

BINARY NEUTRON STAR SIMULATIONS: NEW TOOLS AND INSIGHTS

by

Trevor Vincent

A thesis submitted in conformity with the requirements
for the degree of Doctor of Philosophy
Graduate Department of Physics
University of Toronto

Copyright © 2018 by Trevor Vincent

Abstract

Binary Neutron Star Simulations: New Tools and Insights

Trevor Vincent

Doctor of Philosophy

Graduate Department of Physics

University of Toronto

2018

On September 14th, 2015 LIGO detected gravitational waves from a merging binary black hole. Just under two years later, LIGO detected gravitational waves from merging neutron stars in coincidence with detections of a gamma ray burst and kilonova. Many more detections of this kind are expected in the future. There are several problems that are facing the numerical relativity community now that binary neutron stars (BNS) have been detected by LIGO and this thesis will tackle two of them. We first present a new numerical scheme that aims to lay the groundwork for more realistic BNS simulations in the future. This code uses discontinuous Galerkin numerical methods to solve elliptic problems on curvilinear and non-conforming meshes in parallel on large supercomputers. We test this code on several problems in numerical relativity, including one that mimicks the discontinuities in the phase transitions of a Neutron star, as well as the two and three black-hole initial gravitational data problem. This code will be used in the future to obtain highly accurate initial gravitational data for BNS simulations. Following this, we present a set of twelve new BNS simulations using a current state-of-the-art numerical relativity code. We study the neutrino and matter emission from this set of twelve simulations to establish trends between the source parameters and the emission from the binary. Large sets of simulations of BNS mergers will be crucial for the upcoming LIGO observation runs to maximize the extractions of astrophysical information from the instrument data. Finally we conclude the thesis by overviewing work that is currently being done by collaborators to progress the tools and insights presented in this thesis.

Acknowledgements

When I first began my PhD journey, I never would have dreamed that I would be attending the press conference for the first detection of gravitational waves at the Albert Einstein Institute in Hannover, Germany, or that I would later become a member of the Nobel prize-winning LIGO collaboration. For this and many other things, I have to give a big thank you to my supervisor Professor Harald Pfeiffer. Harald not only gave me the opportunity to research within one of the most exciting fields currently in astrophysics, but provided me with multiple opportunities to present my research abroad. He also gave me the opportunity to attend several summer schools and work overseas at the Albert Einstein Institute in Golm, Potsdam. On top of all of this, Harald has helped me get through multiple tough problems that arose in my research and provided the initial ideas for the projects in this thesis.

Secondly, thank you to Professor Francois Foucart who taught me how to run and analyze binary neutron stars simulations with the neutrino transport scheme in the Spectral Einstein Code. Chapter 3 of this thesis would not be possible without his knowledge, support and constant help with debugging the (seemingly) hundreds of problems that cropped up while running these simulations.

Thirdly, thank you to my committee members Professor Charles Dyer and Professor Amar Vutha who provided me excellent advice both during the committee meeting and outside of our formal meetings. I still miss the General Relativity class parties at Prof. Dyer's house.

Fourthly, thank you to Krystyna Biel for helping me with all the logistical duties required throughout the PhD, her understanding, thoughtfulness and efficiency was greatly appreciated.

Lastly, a big thank you to my family, the constant support of my dad, mother, sister and grandad, and the great friends I made in and outside the physics department at the U of T during my PhD: Gunjan Lakhani, Nazim Boudjada, Amir Abdul-Reda, Daniel Dick, Heather Fong, Adam Lewis, Nick Tacik, Carl-Johan Haster, Aaron Zimmerman, Prayush Kumar, Alex Kostenko, Helen Baxter, Daniel Baker, Fang Xi Lin, Phil Berger, George Stein, Dana Simard, Claire Yu, Mubdi Rahman, Julien Dyer, Alex Bercik, Hannah Fronenberg, Luca Talamo and

Peter Pangarov and out of town friends Jonah Miller, Nils Deppe, Paul Sarte, Nils Fischer, Stephen Green, Peter Zimmerman, Kyohei Kawaguchi, Sergei Ossokine, Roland Haas, Tim Dietrich. These people made the PhD journey unforgettable and I thank them all for the great times.

Contents

1	Introduction	1
1.1	Binary Neutron Stars (BNS): Einstein’s Richest Laboratory	4
1.2	Simulating BNS: General relativistic hydrodynamics	11
1.3	Simulating BNS: Current problems	14
1.4	Simulating BNS: New numerical methods	16
2	A hp-adaptive discontinuous Galerkin solver for elliptic equations in numerical relativity	19
2.1	Chapter Overview	19
2.2	Introduction	20
2.3	Numerical Algorithm	22
2.3.1	DGFEM discretization	22
2.3.2	Penalty Function	32
2.3.3	Norms	32
2.3.4	Multigrid-Preconditioned Newton-Krylov	32
2.3.5	hp-Adaptivity	45
2.3.6	Implementation	50
2.4	Test Examples	51
2.4.1	Poisson with $H^{4-\epsilon}$ solution	51
2.4.2	Constant density star	52
2.4.3	Cubed sphere Meshes and Stretched Boundary Elements	55
2.4.4	Puncture Initial Data	61
2.5	Conclusion and Future Work	69

3 Unequal Mass Binary Neutron Star Simulations with M1 Neutrino Transport: Ejecta and Neutrino Emission	71
3.1 Chapter Overview	71
3.2 Introduction	72
3.3 Numerical Implementation	75
3.3.1 General Overview	75
3.3.2 Initial Data	76
3.3.3 Spacetime Evolution	77
3.3.4 Neutrino Evolution	79
3.3.5 Fluid Evolution	85
3.3.6 Composition Evolution	85
3.3.7 Equation of State (EOS)	86
3.3.8 Initial Models	88
3.3.9 Grid Setup	89
3.3.10 Ejecta Analysis	92
3.3.11 Errors	93
3.4 Numerical Results	94
3.4.1 General Overview	94
3.4.2 Matter outflow	97
3.4.3 Neutrino Emission	106
3.5 Conclusion and Future Work	111
4 Conclusions & Future Work	114
4.1 Conclusions	114
Bibliography	116

List of Tables

2.1	The randomly generated parameters for the three black holes. We list the mass m, the position $(x, y, 0)$, the momentum $(P_x, P_y, 0)$ and the spin $(0, 0, S_z)$ of the black holes.	68
3.1	Parameters for the initial models presented in this paper. M_1 and M_2 are the gravitational masses of the two neutron stars, C_1 and C_2 are the compactness of the respective neutron stars and d_0 is the initial coordinate distance between the centers of stars. The last column gives the post-merger remnant as of 7.5-ms post-merger. If the binary collapsed before 7.5-ms a rough estimate for the time of collapse is shown in brackets.	90
3.2	Finite difference grid sizes for the initial models. All simulations were set so that they would have a 300m resolution during merger. For reference, the DD2 $1.44M_\odot$ star has a $\sim 10.48\text{km}$ radius in our grid coordinates.	91
3.3	The first two wide-columns provide the mass (M_{ej}), average electron fraction ($\langle Y_e \rangle$) and average asymptotic velocity ($\langle v_\infty \rangle$) of the matter labelled unbound from time $t=0$ up to 7.5ms post-merger (or collapse) in the polar and equatorial regions. The last wide-column gives the mass (M_{ej}), average electron fraction ($\langle Y_e \rangle$) and average asymptotic velocity ($\langle v_\infty \rangle$) over all regions.	98

List of Figures

- 2.1 Ingredients into the setup of the domain-decomposition: (a) The macro-mesh \mathbb{E}_0 of a 2-dimensional disk consisting of five macro-elements. (b) A representative mesh derived from the macro-mesh by refining once in the left-most macro-element. (c) The boundary mortar Γ_B . (d) The interior mortar Γ_I 24
- 2.2 A representation of a 2:1 interface in two dimensions. On the left we have a 4th-order element and on the right we have two elements of degree 8 and degree 2 respectively. The data on the faces of these three elements will be used in the mortar integrals of equation 2.5, but since the interface is non-conforming, we need to interpolate the face data to a shared broken polynomial space on the mortar. To ensure that the face data on each element is in a polynomial space which a subset of the polynomial space on the mortar, we demand that the polynomial degree p on the mortar is the maximum of the polynomial degrees on either side. For the lower mortar face (red) this would be $\max(4, 3) = 4$ and for the upper mortar face (green) this would be $\max(4, 6) = 6$. Since the left element must interpolate its face data to a space containing two faces, we use the hp-interpolation operator \mathcal{I}_{hp} . On the right side, each element has to map its data from one face to one face, so we use the p-interpolation operator \mathcal{I}_p 31

2.3	A structural representation of a multigrid v-cycle. The nodes in yellow are actual grid-levels, while the nodes in blue represent surrogate grids which are not necessarily 2:1 balanced. In order for a multigrid v-cycle to represent a symmetric operation, the grid levels along the down arrow are exactly the same as the grid-levels along the up arrow. Level 0 represents the coarsest possible mesh.	36
2.4	A simple 2-D Schwarz subdomain, with no h-nonconforming or p-nonconforming boundaries. In grey is the element e which is the center of the subdomain. The light grey area is the overlap (of size δ_ξ) into the other elements. The subdomain is composed of everything in light and dark grey.	41
2.5	A plot of the Schwarz weighting function defined by Eq. (2.39). The dark grey shaded region is the center element of the Schwarz subdomain e_c and the light grey shaded region is the overlap of width δ_ξ . For this plot, we set $\delta_\xi = .25$. . .	43
2.6	Problem A, Eq. (2.55): Convergence of the energy norm estimator η^2 and the error between the numerical solution u_h and the analytic solution u in the energy norm $\ \cdot\ _*$ (Eq. 2.33) and the L_2 norm $\ \cdot\ _2$ (Eq. 2.34).	52
2.7	Problem A, Eq. (2.55): Visualization of the solution and the hp-refined computational mesh. Top portion: The computational grid, color-coded by polynomial degree, with height representing the solution u . Bottom portion: Color-coded by h -refinement level. A cell on level l has size 2^{-l} of the overall computational domain.	53
2.8	Problem A, Eq. (2.55): Iteration-count of the flexible conjugate gradient (FCG) Krylov subspace solver versus the Multigrid preconditioned FCG solver (MG-FCG). The MG-FCG completes all solves in ≤ 3 iterations.	53
2.9	Problem B, Eq. (2.56): Convergence of the energy norm estimator η^2 and the error between the numerical solution u_h and the analytic solution u in the energy norm $\ \cdot\ _*$ and the $\ \cdot\ _2$ norm	55

2.10 Comparison of the average number of iterations per Newton-Raphson step when using a Chebyshev smoothed Multigrid preconditioner and no preconditioner.	56
2.11 Visualization of hp-refined computational mesh for Problem B, Eq. (2.56). Top portion: The $z = 0$ cross-section of the computational grid, color-coded by the polynomial degree and with height representing the solution ψ . Bottom portion: Volume rendering of the $z \leq 0$ part of the computational domain, color-coded by the h -refinement level. A cell on level l has size 2^{-l} of the overall computational domain.	57
2.12 The mesh structure for a computational domain with spherical outer boundary. This structure consists of 13 macro-meshes (shown in different colors). For clarity, only the $z \leq 0$ part of the mesh is shown.	58
2.13 Problem C, Eq. (2.64): Convergence of the solution as the degree p is uniformly increased across all elements for uniform case and as the degree p is adaptively increased in the amr case. (Dirichlet BC at $R=1000$).	59
2.14 Visualization of the final mesh for problem C. Shown is a xy-plane slice of the mesh which has been warped so that the height of the surface corresponds to the value u of the solution, color coded by the polynomial degree. For ease of visualization, grid-points are mapped onto the compactified grid on which the estimator η_e is computed; the compactified outer radius $R = 3$ corresponds to the physical outer radius $R = 1000$	60
2.15 Problem C, (Eq. 2.64): Convergence of the solution in the L_2 norm (See Eq. 2.34) as the degree p is adaptively refined with a compact and non-compact estimator. (Dirichlet BC at $R=1000$ and $R = 10^9$). All runs use the AMR/dG parameters of the p-amr run in Figure 2.13.	61
2.16 Comparison of four different methods of preconditioning FCG. All runs use the AMR/dG parameters of the p-amr run in Figure 2.13.	62

2.17 Convergence of the SpEC–elliptic solver with manually adopting the domain-decomposition and manual adjustment of resolution to compensate for the singularities. Plotted are differences to the next- <i>lower</i> resolution at four points in the computational domain.	64
2.18 Problem D: Black hole initial data with two punctures. Convergence of the error between the dG solution and the SpEC solution.	65
2.19 Problem D (Black hole initial data with two punctures): Visualization of the hp-refined computational mesh. The bottom portion of the image shows a volume rendering of the $z < 0$ portion of the computational domain, with two blocks removed, and color-coded by the h-refinement level. The top portion of the image shows the $z = 0$ cross-section of the computational grid, color-coded by the polynomial degree, with the height representing the solution u . For ease of visualization, grid-points are mapped onto the compactified grid on which the estimator η_e is computed; the compactified outer radius $R = 3$ corresponds to the physical outer radius $R = 10^{11}$	66
2.20 Problem E (Black hole initial data with three punctures): Visualization of the solution, and the hp-refined computational mesh. Top portion: the $z = 0$ cross-section of the computational grid, color-coded by the polynomial degree, with the height representing the solution u . Middle portion: volume rendering of the $z < 0$ part of the computational domain, with two blocks removed, and color-coded by the h-refinement level. Bottom portion: Zoom into the middle portion, highlighting the region near the three punctures with highest refinement level. For ease of visualization, grid-points are mapped onto the compactified grid on which the estimator η_e is computed; the compactified outer radius $R = 3$ corresponds to the physical outer radius $R = 10^{11}$	67

2.21 Problem D: Black hole initial data with three randomly generated punctures.	
Convergence of the error between AMR steps at four different points, three corresponding to the location of the punctures and one at (100, 0, 0).	68
3.1 M-R curves for each of the equations of states used in this work.	87
3.2 The evolution of rest mass density ρ_0 for the D144144 model. For the first ~ 18 ms the stars orbit each other before merging at ~ 0 ms. At around ~ 3 ms we can see a post-merger remnant almost fully formed with two tidal tails. At ~ 7.5 ms the post-merger remnant has largely settled and most of the unbound material has left the grid.	95
3.5 The electron fraction Y_e for the $1.2M_\odot + 1.32M_\odot$ models at 7.5ms post merger. The black contours encapsulate material that is flagged unbound. The white contours encapsulate material that is above densities $10^9, 10^{11}, 10^{13} g/cm^3$ respectively, with the higher density contours appearing closer to remnant core.	100
3.6 v_∞ distribution of the polar and equatorial ejecta across different EOS for the $1.2M_\odot + 1.32M_\odot$ models.	101
3.7 Y_e distribution of the polar and equatorial ejecta across different EOS for the $1.2M_\odot + 1.32M_\odot$ models.	102
3.8 v_∞ distribution of the polar and equatorial ejecta across different mass ratios of the SFHo EOS.	104
3.9 Y_e distribution of the polar and equatorial ejecta across different mass ratios of the SFHo EOS.	105
3.10 Disk masses at 7.5-ms post-merger or just before collapse if remnant is BH. . .	106

3.11 Anti-electron neutrino energy density (first moment) for the $1.44M_{\odot} + 1.44M_{\odot}$ DD2 model at 3ms post-merger. top portion: A xz-slice of the anti-electron neutrino energy density $E_{\bar{\nu}_e}$ with the arrows showing the effective neutrino transport velocity $v_{\nu}^i = \alpha \frac{E_{\bar{\nu}_e}}{F_{\bar{\nu}_e}^i} - \beta^i$. We see the largest energy density near the polar regions of the remnant, where the density is less and the neutrinos are free to stream. bottom portion: A xy-slice of the anti-electron neutrino energy density $E_{\bar{\nu}_e}$ with the arrows showing the effective neutrino transport velocity. We see the neutrinos advecting with the fluid in the dense regions near the core.	108
3.12 Anti-electron neutrino luminosity for the 12132 models up to around 7.5 post-merger. We see that for the softer SFHo EOS, there is a larger luminosity. Similar trends hold for the other neutrino species.	109
3.13 Neutrino luminosity for DD2 runs up to around 7.5ms across neutrino species.	109
3.14 Neutrino flux density moment as a function of angle for different EOS runs at 7.5ms. The angle is defined with respect to the equatorial plane with 0° being the equator, and $\pm 90^{\circ}$ being the North and South poles. The different neutrino species are color coded.	110
3.15 Electron fraction distribution for the off-grid ejecta of the $1.2M_{\odot} + 1.32M_{\odot}$ SFHo model at 7.5 ms and 10 ms post-merger. We can clearly see that more matter is being ejected at higher Y_e fractions due to neutrino irradiation.	111

Chapter 1

Introduction

The initial proposals for gravitational wave interferometers were constructed in the late 1980s with the scientific goal to detect the inspiral and merger of compact-object binaries. About two decades later, the LIGO (LIGO et al. (2018)), Virgo (Acernese et al. (2015)) and GEO600 (Affeldt et al. (2014)) detectors have been developed and operated as a network from the period of 2005 to 2010. No detections were made during this initial stage of sensitivity. Alongside the early-2000 interferometer development, numerical simulations of the Einstein equations were beginning to gather ground. In 2000, the first binary neutron star merger was simulated (Shibata & Uryū (2000)), five years later the first binary black hole merger was computed after years of trial and error (Pretorius (2005a)) and finally in 2006 the first binary neutron star - black hole simulations were performed (Shibata & Uryū (2006)). A variety of numerical relativity groups started forming at this time all around the globe, the Caltech-Cornell-CITA group (SXS; black-holes.org), the Kyoto/Tokyo group (Nagakura et al. (2014)), the University of Illinois at Urbana-Champaign group (UIUC), to name but a few. These groups started building simulation sets to aid in the parameter estimation studies that were expected to follow from the first detections. On September 9th, 2015, in a astonishing event during a engineering run, LIGO detected the gravitational wave from two coalescing black holes (Abbott et al. (2016b)). Since this day, there have been multiple binary black hole waveform detections (LIGO et al. (2018)). More recently however, LIGO made a landmark detection, GW170817 (Abbott et al. (2017a)).

GW170817 coincided with the detection of a gamma ray burst, GRB 170817A and a series of observations that followed across the electromagnetic spectrum codenamed *atf2017gw* (Villar et al. (2017)). The inferred masses of the bodies and the variety of electromagnetic observations imply that the source was a neutron star binary.

With these detections and the awarding of the 2017 Nobel prize to three LIGO members, the gravitational era of astronomy was born. More detectors are expected in the future. With the upcoming earth-based detectors KAGRA (Somiya (2012)) and LIGO-India (gw-indigo.org), there will be a worldwide detector network capable of precise source localization. On top of this, there are already plans for third generation earth-based detectors such as the Einstein Telescope (Punturo et al. (2010)) and space based detectors such as LISA (elisascience.org) which will cover an entirely different frequency band. With more and more detectors, both space-based and earth-based, we will be able to gather data from the full frequency spectrum of gravitational waves directed towards Earth.

The LIGO detection search and parameter estimator pipeline relies heavily on waveforms computed from numerical relativity. LIGO detects signals using a technique called matched-filtering, which compares instrument data against large catalogs of theoretically modelled waveform templates. These templates are created using a mix of post-Newtonian expansions, semi-analytic models tuned to numerical relativity and full numerical relativity waveforms (Sachdev et al. (2019)). Furthermore, to extract the best astrophysical information from the instrument data, it is crucial to have numerical relativity waveforms. Parameter estimation in the first LIGO detection GW150914, of two coalescing black holes, used multiple semi-analytic models, tuned to numerical relativity waveforms, to determine source characteristics (Abbott et al. (2016a)). For the LIGO detection GW170817, there was both gravitational and electromagnetic emission. Post-Newtonian (PN) waveforms were used to estimate source parameters from the instrument data because in the frequency range detected the binary was well modelled by the PN expansion (Abbott et al. (2017a)). Even though the gravitational waveform did not capture details about the post-merger remnant, the electromagnetic emission from GW170817 provides

information about this stage. Using the EM observations, fully relativistic simulations of binary neutron star mergers were used to help extract information about the remnant and constrain different aspects of the equation of state of nuclear matter (Radice et al. (2017); Shibata et al. (2017)).

Even though numerical relativity has matured greatly since the major developments of the early 2000's, there is still a lot more that needs to be achieved. The biggest problems today in numerical relativity are the simulation of compact object binaries with matter and the simulation of supernovae, both requiring a great amount of microphysics, multi-scale grids, large sets of complex non-linear PDES, fast supercomputers and modern numerical techniques. With the coming of the exascale age of supercomputing there is an increasingly realistic chance of simulating these systems with all the known microphysics. This thesis aims to make progress in two areas of numerical relativity, both aiming at laying the groundwork for future BNS codes and analysis of LIGO BNS detections. First, we seek to improve the computational techniques used to solve the Einstein field equations so that more realistic microphysics may be introduced into the BNS simulations as computing power reaches exascale and beyond. To do this, we develop a new numerical scheme and code, which we test on several problems in numerical relativity, including a problem that mimicks the phase transitions discontinuities of a neutron star and the problem of two and three-black-hole initial gravitational data. This is outlined in Chapter 2. Secondly, we seek to further probe the parameter space of binary neutron star simulations with one of the most state of the art numerical relativity codes to help understand the emission properties of these LIGO sources for the next generation of detections. This is outlined in Chapter 3. Thus, this thesis is dedicated to the improving not only the current understanding of BNS mergers with the best numerical techniques of today, but paving the way for more realistic future BNS merger simulations using new exciting methods. The remaining portion of this introductory chapter will briefly describe the physics and computational techniques needed to understand the context of Chapters 2 and 3. We begin our introduction in Chapter 1 with a broad overview of the astrophysics of binary neutron star mergers. Then in the following

sections, we narrow our focus to the numerical solution of the general-relativistic equations of hydrodynamics for BNS mergers and the problems that current numerical solvers have with these equations. We end Chapter 1 by introducing a new numerical method called discontinuous Galerkin which we plan on using to improve future BNS simulations.

1.1 Binary Neutron Stars (BNS): Einstein’s Richest Laboratory

In the last section we outlined the grand scheme of this thesis and the chapters to follow. Now, we will start delving deeper into the binary neutron stars. In this section, we give a high-level overview of binary neutron star systems and the computational work that has been performed to model them.

The first binary neutron star system discovered was PSR B1913+16 by Russell Hulse and Joseph Taylor in 1974 (Hulse & Taylor (1975)). From observations of radio-wave pulses of the stars, it was determined that the orbit of PSR B1913+16 was shrinking at a rate of 3.5 meters per year, precisely the amount predicted by the loss of energy in gravitational wave emission (Weisberg et al. (1981)). For their work, Hulse and Taylor were awarded the 1993 Nobel Prize in Physics. Since then, radio telescopes have discovered roughly a dozen binary neutron star systems, all within our Galaxy (Baiotti & Rezzolla (2016)). However, the presence of extragalactic short gamma-ray bursts, believed to be powered by binary neutron star mergers, gives indirect evidence for the existence of even more neutron star binaries. Furthermore, with the first BNS detection by LIGO, codenamed GW170817, we can expect many more gravitational wave detections of BNS systems in the future. Using updated models which include the GW170817 observation, LIGO expects a detection rate of $110 - 3840 \text{ Gpc}^{-3} \text{ yr}^{-1}$ (LIGO et al. (2018)). This detection rate coupled with a detection radius of $\sim 107 \text{ Mpc}$ for the LIGO-Hanford observatory (Abbott et al. (2017a)) means a lower bound of ~ 10 BNS mergers could be seen by LIGO per year. With the upcoming LIGO A+ upgrades it is expected that by the 2026, we may be detecting one per week (LIGO Document G1601435-v3).

There are three main stages of a neutron star binary system. These stages can be distinguished using the radius R of the neutron stars and their separation a . For $a \gg R$, the binaries orbit each other with a separation that slowly decreases in a stage known as the **inspiral** (Faber & Rasio (2012); Baumgarte & Shapiro (2010); Shibata (2015); Rezzolla & Zanotti (2013)). In this stage, the stars are well approximated as point particles and orbit each other with speeds not exceeding $\sim 0.2c$ (Shibata (2015)). During the late inspiral, when the stars are orbitting each other multiple times per second ($\mathcal{O}(10)Hz$), they start being visible by the LIGO detector. As the neutron stars approach and $a \sim 5R$, they become tidally deformed. The amount of deformation is governed by the equation of state (EOS) of the nuclear matter. At this stage the stars are orbitting at an extreme rate ($\mathcal{O}(500)Hz$). As $a \sim 30km$ the stars plunge toward each other and merge into a single hypermassive neutron star, ejecting a large amount of material in the process (Shibata (2015)). This ejecta fuses together new nuclei whose decay releases optical radiation in an event called a “kilonova” (Metzger (2016)). This stage is known as the **merger** (Faber & Rasio (2012); Baumgarte & Shapiro (2010); Shibata (2015); Rezzolla & Zanotti (2013)). Whether this newly formed massive neutron star collapses down to a black-hole depends on the total mass of the system and the EOS. For most EOS, a binary of total mass close to $3M_{\odot}$ will collapse within a few milliseconds after merger (Bauswein et al. (2013)). Once the remant has settled into either a black-hole or a massive neutron star, we have reached the **ringdown** stage (Faber & Rasio (2012); Baumgarte & Shapiro (2010); Shibata (2015); Rezzolla & Zanotti (2013)). The dynamics and observational signatures of the ringdown stage are dominated by the left-over accretion disk, which drives magnetic, viscous and neutrino winds and potentially provides the energy source for a collimated polar jet creating one of the most luminous events in the universe, the short gamma ray burst. Both the merger and the ringdown stages are outside the LIGO band, so these stages cannot be probed via gravitational waves until the next-generation of detectors is here (Abbott et al. (2017d)).

