerical Relativity*. Cambridge University Press, June 2010. doi:10.1017/cbo9781139193344.

[29] M. J. Berger and J. Oliger. Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of Computational Physics*, 53(3):484–512, Mar. 1984. doi:10.1016/0021-9991(84)90073-1.

[30] S. Bernuzzi and D. Hilditch. Constraint violation in free evolution schemes: Comparing the BSSNOK formulation with a conformal decomposition of the Z4 formulation. *Physical Review D*, 81(8), Apr. 2010. doi:10.1103/physrevd.81.084003.

[31] C. Bona, J. Massó, E. Seidel, and J. Stela. New formalism for numerical relativity. *Physical Review Letters*, 75(4):600–603, July 1995. doi:10.1103/physrevlett.75.600.

[32] C. Bona, T. Ledvinka, C. Palenzuela, and M. Žáček. General-covariant evolution formalism for numerical relativity. *Physical Review D*, 67(10), May 2003. doi:10.1103/physrevd.67.104005.

[33] G. Bozzola. kuibit: Analyzing Einstein toolkit simulations with Python. *Journal of Open Source Software*, 6(60):3099, Apr. 2021. doi:10.21105/joss.03099.

[34] D. Brown, P. Diener, O. Sarbach, E. Schnetter, and M. Tiglio. Turduckening black holes: An analytical and computational study. *Physical Review D*, 79(4), Feb. 2009. doi:10.1103/physrevd.79.044023.

[35] B. Brügmann. Adaptive mesh and geodesically sliced Schwarzschild spacetime in 3+1 dimensions. *Physical Review D*, 54(12):7361–7372, Dec. 1996. doi:10.1103/physrevd.54.7361.

[36] G. Camelio, T. Dietrich, M. Marques, and S. Rosswog. Rotating neutron stars with nonbarotropic thermal profile. *Physical Review D*, 100(12), Dec. 2019. doi:10.1103/physrevd.100.123001.

[37] T. Celora, I. Hawke, P. C. Hammond, N. Andersson, and G. L. Comer. Formulating bulk viscosity for neutron star simulations. *Physical Review D*, 105(10), May 2022. doi:10.1103/physrevd.105.103016.

[38] E. Chabanat, P. Bonche, P. Haensel, J. Meyer, and R. Schaeffer. A Skyrme parametrization from subnuclear to neutron star densities part II. Nuclei far from stabilities. *Nuclear Physics A*, 635(1-2):231–256, May 1998. doi:10.1016/s0375-9474(98)00180-8.

[39] F. Cipolletta, J. V. Kalinani, E. Giangrandi, B. Giacomazzo, R. Ciolfi, L. Sala, and B. Giudici. Spritz: General relativistic magnetohydrodynamics with neutrinos. *Classical and Quantum Gravity*, 38(8):085021, Mar. 2021. doi:10.1088/1361-6382/abebb7.

[40] P. Colella and M. D. Sekora. A limiter for PPM that preserves accuracy at smooth extrema. *Journal of Computational Physics*, 227(15):7069–7076, July 2008. doi:10.1016/j.jcp.2008.03.034.

[41] P. Colella and P. R. Woodward. The Piecewise Parabolic Method (PPM) for gas-dynamical simulations. *Journal of Computational Physics*, 54(1):174–201, Apr. 1984. doi:10.1016/0021-9991(84)90143-8.

[42] CompOSE Core Team. CompOSE, accessed 2021. URL https://compose.obspm.fr/home.

[43] P. S. Cowperthwaite et al. The electromagnetic counterpart of the binary neutron star merger LIGO/Virgo GW170817. II. UV, optical, and near-infrared light curves and comparison to kilonova models. *The Astrophysical Journal*, 848(2):L17, Oct. 2017. doi:10.3847/2041-8213/aa8fc7.

[44] B. Daszuta, F. Zappa, W. Cook, D. Radice, S. Bernuzzi, and V. Morozova. `GR-Athena++`: Puncture evolutions on vertex-centered oct-tree adaptive mesh refinement. *The Astrophysical Journal Supplement Series*, 257(2):25, Nov. 2021. doi:10.3847/1538-4365/ac157b.

[45] T. J. Dekker. Finding a zero by means of successive linear interpolation. In B. Dejon and P. Henrici, editors, *Constructive Aspects of the Fundamental Theorem of Algebra*, pages 37–48, New York, 1969. Wiley-Interscience. ISBN 978-0-471-20300-1.

[46] N. Deppe et al. Simulating magnetized neutron stars with discontinuous Galerkin methods, 2021. URL https://arxiv.org/abs/2109.12033.

[47] N. Deppe et al. `SpECTRE`, accessed 2022. URL https://spectre-code.org.

[48] T. Dietrich, D. Radice, S. Bernuzzi, F. Zappa, A. Perego, B. Brügmann, S. V. Chaurasia, R. Dudi, W. Tichy, and M. Ujevic. `CoRe` database of binary neutron star merger waveforms. *Classical and Quantum Gravity*, 35(24):24LT01, Nov. 2018. doi:10.1088/1361-6382/aaebc0.

[49] T. Dietrich, M. W. Coughlin, P. T. H. Pang, M. Bulla, J. Heinzel, L. Issa, I. Tews, and S. Antier. Multimessenger constraints on the neutron-star equation of state and the Hubble constant. *Science*, 370(6523):1450–1453, Dec. 2020. doi:10.1126/science.abb4317.

[50] T. Dietrich, T. Hinderer, and A. Samajdar. Interpreting binary neutron star mergers: Describing the binary neutron star dynamics, modelling gravitational waveforms, and analyzing detections. *General Relativity and Gravitation*, 53(3), Mar. 2021. doi:10.1007/s10714-020-02751-6.

[51] D. Eichler, M. Livio, T. Piran, and D. N. Schramm. Nucleosynthesis, neutrino bursts and γ-rays from coalescing neutron stars. *Nature*, 340(6229):126–128, July 1989. doi:10.1038/340126a0.

[52] B. Einfeldt. On Godunov-type methods for gas dynamics. *SIAM Journal on Numerical Analysis*, 25(2):294–318, Apr. 1988. doi:10.1137/0725021.

[53] A. Endrizzi, A. Perego, F. M. Fabbri, L. Branca, D. Radice, S. Bernuzzi, B. Giacomazzo, F. Pederiva, and A. Lovato. Thermodynamics conditions of matter in the neutrino decoupling region during neutron star mergers. *The European Physical Journal A*, 56(1), Jan. 2020. doi:10.1140/epja/s10050-019-00018-6.

[54] Z. Etienne et al. The Einstein Toolkit, accessed 2021. URL http://einsteintoolkit.org/index.html.

[55] Z. B. Etienne, Y. T. Liu, V. Paschalidis, and S. L. Shapiro. General relativistic simulations of black-hole–neutron-star mergers: Effects of magnetic fields. *Physical Review D*, 85(6), Mar. 2012. doi:10.1103/physrevd.85.064029.

[56] M. Evans et al. A horizon study for Cosmic Explorer: Science, observatories, and community, 2021. URL https://arxiv.org/abs/2109.09882.

[57] J. A. Font. Numerical hydrodynamics in general relativity. *Living Reviews in Relativity*, 3(1), May 2000. doi:10.12942/lrr-2000-2.

[58] B. Fore and S. Reddy. Pions in hot dense matter and their astrophysical implications. *Physical Review C*, 101(3), Mar. 2020. doi:10.1103/physrevc.101.035809.

[59] C. F. Gammie, J. C. McKinney, and G. Toth. HARM: A numerical scheme for general relativistic magnetohydrodynamics. *The Astrophysical Journal*, 589(1): 444–457, May 2003. doi:10.1086/374594.

[60] S. K. Godunov. A difference scheme for numerical solution of discontinuous solution of hydrodynamic equations. *Math. Sbornik*, 47:271–306, 1959.

[61] T. Goodale, G. Allen, G. Lanfermann, J. Massó, T. Radke, E. Seidel, and J. Shalf. The Cactus framework and toolkit: Design and applications. In *Vector and Parallel Processing – VECPAR'2002, 5th International Conference, Lecture Notes in Computer Science*, Berlin, 2003. Springer. URL http://edoc.mpg.de/3341.