Each stage of the binary coalescence is marked by differences in the gravitational wave. The early inspiral is represented by a sinusoidal pattern as the neutron stars slowly orbit each other

like point particles (Bauswein & Stergioulas (2019)). As the binary first enters the LIGO band in the late inspiral, the stars become tidally deformed. This tidal deformation induces a quadrupole moment which changes the shape of the waveform (Hinderer et al. (2010)). The magnitude of the tidal deformation is governed by the EOS, so measurement of the tidal deformability can put constraints on the EOS and such constraints were computed for GW170817 (Raithel (2019)). As the stars plunge toward each other, the frequency grows rapidly and reaches its peak at merger, this rapid growth of the frequency is often called the “chirp” because it resembles the chirp sound of a sliding whistle if the frequencies were converted to sound waves. As the cores-bounce together and rotate during merger, the proto-remnant emits waves with a multiltude of frequencies with most of the emission in three major peaks (the largest being from the rotation of the proto-remnant and the other two from the fundamental frequencies of the bouncing spring-like cores) (Takami et al. (2015)). These peaks are EOS dependent and thus can lead to constraints on the EOS, but unfortunately these peaks were outside of the LIGO frequency band for GW170817 (Abbott et al. (2017d)). Lastly as the remnant settles, there is a rapid decay of the amplitude as energy is lost from the remnant and it “rings down”. For GW170817, neither the ringdown nor the merger stage could provide us with constraints the EOS. The late inspiral however provided the a measure of the tidal deformability. Using this measurement, the radius of the primary neutron star if the binary had a mass-ratio $q = 1$ was constrained to be $9.8 < R < 13.2$ with a maximum mass of $2.3M_{\odot}$ if the remnant collapsed to a black hole (Raithel (2019)). However, some studies have argued that the late-time electromagnetic emission from GW170817 can be better explained by a long-lived neutron star remnant (Yu et al. (2018)). Therefore, depending on the assumed fate of the binary, implications for M_{max} may vary significantly. In future LIGO observation runs, post-merger gravitational waves from the remnant may provide a clear indication of the remnant’s fate and thus provide us with better constraints on R and M_{max} .

Until GW170817 there were two big mysteries surrounding BNS. The first mystery involved the so-called rapid neutron capture (r-process) elements. It was unclear for decades how

approximately half of the elements in the Milky Way galaxy heavier than iron were formed. The formation of these elements required environments where the density of free neutrons would be so high that neutron captures on nuclei proceed much faster than β decays. This series of neutron captures was called the r-process and such elements were therefore labelled r-process elements. In 1974, Lattimer and Schramm proposed that r-process elements could be formed in the neutron-rich matter ejected from neutron star - black-hole mergers (Lattimer & Schramm (1974)). Some years later, Symbalisty and Schramm proposed that that a similar mechanism of mass ejection could occur from binary neutron star systems and power the r-process (Symbalisty & Schramm (1982)). It was shown in the decade following that Newtonian BNS simulations could produce ejecta from tidal disruption with the desired neutron-richness, speed and mass to potentially produce the r-process elements (Davies et al. (1994); Ruffert et al. (1996); Rosswog et al. (1998); Freiburghaus et al. (1999)). In the years that followed fully-general relativistic simulations were performed and showed that there was another type of ejecta, matter ejected out the poles of the binary from the shocks of the cores contracting and recoiling upon impact (Oechslin & Janka (2006); Hotokezaka et al. (2013); Sekiguchi et al. (2015); Foucart et al. (2015a)). This type of ejecta was neutron-poor, much faster and in general less massive (Foucart et al. (2015a)). Further simulations have shown that the resulting accretion disk after a remnant settles can also ejecta matter via neutrino, magnetic and viscous winds with a variety of neutron-richness, masses and speeds (Fernández & Metzger (2013)). Alongside these simulations, work was being achieved on the theoretical side of astrophysical r-process emission models. In 1998, Li and Paczynski first proposed that the radioactive ejecta of a NS merger could power a supernova-like thermal transient, but they did not have a realistic model of the radioactive heating (Li & Paczynski (1998)). In 2010, Metzger et al. first calculated the emission luminosities following using a realistic modelling of the radioactive heating of decaying r-process nuclei (Metzger et al. (2010)). They predicted peak luminosities of $3 \times 10^{41} \text{ ergs}^{-1}$ for $10^{-2} M_{\odot}$ of ejecta expanding at $v \approx 0.1c$ with a spectral peak at visual wavelengths. As this was roughly 1000 times more luminous than classical novae, they named these events “kilonovae”.

The color of the kilonova distinguishes the type of ejecta involved. In particular, ejecta with $Y_e \lesssim 0.25$, such as the shocked polar ejecta, lacks enough neutrons to create r-process elements past $A \approx 140$ and a blue-colored fast moving kilonova is produced (Metzger & Fernández (2014)). At least all mergers should have $Y_e \lesssim 0.2$ ejecta from tidal-tail ejecta or disk winds, this produces a red slower-moving kilonova which should be a universal feature of all mergers. Disk wind ejecta tends to be isotropic and produces kilonovae that are both blue and red in nature. For the LIGO GW170817 detection, the electromagnetic counterpart was codenamed AT2017fgo. Over the first few days the transient colors were blue and rapidly-evolving with a spectral peak at visual wavelengths (e.g. Soares-Santos et al. (2017); Nicholl et al. (2017); ?); McCully et al. (2017); Cowperthwaite et al. (2017);). At later times, the colors became substantially redder and slowly evolved on timescales of several days. The total mass of the red ejecta was estimated to be $410^{-2} M$ with a somewhat lower velocity $v \approx 0.1c$ than the blue ejecta (e.g. Cowperthwaite et al. (2017); Chornock et al. (2017); Nicholl et al. (2017)). The quantity of blue ejecta from AT2017fgo was estimated to be $\sim 1 - 2 \times 10^{-2} M_\odot$ with a faster velocity of $v \approx 0.2c$ (Cowperthwaite et al. (2017); Nicholl et al. (2017)), based on fitting the observed light curves to kilonova models (Metzger (2017a)).

The second mystery involves the most luminous events in the universe, the Gamma ray burst (GRB). GRBs were first detected in the late 1960s by U.S military satellites sent into orbit to spy on the Soviet Union nuclear testing (Klebesadel et al. (1973)). The satellites were trained to detect short bursts of gamma-rays from expected nuclear explosions. The satellites started measuring gamma rays very frequently, but this work remained classified until it was determined the bursts were of cosmological origin (Klebesadel et al. (1973)). Very little was known about these gamma-ray bursts until the launch of the Compton Gamma Ray Observatory (CGRO) which operated between 1991 and 2000 (Fishman & Meegan (1995)). The CGRO satellite recorded over 2700 gamma ray bursts with the Burst And Transient Source Experiment (BATSE) (Fishman & Johnson (1989)). This experiment showed that the GRBs were isotropically distributed across the sky and originating outside of our galaxy. The BATSE results also showed the existence

of two distinct populations of GRBs referred to as short and long GRBs (Kouveliotou et al. (1993)). The former category typically last for less than 2 seconds and the latter for more than 2 seconds with distribution peaks at 0.2 seconds and 20 seconds respectively. The two types of bursts clearly had different origins. The main progenitors considered were supernovae and compact-object binaries with at least one neutron star (Levan et al. (2016)). It was found that the long bursts happened in star-forming galaxies and could usually be associated with a type Ic core-collapse supernova, whereas the short bursts could not (Hjorth (2013)). This meant that while supernovae were the likely progenitor for long bursts, the progenitor for short bursts must be different. The prevailing model for the short bursts was either a BNS or NSBH merger, although the latter has not yet been detected. It was only until GW170817 and the associated GRB 170817A that the BNS progenitor model for the short bursts was confirmed. About 1.7 seconds after the gravitational wave hit earth, FERMI and INTEGRAL registered a gamma ray burst which was short in duration (2.0s) but sub-luminous ($3 \times 10^{46} \text{ ergs/s}$) compared to other charted short GRBs (Abbott et al. (2017;c)). The 1.7s delay was lucky because it allowed LIGO to place an upper bound on the speed of gravity, with calculations showing a relative difference between the speed of light and speed of gravity of about $\sim 7 \times 10^{-16}$ (Abbott et al. (2017)). While the sub-luminosity was mysterious, later analysis showed that it could be readily explained by an off-axis jet (20-40 degrees) and thus, if we had viewed GRB 170817A head-on it would have been as luminous as the other charted sGRBs.

On top of solving two mysteries, constraining the EOS and the speed of gravity, GW170817 also put constraints on the expansion rate of the universe, H_0 (Collaboration et al. (2017)). With so much science from one detection, the future appears very bright for gravitational wave astronomy and it is clear why BNS are called Einstein’s Richest Laboratory (not just because of the 100 octillion U.S. dollars worth of gold produced in the ejecta!) (Baiotti & Rezzolla (2016)). However, on the numerical side, a lot still needs to be done. Despite continuous developments, current numerical relativity codes have not yet reached the accuracy required to model the gravitational wave signal and the electromagnetic counterparts at the level required to extract

as much information as possible from future detections (see (Baiotti & Rezzolla (2016)). High accuracy is required because small numerical errors in the initial data solves or the evolution lead to larger errors in the waveform (see e.g. Tsokaros et al. (2016)). High-accuracy is very difficult to obtain currently because of the multi-scale nature of the problem, we have to have high enough resolution to resolve the Neutron star and the MRI instability ($\sim 100 - 200m$ and $\sim 10cm$ respectively), but a large enough grid to extract the waveforms far away from the remnant (1600km). On top of this, most codes do not take into account all of the microphysics relevant to the evolution of the post-merger remnant, including a hot nuclear-theory based equation of state, a neutrino transport scheme accounting for both neutrino-matter and neutrino-neutrino interactions and the evolution of the magnetic fields with enough resolution to resolve the growth of magneto-hydrodynamics instabilities (Foucart et al. (2015a)). That being said, slowly but surely, different levels of micro-physics are being added to the codes. The first papers that studied fully general relativistic BNS simulations including the effects of neutrinos were (Neilsen et al. (2014); Palenzuela et al. (2015)) with a simple leakage scheme and (Sekiguchi et al. (2015)) with a more complex M1 neutrino transport scheme. Both the leakage scheme and M1 transport scheme are approximate methods to deal with neutrinos which are preferred because solving the 7-dimensional Boltzmann transport equation for the neutrino distribution during a BNS simulation is currently numerically intractable. For a review of leakage and the M1 transport scheme see (Foucart et al. (2015b)). These BNS simulations with neutrino cooling and neutrino transport have focused mainly on equal mass systems with $M_{ns} = 1.35M_{\odot}$. Collectively these papers find that the ejected mass is only substantial enough to explain the total mass of r-process heavy elements in our galaxy for r-process nucleo-synthesis in the case of a softer equation of state (e.g., more compact stars). The first paper on BNS mergers with neutrino interactions using the SpEC code looked at $1.2M_{\odot}$ equal mass systems and compared a simple leakage cooling neutrino scheme with the more complicated gray M1 neutrino transport scheme, finding that the more realistic transport scheme had a significant affect on the disk composition and the outflows, producing more neutron rich material that could possibly seed r-process element

creation. (Radice et al. (2016)) examined the effects of eccentricity and neutrino cooling on the matter outflows and remnant disk of a LS220 equal mass binary, finding that both had significant effects, with the absence of a neutrino scheme leading to matter outflows a factor of 2 off. Finally, only very recently have there been studies examining the effects on matter outflows due to mass asymmetry in the initial binaries, with both (Lehner et al. (2016)) and (Sekiguchi et al. (2016a)) finding that mass asymmetry can affect the neutron-richness and total ejecta for both soft and stiff equations of state.

In Chapter 3 of this thesis, we will be contributing to the above growing set of BNS studies, by looking at the effect of mass asymmetry and different EOSs on matter and neutrino emissions. We will be using the SpEC code by the SXS collaboration (black-holes.org) which now has a state-of-the-art neutrino transport scheme.

1.2 Simulating BNS: General relativistic hydrodynamics

In the previous section we gave a broad overview of the physics of binary neutron star systems and the computational work to simulate their merger. In this section we take a closer look at the equations involved to simulate BNS systems, namely, the Einstein field equations and the equations of relativistic hydrodynamics.

In their usual form, space and time are treated on equal footing in the Einstein equations. From the perspective of performing a numerical evolution however, we need to reformulate the problem as an initial value problem where we have a set of initial gravitational and matter data at some time t and a set of evolution equations which we can use to get the updated data at a later time. To do this, we make the ansatz that space-time can be treated as a time sequence of spatial hypersurfaces. With this ansatz, the space-time metric $g_{\mu\nu}$ can be decomposed into

$$ds^2 = g_{\mu\nu}dx^\mu dx^\nu = -\alpha^2 dt^2 + \gamma_{ij}(dx^i + \beta^i dt)(dx^j + \beta^j dt), \quad (1.1)$$

where the spatial metric γ_{ij} is a function of the spatial coordinates x^i and t , α is the lapse function that measures proper time between neighboring hypersurfaces along their timelike unit

normals n^μ and β^i is called the shift, which determines how coordinate labels move between each hypersurface. This is known as the 3+1 decomposition of the metric (Arnowitt et al. (1959)).

Now we need to decompose the Einstein field equations into a set of evolution and constraint equations involving the quantities β^i , α and γ_{ij} . To do this, we introduce the projection operator $\perp_\beta^\alpha = \delta_\beta^\alpha + n^\alpha n_\beta$, which can be easily proven to project the space-time components orthogonal to n^μ out of any space-time vector.

The three possible projections of the Einstein field equations: $n^\mu n^\nu G_{\mu\nu} = 8\pi n^\mu n^\nu T_{\mu\nu}$, $n^\mu \perp_\delta^\nu G_{\mu\nu} = 8\pi n^\mu \perp_\delta^\nu T_{\mu\nu}$, $\perp_\rho^\mu \perp_\delta^\nu G_{\mu\nu} = 8\pi \perp_\rho^\mu \perp_\delta^\nu T_{\mu\nu}$ lead to the three York/ADM 3+1 equations respectively:

$$0 = R + K^2 - K^{mn} K_{mn} - 16\pi\rho, \quad (1.2)$$

$$0 = D_i K - D_m K_i^m + 8\pi j_i, \quad (1.3)$$

$$\begin{aligned} \partial_t K_{ij} = & \beta^m \partial_m K_{ij} + K_{mj} \partial_i \beta^m + K_{im} \partial_j \beta^m - D_i D_j \alpha \\ & + \alpha (R_{ij} + K K_{ij} - 2 K_{im} K_j^m) + 4\pi\alpha [(S - \rho)\gamma_{ij} - 2S_{ij}], \end{aligned} \quad (1.4)$$

where $K_{ij} = \beta^m \partial_m \gamma_{ij} + \gamma_{mj} \partial_i \beta^m + \gamma_{im} \partial_j \beta^m - \partial_t \gamma_{ij}$, is called the extrinsic curvature and it measures the rate at which the hypersurface deforms as it is carried forward along a normal (Baumgarte & Shapiro (2010)). We also introduced the projections of the stress tensor: $\rho = T_{\mu\nu} n^\mu n^\nu$, $j_\alpha = - \perp_\alpha^\nu T_{\mu\nu} n^\mu$, $S_{\alpha\beta} = \perp_\alpha^\mu \perp_\beta^\nu T_{\mu\nu}$ and $S = \gamma^{\mu\nu} S_{\mu\nu}$, while R_{ij} and R denote the Ricci tensor and scalar associated with γ_{ij} .

The 3+1 equations are a set of 10 equations. Equation 1.2 is known as the Hamiltonian constraint equation. Equation 1.3 is called the momentum constraint equation and is composed of three equations. In total, these four elliptic constraint equations play a similar role as the equations $\nabla \cdot \vec{E} = 4\pi\rho$ and $\nabla \cdot \vec{B} = 0$ which constrain the initial \vec{E} and \vec{B} fields in Maxwell's equations. The constraint equations must be solved prior to evolving the initial data and are usually called the **initial data equations**. The last set of six equations, Eqn. 1.4, are the evolution equations. As they stand, the 3+1 equations are ill-posed and still need to be manipulated a bit in order to discretize and solve them on a computer. For the set of evolution equations, the most

used schemes are BSSN (Baumgarte & Shapiro (1998); Shibata & Nakamura (1995)), Z4 (Bona et al. (2003)) and the Generalized Harmonic Decomposition (Pretorius (2005b); technically, the Generalized Harmonic decomposition doesn't start from the ADM equations, see Chapter 3 for more details) which each manipulate the evolution equations into a slightly different well-posed hyperbolic system. For the initial data equations, the most used schemes are conformal transverse traceless (CTT; Bowen & York Jr (1980)), conformal-thin sandwich (CTS; York Jr (1999)) and the extended-conformal thin-sandwich (XCTS; Pfeiffer & York Jr. (2005)) frameworks.

Alongside the field-equations, the matter equations must be solved. These come from the local conservation equations $\nabla_\mu T^{\mu\nu} = 0$ (this changes for the case of radiation-hydrodynamics, see Chapter 3). For Neutron-star matter, a perfect-fluid tensor and an irrotational velocity distribution are usually assumed and are coupled with some choice of equation of state (EOS) to complete the system(Faber & Rasio (2012)). For black-holes, we set the source terms $\rho = j_\alpha = S_{\alpha\beta} = 0$ because there is only vacuum space. On top of the matter-equations, the equations of neutrino transport must be solved, we will look at this in detail in Chapter 3. Furthermore, if magnetic fields are included, we must also solve the equations of magnetohydrodynamics (Baumgarte & Shapiro (2003); Shibata & Sekiguchi (2012)).

To extract the gravitational waveform, the Weyl scalar ψ_4 , which represents the outgoing transverse radiation, is extracted at a large radius away from the simulated binary system (the “wave-zone”). The energy, linear momentum and angular momentum of the gravitational wave are computed by integrating the ψ_4 scalar in time (Kyutoku et al. (2015)).

For a more in-depth review, see (Sperhake (2014)), (Faber & Rasio (2012)) and (Shibata & Taniguchi (2011)) for BBH, NSNS and NSBH systems respectively. In the next section we discuss problems that are currently plaguing solvers for the initial data equations and then in the subsequent section we discuss new numerical methods that we plan to use to fix these problems.

1.3 Simulating BNS: Current problems

To solve the hydrodynamics equations $\nabla_\mu T_{\mu\nu} = 0$ for a binary neutron star simulation, we require an equation of state (EOS). In its most general form the term equation of state is used for any relation between thermodynamic state variables. For the case of a hydrodynamics description of a neutron star fluid, a equation of state relates the pressure of the fluid P , to it's density ρ , electron fraction Y_e and temperature T . In simulations there are two current approaches to modeling neutron stars with temperature-dependent realistic EOSs in binaries. The first is to use EOSs in tabulated form (a set of (P, ρ, Y_e, T) points) and to interpolate between tabular points as required during the simulation. Since this can be computationally expensive, more approximate approaches have been tried. The most common approximant is to use n piece-wise polytropes with a polytropic thermal correction (Deaton et al. (2013); Kyutoku (2013); Bauswein et al. (2014); Kyutoku et al. (2015)). Following this approach, the cold ($T \approx 0$) piece-wise polytropic part of the EOS is defined as follows

$$P_{cold} = K_i \rho^{\Gamma_i}, \quad \epsilon_{cold} = \epsilon_i + \frac{K_i \rho^{\Gamma_i - 1}}{\Gamma_i - 1}, \quad (1.5)$$

where P , ρ , ϵ , K and Γ are the pressure, rest-mass density, specific internal energy, polytropic constant and polytropic index respectively, with i denoting the $i - th$ polytropic piece. It was found that four polytropic pieces ($n = 4$) will approximate most EOSs of cold neutron matter to good accuracy (Read (2008)). However, a piecewise-polytopic EOS is only valid for neutron stars with a temperature $T \approx 0$. In neutron star mergers, temperatures can reach up to $T \sim 50 MeV$, so a polytropic EOS would not be realistic. To include thermal effects when using piecewise polytropes, one splits up the pressure $P = P_{cold} + P_{th}$ and internal energy $\epsilon = \epsilon_{cold} + \epsilon_{th}$ and assumes an ideal gas relationship $P_{th} = \rho \epsilon_{th} (\Gamma_{th} - 1)$ where the ideal gas index Γ_{th} is constant for all ρ and ϵ and usually set to 2. (Bauswein et al. (2010); Takami et al. (2014)).

The main issue with solving the Einstein equations and matter equations with realistic EOSs in tabulated or with an approximate piecewise-analytic form is that the EOS are not smooth due to the multiple phase transitions and the non-analytic behavior or very steep slopes at the surface of

the NS. For example, let's consider the LS220 EOS (a commonly used phenomenological nuclear matter EOS for astrophysical application, see Lattimer & Swesty (1991)) at low temperature and electron fraction. Near the NS center the EOS can be approximated by a stiff $\Gamma \approx 7/2$ polytrope and then at slightly lower densities ($\approx 10^{14} g/cm^3$) the EOS begins to soften with $\Gamma \approx 1/2$ and then at even lower densities, the EOS asymptotically approaches the adiabatic index of a relativistic Fermi gas, $\Gamma \approx 4/3$. In certain coordinates, these changes can happen very close to the neutron star surface and are difficult to resolve (Deaton et al. (2013)). Furthermore, initial data solves for BNS and NSBH binaries have traditionally used multi-domain spectral finite element methods such as the SpEC elliptic solver Spells (Pfeiffer et al. (2003a)) and LORENE (Gourgoulhon et al. (2001)). Spectral methods have the nice feature of exponential convergence when the underlying problem is smooth. However, when the problem is non-smooth such as is the case when we consider tabulated EOSs or piecewise-analytic EOS fits (e.g. piecewise-polytropes), spectral methods show poor convergence. In order to retain convergence, subdomains are often inserted by hand around the non-analytic parts (Deaton et al. (2013)). Finally, for currently unknown reasons, the elliptic solver Spells (Pfeiffer et al. (2003a),Foucart et al. (2008),Henriksson et al. (2014)) used by the SXS collaboration (black-holes.org) does not converge for high compactness NS initial-data without multiple extra corrective iteration schemes and even then, it cannot converge to high accuracy data when the binary objects are very close in separation (Henriksson et al. (2014)). This lack of convergence could be due to mathematical non-uniqueness in solutions of the XCTS formulation of the Einstein constraint equations (Cordero-Carrión et al. (2009)), but (Henriksson et al. (2014)) hints at the possibility that the compactness can be pushed far beyond previous simulations by using refined iteration schemes. In the next section we discuss a new powerful numerical technique that might provide us with a way around these problems.

1.4 Simulating BNS: New numerical methods

In the last section we outlined problems with current numerical solution techniques when handling binary neutron star initial data. In this section, we look at the current numerical techniques in detail and then describe a new method called discontinuous Galerkin that combines the best aspects of these current techniques. Discontinuous Galerkin (DG) methods will allow us to go beyond known problems with current binary neutron star initial data. To introduce DG methods, we must first look at the standard numerical techniques used today in numerical relativity. The most common methods used in Numerical Relativity to solve the evolution or initial data equations are finite difference, finite volume methods and spectral methods (Baumgarte & Shapiro (2010)). We introduce all three methods by examining the following PDE:

$$R(\bar{u}) = 0. \quad (1.6)$$

Here R is some partial differential operator, usually referred to as the residual, that operates on the solution u which could be composed of several fields we are solving for (e.g. the four fields in the initial data equations or the 10 components of the metric). Equation 1.6 represents a continuous equation which must be discretized in order to be solved on a computer. Finite difference, finite volume and spectral methods are different ways of taking continuous equations and making them discrete and therefore they are called discretization methods. Each of the different discretization methods used in numerical relativity demand in a different way, that the discrete version of Eqn. 1.6 be zero. In the finite difference method we set $R(u)$ to be zero at each of the grid points and then we Taylor expand the differential operators and represent the solution as a single number at each grid point. In finite volume methods we set the integral of the residual over an element (usually called a “cell”) of a mesh to be zero in each of the elements and we represent the solution as a single number per cell (the average of the solution over that cell). In the spectral finite element method we break up the domain into a set of elements with simple topologies and expand the solution over a set of basis functions. We demand that the

L_2 -inner product between the residual and each of these basis functions is zero. That is, on each subdomain we have

$$\int R(\bar{u})\psi_j dx = 0 \quad \forall \psi_j. \quad (1.7)$$

Recently, a new discretization method has become popular. Discontinuous Galerkin (DG) methods (Reed & Hill (1973); Hesthaven & Warburton (2008); Cockburn (2001); Cockburn & Shu (1998); Cockburn (1998); Cockburn et al. (2000)) combine the power of spectral methods in smooth regions with the shock-handling features of finite volume methods in discontinuous regions. To accomplish this, DG methods represent the solution on each element as an expansion over a set of basis functions with the residual satisfying Eqn. 1.7. To couple the solution between elements, DG methods use flux terms between neighboring elements which penalize deviation of the solution at the interface. These flux terms allow DG methods to borrow the discontinuity handling techniques of the finite volume method. With these features, DG methods can potentially obtain exponential convergence even when the solution is not smooth over the entire grid (see Chapter 2 for more details). On top of this, DG methods allow us to implement several nice features:

1. **hp-adaptivity:** In DG methods there are two types of refinement, you can increase the number of basis functions (p-refinement), or you can split an element of the mesh into smaller elements (h-refinement). In smooth regions, p-refinement is preferred and in non-smooth regions h-refinement is preferred. The combination of these two types of refinement (hp-refinement) can lead to rapid convergence of the solution.
2. **Minimal communication:** Since each element only needs the nearest-neighbour face data to compute the penalty flux terms, the amount of communication between processors is minimal.
3. **Easy Boundary handling:** Boundary elements can be treated just like interior elements with the boundary conditions being applied through the penalty terms in the flux. This

makes algorithms like Multigrid much easier to implement, since no special treatment is required for boundary elements.

4. Easy geometry handling: Unlike finite-difference methods which require special finite difference stencils for the boundaries of curved domains, DG methods can be applied as is to any type of domain, without any significant changes.

In the next Chapter we will outline a novel discontinuous Galerkin code we developed. We will showcase all of the features listed above by solving several problems in numerical relativity with our code. The future goal of this code will be to solve for BNS initial data with realistic EOS, spins and mass-ratios to high accuracy.

Chapter 2

A hp-adaptive discontinuous Galerkin solver for elliptic equations in numerical relativity

2.1 Chapter Overview

The material in this chapter is based on "A hp-adaptive discontinuous Galerkin solver for elliptic equations in numerical relativity" by Trevor Vincent, Harald Pfeiffer, Nils Fischer being prepared for Phys. Rev. D.

A considerable amount of attention has been given to discontinuous Galerkin methods for hyperbolic problems in Numerical Relativity, showing potential advantages of the methods in dealing with hydrodynamical shocks and other discontinuities. This paper investigates discontinuous Galerkin methods in numerical relativity to solve problems of elliptic nature. We present a novel hp-adaptive numerical scheme for curvilinear and non-conforming meshes. It uses a multigrid preconditioner with a Chebyshev or Schwarz smoother to create a very scalable discontinuous Galerkin code on generic domains. The code employs compactification to move the outer boundary near spatial infinity. We explore the properties of the code on some test

problems, including one mimicking Neutron stars with phase transitions. We also apply it to construct initial data for two or three black holes.

2.2 Introduction

Discontinuous Galerkin (DG) methods (Reed & Hill (1973); Hesthaven & Warburton (2008); Cockburn (2001); Cockburn & Shu (1998); Cockburn et al. (2000)) have matured into standard numerical methods to simulate a wide variety of partial differential equations. In the context of numerical relativity (Baumgarte & Shapiro (2010)), discontinuous Galerkin methods have shown advantages for relativistic hyperbolic problems over traditional discretization methods such as finite difference, finite volume and spectral finite elements (Field et al. (2010); Brown et al. (2012); Field et al. (2009); Zumbusch (2009); Radice & Rezzolla (2011); Mocz et al. (2014); Zanotti et al. (2014); Endeve et al. (2015); Teukolsky (2016); Bugner et al. (2016); Schaal et al. (2015); Zanotti et al. (2015); Miller & Schnetter (2017)) by combining the best aspects of all three methods. As computers reach exa-scale power, new methods like DG are needed to tackle the biggest problems in NR such as realistic supernovae and binary neutron-star merger simulations on these very large machines (Kidder et al. (2017)).

DG efforts in numerical relativity have so far targeted evolutionary problems only. This paper explores DG for elliptic problems in numerical relativity. We present an elliptic solver with the following primary features:

1. It operates on curved meshes, with non-conforming elements.
2. It supports adaptive h and p refinement, driven by a posteriori error estimators.
3. It employs multi-grid for efficient solution of the resulting linear systems.
4. It uses compactified domains to treat boundary conditions at infinity.

While each of these features has appeared in the literature before (Arnold et al. (2002); Stiller (2017); Hesthaven & Warburton (2008); Kronbichler & Wall (2018); Fick (2014); Kozdon &

Wilcox (2018); Kozdon et al. (2019)), to our knowledge, our solver for the first time combines all these elements simultaneously and demonstrate their effectiveness on difficult numerical problems.

Specifically, the present article is a step toward a solver for the Einstein constraint equations, which must be solved to construct initial data for the evolution of compact binary systems (Pfeiffer (2005); Cook (2000); Baumgarte & Shapiro (2010)). The constraint equations are generally rewritten as elliptic equations, and depending on detailed assumptions, this results in one or more coupled non-linear elliptic partial differential equations. Construction of initial data is arguably the most important elliptic problem in numerical relativity, but not the only one: Elliptic equations also occur in certain gauge conditions (Baumgarte & Shapiro (2010)) or for implicit time-stepping to alleviate the computational cost of high-mass-ratio binaries (Lau et al. (2009; 2011)).

The motivation for developing a new solver is multi-fold. First, current spectral methods have difficulty obtaining certain initial data sets, such as binaries at short separation containing a neutron star, where the neutron star has high compactness and a realistic equation of state (Henriksson et al. (2016)). Furthermore, there is a need for a solver which can routinely obtain high-accuracy. Errors from inaccurate initial data sets creep into the evolutions with sizeable effect: Ref. (Tsokaros et al. (2016)) shows that despite only global (local) differences of 0.02% (1%) in the initial data of the two codes COCAL and LORENE for irrotational neutron-star binaries, the gravitational wave phase at the merger time differed by 0.5 radians after 3 orbits. The design of a more accurate code requires adaptive mesh refinement, load-balancing and scalability which a DG code potentially can provide. The present work also complements the DG evolution code presented in Ref. (Kidder et al. (2017)), leading to a complete framework for solving both elliptic and hyperbolic PDEs in numerical relativity.

The organization of the paper is as follows: Section 2.3 presents the components of our discontinuous Galerkin code. Section 2.4 showcases our hp-adaptive multigrid solver on increasingly challenging test-problems, each illustrating the power of the discontinuous Galerkin

method. This section includes a solution for the Einstein constraint equations in the case of a constant density star. This problem has a surface discontinuity which mimicks Neutron stars with phase transitions. Lastly, this section also presents solutions for initial-data of two orbiting non-spinning black-holes to showcase and compare it with the elliptic solver Spells (Pfeiffer et al. (2003b)) as well as solutions for the initial data of three black-holes with random locations, spins and momenta. We close with a discussion in Sec. 2.5.

2.3 Numerical Algorithm

2.3.1 DGFEM discretization

We base our nomenclature on (Arnold et al. (2002); Sherwin et al. (2006); Di Pietro & Ern (2011); Fick (2014); Kozdon & Wilcox (2018); Kozdon et al. (2019)), which we summarize here for completeness.

Mesh

Our purpose is to solve elliptic equations in a computational domain $\Omega \subset \mathbb{R}^d$ in d dimensions. To allow for non-trivial topology and shape of Ω , we introduce a *macro-mesh* (also known as a multi-block mesh) \mathbb{E}_0 consisting of macro-elements (also known as blocks) $e_0 \in \mathbb{E}_0$ such that (1) the macro-elements cover the entire domain, i.e., $\cup_{e_0 \in \mathbb{E}_0} e_0 = \Omega$, (2) the macro-elements touch each other at complete faces and do not overlap; and (3) each $e_0 \in \mathbb{E}_0$ is the image of the reference cube $[-1, 1]^d$ under a mapping $\Phi_{e_0} : [-1, 1]^d \rightarrow e_0$. As an example, Fig. 2.1 shows a macro-mesh of five elements covering a two-dimensional disk. The macro-mesh represents the coarsest level of subsequent mesh-refinement.

The macro-mesh \mathbb{E}_0 is refined by subdividing macro-elements into smaller elements along faces of constant reference cube coordinates. Refinement can be multiple levels deep (i.e. refined elements can be further subdivided), and we do not assume uniform refinement. The refined mesh \mathbb{E} is referred to as the *fine mesh*, which is exemplified in panel (b) of Fig. 2.1. In contrast

to the macro-mesh, the fine mesh will generally be non-conforming at element boundaries, both within one macro-element and at boundaries between macro-elements. We assume that there are at most a 2:1 refinement difference at any element boundary, i.e. the boundary of a coarse element faces at most two smaller elements (per dimension). Each face of each element $e \in \mathbb{E}$ is endowed with an outward-pointing unit-normal \hat{n} . The map Φ_{e_0} in macro-element $e_0 \in \mathbb{E}_0$ induces a map on each fine element e within e_0 , denoted

$$\Phi_e : [-1, 1]^d \rightarrow e. \quad (2.1)$$

The reference coordinates of Φ_e are linearly related to the reference coordinates of Φ_{e_0} .

Turning to boundaries, we define the set of all element boundaries (internal and external), $\Gamma = \cup_{e \in \mathbb{E}} \partial e$, where ∂e is the boundary of element e of the fine mesh. Γ is called the *mortar*, and it decomposes into a finite set of $(d - 1)$ -dimensional mortar elements $m \in \mathbb{M}$, arising from the faces of each mesh element $e \in \mathbb{E}$, s.t. $\Gamma = \cup_{m \in \mathbb{M}} m$ and each mortar element intersects at most at the boundary of two elements. \mathbb{M} splits into *interior mortar elements* \mathbb{M}_I and *exterior mortar elements* \mathbb{M}_B , with $\mathbb{M}_I \cap \mathbb{M}_B = 0$. Similarly, Γ splits into *interior mortar* $\Gamma_I = \cup_{m \in \mathbb{M}_I} m$ and *exterior mortar* $\Gamma_B = \cup_{m \in \mathbb{M}_B} m$, cf. Fig. 2.1, which intersect at $(d - 2)$ -dimensional edges where the interior mortar touches the exterior boundary. We partition the exterior mortar elements further into elements where we apply Dirichlet boundary conditions, \mathbb{M}_D , Neumann boundary conditions, \mathbb{M}_N , and Robin boundary conditions, \mathbb{M}_R , respectively. This induces sets of boundary points via $\Gamma_D = \cup_{m \in \mathbb{M}_D} m$, $\Gamma_N = \cup_{m \in \mathbb{M}_N} m$ and $\Gamma_R = \cup_{m \in \mathbb{M}_R} m$. We assume Γ_D , Γ_N and Γ_R are disjoint, i.e. one type of boundary condition is employed on each connected part of the boundary. Finally, we introduce two definitions to help us simplify equations later in the text. Firstly, because internal boundaries and external Dirichlet boundaries are often treated similarly, it is convenient to define $\mathbb{M}_{ID} = \mathbb{M}_I \cup \mathbb{M}_D$ and $\Gamma_{ID} = \Gamma_I \cup \Gamma_D$. Secondly, we refer to the set of mortar elements surrounding an element e as \mathbb{M}_e .

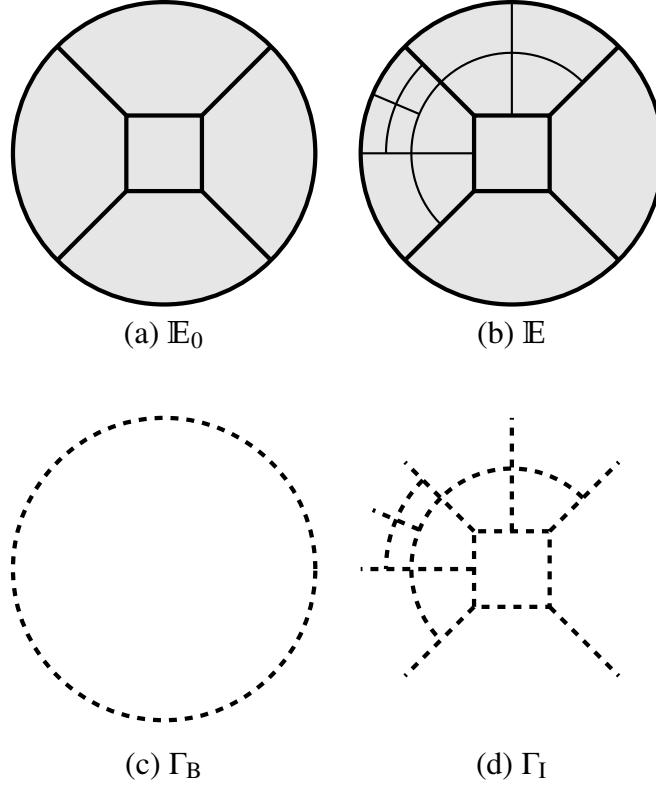


Figure 2.1: Ingredients into the setup of the domain-decomposition: (a) The macro-mesh \mathbb{E}_0 of a 2-dimensional disk consisting of five macro-elements. (b) A representative mesh derived from the macro-mesh by refining once in the left-most macro-element. (c) The boundary mortar Γ_B . (d) The interior mortar Γ_I .

Weak Equations

On the fine mesh \mathbb{E} , we wish to discretize the following model elliptic problem with the symmetric interior penalty discontinuous Galerkin method (Arnold et al. (2002)):

$$\partial_i \partial_i u(\mathbf{x}) + f(\mathbf{x})u = g(\mathbf{x}) \quad \mathbf{x} \in \Omega, \quad (2.2a)$$

$$u = g_D(\mathbf{x}) \quad \mathbf{x} \in \Gamma_D, \quad (2.2b)$$

$$\hat{n}_i \partial_i u = g_N(\mathbf{x}) \quad \mathbf{x} \in \Gamma_N, \quad (2.2c)$$

$$\hat{n}_i \partial_i u + \gamma u = g_R(\mathbf{x}) \quad \mathbf{x} \in \Gamma_R. \quad (2.2d)$$

In Eqs. (2.2) and in the following, we employ the Einstein sum convention, so that the first term in Eq. (2.2a) represents $\sum_{i=1}^d \partial_i \partial_i u$, and similar for the left hand side of Eqs. (2.2c) and (2.2d). In accordance with the underlying ideas of DG, we seek to approximate the solution of Eqs. (2.2) with polynomials on each element $e \in \mathbb{E}$, without strictly enforcing continuity across elements. Let V_{h,e,p_e} denote the set of polynomials on the reference cube $[-1, 1]^d$ up to degree p_e mapped to element $e \in \mathbb{E}$. We assume the same maximum polynomial order along each dimension; it is straightforward to extend to different polynomial order along different dimensions. The functions in V_{h,e,p_e} are understood to be extended by 0 outside of e (i.e. on other elements). The global function space is the direct sum of the per-element polynomial spaces,

$$V_h = \bigoplus_{e \in \mathbb{E}} V_{h,e,p_e}, \quad (2.3)$$

where the polynomial order may vary between elements. Because neighboring element touch, on internal boundaries Γ_I the discretized solution $u_h \in V_h$ will be represented twice on touching elements with generally different values on either element (this is origin of the term 'discontinuous' in DG).

The discretized solution $u_h \in V_h$ is determined, such that the residual of Eqs. (2.2a) is orthogonal to the function space V_h . Within the symmetric interior penalty discontinuous Galerkin discretization (Arnold et al. (2002); Di Pietro & Ern (2011)), this yields

$$L_h(u_h, v_h) = F_h(v_h) \quad \forall v_h \in V_h, \quad (2.4)$$

where

$$\begin{aligned} L_h(u, v) = & \int_{\mathbb{E}} \partial_i v \partial_i u \, dx - \int_{\Gamma_{ID}} [\![u]\!]_i \{\partial_i v\} \, ds \\ & - \int_{\Gamma_{ID}} [\![v]\!]_i \{\partial_i u\} \, ds + \int_{\Gamma_{ID}} \sigma [\![u]\!]_i [\![v]\!]_i \, ds \\ & + \int_{\mathbb{E}} f u v \, dx + \int_{\Gamma_R} \gamma u v \, ds, \end{aligned} \quad (2.5)$$

and

$$\begin{aligned}
F_h(v) = & \int_{\mathbb{E}} gv \, dx - \int_{\Gamma_D} g_D \hat{n}_i \partial_i v \, ds + \int_{\Gamma_D} \sigma g_D v \, ds \\
& + \int_{\Gamma_N} g_N v \, ds - \int_{\Gamma_D} g_R v \, ds.
\end{aligned} \tag{2.6}$$

For internal boundaries, the operators $\llbracket \cdot \rrbracket$ and $\{\!\{ \cdot \}\!\}$ are defined by

$$\llbracket q \rrbracket = q^+ \mathbf{n}^+ + q^- \mathbf{n}^- \quad \text{on } \Gamma_I, \tag{2.7}$$

$$\{\!\{ q \}\!\} = \frac{1}{2} (q^+ + q^-) \quad \text{on } \Gamma_I. \tag{2.8}$$

Here q is a scalar function, '+' and '-' is an arbitrary labeling of the two elements e^+ and e^- touching at the interface, and q^\pm and \mathbf{n}^\pm are the function values and the outward pointing unit normal on the two elements that share the interface. These operators are extended to external boundaries by

$$\llbracket q \rrbracket = q \mathbf{n} \quad \text{on } \Gamma_B, \tag{2.9}$$

$$\{\!\{ q \}\!\} = q \quad \text{on } \Gamma_B. \tag{2.10}$$

Breaking up the integrals in Eqs. (2.5) and (2.6) into integrals over individual mesh- and mortar-elements, one finds

$$\begin{aligned}
L_h(u, v) = & \sum_{e \in \mathbb{E}} \int_e \partial_i v \partial_i u \, dx - \sum_{m \in \mathbb{M}_{ID}} \int_m \llbracket u \rrbracket_i \{\!\{ \partial_i v \}\!\} \, ds \\
& - \sum_{m \in \mathbb{M}_{ID}} \int_m \llbracket v \rrbracket_i \{\!\{ \partial_i u \}\!\} \, ds + \sum_{m \in \mathbb{M}_{ID}} \int_m \sigma \llbracket u \rrbracket_i \llbracket v \rrbracket_i \, ds \\
& - \sum_{m \in \mathbb{M}_R} \int_m \gamma u v \, ds + \sum_{e \in \mathbb{E}} \int_e f u v \, dx,
\end{aligned} \tag{2.11}$$

and

$$\begin{aligned}
F_h(v) = & \sum_{e \in \mathbb{E}} \int_e gv \, dx - \sum_{m \in \mathbb{M}_D} \int_m g_D \hat{n}_i \partial_i v \, ds + \sum_{m \in \mathbb{M}_D} \int_m \sigma g_D v \, ds \\
& + \sum_{m \in \mathbb{M}_N} \int_m g_N v \, ds - \sum_{m \in \mathbb{M}_R} \int_m g_R v \, ds.
\end{aligned} \tag{2.12}$$

For later reference, we will refer to the first four integrals in Eq. (2.11) as the stiffness, consistency, symmetry and penalty integrals respectively and denote them $L_h^{\text{stiff}}(u, v)$, $L_h^{\text{con}}(u, v)$, $L_h^{\text{sym}}(u, v)$ and $L_h^{\text{pen}}(u, v)$ so that we may write

$$\begin{aligned} L_h(u, v) = & L_h^{\text{stiff}}(u, v) + L_h^{\text{con}}(u, v) + L_h^{\text{sym}}(u, v) + L_h^{\text{pen}}(u, v) \\ & + \sum_{e \in \mathbb{E}} \int_e fuv \, dx - \int_{\Gamma_R} \gamma uv \, ds. \end{aligned} \quad (2.13)$$

Basis Functions

So far, we have not yet specified a concrete basis for the polynomial spaces V_{h,e,p_e} introduced in Sec. 2.3.1. We do so now. Recall that curvilinear elements $e \in \mathbb{E}$ are mapped to reference cubic elements $[-1, 1]^d$. That is, each point $\mathbf{x} \in e$ corresponds to a reference cube coordinate $\boldsymbol{\xi} = \Phi_e^{-1}(\mathbf{x})$, cf. Eq. (2.1).

Along each dimension $\xi_i \in [-1, 1]$, where the subscript $i \in \{1, \dots, d\}$ denotes dimension, we first choose $N_i + 1$ Legendre-Gauss-Lobatto collocation points $\xi_{\alpha_i}^{\text{LGL}}$. The index $\alpha_i \in \{1, \dots, N_i + 1\}$ identifies the point along dimension i . We take $N_i \geq 1$ so that the points with $\xi_i = -1$ and $\xi_i = 1$, that lie on the faces of the cube, are always collocation points. The collocation points in all d dimensions form a tensor product grid of $\prod_{i=1}^d (N_i + 1)$ d -dimensional collocation points $\boldsymbol{\xi}_{\alpha}^{\text{LGL}} = (\xi_{\alpha_1}^{\text{LGL}}, \dots, \xi_{\alpha_d}^{\text{LGL}})$ that we index by $\alpha \in \{1, \dots, \prod_{i=1}^d (N_i + 1)\}$ that identifies a point regardless of dimension. With respect to these collocation points we can now construct the one-dimensional interpolating Lagrange polynomials

$$l_{\alpha_i}(\xi) := \prod_{\substack{\beta_i=1 \\ \beta_i \neq \alpha_i}}^{N_i+1} \frac{\xi - \xi_{\beta_i}^{\text{LGL}}}{\xi_{\alpha_i}^{\text{LGL}} - \xi_{\beta_i}^{\text{LGL}}}, \quad \xi \in [-1, 1] \quad (2.14)$$

and employ their tensor product to define the d -dimensional basis functions

$$\psi_{\alpha}(\boldsymbol{\xi}) = \prod_{i=1}^d l_{\alpha_i}(\xi_i). \quad (2.15)$$

When evaluating ψ_{α} in the physical coordinates $\mathbf{x} \in e$, one uses $\Phi_e^{-1} : \mathbf{x} \rightarrow \boldsymbol{\xi}$ to map the function argument. For instance, the set of test-functions on element $e \in \mathbb{E}$ becomes

$$V_{h,e,p_e} = \text{span} \{ \Phi_e^{-1} \circ \psi_{\alpha} \}. \quad (2.16)$$

Furthermore, because the ψ_α form a nodal basis, the expansion coefficients are the function values at the nodal points ξ^α , and each test-function can be written as

$$u_h^e = \sum_{\alpha} u_h^e(\xi_\alpha^{\text{LGL}}) \psi_\alpha = \sum_{\alpha} u_\alpha^e \psi_\alpha, \quad (2.17)$$

where $u_\alpha^e := u_h^e(\xi_\alpha^{\text{LGL}})$.

Semi-Discrete Global Matrix Equations

The global solution over the entire mesh is the direct sum of the solutions on each element, that is

$$u_h = \bigoplus_{e \in \mathbb{E}} u_h^e = \bigoplus_{e \in \mathbb{E}} \sum_{\alpha} u_\alpha^e \psi_\alpha^e. \quad (2.18)$$

Thus, with a chosen global ordering of elements $e \in \mathbb{E}$ and local ordering of basis functions ψ_α^e , we may prescribe a global index α' to each of the local expansion coefficients u_α^e . With this, we may now turn Eqs. (2.11) and (2.12) into a global linear system of equations. Let $\mathbb{L} = L_h(\psi_{\alpha'}, \psi_{\beta'}) =: L_{\alpha' \beta'}$, $\mathbf{u} = u_{\beta'}$ and $\mathbf{F} = F(\psi_{\alpha'}) =: F_{\alpha'}$, then the global linear system is

$$\mathbb{L}\mathbf{u} = \mathbf{F}. \quad (2.19)$$

Instead of forming the full global matrix \mathbb{L} and performing the matrix-vector operator $\mathbb{L}\mathbf{u}$ over the global space, we can restrict the integrals in (2.11) and (2.12) to those pertaining to element e and the mortar elements m of ∂e , and perform elemental matrix-vector operations.

Equation 2.19 contains integrals and therefore represents only a semi-discrete set of equations. In the next section we show how to make Eq. (2.19) fully discrete by using quadrature to approximate the integrals.

Quadrature

The integrals in Eqs. (2.11) and (2.12) depend on various geometric objects: We define the Jacobian matrix with respect to the mapping $\phi_e : \xi \rightarrow x$ on an element as $e \in \mathbb{E}$

$$(\mathbf{J})_j^i = \frac{\partial \mathbf{x}_i}{\partial \xi_j}, \quad (2.20)$$

with an inverse given by

$$(\mathbf{J}^{-1})_j^i = \frac{\partial \boldsymbol{\xi}_i}{\partial \mathbf{x}_j}. \quad (2.21)$$

The Jacobian determinant is

$$J = \det \mathbf{J} \quad (2.22)$$

Similarly, the surface Jacobian for a mortar element $m \in \mathbb{M}$ is

$$S^m = J \sqrt{\frac{\partial \xi_j}{\partial \mathbf{x}} \cdot \frac{\partial \xi_j}{\partial \mathbf{x}}}, \quad (2.23)$$

where j is the index of the reference coordinate which is constant on the surface under consideration (no sum-convention). For a mortar on a macro-element boundary, S^m may change depending on which of the two macro-element mappings are used to compute Eqn. (2.23). This ambiguity is not problematic as long as all quantities in a mortar integrand are transformed to the macro-element coordinate system in which S^m is being computed. The normal is computed by

$$\hat{\mathbf{n}} = \text{sgn}(\xi_j) \frac{J}{S^m} \frac{\partial \xi_j}{\partial \mathbf{x}}, \quad (2.24)$$

where sgn is the signum function, where j labels the dimension that is constant on the face under consideration (no sum-convention).

Let us now consider the stiffness integral in Eq. (2.11). Because basis-functions are local to each element, we can consider each element e individually, and we will omit the superscript e to lighten the notational load. Substituting the expansion of the solution $u_h = u_\alpha \psi_\alpha$, as well as $v = \psi_\beta$, the stiffness integral becomes

$$L_{h,e}^{\text{stiff}}(u_h, \psi_\beta) = \int_e u_\alpha \frac{\partial \psi_\alpha}{\partial x_k} \frac{\partial \psi_\beta}{\partial x_k} d\mathbf{x}. \quad (2.25)$$

We recall that we employ the Einstein sum convention, i.e. Eq. (2.25) has implicit sums over α and k . The derivatives $\partial \psi_\alpha / \partial \xi_l$ are just polynomials in $\boldsymbol{\xi}$, therefore, they can be re-expanded in our nodal basis as $\partial \psi_\alpha / \partial \xi_l = D_{\alpha\beta}^l \psi_\beta$ with $D_{\alpha\beta}^l := \partial \psi_\alpha / \partial \xi_l(\boldsymbol{\xi}^\beta)$. The physical derivative $\frac{\partial \psi_\alpha}{\partial x_k}$ is then $\frac{\partial \psi_\alpha}{\partial x_k} = (J^{-1})_k^l D_{\alpha\beta}^l \psi_\beta$. The tensor-grid in Eq. (2.15) implies that the matrices $D_{\alpha\beta}^l$ are sparse

for $d > 1$. Using these expressions, and converting to an integral in the reference coordinates, we obtain

$$L_{h,e}^{\text{stiff}}(u_h, v_h) = u_\alpha \int_{[-1,1]^d} (J^{-1})_k^l D_{\alpha\gamma}^l \psi_\gamma (J^{-1})_k^m D_{\beta\delta}^m \psi_\delta J \, d\xi \quad (2.26)$$

$$= u_\alpha L_{\alpha\beta}^{\text{stiff},e}. \quad (2.27)$$

We evaluate the integral in Eq. (2.26) with Gauss-Legendre (GL) quadrature. This choice follows (Mengaldo et al. (2015)) in the use of a stronger set of quadrature points (higher order GL grids) to decrease the error in geometric aliasing. We denote the GL abscissas and weights by $\xi_{\text{GL}}^{(c)}$ and $w_{\text{GL}}^{(c)}$, respectively, where $c = 1, \dots, N_{\text{GL}}$. In multiple dimensions these will be a tensor product of the 1-d GL abscissas and weights denoted by ξ_{GL}^σ and w_{GL}^σ , respectively. All non-polynomial functions, including geometric quantities such $J, (J^{-1})_j^i, S^m, \hat{n}$, are evaluated directly at ξ_{GL}^σ , whereas polynomial functions like the trial function u_h and the test function ψ_h —which naturally are represented on the Legendre-Gauss-Lobatto grid— are interpolated to the GL-quadrature points. Denoting the interpolation matrix by $I_{\sigma\alpha} := \psi_\alpha(\xi_{\text{GL}}^\sigma)$, we find

$$L_{\alpha\beta}^{\text{stiff},e} \approx \sum_\sigma w_\sigma \left(J (J^{-1})_k^l D_{\alpha\gamma}^l \psi_\gamma (J^{-1})_k^m D_{\beta\delta}^m \psi_\delta \right) \Big|_{\xi_{\text{GL}}^\sigma} \quad (2.28)$$

$$= D_{\alpha\gamma}^l I_{\sigma\gamma} (J_\sigma^{-1})_k^l w_\sigma J_\sigma (J_\sigma^{-1})_k^m I_{\sigma\delta} D_{\beta\delta}^m. \quad (2.29)$$

Here $I_{cm} = \psi_m(\xi_{\text{GL}}^\zeta)$ is an interpolation matrix from the GLL points to the GL points. The other volume integrals in Eq. (2.13) are computed in a similar manner.