[62] S. Goriely, N. Chamel, and J. M. Pearson. Further explorations of Skyrme-Hartree-Fock-Bogoliubov mass formulas. XIII. The 2012 atomic mass evaluation and the symmetry coefficient. *Physical Review C*, 88(2), Aug. 2013. doi:10.1103/physrevc.88.024308.

[63] E. Gourgoulhon, P. Grandclément, and J. Novak. LORENE, accessed 2022. URL https://lorene.obspm.fr/.

[64] P. Grete, F. W. Glines, and B. W. O'Shea. `K-Athena`: A performance portable structured grid finite volume magnetohydrodynamics code. *IEEE Transactions on Parallel and Distributed Systems*, 32(1):85–97, Jan. 2021. doi:10.1109/tpds.2020.3010016.

[65] C. Gundlach and R. J. Leveque. Universality in the run-up of shock waves to the surface of a star. *Journal of Fluid Mechanics*, 676:237–264, Apr. 2011. doi:10.1017/jfm.2011.42.

[66] C. Gundlach, G. Calabrese, I. Hinder, and J. M. Martín-García. Constraint damping in the Z4 formulation and harmonic gauge. *Classical and Quantum Gravity*, 22(17):3767–3773, Aug. 2005. doi:10.1088/0264-9381/22/17/025.

[67] P. Haensel. Non-equilibrium neutrino emissivities and opacities of neutron star matter. *Astronomy and Astrophysics*, 262:131–137, Aug. 1992. URL https://ui.adsabs.harvard.edu/abs/1992A&A...262..131H.

[68] P. Hammond, I. Hawke, and N. Andersson. Thermal aspects of neutron star mergers. *Physical Review D*, 104(10), Nov. 2021. doi:10.1103/physrevd.104.103006.

[69] P. Hammond, I. Hawke, and N. Andersson. Detecting the impact of nuclear reactions on neutron star mergers through gravitational waves, 2022. URL https://arxiv.org/abs/2205.11377. Submitted to *Physical Review Letters*.

[70] A. Harten, P. D. Lax, and B. van Leer. On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM Review*, 25(1):35–61, Jan. 1983. doi:10.1137/1025002.

[71] M. Hempel and J. Schaffner-Bielich. A statistical model for a complete supernova equation of state. *Nuclear Physics A*, 837(3-4):210–254, June 2010. doi:10.1016/j.nuclphysa.2010.02.010.

[72] A. Hewish, S. J. Bell, J. D. H. Pilkington, P. F. Scott, and R. A. Collins. Observation of a rapidly pulsating radio source. *Nature*, 217(5130):709–713, Feb. 1968. doi:10.1038/217709a0.

[73] S. Hild et al. Sensitivity studies for third-generation gravitational wave observatories. *Classical and Quantum Gravity*, 28(9):094013, Apr. 2011. doi:10.1088/0264-9381/28/9/094013.

[74] S. Husa, I. Hinder, and C. Lechner. `Kranc`: a Mathematica package to generate numerical codes for tensorial evolution equations. *Computer Physics Communications*, 174(12):983–1004, June 2006. doi:10.1016/j.cpc.2006.02.002.

[75] J. V. Kalinani, R. Ciolfi, W. Kastaun, B. Giacomazzo, F. Cipolletta, and L. Ennoggi. Implementing a new recovery scheme for primitive variables in the general relativistic magnetohydrodynamic code Spritz. *Physical Review D*, 105 (10), May 2022. doi:10.1103/physrevd.105.103031.

[76] W. Kastaun. PyCactus Tools, accessed 2021. URL https://github.com/wokast/PyCactus.

[77] W. Kastaun, R. Ciolfi, and B. Giacomazzo. Structure of stable binary neutron star merger remnants: A case study. *Physical Review D*, 94(4), Aug. 2016. doi:10.1103/physrevd.94.044060.

[78] W. Kastaun, J. V. Kalinani, and R. Ciolfi. Robust recovery of primitive variables in relativistic ideal magnetohydrodynamics. *Physical Review D*, 103(2), Jan. 2021. doi:10.1103/physrevd.103.023018.