The mortar integrals are a bit more involved owing to the extra book-keeping arising from the two elements (named '+' and '-') that touch at the boundary $m \in \mathbb{M}$, each with their own local basis-functions, denoted by ψ_α^- and ψ_α^+ . Taking the penalty-integral as an example, for test-functions v_h that are a basis-function of the '-' element, the definition Eq. (2.7) yields

$$L_{m,h}^{\text{pen}}(u_h, \psi_\alpha^-) = \int_m \sigma [\![u_h]\!]_k [\![\psi_\alpha^-]\!]_k \, ds \quad (2.30)$$

$$= \int_{[-1,1]^{d-1}} \sigma (u_h^- - u_h^+) \psi_\alpha^- S^m \, d\xi. \quad (2.31)$$

Equation (2.31) contains both u_h^- and u_h^+ ; therefore, when substituting in their respective local expansions, $u_h^\pm = u_\alpha^\pm \psi_\alpha^\pm$, and pulling the coefficients outside the integral, we see that the penalty term will result in entries of the global matrix equation (2.19) that couple the two elements e^\pm . Once the coefficients u_α^\pm are moved outside the integral, the remaining integral only depends on basis-functions ψ_α^\pm and geometric quantities. These integrals are evaluated with GL-quadrature, which is expressed in terms of interpolation matrices $I_{\alpha\rho}^\pm := \psi_\alpha^\pm(\xi_{\text{GL}}^\rho)$, where ρ labels the GL collocation points on the boundary $[-1, 1]^{d-1}$ and α labels the local basis-functions in e^\pm , which are to be evaluated in the suitably mapped ξ_{GL} locations.

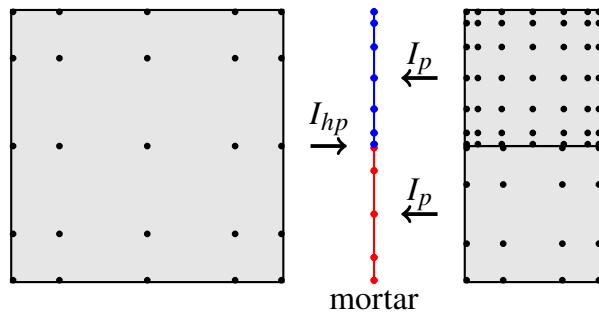


Figure 2.2: A representation of a 2:1 interface in two dimensions. On the left we have a 4th-order element and on the right we have two elements of degree 8 and degree 2 respectively. The data on the faces of these three elements will be used in the mortar integrals of equation 2.5, but since the interface is non-conforming, we need to interpolate the face data to a shared broken polynomial space on the mortar. To ensure that the face data on each element is in a polynomial space which a subset of the polynomial space on the mortar, we demand that the polynomial degree p on the mortar is the maximum of the polynomial degrees on either side. For the lower mortar face (red) this would be $\max(4, 3) = 4$ and for the upper mortar face (green) this would be $\max(4, 6) = 6$. Since the left element must interpolate it's face data to a space containing two faces, we use the hp-interpolation operator I_{hp} . On the right side, each element has to map its data from one face to one face, so we use the p-interpolation operator I_p .

2.3.2 Penalty Function

The penalty parameter σ in Eqs. (2.5) and (2.11) is a spatially dependent function on boundaries $m \in \mathbb{M}$, defined by

$$\sigma = C \frac{p_m^2}{h_m}, \quad (2.32)$$

where p_m and h_m represent a typical polynomial degree and a typical length-scale of the elements touching at the boundary, respectively, and where the parameter $C > 0$ is large enough such that $L(\cdot, \cdot)$ is coercive. We choose $p_m = \max(p^+, p^-)$ and we set $h_m = \min(J^+/S^m, J^-/S^m)$ as this has yielded the best results empirically. Here J^\pm is the volume Jacobian on e^\pm mapping $[0, 1]^d$ to e^\pm , and S^m is the surface Jacobian mapping $[0, 1]^{d-1}$ to the boundary face of e^\pm .

2.3.3 Norms

Throughout the following sections we will use the energy norm,

$$\|\nu\|_* = \left(\sum_{e \in \mathbb{E}} \int_e |\nabla \nu|^2 dx + \sum_{m \in \Gamma} \int_m |\sqrt{\sigma} [\nu]|^2 ds \right)^{\frac{1}{2}}, \quad (2.33)$$

and the L_2 norm,

$$\|\nu\|_2 = \left(\int_{\Omega} \nu^2 dx \right)^{\frac{1}{2}} = \left(\sum_{e \in \mathbb{E}} \int_e \nu^2 ds \right)^{\frac{1}{2}}. \quad (2.34)$$

Here, ν is a scalar function on Ω .

2.3.4 Multigrid-Preconditioned Newton-Krylov

Newton-Krylov

In general, a set of elliptic partial differential equations can be written in the form

$$R(\mathbf{u}) = 0, \quad (2.35)$$

where \mathbf{u} is the solution and R is a non-linear elliptic operator. Defining the Jacobian of the system by $J_R(\mathbf{u}) \equiv \frac{\partial R}{\partial \mathbf{u}}(\mathbf{u})$, Newton-Raphson iteratively refines an initial guess for the solution

$\mathbf{u}^{(k)}$ by solving the following linear system for the correction $\delta\mathbf{u}^{(k)} = \mathbf{u}^{(k+1)} - \mathbf{u}^{(k)}$:

$$\mathbf{J}_R(\mathbf{u}^{(k)})\delta\mathbf{u}^{(k)} = -\mathbf{R}(\mathbf{u}^{(k)}). \quad (2.36)$$

Upon discretization as described in Sec. 2.3.1, Eq. (2.36) results in a $N \times N$ linear system. Once Eq. (2.36) is solved, the improved solution is given by $\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \delta\mathbf{u}^{(k)}$. The Newton-Raphson iterations are continued until the residual $\mathbf{R}(\mathbf{u}^{(k)})$ is sufficiently small. At each step, the linear system is solved by a linear solver we abbreviate as *LSOLVE*. The full algorithm is described in Algorithm 1.

Algorithm 1 Multigrid Preconditioned Newton-Krylov: Solves Eq. (2.35). $N_{\text{iter,NK}}$ and tol_{NK} are user-specified parameters.

```

1: procedure NK( $u_0$ )
2:    $k \leftarrow 0$ 
3:   while  $k \leq N_{\text{iter,NK}}$  or  $\|\mathbf{R}(u_k)\| \geq \text{tol}_{\text{NK}}$  do
4:      $\delta u_k \leftarrow \text{LSOLVE}(u_k, \mathbf{R}(u_k), \mathbf{J}_R(u_k))$ 
5:      $u_{k+1} \leftarrow u_k + \delta u_k$ 
6:      $k \leftarrow k + 1$ 
7:   end while
8:   return  $u_k$ 
9: end procedure

```

For three-dimensional problems with a large number N of degrees of freedom, the Jacobian \mathbf{J} is too large to form fully and partition across processes. For building a scalable solver, we are therefore limited to iterative solvers, the most popular class of which are the Krylov solvers (Saad (2003)) such as the Conjugate Gradient method or GMRES, which only involve matrix-vector operations. In this paper we use the flexible conjugate gradient Krylov (Notay (2000)) solver at each Newton-Raphson step, as summarized in Algorithm 2, where u_0 denotes the initial guess of the linear solver, and N_{its} is the number of iterations.

Algorithm 2 Flexible Conjugate Gradients: Solves Eq. (2.36), with the variable u representing $\delta u^{(k)}$. $N_{\text{iter,FCG}}$ and tol_{FCG} are user-specified parameters.

```

1: function LSOLVE( $u_0, R, J$ )
2:    $r_0 \leftarrow Ju_0 + R$ 
3:    $k \leftarrow 0$ 
4:   while  $k++ \leq N_{\text{iter,FCG}}$  or  $\|r_k\| \geq \text{tol}_{\text{FCG}}$  do
5:      $v_k \leftarrow \text{VCYCLE}(r_k, Jv_k)$ 
6:      $w_k \leftarrow Jv_k$ 
7:      $\alpha_k \leftarrow v_k \cdot r_k$ 
8:      $\beta_k \leftarrow v_k \cdot w_k$ 
9:     if  $k = 0$  then
10:     $d_k \leftarrow v_k$ 
11:     $q_k \leftarrow w_k$ 
12:     $\rho_k \leftarrow \beta_k$ 
13:  else
14:     $\gamma_k \leftarrow v_k \cdot q_{k-1}$ 
15:     $d_k \leftarrow v_k - (\gamma_k / \rho_{k-1})d_{k-1}$ 
16:     $q_k \leftarrow w_k - (\gamma_k / \rho_{k-1})q_{k-1}$ 
17:     $\rho_k \leftarrow \beta_k - \gamma_k^2 / \rho_{k-1}$ 
18:  end if
19:   $u_{k+1} \leftarrow u_k + (\alpha_k / \rho_k)d_k$ 
20:   $r_{k+1} \leftarrow r_k - (\alpha_k / \rho_k)q_k$ 
21: end while
22: return  $u_k$ 
23: end function

```

A typical Krylov solver will take $O(\sqrt{\kappa})$ iterations to reach a desired accuracy, where the condition number κ is defined as the ratio of the maximum eigenvalue and minimum eigenvalue of J_R . For the discontinuous galerkin method, the discretized Laplacian matrix has a condition number that grows with p-refinement and h-refinement (see (Antonietti & Houston (2011); Hesthaven & Warburton (2008)) for example) and therefore the number of iterations will grow with each AMR step unless Eq. (2.36) is preconditioned.

Multigrid preconditioner

We use a multigrid V-cycle (Briggs et al. (2000)) as a preconditioner for each FCG solve (called on line 4 in Algorithm 2). Multigrid is an multi-level iterative method aimed at achieving mesh-independent error reduction rates through a clever method of solving for the error corrections on coarser grids and prolonging the corrections to the finer grids. For a detailed overview of multigrid methods see (Briggs et al. (2000)). The main drawback of multigrid is its complexity. In this section we will briefly describe the components of the multigrid algorithm employed in solving problems in this paper with hp-grids in parallel with the interior penalty discontinuous Galerkin method.

Multigrid Meshes

Multigrid uses a hierarchy of coarsened meshes, labeled with $l = 0, \dots, L$, where $l = L$ represents the fine mesh \mathbf{E} on which the solution is desired, and $l = 0$ represents the coarsest mesh. One V-cycle proceeds from the fine grid to the coarsest grid, and back to the fine grid, as indicated in Fig. 2.3. We construct the coarse meshes $l = L - 1, \dots, 0$ by successively coarsening. Coarsening by combining 2^d elements into one element can lead to interfaces with a 4:1 balance, even if the original mesh is 2:1 balanced. We avoid such 4:1 balance with surrogate meshes, which have been used before in large-scale FEM multigrid solvers, see for example (Sampath et al. (2008); Sundar et al. (2012)). A surrogate mesh is the naively h-coarsened mesh, as indicated in blue in Fig. 2.3. If the surrogate mesh has indeed interfaces with 4:1 balance, the desired 2:1

balance is enforced by refining at the unbalanced interfaces. This results in the coarsened mesh on Level $l = L - 1$, and iteratively down to $l = 0$. It is easy to show that at each coarsening the coarse level $l - 1$ mesh is strictly coarser than the level l fine mesh. Following this, we must make a decision as to what polynomial degree we choose for the coarsened elements. We take the minimum polynomial degree of the 2^d children elements for each parent element of the coarsened mesh, ensuring that functions on the coarse grid can be represented exactly on the fine mesh, a property we will utilize below in Eq. (2.42). Lastly, it is important to note that we never purely p-coarsen on any of the multigrid levels. We do not store the multigrid meshes because we have empirically shown that the coarsening, refining and balancing is not the bottleneck for the multigrid algorithm.

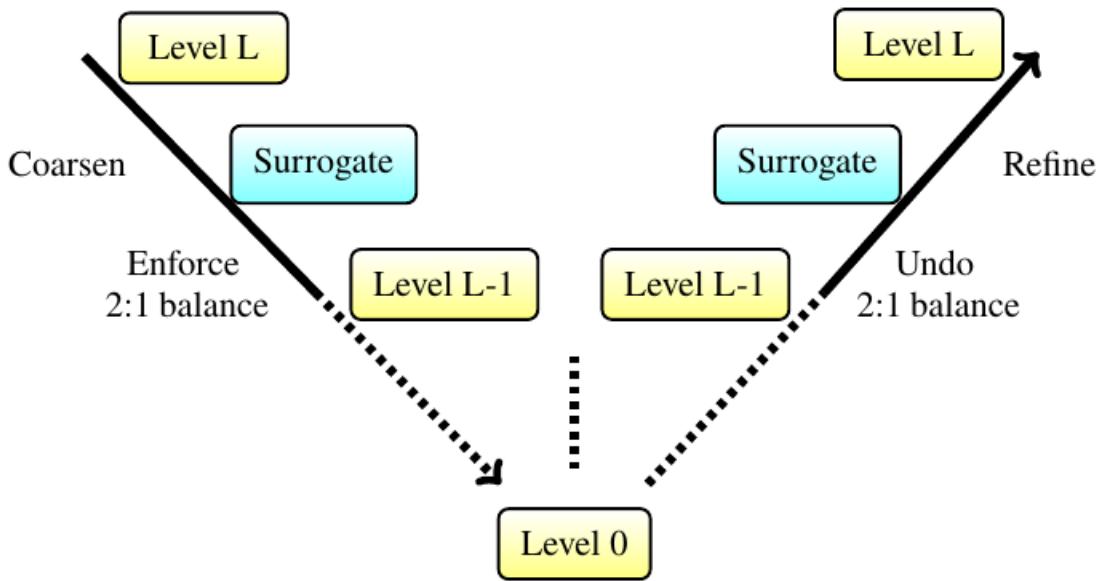


Figure 2.3: A structural representation of a multigrid v-cycle. The nodes in yellow are actual grid-levels, while the nodes in blue represent surrogate grids which are not necessarily 2:1 balanced. In order for a multigrid v-cycle to represent a symmetric operation, the grid levels along the down arrow are exactly the same as the grid-levels along the up arrow. Level 0 represents the coarsest possible mesh.

Multigrid algorithm

Algorithm 3 Multigrid Preconditioner Vcycle.

L denotes the number of multigrid Levels.

```

1: function VCYCLE( $f_L, J_L$ )
2:   for  $l = L, 1$  do
3:      $v_l \leftarrow 0$ 
4:      $v_l \leftarrow \text{SMOOTHER}(v_l, f_l, J_l)$ 
5:     Coarsen grid
6:     Balance grid
7:      $f_{l-1} \leftarrow I_l^T(f_l - J_l v_l)$  ▷ Restriction
8:   end for
9:    $v_0 \leftarrow 0$ 
10:   $v_0 \leftarrow \text{SMOOTHER}(v_0, f_0, J_0)$ 
11:  for  $l = 1, L$  do
12:    Refine grid
13:     $v_l \leftarrow I_l(v_{l-1}) + v_l$  ▷ Prolongation
14:     $v_l \leftarrow \text{SMOOTHER}(v_l, f_l, J_l)$ 
15:  end for
16:  return  $v_L$ 
17: end function

```

Having constructed the coarse meshes, we can now turn to the multigrid algorithm, summarized in Algorithm 3. It consists of several important components: Smoothing (line 7 and 16), coarsening and balancing of the mesh (line 8 and 9), restriction (line 10), a bottom solver (line 12) and prolongation (line 15). We now look at each of these in turn.

Multigrid Smoother

We explore two different smoothers, Chebyshev smoothing and a Schwarz smoother. Both of these avoid explicit storage of the matrix, as required by smoothers like the Gauss-Seidel method (Shang (2009)).

Chebyshev smoothing (Li et al. (2011)), a type of polynomial smoother (Saad (2003)) requires only matrix-vector operations and has been shown to work satisfactorily in scalable multigrid solvers (Ghysels et al. (2012)). Our implementation of Chebyshev smoothing (Li et al. (2011)) is presented in Algorithm 4. For this algorithm there are two user-defined parameters: $N_{\text{iter,Cheb}}$ and Λ . Typical values we use for $N_{\text{iter,Cheby}}$ are in the range 8 – 15 and Λ is usually set in the range 10 – 30. Chebyshev polynomial smoothers require a spectral bound on the eigenvalues of the linear operator. We use conjugate gradients to estimate the eigenvalues of a general symmetric linear operator, as implemented in Algorithm 5. It can be shown (Scales (1989)) that each iteration of conjugate gradients obtains one row of the underlying linear operator in tri-diagonal form:

$$\left(\begin{array}{ccccc} \frac{1}{\alpha_1} & -\frac{\sqrt{\beta_2}}{\alpha_1} & 0 & \cdots & 0 \\ -\frac{\sqrt{\beta_2}}{\alpha_1} & \frac{1}{\alpha_2} + \frac{\beta_2}{\alpha_1} & -\frac{\sqrt{\beta_3}}{\alpha_2} & & \vdots \\ 0 & & \ddots & & 0 \\ \vdots & & -\frac{\sqrt{\beta_{k-1}}}{\alpha_{k-2}} & \frac{1}{\alpha_{k-1}} + \frac{\beta_{k-1}}{\alpha_{k-2}} & -\frac{\sqrt{\beta_k}}{\alpha_{k-1}} \\ 0 & \cdots & 0 & -\frac{\sqrt{\beta_k}}{\alpha_{k-1}} & \frac{1}{\alpha_k} + \frac{\beta_k}{\alpha_{k-1}} \end{array} \right) \quad (2.37)$$

The values α_k and β_k in this expression are obtained from the conjugate gradients algorithm 5 on lines 6 and 9. Furthermore, the eigenvalues of each of the sub tri-diagonal matrices is a subset of the eigenvalues of the full matrix. So at each iteration we can get an estimate of the bound by using the Gershgorin circle theorem (Bell (1965)). Our Alg. 5 combines the CG steps with the estimation of the bound λ_{\max} of the largest eigenvalue.

Besides estimating the spectral radius, we also use the CG iterations to further smooth the solution. This adds robustness to multigrid algorithms (Elman et al. (2001)).

Algorithm 4 Chebyshev Smoothing

$N_{\text{iter,Cheb}}$ and Λ are user-defined parameters.

```

1: function SMOOTH(x, b, J)
2:    $p \leftarrow 0$ 
3:    $\lambda_{\max}, x \leftarrow \text{CGEIGS}(x, b, J)$ 
4:    $\lambda_{\min} \leftarrow \lambda_{\max}/\Lambda$ 
5:    $c \leftarrow (\lambda_{\max} - \lambda_{\min})/2$ 
6:    $d \leftarrow (\lambda_{\max} + \lambda_{\min})/2$ 
7:   for  $k = 1 \dots N_{\text{iter,Cheb}}$  do
8:      $r \leftarrow b - Jx$ 
9:      $\alpha \leftarrow \begin{cases} d^{-1} & k = 1 \\ 2d(2d^2 - c^2)^{-1} & k = 2 \\ (d - \alpha c^2/4)^{-1} & k \neq 1, 2 \end{cases}$ 
10:     $\beta \leftarrow \alpha d - 1$ 
11:     $p \leftarrow \alpha r - \beta p$ 
12:     $x \leftarrow x + p$ 
13:   end for
14:   return  $x$ 
15: end function

```

While the Chebyshev smoother is easy to implement, it is reliant on a robust estimate of the largest eigenvalue and this may not be always true in our case. Thus we also implement an additive Schwarz smoother in the manner of (Stiller (2017)), which is much more robust. The Schwarz smoother works by performing local solves on element-centered subdomains and then adding a weighted sum of these local solves to obtain the smoothed solution. A simple example of one of these subdomains (where the grid is both p and h-uniform) is shown in Figure 2.4. More generally, the Schwarz subdomain centered on element e_c of the mesh is

Algorithm 5 CG Spectral Bound Solver

$N_{\text{iter,eigs}}$ is a user-defined parameter.

```

1: function CGEIGS( $x_1, b, J$ )
2:    $r_1 \leftarrow b - Jx_1$ 
3:    $p_1 \leftarrow r_1$ 
4:    $\lambda_{\max} \leftarrow 0$ 
5:   for  $k = 1, N_{\text{iter,eigs}}$  do
6:      $\alpha_k \leftarrow (r_k \cdot r_k)/(p_k \cdot Jp_k)$ 
7:      $x_{k+1} \leftarrow x_k + \alpha_k p_k$ 
8:      $r_{k+1} \leftarrow r_k - \alpha_k Jp_k$ 
9:      $\beta_k \leftarrow (r_{k+1} \cdot r_{k+1})/(r_k \cdot r_k)$ 
10:     $p_{k+1} \leftarrow r_{k+1} + \beta_k p_k$ 
11:     $\lambda_k \leftarrow \begin{cases} \frac{1}{a_1} + \frac{\sqrt{\beta_1}}{a_1} & \text{if } k = 1 \\ \frac{1}{a_k} + \frac{\beta_{k-1}}{a_{k-1}} + \frac{\sqrt{\beta_{k-1}}}{\alpha_{k-1}} & \text{if } k \neq 1 \end{cases}$ 
12:     $\lambda_{\max} = \max(\lambda_k, \lambda_{\max})$ 
13:   end for
14: return  $\lambda_{\max}, x_{k+1}$ 
15: end function

```

constructed as follows: Starting with all collocation points on e_c , one adds all boundary points of neighboring elements which coincide with faces, vertices or corners of e_c . Around these initial set of collocation points on neighboring elements, one then adds $N_{\text{overlap}} - 1$ layers of additional collocation points (in Fig. 2.4, $N_{\text{overlap}} = 2$). If the mesh has non-uniform h or p refinement, the resulting set of collocation points will have ragged boundaries.

The solutions on the individual Schwarz subdomains are combined as a weighted sum. The weights differ with each collocation point of each subdomain. In 1-D, for a Schwarz subdomain

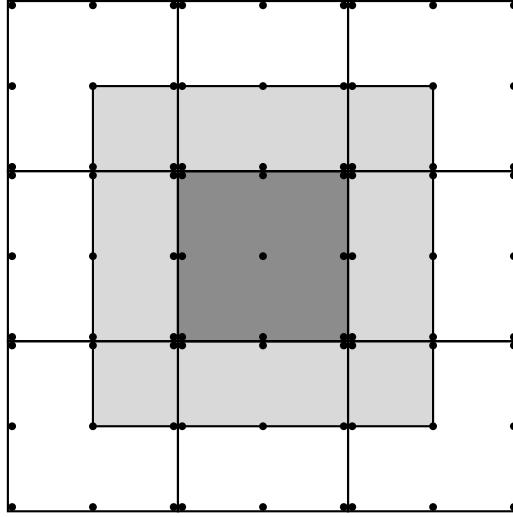


Figure 2.4: A simple 2-D Schwarz subdomain, with no h-nonconforming or p-nonconforming boundaries. In grey is the element e which is the center of the subdomain. The light grey area is the overlap (of size δ_ξ) into the other elements. The subdomain is composed of everything in light and dark grey.

centered on element e_c with left and right neighbours e_{c-1} and e_{c+1} we define an extended LGL coordinate ξ_{ext} for a collocation point x as follows:

$$\xi_{\text{ext}}(x) = \begin{cases} \xi^* & x \in e_c, \\ \xi^* \pm 2 & x \in e_{c\pm 1}, \end{cases} \quad (2.38)$$

where ξ^* is the LGL coordinate of the collocation point x in the reference coordinate system of the macro-element containing e_c . This definition takes care of the case when a Schwarz subdomain contains a face that is on a tree boundary.

We denote the overlap size as δ_ξ and compute it as the width of the Schwarz subdomain overlap in the coordinate ξ_{ext} . With these definitions we compute the weights for this 1-D subdomain using the function $w_h : \mathbb{R} \rightarrow \mathbb{R}$ defined as

$$w_h(\xi_{\text{ext}}) = \frac{1}{2} \left(\phi \left(\frac{\xi_{\text{ext}} + 1}{\delta_\xi} \right) - \phi \left(\frac{\xi_{\text{ext}} - 1}{\delta_\xi} \right) \right), \quad (2.39)$$

where the ϕ function is given by

$$\phi(\xi) = \begin{cases} \text{sgn}(\xi), & |\xi| > 1, \\ \frac{1}{8}(15\xi - 10\xi^3 + 3\xi^5), & |\xi| \leq 1. \end{cases} \quad (2.40)$$

We have plotted the weighting function w_h in Figure 2.5.

For Schwarz subdomains in d -dimensions, the weights are computed as a product of 1-dimensional weights along each dimension,

$$W = \prod_{i=1}^d w_h(\xi_{ext}^i). \quad (2.41)$$

For the Schwarz subdomain solves we need to define the following operators: R_s is the restriction operator for a Schwarz subdomain, it reduces the data on the mesh to the nodes of the subdomain. R_s^T is the transpose of this operator. W_s are the weights for a Schwarz subdomain, computed by evaluating Eq. (2.39) on the nodes of the subdomain. With these definitions, the Schwarz smoother algorithm is listed in Alg. 6. In Alg. 6, N_s indicates the number of smoothing cycles (typically, $N_s = 3$), N_{subs} indicates the number of subdomains which for our implementation is equal to the number of elements. The linear system on line 6 of Alg. 6 is of the size of the Schwarz-subdomain; we solve it with conjugate gradients to a relative tolerance of 10^{-3} .

Multigrid Inter-mesh Operators

Let us now turn to the implementation of restriction and prolongation operators. For clarity, we use the following notation: a superscript lowercase h refers to the children elements and an uppercase H refers to the parent element.

We recall our requirement that the polynomial order of a coarse element is smaller than or equal to the polynomial orders of its children elements (cf. Sec. 2.3.4). This ensures that a parent element's Lagrange polynomial space is always embedded in the broken Lagrange polynomial space of its children. In particular, the coarse-mesh basis-functions can be written, exactly, as

$$\psi_\alpha^H = (I_h^H)_{\beta\alpha} \psi_\beta^h, \quad (2.42)$$

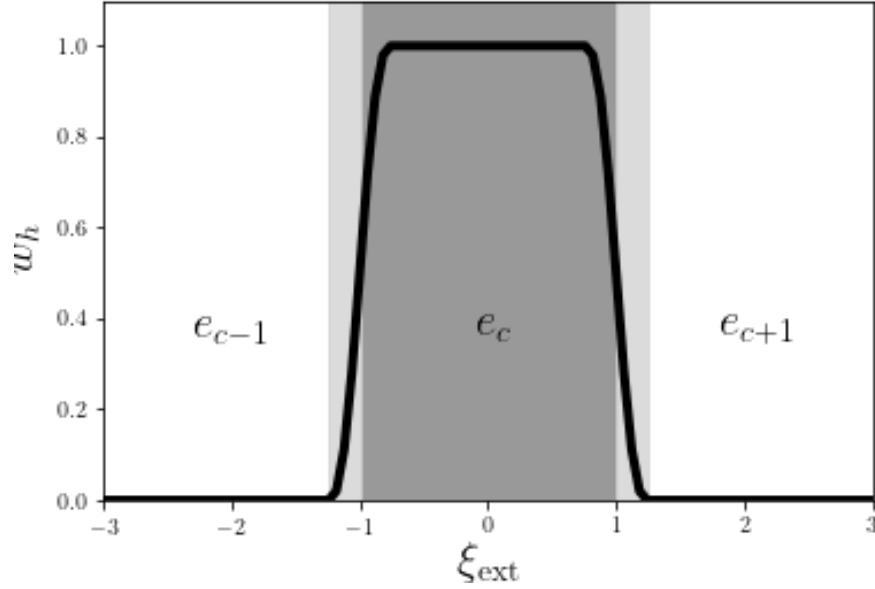


Figure 2.5: A plot of the Schwarz weighting function defined by Eq. (2.39). The dark grey shaded region is the center element of the Schwarz subdomain e_c and the light grey shaded region is the overlap of width δ_ξ . For this plot, we set $\delta_\xi = .25$.

where $(I_h^H)_{\beta\alpha} = \psi_\alpha^H(\xi_h^\beta)$ is the interpolation-matrix of the coarse-mesh basis-functions to the fine-mesh collocation points, and where we utilized our choice of a *nodal* basis. For h -refinement, each ψ_β^h has support in only one of the child-elements and, consequently, the implicit sum over β in Eq. (2.42) goes over the basis-elements of all children elements.

The preceding paragraph immediately suggests the natural definition of the prolongation operator from a coarse-grid function $u^H = u_\alpha^H \psi_\alpha^H$ to a fine-grid function $u^h = u_\beta^h \psi_\beta^h$ such that $u^h \equiv u^H$. This yields

$$u_\beta^h = u^H(\xi_h^\beta) = (I_h^H)_{\beta\alpha} u_\alpha^H, \quad (2.43)$$

so that (I_h^H) represents the prolongation operator.

Because the left hand side operator for the weak equations —e.g. $L(\cdot, \cdot)$ in Eqn. (2.13)— is a bilinear form, the residual will also be a bilinear form, which we will denote $B(\cdot, \cdot)$; on the coarse grid. Equation (2.42) therefore implies

$$B(\cdot, \psi_\alpha^H) = (I_h^H)_{\beta\alpha} B(\cdot, \psi_\beta^h), \quad (2.44)$$

Algorithm 6 Schwarz Smoother

$N_{\text{iter,Sch}}$ is a user-defined parameter.

```

1: function SMOOTH(x, b, J)
2:   for  $k = 1 \dots N_{\text{iter,Sch}}$  do
3:      $r \leftarrow b - Jx$ 
4:     for  $s = 1 \dots N_{\text{subs}}$  do
5:        $r_s \leftarrow R_s r$ 
6:       Solve  $R_s J R_s^T \delta x_s = r_s$ 
7:     end for
8:      $x \leftarrow x + \sum_s R_s^T W_s \delta x_s$ 
9:   end for
10:  return  $x$ 
11: end function

```

which defines our coarse grid residual.

Lastly, let us consider restriction of the operators itself. After linearization around the current solution u_0 , the problems we consider here take the form

$$\nabla^2 \delta u + f(u_0) \delta u = 0, \quad (2.45)$$

for some function f . The associated weak form reads

$$L_h(\delta u, \psi_\alpha) + \int f(u_0) \psi_\alpha \delta u \, d\mathbf{x} = 0 \quad \forall \psi_\alpha. \quad (2.46)$$

The weak Laplacian operator ($L_h(\cdot, \cdot)$) can be computed on any coarse grid via Eqn. (2.13) without any need for a restriction operator. The second term will require either a restriction operator on $f(u_0)$, or a restriction operator on $O_{\alpha\beta}^h \equiv \int f(u_0) \psi_\beta^h \psi_\alpha^h \, d\mathbf{x}$. We choose the latter. By Eq. (2.42), it holds that

$$\int f(u_0) \psi_\alpha^h \psi_\beta^h \, d\mathbf{x} = (I_h^H)_{\gamma\alpha} (I_h^H)_{\delta\beta} \int f(u_0) \psi_\gamma^h \psi_\delta^h \, d\mathbf{x}, \quad (2.47)$$

so that the restricted operator in matrix form becomes

$$O^H = (I_h^H)^T O^h (I_h^H). \quad (2.48)$$

2.3.5 hp-Adaptivity

One of the great advantages of the discontinuous Galerkin method is that it naturally allows for two different methods of refining the grid, h-adaptivity, where one subdivides the element into smaller sub-elements, and p-adaptivity, where one increases the order p_e of the polynomial basis on an element. The combination of both h-adaptivity and p-adaptivity is called hp-adaptivity. If one strategically p-refines in smooth regions of the mesh and h-refines in discontinuous regions of the mesh, then it is possible to regain exponential convergence in particular error norms for problems with non-smooth functions. In the next four sections we will discuss the four components of our hp-adaptive scheme. These are: (1) The expected convergence of the solution on a series of adaptively refined hp-meshes (2) an a posteriori error estimator to decide which elements will be refined; (3) a driving strategy that determines based on the convergence of the error estimator whether to h-refine or p-refine; (4) an efficient method to apply discrete operators in multi-dimensions on an hp-grid. We discuss these four items in order in the follow subsections.

Expected Convergence

For quasi-linear problems with piecewise-analytic solutions u on polygonal domains, the convergence of the energy norm (Eq. 2.33) of the analytical error for the numerical solution u_h is

$$\|u - u_h\|_* \leq C \left(\sum_{e \in \mathbb{E}} \frac{h_e^{2s_e-2}}{P_e^{2k_e-3}} \|u\|_{H^{k_e}}^2 \right)^{1/2}. \quad (2.49)$$

Here H^{k_e} is the Sobolev space on element e with Sobolev order k_e and $1 \leq s_e \leq \min(p_e + 1, k_e)$ (Houston et al. (2007; 2008; 2005b); Bi et al. (2015)). For uniform refinement and uniform Sobolev order k across elements, we expect $\|u - u_h\|_* \leq Ch^{\min(p+1,k)-1}$. Thus, for smooth problems we expect $\|u - u_h\|_* \leq Ch^p$.

One can show (Schötzau et al. (2014)) that there exists a series of hp-adaptive refinement steps where the convergence of the energy norm is asymptotically bounded by

$$\|u - u_h\|_* \leq C_1 \exp(-C_2 \text{DOF}^{1/(2d-1)}). \quad (2.50)$$

Here C_1 and C_2 are constants, d is the dimension of the mesh and DOF is the number of degrees of freedom, in other words the number of grid points on the mesh.

A Posteriori Error Estimator

Ref. (Bi et al. (2015)) derives a local a posteriori error estimator for an interior penalty hp-DG discretization of the following class of strongly nonlinear elliptic problems with Dirichlet boundary conditions

$$-\nabla \cdot \mathbf{a}(\mathbf{x}, u, \nabla u) + f(\mathbf{x}, \nabla u) = 0. \quad (2.51)$$

The error estimator can be computed locally on each element by the following prescription. Given a discretized solution u_h , first define quantities on each element e or mortar element m by

$$R_e \equiv f(u_h, \nabla u_h) - \nabla \cdot \mathbf{a}(u_h, \nabla u_h), \quad e \in \mathbb{E}, \quad (2.52a)$$

$$J_{m,1} \equiv [\![\mathbf{a}(u_h, \nabla u_h)]\!]_m, \quad m \in \mathbb{M}_{\mathbb{I}}, \quad (2.52b)$$

$$J_{m,2} \equiv [\![u_h]\!]_m, \quad m \in \mathbb{M}_{\mathbb{I}}, \quad (2.52c)$$

$$J_{m,3} \equiv [\![u_h - g_D]\!]_m, \quad m \in \mathbb{M}_{\mathbb{D}}. \quad (2.52d)$$

From these quantities we compute the followings integrals on an element e and mortar m :

$$\eta_e^2 = h_e^2 p_e^2 \|R_e\|_{0,e}^2, \quad (2.53a)$$

$$\eta_{m,1}^2 = h_m p_m^{-1} \|J_{m,1}\|_{0,m}^2, \quad (2.53b)$$

$$\eta_{m,2}^2 = \sigma h_m^{-1} p_m^2 \|J_{m,2}\|_{0,m}^2, \quad (2.53c)$$

$$\eta_{m,3}^2 = \sigma h_m^{-1} p_m^2 \|J_{m,3}\|_{0,m}^2, \quad (2.53d)$$

Here, $\|.\|_{0,e}^2 = \int_e (\cdot)^2 d\mathbf{x} = \int_{[-1,1]^d} J(\cdot)^2 d\boldsymbol{\xi}$ and $\|.\|_{0,m}^2 = \int_m (\cdot)^2 d\mathbf{s} = \int_{[-1,1]^{d-1}} S^m(\cdot)^2 d\boldsymbol{\xi}$. Furthermore, h_e is the diameter of the element e , p_m , h_m and σ are defined in Eq. (2.32) and J , S^m are defined in Eq. (2.22) and Eq. (2.23).

Following (Bi et al. (2015)), we take the local estimator on the element to be

$$\eta^2(e) = \eta_e^2 + \sum_{m \in \mathbb{M}_e} \eta_{m,1}^2 + \sum_{m \in \mathbb{M}_e \setminus \mathbb{M}_D} \eta_{m,2}^2 + \sum_{m \in \mathbb{M}_e \cap \mathbb{M}_D} \eta_{m,3}^2, \quad (2.54)$$

where \mathbb{M}_e and \mathbb{M}_D are the sets of mortar elements touching the element e and $\partial\Omega_d$ respectively (see Sec.2.3.1). The estimator provides an estimate of the error in the energy norm, $\|u - u_h\|_*$, see Eq. (2.33). Similar error estimators for various classes of linear and non-linear elliptic PDEs can be found in (Houston et al. (2005c); Hansbo & Larson (2011); Zhu et al. (2011); Houston et al. (2007); Schötzau & Zhu (2009); Houston et al. (2005a); Lovadina & Marini (2009)). Equations (2.53) are typically computed on the physical grid on which the elliptic PDE is solved. For highly stretched grids (for instance, when a mapping inverse in radius is used to push the outer boundary to very large radius $R \sim 10^3 \dots 10^{10}$), the geometric factors in Eqs. (2.53) can distort the error estimates. For those cases, we will sometimes introduce a fiducial grid of identical structure and connectivity, that avoids or mitigates excessive grid-stretching. Once a fiducial grid is chosen, its geometric properties can be used in Eqs. (2.53) and the corresponding norms. Below, we demonstrate that such a “fiducial-grid” error-estimator allows efficient hp-refinement even in highly stretched computational domains. This problem is encountered below in Sec. 2.4.3 and discussed there in greater detail.

Driving Strategy

Given the a posteriori estimator Eq. (2.54) we next need to determine if to h-refine or p-refine elements with a large error $\eta(e)$. In the survey paper (Mitchell & McClain (2011)), Mitchell and McClain looked at 15 different hp-adaptive strategies with a finite elements scheme. Their results indicate that the strategy known as smooth-pred, first introduced in (Melenk & Wohlmuth (2001)) performs quite well, if not the best under the example problems they tested. For this

reason alone we use it, but our code is flexible enough to use any of strategies studied in Mitchell and McClain's paper.

The idea behind the smooth-pred strategy is based on the observation that for a locally smooth solution, the energy norm Eq. (2.33) $\eta(e)$ will converge as h^p , (see Sec. 2.3.5). To take advantage of this observation, we first predict the error in the next refinement step and test if this prediction satisfies the smooth error convergence law, if so we p-refine, otherwise we h-refine. The full algorithm is given in Algorithm 7.

Algorithm 7 hp-AMR Driving Strategy

γ_h and γ_p are user-defined parameters.

```

1: procedure SMOOTH-PRED
2:   if  $\eta^2(e)$  is large then
3:     if  $\eta^2(e) > \eta_{\text{pred}}^2(e)$  then
4:       h-refine element
5:        $\eta_{\text{pred}}^2(e_{\text{children}}) \leftarrow \gamma_h \left(\frac{1}{2}\right)^d \left(\frac{1}{2}\right)^{2p_e} \eta^2(e)$ 
6:     else
7:       p-refine element
8:        $\eta_{\text{pred}}^2(e) \leftarrow \gamma_{p_e} \eta^2(e)$ 
9:     end if
10:   end if
11: end procedure

```

In Algorithm 7, $\eta^2(e)$ is the square of the error estimator, Eq. (2.54) of the element under consideration, and $\eta_{\text{pred}}^2(e)$ is the predicted error estimator from the last AMR step assuming the solution on the element is smooth. We always start with $\eta_{\text{pred}}(e) = 0$, so that each element will first be p -refined, before h -refinement is considered.

The parameters γ_p and γ_h influence the behavior of Alg. 7 as follows: As long as the actual $\eta^2(e)$ is *small* compared to the predicted $\eta_{\text{pred}}^2(e)$, Line 3 implies continued p -refinement. Since

Line 8 reduces the predicted η_{pred}^2 by a factor γ_p , this means, that p -refinement continues as long as each increment in p_e reduces the error-estimator $\eta^2(e)$ by a factor γ_p , i.e. as long as exponential convergence is obtained with convergence rate better than $\sqrt{\gamma_p}$. If this convergence-rate is not observed, the driver switches to h -refinement, and γ_h begins to matter. In this case, a large γ_h will preferably switch back to p -refinement, whereas a small γ_h will prefer continued h -refinement. All runs shown in the remainder of this paper use $\gamma_p = 0.1$. The runs differ in γ_h , which is used to tune h- vs p-refinement, and in how many elements are refined in each AMR-iteration, which is used to control how quickly AMR increases the number of degrees of freedom. However for the majority of the runs we find that $\gamma_h = 0.25$ is a good choice.

In the numerical examples of this paper we use the following two alternatives for the conditional on line 2 of Alg. 7, i.e. to decide which elements should be refined: In the first test-problem, we refine if η^2 is greater than some constant factor times the mean $\bar{\eta}^2$ across all elements e . This criterion was used in the original description of the hp-AMR scheme (See (Melenk & Wohlmuth (2001))). In the remaining problems we refine a percentage of elements with the largest $\eta(e)$. We change criterion because the estimator may vary over orders of magnitude and this variation may change at each refinement step. In such cases, the percentage criterion robustly captures more of the elements with a large estimator than thresholding on a constant times the average estimator). However, the constant factor times the mean method has the advantage that it is computationally cheap and does not require a global sort of the estimator over all cores like the percentage method does. When we use the percentage indicator, we make use of the highly parallel global sort outlined in (Feng et al. (2015)).

On-the-fly hp Tensor-Product Operations

To solve the elliptic equations with multigrid on grids that are hp-adaptively refined, a large number of different operators must be generated at run-time. In no specific order, we need at least the following linear operators

- interpolation operators on faces, edges and volumes of size $(p' + 1) \times (p + 1)$ in 1-D, where

p is the order of the original data and p' is the order of the new interpolated data

- restriction operators on faces, edges and volumes of size $(p + 1) \times (p' + 1)$ in 1-D, where p' is the order of the original data and p is the order of the new restricted data
- derivative operator of order p on the reference element, which has size $(p + 1) \times (p + 1)$ in 1-D.

We can take advantage of the tensor product nature of the reference element in order to efficiently generate these operators on demand. All of the operators listed can be reduced to a tensor product of 1-D operators when applied in $d \geq 2$ dimensions. Upon requiring a certain operator in a calculation, the process is then as follows

1. Check if the 1-D version of the operator has already been computed in the database, if it has been computed, retrieve and goto step 3, if not goto step 2.
2. Compute the 1-D version of the operator and store in the MPI process-local database.
3. Apply the 2-D or 3-D version of the operator on a vector by using optimized Kronecker product matrix vector operation on the nodal vector v : $(A_{1D} \otimes B_{1D})\vec{v}$ for $d = 2$, or $(A_{1D} \otimes B_{1D} \otimes C_{1D})\vec{v}$ for $d = 3$. Here A, B, C denote the 1-D matrices of the respective operator, as applied to the first, second, and third dimension.

A Kronecker product matrix vector operation can be efficiently performed using a series of BLAS matrix-multiply calls, see (Buis & Dyksen (1996)) for example.

2.3.6 Implementation

To implement the multi-block adaptive mesh refinement we use the p4est library, which has been shown to scale to $O(100,000)$ cores (Burstedde et al. (2011)). We use PETSc (Balay et al. (2001)) for the Krylov subspace linear solves and the Newton Raphson iterations. The components of the multigrid algorithm is written by the authors and does not use PETSc.

2.4 Test Examples

We examine the components of our code through three test examples, the first a linear Poisson problem solved on a square grid where the solution is only C^2 -smooth at the $(0, 0)$ grid point. The second example is a non-linear elliptic problem, where we solve the Einstein constraint equations for the initial data of a constant density star. We then end the test examples section with a linear problem on a cubed-sphere with stretched outer boundary and a solution that falls off as r^{-1} with radial coordinate r as $r \rightarrow \infty$. Each test is aimed at isolating different aspects of the puncture black-hole problem, whose solution contains points that are C^2 -smooth and falls off as r^{-1} . The puncture black-hole problem further requires us to solve nonlinear Einstein constraint elliptic equations on a cubed-sphere mesh.

2.4.1 Poisson with $H^{4-\epsilon}$ solution

The first test problem we will investigate numerically is taken from (Stamm & Wihler (2010)), where the authors solve $\nabla^2 u = f$ on $\Omega = (0, 1)^2$ with the solution chosen as

$$u = x(1-x)y(1-y) \left[\left(x - \frac{1}{2} \right)^2 + \left(y - \frac{1}{2} \right)^2 \right]^{3/2}. \quad (2.55)$$

Here $u \in H^{4-\epsilon}$, where $\epsilon > 0$ so from Eq. (2.49) we expect third order convergence for uniform refinement. We have confirmed this numerically. However, with hp-adaptivity, it is possible to achieve $\|u - u_h\|_* \sim \exp(\text{DOF}^{\frac{1}{3}})$, as we show in Fig. 2.6. Notice also the close agreement between the estimator and energy norm $\|u - u_h\|_*$. The final mesh is shown in Fig. 2.7. Of note is the fact that the elements with the lowest polynomial order ($p=4$) are only in the vicinity of the C^2 -smooth $(0, 0)$ grid-point and the refinement level of the mesh is locally much higher here as well. Here we used AMR parameters $\gamma_h = 10$, $\gamma_p = 0.1$; at each AMR iterations all cells were refined which have $\eta^2(e)$ larger than $1/4$ of the overall mean of all $\eta^2(e)$. For this problem, we performed a survey of different choices for γ_h and γ_p . We found rather modest dependence on γ_h within $0.25 \lesssim \gamma_h \lesssim 10$, with $\gamma_h = 10$ leading to the good balance

between h - and p -refinement shown in Fig. 2.7. Our preferred value for the rest of the paper ($\gamma_h = 0.25$) also yields good convergence, but leads to a more uniform polynomial degree through the mesh.

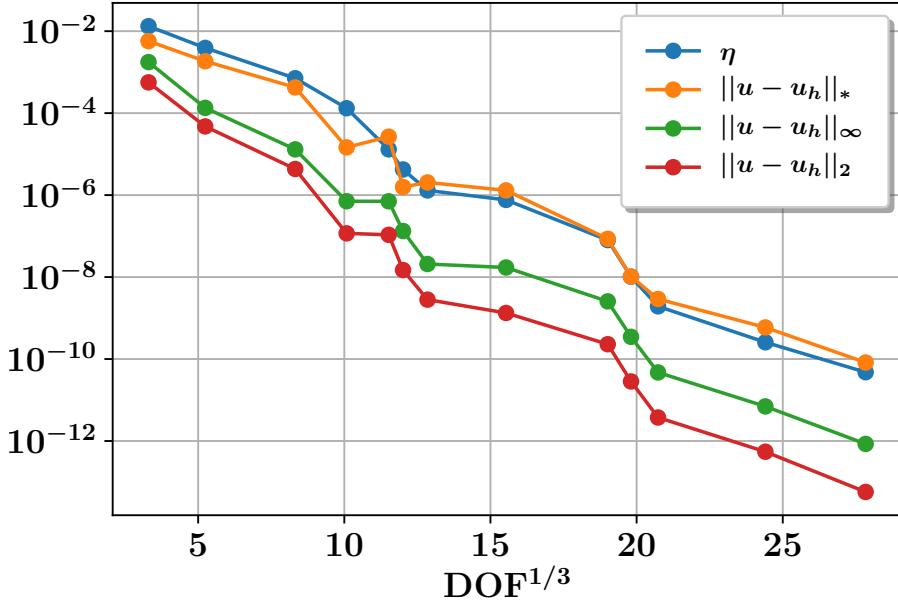


Figure 2.6: Problem A, Eq. (2.55): Convergence of the energy norm estimator η^2 and the error between the numerical solution u_h and the analytic solution u in the energy norm $\|\cdot\|_*$ (Eq. 2.33) and the L_2 norm $\|\cdot\|_2$ (Eq. 2.34).

2.4.2 Constant density star

For the next test problem, taken from (Baumgarte et al. (2007)), we solve the Einstein constraints in the simplest possible scenario, a constant density star. The goal of this test problem is to investigate how the elliptic solver behaves for problems that contain surface discontinuities that mimick surface and phase transition discontinuities in a Neutron star. The Einstein constraint equations for the case of a constant density star reduce to

$$\nabla^2\psi + 2\pi\rho\psi^5 = 0, \quad (2.56)$$

where ρ is the density of the star and ψ is the conformal factor which describes the deviation of the space from flat space. In (Baumgarte et al. (2007)) the authors solve the above equation for

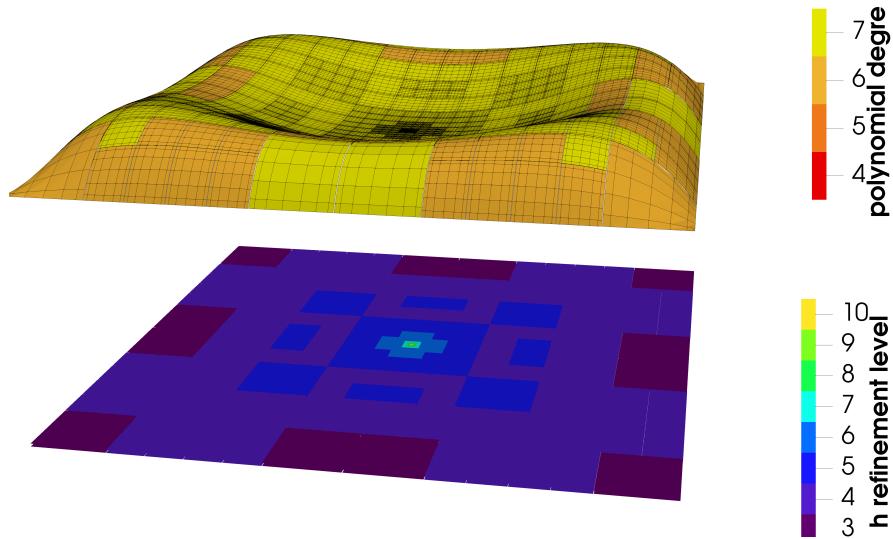


Figure 2.7: Problem A, Eq. (2.55): Visualization of the solution and the hp-refined computational mesh.

Top portion: The computational grid, color-coded by polynomial degree, with height representing the solution u . **Bottom portion:** Color-coded by h -refinement level. A cell on level l has size 2^{-l} of the overall computational domain.