[79] L. Kidder et al. SpEC: Spectral Einstein code, accessed 2022. URL https://www.black-holes.org/code/SpEC.html.

[80] P. Kumar, K. Barkett, S. Bhagwat, N. Afshari, D. A. Brown, G. Lovelace, M. A. Scheel, and B. Szilágyi. Accuracy and precision of gravitational-wave models of inspiraling neutron star-black hole binaries with spin: Comparison with matter-free numerical relativity in the low-frequency regime. *Physical Review D*, 92(10), Nov. 2015. doi:10.1103/physrevd.92.102001.

[81] C. B. Laney. *Computational Gasdynamics*. Cambridge University Press, June 1998. doi:10.1017/cbo9780511605604.

[82] J. M. Lattimer, F. Mackie, D. G. Ravenhall, and D. N. Schramm. The decompression of cold neutron star matter. *The Astrophysical Journal*, 213:225, Apr. 1977. doi:10.1086/155148.

[83] J. M. Lattimer, C. J. Pethick, M. Prakash, and P. Haensel. Direct Urca process in neutron stars. *Physical Review Letters*, 66(21):2701–2704, May 1991. doi:10.1103/physrevlett.66.2701.

[84] F. Lekien and J. Marsden. Tricubic interpolation in three dimensions. *International Journal for Numerical Methods in Engineering*, 63(3):455–471, 2005. doi:10.1002/nme.1296.

[85] L. Lindblom, M. A. Scheel, L. E. Kidder, R. Owen, and O. Rinne. A new generalized harmonic evolution system. *Classical and Quantum Gravity*, 23(16): S447–S462, July 2006. doi:10.1088/0264-9381/23/16/s09.

[86] L. Lindblom, B. J. Owen, and D. A. Brown. Model waveform accuracy standards for gravitational wave data analysis. *Physical Review D*, 78(12), Dec. 2008. doi:10.1103/physrevd.78.124020. URL https://doi.org/10.1103/physrevd.78.124020.

[87] F. Löffler, J. Faber, E. Bentivegna, T. Bode, P. Diener, R. Haas, I. Hinder, B. C. Mundim, C. D. Ott, E. Schnetter, G. Allen, M. Campanelli, and P. Laguna. The Einstein Toolkit: A community computational infrastructure for relativistic astrophysics. *Classical and Quantum Gravity*, 29(11):115001, May 2012. doi:10.1088/0264-9381/29/11/115001.

[88] M. Macleod. Fast nearly ML estimation of the parameters of real or complex single tones or resolved multiple tones. *IEEE Transactions on Signal Processing*, 46(1): 141–148, 1998. doi:10.1109/78.651200.

[89] M. Maggiore, C. V. D. Broeck, N. Bartolo, E. Belgacem, D. Bertacca, M. A. Bizouard, M. Branchesi, S. Clesse, S. Foffa, J. García-Bellido, S. Grimm, J. Harms, T. Hinderer, S. Matarrese, C. Palomba, M. Peloso, A. Ricciardone, and M. Sakellariadou. Science case for the Einstein telescope. *Journal of Cosmology and Astroparticle Physics*, 2020(03):050–050, Mar. 2020. doi:10.1088/1475-7516/2020/03/050.

[90] J. M. Martí and E. Müller. Numerical hydrodynamics in special relativity. *Living Reviews in Relativity*, 2(1), Dec. 1999. doi:10.12942/lrr-1999-3.

[91] J. M. Martí, J. M. Ibáñez, and J. A. Miralles. Numerical relativistic hydrodynamics: Local characteristic approach. *Physical Review D*, 43(12):3794–3801, June 1991. doi:10.1103/physrevd.43.3794.

[92] S. T. McWilliams, B. J. Kelly, and J. G. Baker. Observing mergers of nonspinning black-hole binaries. *Physical Review D*, 82(2), July 2010. doi:10.1103/physrevd.82.024014.

[93] M. C. Miller et al. The radius of PSR J0740+6620 from NICER and XMM-Newton data. *The Astrophysical Journal Letters*, 918(2):L28, Sept. 2021. doi:10.3847/2041-8213/ac089b.

[94] C. Misner, K. Thorne, and J. Wheeler. *Gravitation*. W. H. Freeman, 1973. ISBN 9780716703341.

[95] E. R. Most, L. J. Papenfort, and L. Rezzolla. Beyond second-order convergence in simulations of magnetized binary neutron stars with realistic microphysics. *Monthly Notices of the Royal Astronomical Society*, 490(3):3588–3600, Oct. 2019. doi:10.1093/mnras/stz2809.

[96] E. R. Most, S. P. Harris, C. Plumberg, M. G. Alford, J. Noronha, J. Noronha-Hostler, F. Pretorius, H. Witek, and N. Yunes. Projecting the likely importance of weak-interaction-driven bulk viscosity in neutron star mergers. *Monthly Notices of the Royal Astronomical Society*, 509(1):1096–1108, Oct. 2021. doi:10.1093/mnras/stab2793.

[97] P. Mösta, B. C. Mundim, J. A. Faber, R. Haas, S. C. Noble, T. Bode, F. Löffler, C. D. Ott, C. Reisswig, and E. Schnetter. `GRHydro`: A new open-source general-relativistic magnetohydrodynamics code for the Einstein toolkit. *Classical and Quantum Gravity*, 31(1):015005, Nov. 2013. doi:10.1088/0264-9381/31/1/015005.

[98] T. Nakamura, K. Oohara, and Y. Kojima. General relativistic collapse to black holes and gravitational waves from black holes. *Progress of Theoretical Physics Supplement*, 90:1–218, 1987. doi:10.1143/ptps.90.1.

[99] E. O'Connor and C. Sullivan. NuLib open-source neutrino interaction library, accessed 2021. URL https://github.com/evanoconnor/NuLib.

[100] J. R. Oppenheimer and G. M. Volkoff. On massive neutron cores. *Physical Review*, 55(4):374–381, Feb. 1939. doi:10.1103/physrev.55.374.

[101] C. Palenzuela, S. L. Liebling, D. Neilsen, L. Lehner, O. Caballero, E. O'Connor, and M. Anderson. Effects of the microphysical equation of state in the mergers of magnetized neutron stars with neutrino cooling. *Physical Review D*, 92(4), Aug. 2015. doi:10.1103/physrevd.92.044045.

[102] C. Palenzuela et al. Turbulent magnetic field amplification in binary neutron star mergers, 2021. URL https://arxiv.org/abs/2112.08413.

[103] A. Perego, S. Bernuzzi, and D. Radice. Thermodynamics conditions of matter in neutron star mergers. *The European Physical Journal A*, 55(8), Aug. 2019. doi:10.1140/epja/i2019-12810-7.

[104] E. Pian et al. Spectroscopic identification of r-process nucleosynthesis in a double neutron-star merger. *Nature*, 551(7678):67–70, Oct. 2017. doi:10.1038/nature24298.

[105] A. Prakash, D. Radice, D. Logoteta, A. Perego, V. Nedora, I. Bombaci, R. Kashyap, S. Bernuzzi, and A. Endrizzi. Signatures of deconfined quark phases in binary neutron star mergers. *Physical Review D*, 104(8), Oct. 2021. doi:10.1103/physrevd.104.083029.

[106] M. Punturo et al. The Einstein Telescope: A third-generation gravitational wave observatory. *Classical and Quantum Gravity*, 27(19):194002, Sept. 2010. doi:10.1088/0264-9381/27/19/194002.

[107] M. Pürrer and C.-J. Haster. Gravitational waveform accuracy requirements for future ground-based detectors. *Physical Review Research*, 2(2), May 2020. doi:10.1103/physrevresearch.2.023151.

[108] B. Quinn. Estimation of frequency, amplitude, and phase from the DFT of a time series. *IEEE Transactions on Signal Processing*, 45(3):814–817, Mar. 1997. doi:10.1109/78.558515.

[109] D. Radice. Binary neutron star merger simulations with a calibrated turbulence model. *Symmetry*, 12(8):1249, July 2020. doi:10.3390/sym12081249.