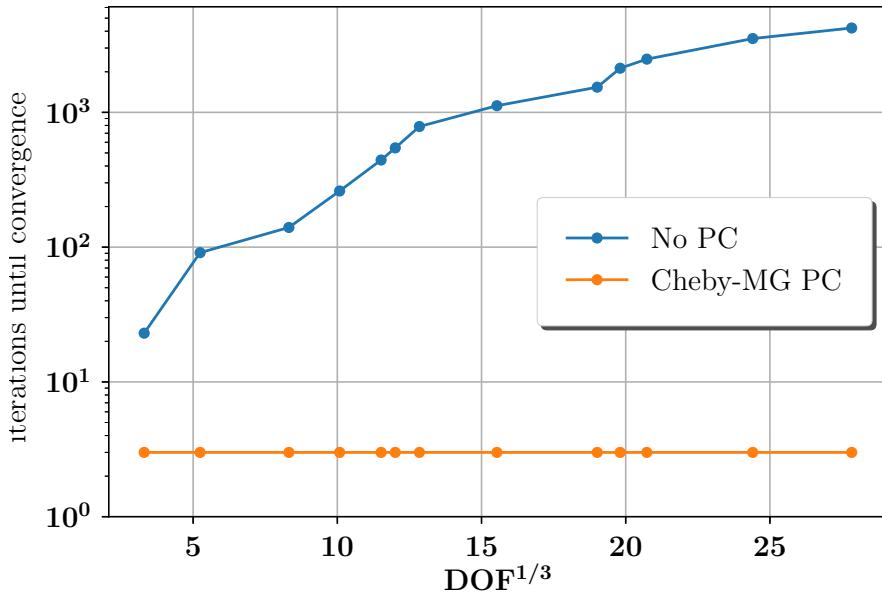


Figure 2.8: Problem A, Eq. (2.55): Iteration-count of the flexible conjugate gradient (FCG) Krylov subspace solver versus the Multigrid preconditioned FCG solver (MG-FCG). The MG-FCG completes all solves in ≤ 3 iterations.

the case of a star with radius r_0 and mass-density

$$\rho = \begin{cases} \rho_0 & r \leq r_0 \\ 0 & r > r_0, \end{cases} \quad (2.57)$$

where r is the radial spherical polar coordinate. Since the star is in isolation, the boundary condition at infinity is $\psi = 1$, corresponding to a asymptotically-flat space. For such a problem, there is an analytic solution given by

$$\psi = \begin{cases} Cu_\alpha(r) & r \leq r_0 \\ \frac{\beta}{r} + 1 & r > r_0. \end{cases} \quad (2.58)$$

with $C = (2\pi\rho_0/3)^{-1/4}$ and

$$u_\alpha(r) \equiv \frac{(\alpha r_0)^{1/2}}{(r^2 + (\alpha r_0)^2)^{1/2}}. \quad (2.59)$$

The parameters α and β are determined from the continuity of ψ and it's first derivative at the surface of the star, and are given by

$$\beta = r_0(Cu_\alpha(r_0) - 1) \quad (2.60)$$

$$\rho_0 r_0^2 = \frac{3}{2\pi} \frac{\alpha^{10}}{(1 + \alpha^2)^3} \quad (2.61)$$

We solve the above problem on a cubic domain with $\rho_0 = 0.001$ and analytic Dirichlet boundary conditions on a boundary at $8r_0$. The non-linear term in Eqn. 2.56 is handled by first interpolating ψ onto the GL quadrature points of an element, evaluating ψ^5 at the GL quadrature points and then performing the necessary Gaussian quadrature sum on each element. Figure 2.9 we showcase the convergence of the solution and note the nice agreement between the energy norm estimator η^2 and the energy norm of the analytic error. To achieve this convergence we used the following AMR parameters, $\gamma_h = 0.25$, $\gamma_p = 0.1$. At each AMR refinement iteration, 10% of the elements are refined. In Figure 2.10 we show a comparison between

multigrid-preconditioned FCG iterations and unpreconditioned FCG iterations. We notice that for the preconditioned case, the iteration count is roughly constant with increases in DOF, whereas the unpreconditioned iterations grow with DOF. Finally, in Figure 2.11 we show the mesh after the last AMR step. This mesh showcases the highest h-refinement around the star boundary (yellow circle) as expected, since this area is not smooth.

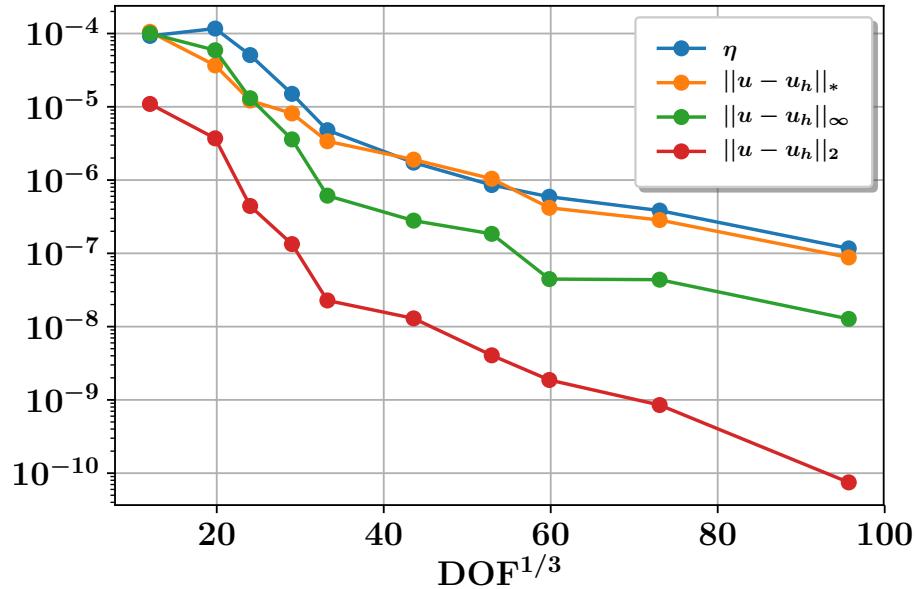


Figure 2.9: Problem B, Eq. (2.56): Convergence of the energy norm estimator η^2 and the error between the numerical solution u_h and the analytic solution u in the energy norm $\|\cdot\|_*$ and the $\|\cdot\|_2$ norm

2.4.3 Cubed sphere Meshes and Stretched Boundary Elements

So far, we have only investigated meshes with a regular, Cartesian structure and with a rectangular outer boundary at close distance. We will now investigate scenarios with a spherical outer boundary, where we use a macro-mesh arising from the 3-dimensional generalization of Fig. 2.1 (a). We will also place the outer boundary at very large radius, typically 10^9 , to approximate boundary conditions at infinity. Typically, such problems have solutions which fall off as a power series in $1/r$.

Figure 2.12 shows the structure of the mesh we will use: a central cube, surrounded by *two* layers of six deformed macro-elements each. The inner layer interpolates from the cubical inner

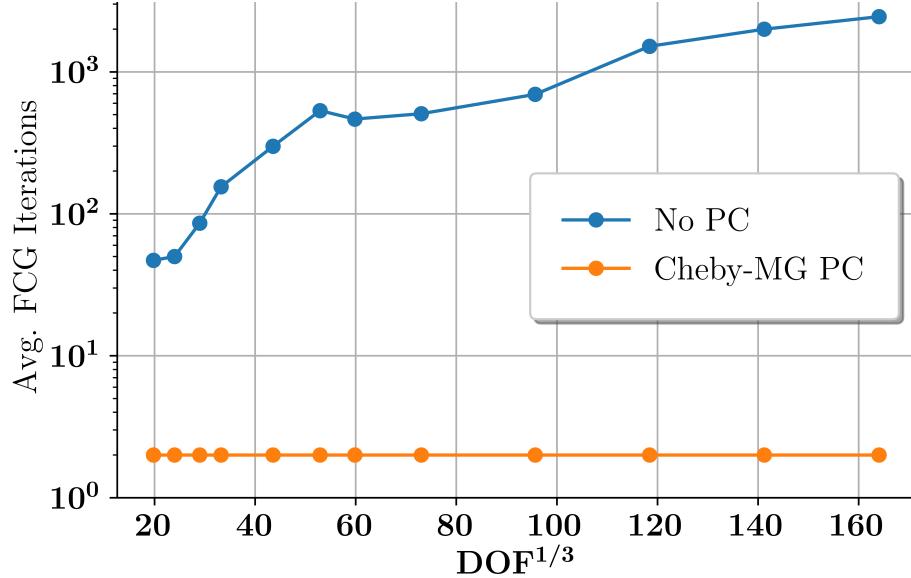


Figure 2.10: Comparison of the average number of iterations per Newton-Raphson step when using a Chebyshev smoothed Multigrid preconditioner and no preconditioner.

region to a spherical outer region. The outer layer has spherical boundaries both at its inner and outer surface, and thus radial coordinate lines that are always orthogonal to the angular coordinate lines. This allows to apply a *radial coordinate transformation* in the outer layer, to move its outer boundary to near infinity. Because we know the solution will fall off as $1/r$ we use an inverse mapping in the outer six spherical wedges of the cubed-sphere (blocks 0-5 in Fig. 2.12). This mapping is defined as follows. Denote the physical grid variable with $r \in [r_1, r_2]$ and the collocation-point integration variable as $x \in [x_1, x_2]$, then the inverse mapping is defined by

$$r = \frac{m}{x - t}, \quad (2.62)$$

where

$$m = \frac{x_2 - x_1}{\frac{1}{r_2} - \frac{1}{r_1}}, \quad t = \frac{x_1 r_1 - x_2 r_2}{r_1 - r_2}. \quad (2.63)$$

We will investigate the discontinuous Galerkin method on the following test problem which captures many of the above features:

$$\nabla^2 u = 3(1 + x^2 + y^2 + z^2)^{-\frac{5}{2}}, \quad (2.64)$$

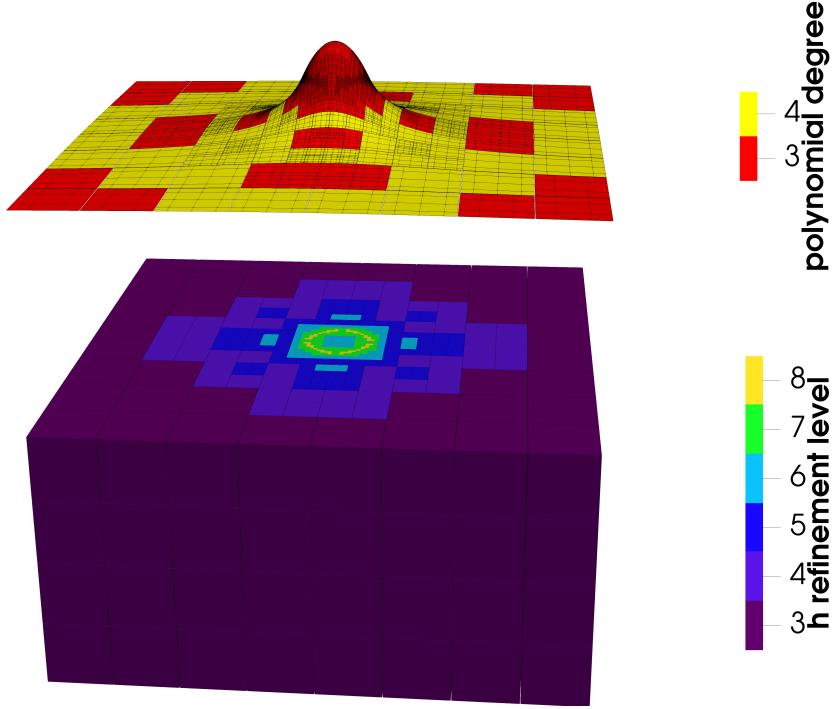


Figure 2.11: Visualization of hp-refined computational mesh for Problem B, Eq. (2.56). **Top portion:** The $z = 0$ cross-section of the computational grid, color-coded by the polynomial degree and with height representing the solution ψ . **Bottom portion:** Volume rendering of the $z \leq 0$ part of the computational domain, color-coded by the h -refinement level. A cell on level l has size 2^{-l} of the overall computational domain.

with $\Omega = \{(x, y, z) : x^2 + y^2 + z^2 < R\}$ and Dirichlet boundary conditions given by the analytical solution, which is a Lorentzian function $u = (1 + r^2)^{-1/2}$. The Lorentzian function falls off as a power series in $1/r$ as $r \rightarrow \infty$.

To solve this test problem we run two schemes: uniform p -refinement and adaptive p -refinement. We run only with p-AMR because the underlying solution is smooth everywhere so there is very little to no benefit in running with hp-AMR for this problem (which we also found to be the case empirically). For the uniformly refined run, we start with refinement level $l = 4$, i.e. with $2^{3l} = 4096$ elements in each of the 13 macro-elements, and increase p from 2 to 11.

To achieve pure p-AMR with the hp-AMR scheme outlined in Sec. 2.3.5, we start with refinement level $l = 2$, i.e. with $2^{3l} = 64$ elements in each of the 13 macro-elements, and with

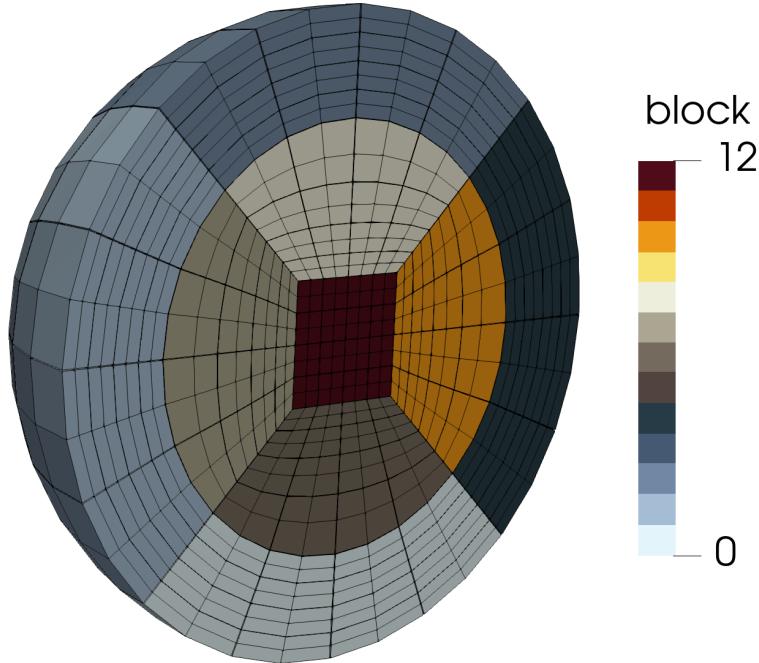


Figure 2.12: The mesh structure for a computational domain with spherical outer boundary. This structure consists of 13 macro-meshes (shown in different colors). For clarity, only the $z \leq 0$ part of the mesh is shown.

polynomial order $p_e = 1$ in all elements. We set $\gamma_h = 10^6$, $\gamma_p = 10^6$ and in each AMR iteration, we refine the 25% elements with the largest estimator. Figure 2.13 shows the convergence of the p-AMR scheme versus the p-uniform scheme. We stop at the maximum polynomial order currently allowed in the code, $p = 19$. Comparing the L_∞ norm between the p-AMR and p-uniform runs, the number of degrees of freedom per dimension, $\text{DOF}^{1/3}$, is roughly cut in half. The L_2 norm is slower to converge because it is dominated by the contributions of the outer stretched wedges, and this region is less aggressively refined by the AMR. Figure 2.14 shows the final mesh for the problem. The AMR algorithm increases the polynomial order p predominantly in the centre, where the solution has the most structure.

Strong coordinate stretching, as performed via the inverse map Eq. (2.62) leads to the following problem: The error estimator $\eta^2(e)$ utilizes integrals in physical coordinates in Eqs. (2.52). For strongly stretched grids (e.g. with inverse mappings where R is orders of magnitude larger than other length scales in the problem) these volume integrals will place a large

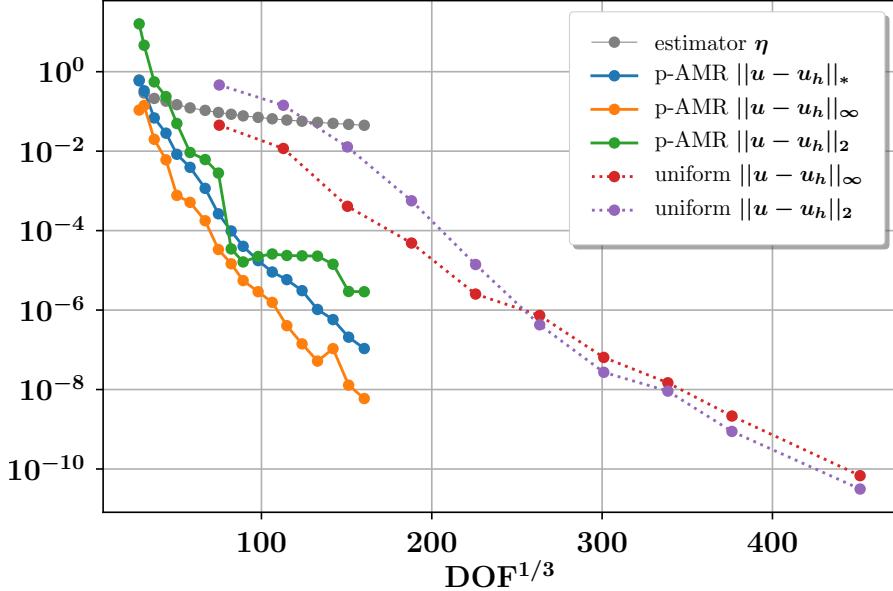


Figure 2.13: Problem C, Eq. (2.64): Convergence of the solution as the degree p is uniformly increased across all elements for uniform case and as the degree p is adaptively increased in the amr case. (Dirichlet BC at $R=1000$).

emphasis on the regions at large distance. AMR will then aggressively refine in the stretched region despite the pointwise errors (as measured by the L_∞ norm for individual elements) being very small. Such regions tend to have a very low L_∞ because the numerical solution is very accurate there, but a high L_2 because of the large volume in the stretched region, thus also explaining why the L_2 norm of the p-AMR run in Fig. 2.13 fails to converge as smoothly as the L_∞ norm.

We solve this problem by introducing a “compactified grid”, which has the same structure as the physical grid, but without the compactification in the physical grid. We then compute the integrals for the estimator in Eqs. (2.53) on this compactified grid. The integrands, Eqs. (2.52), are as before computed on the physical grid. In essence, this procedure merely changes the weighting of the different regions of the grid via the Jacobians J in the integrals and the parameters h_e, h_m which are now computed on the compactified grid. In practice, we use as compactified grid a cubed-sphere mesh where the outer spherical shell extends from radius 2 to radius 3, and where the middle cube has a side-length of $2/\sqrt{3}$.

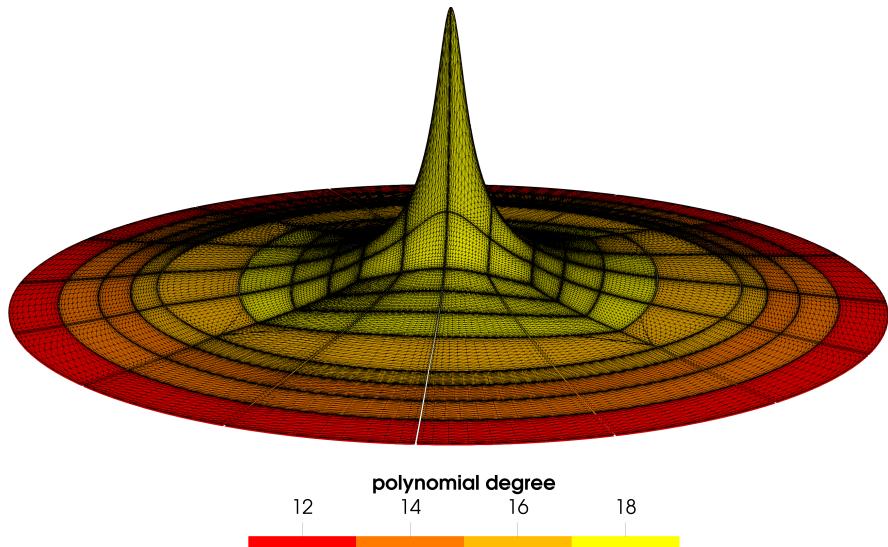


Figure 2.14: Visualization of the final mesh for problem C. Shown is a xy -plane slice of the mesh which has been warped so that the height of the surface corresponds to the value u of the solution, color coded by the polynomial degree. For ease of visualization, grid-points are mapped onto the compactified grid on which the estimator η_e is computed; the compactified outer radius $R = 3$ corresponds to the physical outer radius $R = 1000$.

Figure 2.15 shows the convergence of the L_∞ norm using the compactified grid versus the non-compactified grid. For $R = 1000$, the standard estimator (“non-compact η ”) converges well for the first 10 AMR iterations, but then stalls. For $R = 10^9$, the standard estimator fails to yield convergence of the solution at all. For both cases, the new estimator (“compact η ”) results in good convergence. We stress that the compactified estimator is only used in driving the AMR refinement. The DG-scheme is always formulated in the physical domain, and also the error plotted in Fig. 2.15 is computed from the solution on the physical domain with outer boundary $R = 1000$ or 10^9 . The p-AMR run shown in Fig. 2.13 uses the compactified estimator.

Finally, we investigate the efficiency of preconditioners for the p-AMR run with $R = 1000$ and the compactified estimator, i.e. the run plotted in orange in Figs. 2.13 and 2.15. Figure 2.16 presents the iteration counts for four different kinds of preconditioning. Chebyshev-smoothed multigrid preconditioner loses efficiency on cubed spheres with stretched boundaries, possibly due to a poorly estimated upper eigenvalue in Alg. 5. However, using the more powerful

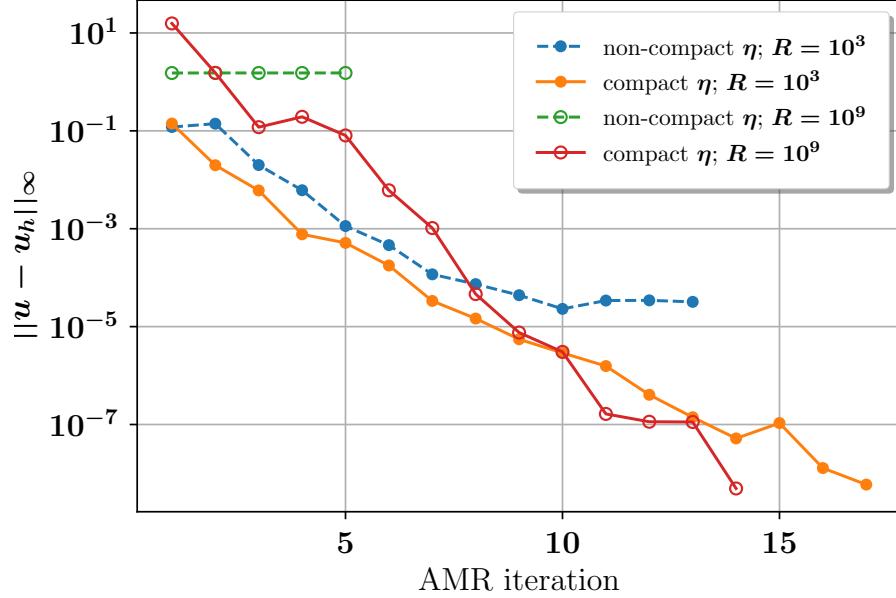


Figure 2.15: Problem C, (Eq. 2.64): Convergence of the solution in the L_2 norm (See Eq. 2.34) as the degree p is adaptively refined with a compact and non-compact estimator. (Dirichlet BC at $R=1000$ and $R = 10^9$). All runs use the AMR/dG parameters of the p-amr run in Figure 2.13.

domain decomposition additive Schwarz method, we can regain the efficiency of the multigrid-preconditioner seen in the previous two sections. Here, the Schwarz subdomains have $N_{\text{overlap}} = 2$, and each multi-grid iteration employs $N_{\text{iter,Sch}} = 3$ iterations of Schwarz-smoothing with $\text{rtol}=1\text{e-}3$. Chebyshev uses $N_{\text{iter,eigs}} = 15$ iterations for the eigenvalue estimate, and $N_{\text{iter,Cheb}} = 15$ iterations in the Chebyshev smoother.

2.4.4 Puncture Initial Data

There are various approaches to solving for binary black hole initial data sets and these approaches are primarily distinguished by the initial choice of hypersurface and how the physical singularity inside the black holes is treated. One possibility when considering two black holes is to work on \mathbb{R}^3 with two balls excised(Cook (1994); Cook & Pfeiffer (2004); Caudill et al. (2006)). This approach has been shown to work well with the spectral finite element method (e.g. (Pfeiffer et al. (2003a))), but is more problematic for finite-difference codes because special stencils must be created near the curved boundaries. Another popular approach, which is more amenable

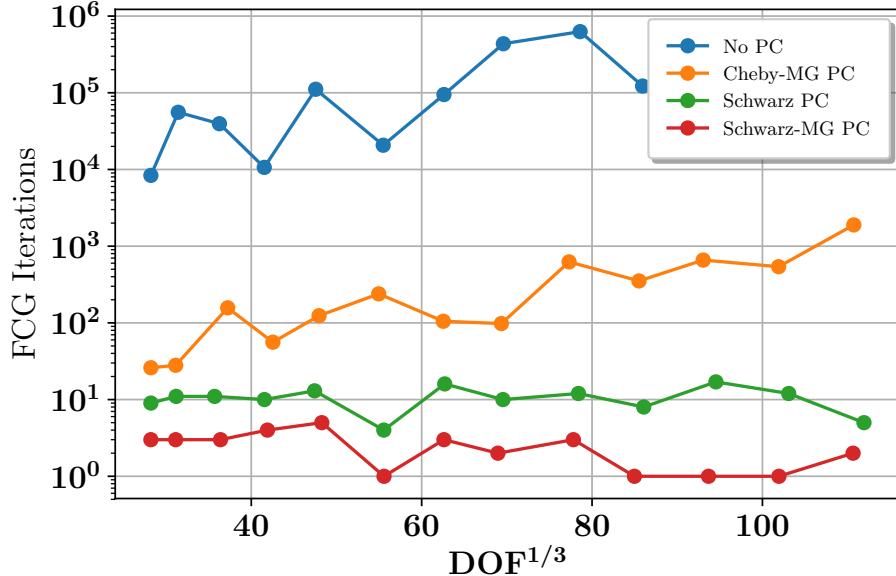


Figure 2.16: Comparison of four different methods of preconditioning FCG. All runs use the AMR/dG parameters of the p-amr run in Figure 2.13.

to finite difference codes is the puncture method (Brandt & Brügmann (1997)), where an elliptic equation is solved on \mathbb{R}^3 , with two points where the solution becomes singular. These points represent the inner asymptotically flat infinity (Brill-Lindquist topology). The puncture method simplifies the numerical method because no special inner boundary condition has to be considered, however the solution is only C^4 smooth at the puncture points (Brandt & Brügmann (1997)). Without using contrived coordinate systems to remove the C^4 -smooth nature of the punctures (See (Ansorg et al. (2004)) for example), spectral methods cannot perform optimally, because they would require the solution to be smooth on the computational domain in order to obtain exponential convergence. We choose to solve for binary black hole initial-data with the puncture method in this paper for two reasons. The first is that the method does not couple well with traditional spectral schemes as discussed above and this allows us to compare the discontinuous Galerkin method to the spectral method. Secondly, the equation we must solve is less complicated than the excision case because it only involves a solve for a single field, the conformal factor, as opposed to the 5 fields one must solve for with the excision method (see e.g. (Pfeiffer et al. (2003a))), so it is easier to implement numerically.