[110] D. Radice, F. Galeazzi, J. Lippuner, L. F. Roberts, C. D. Ott, and L. Rezzolla. Dynamical mass ejection from binary neutron star mergers. *Monthly Notices of the Royal Astronomical Society*, 460(3):3255–3271, May 2016. doi:10.1093/mnras/stw1227.

[111] D. Radice, S. Bernuzzi, and A. Perego. The dynamics of binary neutron star mergers and GW170817. *Annual Review of Nuclear and Particle Science*, 70(1): 95–119, Oct. 2020. doi:10.1146/annurev-nucl-013120-114541.

[112] D. Radice, S. Bernuzzi, A. Perego, and R. Haas. A new moment-based general-relativistic neutrino-radiation transport code: Methods and first applications to neutron star mergers. *Monthly Notices of the Royal Astronomical Society*, 512(1): 1499–1521, Mar. 2022. doi:10.1093/mnras/stac589.

[113] C. Reisswig and D. Pollney. Notes on the integration of numerical relativity waveforms. *Classical and Quantum Gravity*, 28(19):195015, Sept. 2011. doi:10.1088/0264-9381/28/19/195015.

[114] D. Reitze, R. X. Adhikari, S. Ballmer, B. Barish, L. Barsotti, G. Billingsley, D. A. Brown, Y. Chen, D. Coyne, R. Eisenstein, M. Evans, P. Fritschel, E. D. Hall, A. Lazzarini, G. Lovelace, J. Read, B. S. Sathyaprakash, D. Shoemaker, J. Smith, C. Torrie, S. Vitale, R. Weiss, C. Wipf, and M. Zucker. Cosmic Explorer: The U.S. contribution to gravitational-wave astronomy beyond LIGO. *Bulletin of the AAS*, 51(7), Sept. 2019. URL https://baas.aas.org/pub/2020n7i035.

[115] L. Rezzolla and O. Zanotti. *Relativistic Hydrodynamics*. Oxford University Press, Sept. 2013. doi:10.1093/acprof:oso/9780198528906.001.0001.

[116] L. Rezzolla, L. Baiotti, B. Giacomazzo, D. Link, and J. A. Font. Accurate evolutions of unequal-mass neutron-star binaries: Properties of the torus and short GRB engines. *Classical and Quantum Gravity*, 27(11):114105, May 2010. doi:10.1088/0264-9381/27/11/114105.

[117] T. E. Riley et al. A NICER view of the massive pulsar PSR J0740+6620 informed by radio timing and XMM-Newton spectroscopy. *The Astrophysical Journal Letters*, 918(2):L27, Sept. 2021. doi:10.3847/2041-8213/ac0a81.

[118] G. B. Rybicki and A. P. Lightman. *Radiative Processes in Astrophysics*. Wiley-VCH Verlag GmbH & Co. KGaA, May 1985. doi:10.1002/9783527618170.

[119] N. Sanchis-Gual, P. J. Montero, J. A. Font, E. Müller, and T. W. Baumgarte. Fully covariant and conformal formulation of the Z4 system in a reference-metric approach: Comparison with the BSSN formulation in spherical symmetry. *Physical Review D*, 89(10), May 2014. doi:10.1103/physrevd.89.104033.

[120] A. Schmitt and P. Shternin. Reaction rates and transport in neutron stars. In *The Physics and Astrophysics of Neutron Stars*, pages 455–574. Springer International Publishing, 2018. doi:10.1007/978-3-319-97616-7_9.

[121] A. S. Schneider, L. F. Roberts, and C. D. Ott. Open-source nuclear equation of state framework based on the liquid-drop model with Skyrme interaction. *Physical Review C*, 96(6), Dec. 2017. doi:10.1103/physrevc.96.065802.

[122] A. S. Schneider, C. Constantinou, B. Muccioli, and M. Prakash. Akmal-Pandharipande-Ravenhall equation of state for simulations of supernovae, neutron stars, and binary mergers. *Physical Review C*, 100(2), Aug. 2019. doi:10.1103/physrevc.100.025803.

[123] E. Schnetter, S. Brandt, S. Cupp, R. Haas, P. Mösta, and S. Shankar. CarpetX, accessed 2022. URL https://zenodo.org/record/6131529.

[124] M. Shibata. *Numerical Relativity*. World Scientific, May 2015. doi:10.1142/9692.

[125] M. Shibata and T. Nakamura. Evolution of three-dimensional gravitational waves: Harmonic slicing case. *Physical Review D*, 52(10):5428–5444, Nov. 1995. doi:10.1103/physrevd.52.5428.

[126] M. Shibata, K. Taniguchi, and K. Uryū. Merger of binary neutron stars with realistic equations of state in full general relativity. *Physical Review D*, 71(8), Apr. 2005. doi:10.1103/physrevd.71.084021.

[127] D. M. Siegel, P. Mösta, D. Desai, and S. Wu. Recovery schemes for primitive variables in general-relativistic magnetohydrodynamics. *The Astrophysical Journal*, 859(1):71, May 2018. doi:10.3847/1538-4357/aabcc5.

[128] L. Smarr and J. W. York. Kinematical conditions in the construction of spacetime. *Physical Review D*, 17(10):2529–2551, May 1978. doi:10.1103/physrevd.17.2529.

[129] A. W. Steiner, M. Hempel, and T. Fischer. Core-collapse supernova equations of state based on neutron star observations. *The Astrophysical Journal*, 774(1):17, Aug. 2013. doi:10.1088/0004-637x/774/1/17.

[130] J. M. Stone et al. Athena++, accessed 2022. URL https://www.athena-astro.app/.

[131] B. Szilágyi. Key elements of robustness in binary black hole evolutions using spectral methods. *International Journal of Modern Physics D*, 23(07):1430014, June 2014. doi:10.1142/s0218271814300146.

[132] K. Takami, L. Rezzolla, and L. Baiotti. Spectral properties of the post-merger gravitational-wave signal from binary neutron stars. *Physical Review D*, 91(6), Mar. 2015. doi:10.1103/physrevd.91.064001.

[133] The Cosmic Explorer Consortium. Cosmic Explorer, accessed 2022. URL https://cosmicexplorer.org/.

[134] The Einstein Telescope Collaboration. The Einstein Telescope, accessed 2022. URL http://www.et-gw.eu/.

[135] J. Thornburg. *Numerical relativity in black hole spacetimes*. PhD thesis, The University of British Columbia, 1993. URL https://doi.library.ubc.ca/10.14288/1.0085621.

[136] TOP500. June 2022 | TOP500, accessed 2022. URL https://www.top500.org/lists/top500/2022/06/.

[137] E. F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer Berlin Heidelberg, 2009. doi:10.1007/b79761.

[138] S. Typel, G. Röpke, T. Klähn, D. Blaschke, and H. H. Wolter. Composition and thermodynamics of nuclear matter with light clusters. *Physical Review C*, 81(1), Jan. 2010. doi:10.1103/physrevc.81.015803.

[139] R. M. Wald. *General Relativity*. University of Chicago Press, 1984. doi:10.7208/chicago/9780226870373.001.0001.

[140] J. R. Wilson and G. J. Mathews. *Relativistic Numerical Hydrodynamics*. Cambridge University Press, Nov. 2003. doi:10.1017/cbo9780511615917.

[141] A. J. Wright and I. Hawke. Resistive and multi-fluid RMHD on graphics processing units. *The Astrophysical Journal Supplement Series*, 240(1):8, Jan. 2019. doi:10.3847/1538-4365/aaf1b0.

[142] T. Yamamoto, M. Shibata, and K. Taniguchi. Simulating coalescing compact binaries by a new code (SACRA). *Physical Review D*, 78(6), Sept. 2008. doi:10.1103/physrevd.78.064054.

[143] W. Zhang, A. Myers, K. Gott, A. Almgren, and J. Bell. AMReX: Block-structured adaptive mesh refinement for multiphysics applications. *The International Journal of High Performance Computing Applications*, 35(6):508–526, June 2021. doi:10.1177/10943420211022811.