Nevertheless, puncture data provides a testing ground for many of the new techniques developed here: Singular points which benefit from h-refinement; smooth regions that benefit from p-refinement; a spherical outer boundary requiring the cubed-sphere domain shown in Fig. 2.12; and a boundary at infinity (or near infinity) which requires a compactified radial coordinate. Moreover, for testing of adaptivity, it is easy to add arbitrarily many black holes each represented by its own singularity, at arbitrary coordinates with arbitrary spins.

For the case of puncture data, the initial data equations of general relativity reduce down to a single equation (Brandt & Brügmann (1997)):

$$-\nabla^2 u = \frac{1}{8} \bar{A}^{ij} \bar{A}_{ij} \psi^{-7} \quad (2.65)$$

where \bar{A}_{ij} is a spatially dependent function given by

$$\bar{A}_{ij} = \frac{3}{2} \sum_I \frac{1}{r_I^2} [2P_I^I n_j^l - (f_{ij} - n_i^l n_j^l) P_I^k n_k^I + \frac{4}{r_I} n_{(i}^l \epsilon_{j)kl} S_I^k n_I^l]. \quad (2.66)$$

Here, n_I^i are the spatially varying components of the radial unit-vector $\hat{n}_I(\vec{x}) = (\vec{x} - \vec{c}_I)/|\vec{x} - \vec{c}_I|$ relative to the position \vec{c}_I of the I -th black hole (Brandt & Brügmann (1997)). The constant vectors \mathbf{P}_I and \mathbf{S}_I quantify the momentum and spin of the I -th black-hole and $\psi = 1 + \sum_I \frac{m_I}{2r_I} + u$.

The boundary condition is

$$u \rightarrow 0, \quad |\vec{x}| \rightarrow \infty. \quad (2.67)$$

We solve the above elliptic PDE using first the spectral code SpEC and then the dG solver presented in this paper. We solve for the case of two orbiting equal mass black holes with momenta ± 2 , zero spin and initial positions $(\pm 3, 0, 0)$ in units of total mass M . This is the test case used in (Ansorg (2007)). Since there is no analytical solution, we will compare the solutions between refinement levels at four reference points on the x-axis. These are $(0, 0, 0)$, $(3, 0, 0)$, the location of the right-most puncture, $(10, 0, 0)$ and $(100, 0, 0)$.

The SpEC solver was already used to solve for puncture data in (Dennison et al. (2006); Lovelace et al. (2008)). This spectral code is apriori not well suited for puncture data which results in a non-smooth solution $u(\vec{x})$. Because we know where the singularities of the punctures

are, this can be overcome manually, by covering the punctures with very small rectangular blocks, at high enough resolution, to compensate for the loss of exponential convergence. Figure 2.17 shows the difference in the solution at the four reference points as the resolution is manually increased. We emphasize that this solution obtained with SpEC depends on (i) *prior knowledge* of the locations of the singularities; and (ii) tuning of SpEC’s mesh and resolution *by hand*.

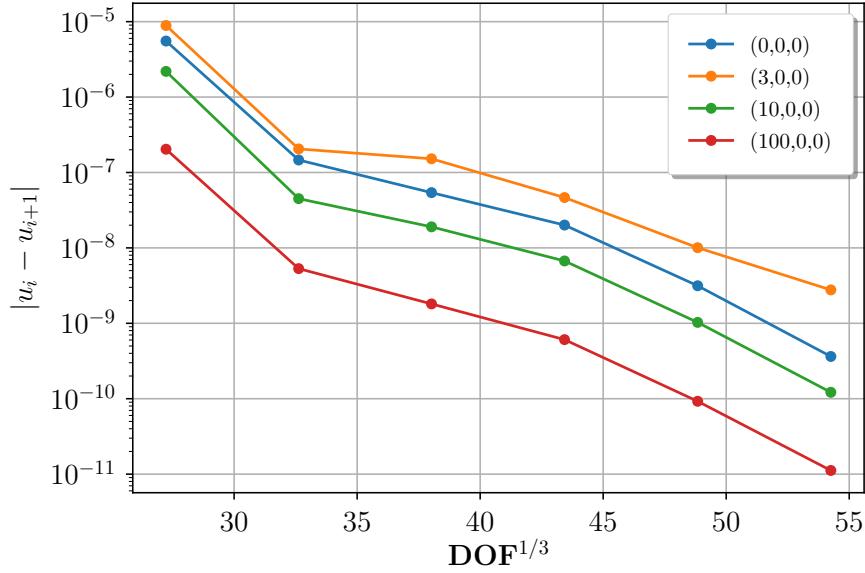


Figure 2.17: Convergence of the SpEC–elliptic solver with manually adopting the domain-decomposition and manual adjustment of resolution to compensate for the singularities. Plotted are differences to the next-*lower* resolution at four points in the computational domain.

For the dG solver, we start with a uniformly refined cubed-sphere mesh at level $l = 1$, i.e. 13 blocks, each consisting of eight cells. The outer radius at 10^{11}M and the size of the inner cube is 10M . We start further with elements of polynomial order $p = 2$. The location of the punctures is *not* utilized in the dG code, and all mesh-refinement is automatic, driven by Alg. 7 with parameters $\gamma_h = .25$ and $\gamma_p = .1$ and we refine the top 12.5% of elements. In order to run a Schwarz smoother for this problem we would need to transfer ghost-data for the operator described by Eqn. (2.48) whenever a Schwarz subdomain contains a ghost element. While this is by no means problematic, we have not yet implemented the infrastructure to do it, so we just

use a Chebyshev smoother when we precondition this problem with Multigrid. Figure 2.18 shows the convergence of the four reference points with respect to the finest grid SpEC solution between AMR levels. We first see that in terms of overall DOF, the dG solver doesn't do much worse than the finely tuned SpEC solver, even though the dG solver has to adaptively find the punctures, has a larger initial error and has no h-or-p coarsening, so mistakes in the refinement cannot be fixed. Thus, taken all of this into account, the convergence is highly satisfactory. The bounce in the $(3, 0, 0)$ at the second to last iteration arises because the dG-solution oscillates around the SpEC solution and coincidentally is shown near a zero-crossing. Figure 2.20 shows the solution on the final mesh, which has the highest h-refinement exactly at the points of the punctures, as desired.

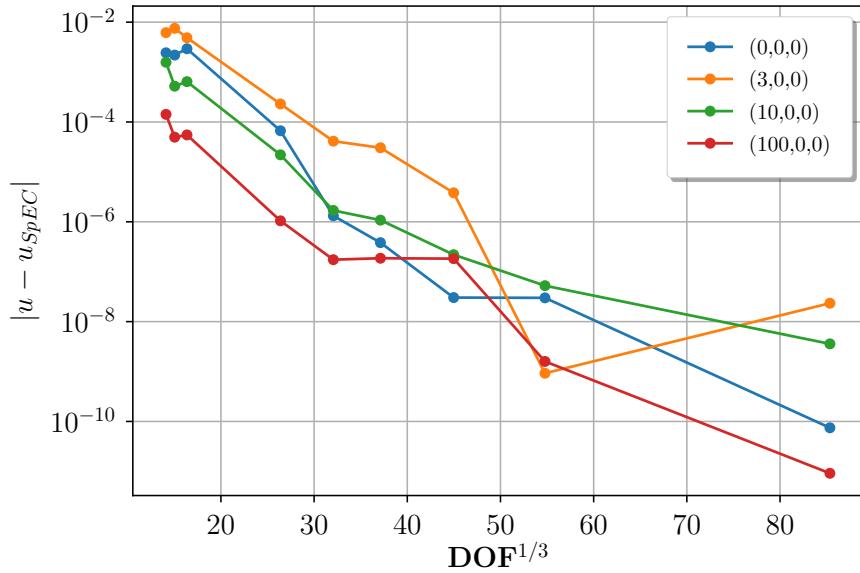


Figure 2.18: Problem D: Black hole initial data with two punctures. Convergence of the error between the dG solution and the SpEC solution.

Next, we solve for the puncture initial data of three black-holes randomly located in the xy -plane, with random spins and random momenta. Spectral solvers such as SpEC cannot perform well when the singular points on the grid are not known in advance. Thus, we end this paper showcasing a problem that our discontinuous Galerkin code can solve, but SpEC cannot. Table ?? illustrates the parameters for the randomly placed punctures and their spin and

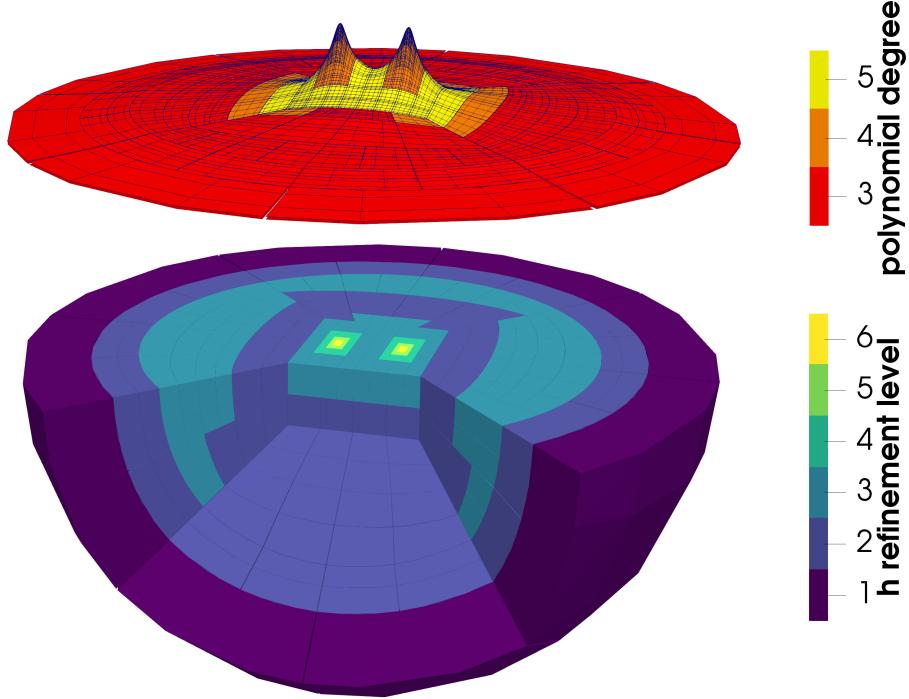


Figure 2.19: Problem D (Black hole initial data with two punctures): Visualization of the hp-refined computational mesh. The bottom portion of the image shows a volume rendering of the $z < 0$ portion of the computational domain, with two blocks removed, and color-coded by the h -refinement level. The top portion of the image shows the $z = 0$ cross-section of the computational grid, color-coded by the polynomial degree, with the height representing the solution u . For ease of visualization, grid-points are mapped onto the compactified grid on which the estimator η_e is computed; the compactified outer radius $R = 3$ corresponds to the physical outer radius $R = 10^{11}$.

momenta.

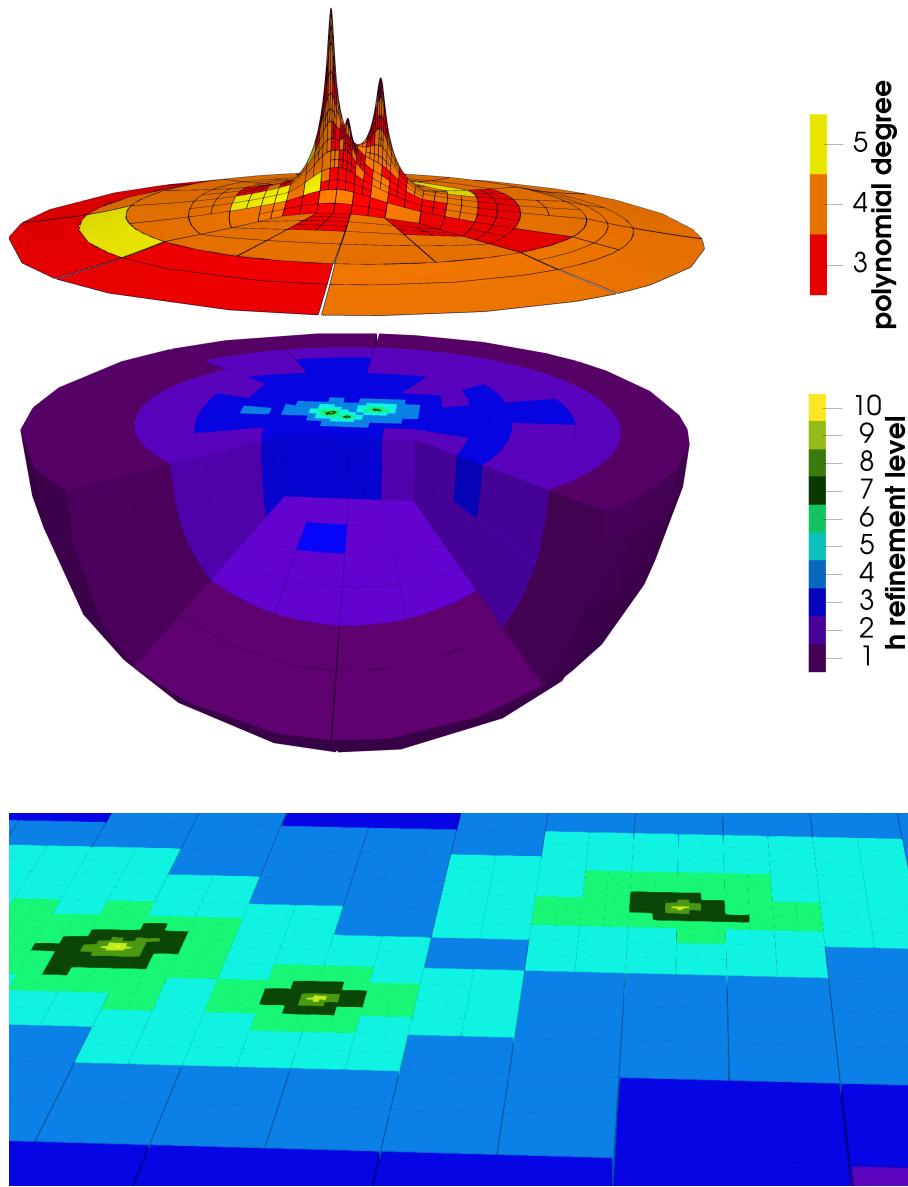


Figure 2.20: Problem E (Black hole initial data with three punctures): Visualization of the solution, and the hp-refined computational mesh. **Top portion:** the $z = 0$ cross-section of the computational grid, color-coded by the polynomial degree, with the height representing the solution u . **Middle portion:** volume rendering of the $z < 0$ part of the computational domain, with two blocks removed, and color-coded by the h-refinement level. **Bottom portion:** Zoom into the middle portion, highlighting the region near the three punctures with highest refinement level. For ease of visualization, grid-points are mapped onto the compactified grid on which the estimator η_e is computed; the compactified outer radius $R = 3$ corresponds to the physical outer radius $R = 10^{11}$.

	Puncture 1	Puncture 2	Puncture 3
m	0.2691	0.4063	0.3245
x	0.0152	-2.316	-1.0279
y	-0.6933	1.8274	-2.2711
P_x	0.0585	-0.0284	0.1640
P_y	0.0082	-0.1497	0.0515
S_z	-0.0134	-0.0332	-0.0708

Table 2.1: The randomly generated parameters for the three black holes. We list the mass m , the position $(x, y, 0)$, the momentum $(P_x, P_y, 0)$ and the spin $(0, 0, S_z)$ of the black holes.

Figure 2.21 shows the convergence of four points, three at the location of the punctures, and one at $(100, 0, 0)$. We use amr parameters $\gamma_h = 0.25$, $\gamma_p = 0.1$, $F_{\text{refined}} = 0.125$ for this run.

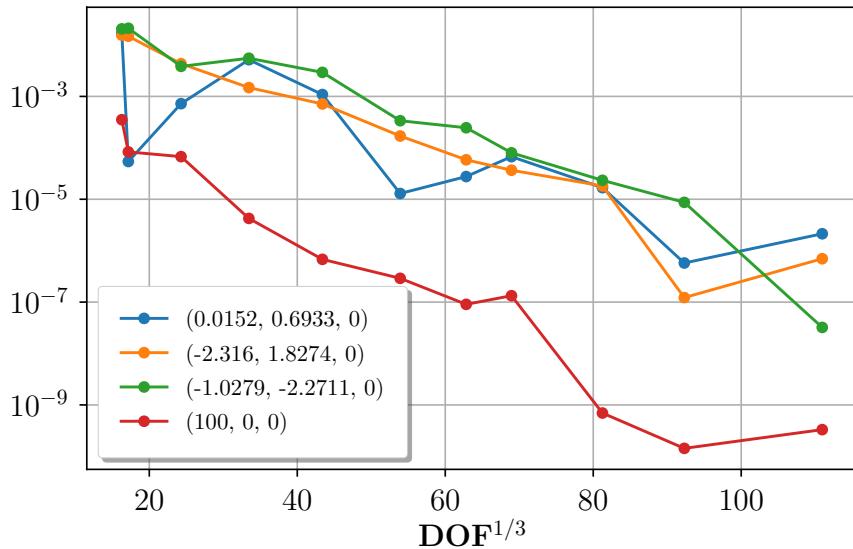


Figure 2.21: Problem D: Black hole initial data with three randomly generated punctures. Convergence of the error between AMR steps at four different points, three corresponding to the location of the punctures and one at $(100, 0, 0)$.

2.5 Conclusion and Future Work

We presented a new code for solving elliptic equations intended for numerical relativity. The methodology we use differs from other codes in the field in many important respects. In particular, we use a discontinuous Galerkin method to discretize the equations, an hp-adaptive mesh refinement scheme driven by an a posteriori estimator and a matrix-free, scalable Multigrid preconditioned Newton-Krylov solver. Individually, many of the features of our code have been implemented before (Kozdon & Wilcox (2018); Kozdon et al. (2019); Stiller (2017); Sundar et al. (2012); Hesthaven & Warburton (2008)), but they have never been combined together to create a general dG solver. In particular, the combination of curved meshes (cf. Fig. 2.12) and non-conforming elements is novel and is crucial for generic solution-driven AMR. Moreover, the compactified AMR-driver introduced in Sec. 2.4.3 is also new, allowing for compactified computational domains with outer boundary near infinity. Lastly, the use of a multigrid-preconditioned solver with a Schwarz (or Chebyshev) smoother on non-polygonal meshes has not been investigated in a dG setting until this paper.

For BBH puncture data, our new code approaches the accuracy of existing, specialized codes like SpEC. In addition, the automatic AMR in the new code does not require manual tuning of the computational mesh, and does not independent of prior knowledge of features of the solution like the location of black hole punctures. The new code already improves on the more specialized codes by being able to handle an arbitrary number of puncture-black holes. AMR can also automatically resolve discontinuities without prior knowledge of their existence (cf. Fig. 2.11).

Moving forward, there are still several areas of improvement our code could possibly benefit from:

1. **Load balancing:** For problems that require adaptive mesh refinement and multi-grid, there will naturally be a unbalanced number of degrees of freedom (DOF) across processors and this can slow down the Krylov iterations substantially. This can be ammended by

incorporating a task-based parallelism framework for load balancing. We plan on reworking our elliptic solver into the task-based parallel code SpECTRE, which is concurrently being developed (Kidder et al. (2017)).

2. **Anisotropic refinement:** Most realistic problems have some anisotropy and therefore a solver would benefit from anisotropic mesh refinement. Indeed, most of the problems in this paper could have had better convergence with anisotropic refinement, for instance, Problem C is spherically symmetric, and puncture data is approximately spherically symmetric at large distance. We use the p4est framework for mesh refinement and while it has support for anisotropic refinement, the direction of the anisotropy has to be known a priori. We look to go beyond this and have general refinement in a future edition of our code.
3. **Hybridizable dG:** The discontinuous Galerkin method can be quite expensive in terms of the amount of DOF it requires to converge to a certain error. Recently, a method called Hybridizable dG has been coupled with matrix-free multigrid methods to solve elliptic problems with substantially reduced DOF over the classical dG method (Fabien et al. (2019); Muralikrishnan et al. (2019)). Whether this method can be fully incorporated into the complex scheme presented in this paper, will be an area of further inquiry.

In future work, we plan to use this solver to expand the physics in compact object initial data, for instance, neutron star initial data for equation of state with phase-transitions, neutron stars with very high compactness (where current solvers fail (Henriksson et al. (2016))), or compact objects in alternative theories of gravity, or with novel matter fields like boson stars.

Chapter 3

Unequal Mass Binary Neutron Star Simulations with M1 Neutrino Transport: Ejecta and Neutrino Emission

3.1 Chapter Overview

The material in this chapter is based on "Unequal Mass Binary Neutron Star Simulations with M1 Neutrino Transport: Ejecta and Neutrino Emission" by Trevor Vincent, Francois Foucart, Roland Haas, Matthew Duez, Lawrence Kidder, Harald Pfeiffer, Mark Scheel being prepared for Phys. Rev. D.

We present twelve new simulations of unequal mass neutron star mergers. The simulations were preformed with the SpEC code, and utilize nuclear-theory based equations of state and a two-moment gray neutrino transport scheme with an improved energy estimate based on evolving the number density. We model the neutron stars with the SFHo, LS220 and DD2 equations of state (EOS) and we study the neutrino and matter emission of all twelve models to search for robust trends between binary parameters and emission characteristics. We find that the total mass of the dynamical ejecta exceeds $0.01M_{\odot}$ only for SFHo with weak dependence on the mass-ratio across all models. We find that the ejecta have a broad Y_e distribution ($\approx 0.06 - 0.48$),

with mean 0.2. Y_e increases with neutrino irradiation over time, but decreases with increasing binary asymmetry. We also find that the models have ejecta with a broad asymptotic velocity distribution ($\approx 0.05 - 0.7c$). The average velocity lies in the range $0.2c - 0.3c$ and decreases with binary asymmetry. Furthermore, we find that disk mass increases with binary asymmetry and stiffness of the EOS. Y_e of the disk increases with softness of the EOS. Finally, the strongest neutrino emission occurs for the models with soft EOS and we find no significant dependence of the magnitude or angular distribution of the neutrino luminosity with mass-ratio except in the case of the heavier (non-electron) neutrino species.

3.2 Introduction

The binary NS merger GW170817 was a landmark event, combining the first detection of gravitational waves with an observation of a short gamma ray burst and a kilonova (Abbott et al. (2017b;c); Abbott et al. (2017); Ajello et al. (2018)). Following the event, studies emerged analyzing many aspects of GW170817, from the internal structure of neutron stars (Read et al. (2009); Del Pozzo et al. (2013); Lackey & Wade (2015); The LIGO Scientific Collaboration et al. (2018a;b)), to the production of short gamma-ray bursts (Mochkovitch et al. (1993); Lee & Kluzniak (1999); Janka et al. (1999); Abbott et al. (2017c); Abbott et al. (2017); Mooley et al. (2018)) and the synthesis of r-process elements (Li & Paczynski (1998); Lattimer & Schramm (1976); Rosswog et al. (1999); Kulkarni (2005); Metzger et al. (2010); Metzger (2017a); Kasliwal et al. (2017); Coulter et al. (2017); Chornock et al. (2017); Coulter et al. (2017); Cowperthwaite et al. (2017); Kasen et al. (2017); Kilpatrick et al. (2017); McCully et al. (2017); Nicholl et al. (2017); Pian et al. (2017); Smartt et al. (2017); Soares-Santos et al. (2017); Tanvir et al. (2017); Evans et al. (2017)). with the increasing sensitivity of advanced gravitational wave interferometers many binary neutron star (BNS) detections will be made in the next decade (LIGO et al. (2018)).

Numerical simulations of mergers play a crucial role in efforts to model the gravitational wave signal, predict the properties of its electromagnetic counterparts, and estimate the production of

r-process elements from the merger. In this work, we focus on the matter and neutrino emissions from BNS mergers. In particular we look at the ejecta and neutrino emission from a new set of twelve asymmetric-mass ratio BNS simulations which extends a previous set of 4 equal mass ratio BNS simulations (Foucart et al. (2016b)). While asymmetric-mass ratio BNS simulations have been studied before in the context of general-relativistic-radiation hydrodynamics, e.g. (Sekiguchi et al. (2016a)), (Lehner et al. (2016)), (Radice et al. (2016)), these previous studies use either a less advanced neutrino scheme and/or different mass-ratios and EOS. Thus this paper adds to the ongoing effort to simulate BNS systems and characterize their observables.

Neutrino interactions were first included in general relativistic simulations of neutron star mergers through a simple leakage scheme (Sekiguchi et al. (2011)), based on approximate methods developed for Newtonian simulations (Ruffert et al. (1996); Rosswog & Liebendörfer (2003)). A leakage scheme uses the local properties of the fluid and an estimate of the neutrino optical depth to determine the amount of energy lost locally to neutrino-matter interactions, and the associated change in the composition of the fluid. Leakage schemes provide an order-of-magnitude accurate estimate of neutrino cooling in the post-merger remnant, and have thus been used to capture the first-order effect of neutrino-matter interactions in general relativistic simulations of compact binary mergers (Sekiguchi et al. (2011); Wanajo et al. (2014); Lehner et al. (2016); Radice et al. (2016); Palenzuela et al. (2015); Deaton et al. (2013); Foucart et al. (2014)). The inclusion of neutrino-matter effects with the leakage scheme, while crude, significantly affected the composition, morphology and total mass of the outflows, with some studies showing a factor 2 difference in total ejecta mass (Radice et al. (2016)). However, most implementations of leakage do not account for irradiation of low density regions by neutrinos emitted from hot, dense regions. This potentially leads to large errors in the composition of the outflows, mostly by underestimating the number of protons (Foucart et al. (2016); Foucart et al. (2015b)). Accordingly, the simplest leakage schemes are very inaccurate when attempting to predict the properties of post-merger electromagnetic signals. The only general relativistic simulations going beyond leakage use a moment formalism with an analytic closure to approximate the

Boltzmann equation (Thorne (1981); Shibata et al. (2011)). In particular, neutron star merger simulations have been performed with a gray M1 scheme (Foucart et al. (2016); Foucart et al. (2015b); Sekiguchi et al. (2015; 2016a)), in which the energy density and flux density of each neutrino species are evolved. In BNS mergers, the use of this moment formalism showed that a range of compositions and thus of nucleosynthesis outcomes, exists in the material ejected by the merger (Wanajo et al. (2014)).

Previous studies of asymmetric mass-ratio BNS systems with fully general-relativistic radiation-hydrodynamics include (Sekiguchi et al. (2016a); Lehner et al. (2016); Radice et al. (2018)). Sekiguchi et al. (Sekiguchi et al. (2016a)) studied two EOS, SFHo and DD2 with mass ratios between .86 and 1.0 and a fixed total mass of $2.7M_{\odot}$ to around 30-ms post-merger using a M1 neutrino transport scheme. They found that for SFHo the ejecta mass depended weakly on mass-ratio, but the average electron number per baryon decreased with mass ratio. For DD2 these trends were reversed. They also found that only the soft EOS, SFHo, produced ejecta mass above $0.01M_{\odot}$. Lehner et al. (Lehner et al. (2016)) studied three EOS, NL3, SFHo and DD2 with mass ratios between .76 and 1.0 and a fixed total mass of $2.7M_{\odot}$ at 3-ms post-merger using a neutrino leakage scheme. They found that there was a greater ejecta mass with increasing binary asymmetry. Unlike Sekiguchi et al., Lehner et. al found that none of the EOS produced ejecta above $0.01M_{\odot}$. Finally, Radice et al. (Radice et al. (2018)) studied four EOS, BHB $\Lambda\phi$, SFHO, DD2 and LS220, with mass ratios between .85 and 1.0 and a fixed total mass of $2.7M_{\odot}$ to around 20-ms post-merger using a neutrino leakage scheme and a viscous hydrodynamics scheme. Radice et al./ found that none of their models produced ejecta over $0.01M_{\odot}$ and their numbers agreed with Lehner et al. (Lehner et al. (2016)). The discrepancy between these results could however be due different choices for the definition of the unbound material in these studies. This paper adds onto previous works in the following ways. First, we look at a different set of parameters not found in the above studies. We use the SFHo, LS220 and DD2 EOS to study the effects of EOS on the merger emissions. We use mass-ratios ranging from $q \sim .76 - 1$ to study the effects of mass asymmetry on emissions. Unlike the previous studies, we do not

fix the total mass and allow it to vary from $\sim 2.5 - 2.9M_{\odot}$. On top of this, we use a new M1 neutrino transport scheme which evolves the number density, allowing for consistent lepton number evolution (Foucart et al. (2016)).

We organize the paper as follows. In Section 3.3, we discuss the numerical implementation we use and the equations we solve, including the gray-M1 scheme for neutrino transport. In the following sections of the paper, we discuss the matter and neutrino emission from a new set of twelve binary neutron star merger simulations, ranging in mass ratio and equation of state. Finally we conclude with ideas for future work. We use a system of units such that $c = G = M_{\odot} = 1$, where c is the speed of light in vacuum, G is the gravitational constant, and M_{\odot} is the mass of the Sun. We use Einstein's convention of summation over repeated indices. Latin indices run over 1, 2, 3, while Greek indices run over 0, 1, 2, 3. The spacetime metric signature is $(-, +, +, +)$.

3.3 Numerical Implementation

3.3.1 General Overview

We evolve Einstein's equations and the general relativistic equations of ideal radiation-hydrodynamics using the Spectral Einstein Code (SpEC)(spe (2019)). SpEC evolves those equations on two separate grids: a pseudospectral grid for Einstein's equations, written in the generalized harmonic formulation (Lindblom et al. (2006)), and a finite volume grid for the general relativistic equations of neutrino-hydrodynamics, written in conservative form. The latter makes use of an approximate Riemann solver (HLL (Harten et al. (1983))) and high-order shock capturing methods (fifth order WENO scheme (Liu et al. (1994); Jiang & Shu (1996))), resulting in a second-order accurate evolution scheme. For the time evolution, we use a third-order Runge-Kutta algorithm. Finally, after each time step, the two grids communicate the required source terms, using a third-order accurate spatial interpolation scheme. Those source terms are the metric and its derivatives (from the pseudospectral grid to the finite volume grid) and the

fluid variables, which are rest-mass density, pressure, spatial velocity, the Lorentz factor and enthalpy. The following sections will give more detail on individual segments of this numerical method.

In the following sections, we make use the 3+1 decomposition of the metric

$$ds^2 = g_{\alpha\beta} dx^\alpha dx^\beta \quad (3.1)$$

$$= -\alpha^2 dt^2 + \gamma_{ij}(dx^i + \beta^i)(dx^j + \beta^j) \quad (3.2)$$

where α is the lapse, β^i the shift, and γ_{ij} the 3-metric on a slice of constant coordinate t . The extension of γ_{ij} to the full 4-dimensional space is the projection operator:

$$\gamma_{\alpha\beta} = g_{\alpha\beta} + n_\alpha n_\beta. \quad (3.3)$$

3.3.2 Initial Data

Initial data for this simulation was produced by an BNS initial data solver based on the work of Foucart *et al.* for BHNS systems (Foucart et al. (2008)), which was built upon the elliptic solver Spells (Pfeiffer et al. (2003b)) and further improved for BNS systems in (Tacik et al. (2015)) and (Haas et al. (2016)). We start by considering systems in quasi-equilibrium, where time derivatives vanish in a corotating frame. We take the metric to be conformally flat and solve for the lapse, shift, and conformal factor using the extended conformal thin sandwich (XCTS) equations (Pfeiffer & York (2003)). The matter in the stars is modeled as a cold perfect fluid with an irrotational velocity profile.

Low eccentricity can be achieved through an iterative procedure requiring the evolution of the system for 2-3 orbits in each iteration (Pfeiffer et al. (2007)). All of the simulations in this paper use that algorithm to achieve estimated eccentricities of approximately $e \sim 0.001$.

Neutrinos are initialized in thermal equilibrium with the fluid. For more details see (Foucart et al. (2016a)).

3.3.3 Spacetime Evolution

The Einstein field equations can be written in the form

$$R_{\alpha\beta} = 8\pi \left(T_{\alpha\beta} - \frac{1}{2} \psi_{\alpha\beta} T \right), \quad (3.4)$$

where $R_{\alpha\beta}$ is the Ricci tensor, $\psi_{\alpha\beta}$ is the metric tensor and $T_{\alpha\beta}$ is the stress energy tensor with trace T . SpEC uses the generalized harmonic decomposition (Lindblom et al. (2006)) to write the Einstein field equations in a form that allows stable numerical computation.

In the generalized harmonic formalism, the evolution of the coordinates follows the wave equation

$$\psi_{ab} \nabla^c \nabla_c x^b = H_a(x^c), \quad (3.5)$$

where ψ_{ab} is the spacetime metric, and $H_a(x^c)$ a set of four arbitrary functions. Using Eq. (3.5) we can rewrite Eq. (3.4) as (Pretorius (2005b))

$$\begin{aligned} \psi^{\delta\gamma} \partial_\gamma \partial_\delta \psi_{\alpha\beta} + \partial_\beta \psi^{\gamma\delta} \partial_\gamma \psi_{\alpha\delta} + \partial_\alpha \psi^{\gamma\delta} \partial_\gamma \psi_{\beta\delta} + 2\partial_{(\beta} H_{\alpha)} \\ - 2H_\delta \Gamma_{\alpha\beta}^\delta + 2\Gamma_{\delta\beta}^\gamma \Gamma_{\gamma\alpha}^\delta = -8\pi(2T_{\alpha\beta} - g_{\alpha\beta} T), \end{aligned} \quad (3.6)$$

where the Christoffel symbols $\Gamma_{\alpha\beta}^\delta$ are defined by

$$\Gamma_{\alpha\beta}^\gamma = \frac{1}{2} \psi^{\gamma\epsilon} (\partial_\beta \psi_{\alpha\epsilon} + \partial_\alpha \psi_{\beta\epsilon} \partial_\epsilon \psi_{\alpha\beta}). \quad (3.7)$$

Eq. (3.6) introduces four independent gauge functions H_α , which need to be chosen. Before the merger of the neutron stars, we set

$$H_\alpha(x_c^i, t) = H_\alpha(x_c^i, 0) \exp\left(-\frac{t^2}{\tau^2}\right), \quad (3.8)$$

with $\tau = \sqrt{d_0^3/M}$, d_0 the initial separation, M is the total mass of the binary at infinite separation and x_c^i are comoving spatial coordinates which follow the rotation and inspiral of the binary. The initial data is constructed in a gauge $H_a^{initial}$ that assumes the time derivatives in the comoving frame are zero. At the beginning of the simulation, we set $H_a(x_c^i, 0) = \hat{H}_a$, where \hat{H}_a is a tensor that agrees with $H_a^{initial}$ in a frame comoving with the grid and is constant in time. The gauge

will thus evolve into the harmonic condition $H_a = 0$. During merger, we find that transitioning to the “damped harmonic” gauge condition is better (Szilágyi (2014)). Defining

$$H_\alpha = \left(\log \frac{\sqrt{\gamma}}{\alpha} \right)^2 \left(\frac{\sqrt{\gamma}}{\alpha} t_\alpha - \gamma_{\alpha i} \frac{\beta^i}{\alpha} \right), \quad (3.9)$$

we transition according to

$$H_\alpha(t) = H_\alpha \left(1 - \exp \frac{-(t - t_{DH})^2}{\tau_m^2} \right), \quad (3.10)$$

with t_{DH} the time at which we turn on the damped harmonic gauge and $\tau_m = 100M$.

With these gauge choices we solve a first order representation of the generalized harmonic system (Eq. 3.6) in which the fundamental variables are the spacetime metric ψ_{ab} , its spatial first derivatives ϕ_{iab} and its first derivatives in the direction normal to the $t =$ constant slice Π_{ab} . The generalized harmonic first order system is then:

$$\partial_t \psi_{ab} - (1 + \gamma_1) N^k \partial_k \psi_{ab} = -N \Pi_{ab} - \gamma_1 N^i \Phi_{iab}, \quad (3.11)$$

$$\partial_t \Pi_{ab} - N^k \partial_k \Pi_{ab} + N g^{ki} \partial_k \Phi_{iab} - \gamma_1 \gamma_2 N^k \partial_k \psi_{ab} \quad (3.12)$$

$$\begin{aligned} &= 2N\psi^{cd}(g^{ij}\Phi_{ica}\Phi_{jdb} - \Pi_{ca}\Pi_{db} - \psi^{ef}\Gamma_{ace}\Gamma_{bdf}) \\ &\quad - 2N\nabla_{(a}H_{b)} - \frac{1}{2}Nt^c t^d \Pi_{cd} \Pi_{ab} - Nt^c \Pi_{ci} g^{ij} \Phi_{jab} \\ &\quad + N\gamma_0[2\delta_{(a}^c t_{b)} - \psi_{ab} t^c](H_c + \Gamma_c) - \gamma_1 \gamma_2 N^i \Phi_{iab} \\ &\quad - 2\alpha(T_{ab} - \frac{1}{2}\psi_{ab} T^{cd} \psi_{cd}), \end{aligned}$$

$$\begin{aligned} \partial_t \phi_{iab} - N^k \partial_k \phi_{iab} + N \partial_i \Pi_{ab} - \gamma_2 N \partial_i \psi_{ab} \quad (3.13) \\ = \frac{1}{2}Nt^c t^d \Phi_{icd} \Pi_{ab} + N g^{jk} t^c \Phi_{ijc} \Phi_{kab} - N \gamma_2 \Phi_{iab}, \end{aligned}$$

where

$$\partial_t \psi_{ab} := -N \Pi_{ab} + N^i \phi_{iab}, \quad (3.14)$$

$$\partial_i \psi_{ab} := \phi_{iab}. \quad (3.15)$$

This amounts to a symmetric hyperbolic system of 50 coupled nonlinear equations. The constraint damping parameters $\{\gamma_0, \gamma_1, \gamma_2\}$ are additional free parameters which dampen any constraint violating modes which may grow due to small numerical errors in the evolution. To set the γ parameters we use a trial-and-error process which concludes when we find that the constraint-violating modes do not grow significantly over time. In practice, we set the parameters to

$$\gamma_1 = 0.999(f(r_c, 10d) - 1), \quad (3.16)$$

$$\begin{aligned} \gamma_2 = & \frac{.005}{M} + \frac{3}{M_1} f(r_1^i, 2.5R_1) \\ & + \frac{3}{M_2} f(r_2^i, 2.5R_2) + \frac{.075}{M_2} f(r_c^i, 2.5d), \end{aligned} \quad (3.17)$$

$$\begin{aligned} \gamma_2 = & \frac{.005}{M} + \frac{.2}{M_1} f(r_1^i, 2.5R_1) \\ & + \frac{.2}{M_2} f(r_2^i, 2.5R_2) + \frac{.075}{M_2} f(r_c^i, 2.5d), \end{aligned} \quad (3.18)$$

$$f(r^i, w) = \exp(-|r^i|^2/w^2), \quad (3.19)$$

where $r_{1,2,c}^i$ correspond to the coordinate locations of the first neutron star, second neutron star and center of mass respectively, R_1/M_1 is the radius and mass of the 1st neutron star, R_2/M_2 is the radius and mass of the 2nd neutron star, $M = M_1 + M_2$ and d is the separation of the two neutron star centers.

3.3.4 Neutrino Evolution

We use a gray two moment M1 scheme for neutrino transport introduced in (Foucart et al. (2016a)). To recap, the new scheme has the advantages of exactly conserving the total lepton number, and taking into account spatial variations in the neutrino energy. In this section we will give an overview of the scheme we use.

For each neutrino ν_i we can describe the neutrinos by their distribution function $f_\nu(x^\mu, p^\mu)$ where $x^\mu = (t, x^i)$ gives the time and the position of the neutrinos and p^α is the 4-momentum of

the neutrinos. The distribution function $f_{(\nu)}$ evolves according to the Boltzmann equation:

$$p^\alpha \left[\frac{\partial f_{(\nu)}}{\partial x^\alpha} - \Gamma_{\alpha\gamma}^\beta \frac{\partial f_{(\nu)}}{\partial p^\beta} \right] = \left[\frac{df_{(\nu)}}{d\tau} \right]. \quad (3.20)$$

In general, this is a 7-dimensional problem which is extremely expensive to solve numerically. Approximations to the Boltzmann equation have thus been developed for numerical applications. In this work, we consider the moment formalism developed by Thorne (Thorne (1981)) and Shibata et al. (Shibata et al. (2011)), in which only the lowest moments of the distribution function in momentum space are evolved. We limit ourselves to the use of this formalism in the gray approximation, that is we only consider energy-integrated moments. We will consider three independent neutrino species: the electron neutrino ν_e , the electron antineutrinos $\bar{\nu}_e$, and the heavy lepton neutrinos ν_X . The latter represents the 4 species $(\nu_\mu, \bar{\nu}_\mu, \nu_\tau, \bar{\nu}_\tau)$. This merging is justified because the temperatures and neutrino energies reached in our merger calculations are low enough to suppress the formation of the corresponding heavy leptons. The presence of heavy leptons would then require including the charged current neutrino interactions that differentiate between these individual species.

In the gray approximation with the first two moments of the distribution function, we evolve for each species projections of the stress-energy tensor of the neutrino radiation $T_{rad}^{\mu\nu}$. We write

$$T_{rad}^{\mu\nu} = Ju^\mu u^\nu + H^\mu u^\nu + H^\nu u^\mu + S^{\mu\nu}, \quad (3.21)$$

with $H^\mu u_\mu = S^{\mu\nu} u_\mu = 0$ and u^μ the 4-velocity of the fluid. We can decompose the momentum as follows

$$p^\alpha = \nu(u^\alpha + l^\alpha), \quad (3.22)$$

with $l^\alpha u_\alpha = 0$ and $l^\alpha l_\alpha = 1$. With this decomposition, the energy J , flux H^μ and stress tensor $S^{\mu\nu}$ of the neutrino radiation as observed by an observer comoving with the fluid are related to the neutrino distribution function by

$$J = \int_0^\infty d\nu \nu^3 \int d\Omega f_{(\nu)}(x^\alpha, \nu, \Omega) \quad (3.23)$$

$$H^\mu = \int_0^\infty d\nu \nu^3 \int d\Omega f_{(\nu)}(x^\alpha, \nu, \Omega) l^\mu \quad (3.24)$$

$$S^{\mu\nu} = \int_0^\infty d\nu \nu^3 \int d\Omega f_{(\nu)}(x^\alpha, \nu, \Omega) l^\mu l^\nu, \quad (3.25)$$

where ν is the neutrino energy in the fluid frame, $\int d\Omega$ denotes integrals over solid angle on a unit sphere in momentum space, and We also utilize the decomposition of $T_{rad}^{\mu\nu}$ in terms of the energy, flux and stress tensor observed by an inertial observer,

$$T_{rad}^{\mu\nu} = E n^\mu n^\nu + F^\mu n^\nu + F^\nu n^\mu + P^{\mu\nu}, \quad (3.26)$$

with $F^\mu n_\nu = P^{\mu\nu} n_\mu = F^t = P^{t\nu} = 0$, and n^α the unit normal to a $t = \text{constant}$ slice.

We define a projection operator on the reference frame of an observer comoving with the fluid:

$$h_{\alpha\beta} = g_{\alpha\beta} + u_\alpha u_\beta. \quad (3.27)$$

We can then write equations relating the fluid frame variables to the inertial frame variables

$$E = W^2 J + 2W v_\mu H^\mu + v_\mu v_\nu S^{\mu\nu}, \quad (3.28)$$

$$F_\mu = W^2 v_\mu J + W(g_{\mu\nu} - n_\mu v_\nu) H^\nu \quad (3.29)$$

$$\begin{aligned} &+ W v_\mu v_\nu H^\nu + (g_{\mu\nu} - n_\mu v_\nu) H^\nu + W v_\mu v_\nu H^\nu \\ &+ (g_{\mu\nu} + n_\mu v_\nu) v_\rho S^{\nu\rho}, \end{aligned}$$

$$\begin{aligned} P_{\mu\nu} = & W^2 v_\mu v_\nu J + W(g_{\mu\nu} - n_\mu v_\rho) v_\nu H^\rho \\ &+ (g_{\mu\rho} - n_\mu v_\rho)(g_{\nu\kappa} - n_\nu v_\kappa) S^{\rho\kappa} \\ &+ W(g_{\rho\nu} - n_\rho v_\nu) v_\mu H^\rho \end{aligned} \quad (3.30)$$

using the decomposition of the 4-velocity

$$u^\mu = W(n^\mu + v^\mu), \quad (3.31)$$

where $v^\mu n_\mu = 0$ and $W = \sqrt{1 + v_i v^i}$.

By taking moments of the Boltzmann equation, the evolution equations for $\tilde{E} = \sqrt{\gamma}E$ and $\tilde{F} = \sqrt{\gamma}F^i$ can then be written in conservative form

$$\partial_t \tilde{E} + \partial_j(\alpha \tilde{F}^j - \beta^j \tilde{E}) \quad (3.32)$$

$$= \alpha(\tilde{P}^{ij} K_{ij} - \tilde{F}^j \partial_j \ln \alpha - \tilde{S}_{\text{rad}}^\alpha n_\alpha),$$

$$\partial_t \tilde{F}_i + \partial_j(\alpha \tilde{P}_i^j - \beta^j \tilde{F}_i) \quad (3.33)$$

$$= (-\tilde{E} \partial_i \alpha + \tilde{F}_k \partial_i \beta^k + \frac{\alpha}{2} \tilde{P}^{jk} \partial_i \gamma_{jk} + \alpha \tilde{S}_{\text{rad}}^\alpha \gamma_{i\alpha}),$$

where γ is the determinant of γ_{ij} , $\tilde{P}_{ij} = \sqrt{\gamma} P_{ij}$, and $\tilde{S}_{\text{rad}}^\alpha = \sqrt{\gamma} S_{\text{rad}}^\alpha$ includes all collisional source terms. Additionally, we consider the number current density for each species of neutrino:

$$N^\mu = N n^\mu + \mathcal{F}^\mu, \quad (3.34)$$

where N is the number density of neutrinos, and \mathcal{F}^μ the number flux density. We can define the average neutrino energy in the fluid frame by

$$N^\mu = \frac{Ju^\mu + H^\mu}{\langle \nu \rangle}, \quad (3.35)$$

which then gives us an estimate for the energy

$$\langle \nu \rangle = W \frac{(E - F_i v^i)}{N}, \quad (3.36)$$

where W is the Lorentz factor introduced in Eq. 3.31. The evolution equation for $\tilde{N} := \sqrt{\gamma}N$ is

$$\partial_t \tilde{N} + \partial_j(\alpha \sqrt{\gamma} \mathcal{F}^j - \beta^j \tilde{N}) = \alpha \sqrt{\gamma} C_{(0)}. \quad (3.37)$$

To close this system of equations we need three additional ingredients: a prescription for the computation of $P^{ij}(E, F_i)$ which is called the closure relation, a prescription for the computation of the number flux \mathcal{F}^j (specific to the evolution of the number density N in this paper) and the collisional source terms $\tilde{S}^\alpha, C_{(0)}$.

For $P^{ij}(E, F_i)$ we use the M1 approach and interpolate between optically thick and thin limits:

$$P^{ij} = \frac{3p-1}{2} P_{\text{thin}}^{ij} + \frac{3(1-p)}{2} P_{\text{thick}}^{ij}. \quad (3.38)$$

Here the parameter p is known as the variable Eddington factor and the functional form of p in terms of the lower moments H, J is known as the M1 closure. Our choices for $P_{\text{thin}}^{ij}, P_{\text{thick}}^{ij}$ and p are discussed in the Appendix of (Foucart et al. (2015)).

For \mathcal{F}^i , by definition we have

$$\mathcal{F}^i = \frac{JWv^i}{\langle v \rangle} + \frac{\gamma_\mu^i H^\mu}{\langle v^F \rangle} \quad (3.39)$$

where the flux-weighted average neutrino energy, $\langle v^F \rangle$ is computed in such a way to take the effects of a finite optical depth on the spectrum into account, see (Foucart et al. (2016b)) for details.

For the source terms \tilde{S}^α , we assume the fluid has an energy-integrated emissivity $\bar{\eta}$ due to the charged-current reactions



as well as electron-positron pair annihilation



plasmon decay



and nucleon-nucleon Bremsstrahlung



The inverse reactions are responsible for an energy-averaged absorption opacity $\bar{\kappa}_a$. We also consider an energy-averaged scattering opacity $\bar{\kappa}_s$ due to elastic scattering of neutrinos on

nucleons and heavy nuclei. Neglecting other reactions (e.g. inelastic scatterings and $\nu\bar{\nu}$ annihilation) the source terms S^α can then be shown to be (Shibata et al. (2011))

$$\tilde{S}_{\text{rad}}^\alpha = \sqrt{\gamma}[\bar{\eta}u^\alpha - \bar{\kappa}_\alpha Ju^\alpha - (\bar{\kappa}_a + \bar{\kappa}_s)H^\alpha]. \quad (3.45)$$

The collisional source term for the number density \bar{N} is given by

$$C_{(0)} = \bar{\eta}_N - \bar{\kappa}_N \frac{J}{\langle \nu \rangle} = \bar{\eta}_N - \frac{\bar{\kappa}_N J \bar{N}}{W(\bar{E} - \bar{F}_i v^i)}. \quad (3.46)$$

Thus we need the energy integrated emissivities $(\bar{\eta}, \bar{\eta}_N)$ and the energy-averaged opacities $(\bar{\kappa}_A, \bar{\kappa}_S, \bar{\kappa}_N)$ to compute the source terms. To do this we need the equilibrium energy-integrated emissivities $(\bar{\eta}^{eq}, \bar{\eta}_N^{eq})$ and equilibrium opacities $(\bar{\kappa}_A^{eq}, \bar{\kappa}_S^{eq})$ which assume a Fermi-Dirac distribution with temperature T_{fluid} . When they are available, we use the equilibrium emissivities and opacities from Ruffert et al. (Ruffert et al. (1996)), except for nucleon-nucleon Bremsstrahlung for which the emissivity is computed following Burrows et al. (Burrows et al. (2006)). When either an equilibrium emissivity or opacity is not available for a reaction, we compute whichever is unknown using Kirchoff's Law:

$$\bar{\eta}^{eq} = \bar{\kappa}^{eq} \int B_{(\nu)} d\nu, \quad (3.47)$$

where $B_{(\nu)}$ is the blackbody spectrum in equilibrium with the fluid. See (Foucart et al. (2016a)) for more details. After computing the equilibrium quantities, we make the choices

$$\bar{\eta} = \bar{\eta}^{eq}, \quad (3.48)$$

$$\bar{\eta}_N = \bar{\eta}_N^{eq}, \quad (3.49)$$

$$\bar{\kappa}_A = \bar{\kappa}_A^{eq} \frac{T_\nu^2}{T_{\text{fluid}}^2}, \quad (3.50)$$

$$\bar{\kappa}_S = \bar{\kappa}_S^{eq} \frac{T_\nu^2}{T_{\text{fluid}}^2}, \quad (3.51)$$

where T_ν is the neutrino temperature computed from the neutrino energy and number density, assuming a blackbody spectrum (see (Foucart et al. (2016a)) for more details). Lastly we set

$$\bar{\kappa}_N = \bar{\kappa}_A \frac{\bar{\eta}_N}{\bar{\eta}} \frac{F_3(\eta_\nu) T_{\text{fluid}}}{F_2(\eta_\nu)}, \quad (3.52)$$

where $\eta_\nu = \mu_\nu/T$ with μ_ν the chemical potential of neutrinos in equilibrium with the fluid and $F_k(\eta_\nu)$ the Fermi integral:

$$F_k(\eta_\nu) = \int_0^\infty \frac{x^k}{1 + \exp(x - \eta_\nu)} dx. \quad (3.53)$$

3.3.5 Fluid Evolution

The neutron stars are described by ideal fluids with stress tensor

$$T_{\mu\nu} = \rho_0 h u_\mu u_\nu + P g_{\mu\nu}, \quad (3.54)$$

where ρ_0 is the rest mass density, h the specific enthalpy, P the pressure and u^μ the 4-velocity. The general relativistic equations of hydrodynamics are evolved in conservative form, using the conservative variables

$$\rho_* = -\sqrt{\gamma} n_\mu u^\mu \rho_0, \quad (3.55)$$

$$\tau = \sqrt{\gamma} n_\mu n_\nu T^{\mu\nu} - \rho_*, \quad (3.56)$$

$$S_k = -\sqrt{\gamma} n_\mu T_k^\mu. \quad (3.57)$$

The $3 + 1$ stress-energy conservation equations for radiation-hydrodynamics $\nabla_\nu T^{\mu\nu} = -S_{\text{rad}}^\mu$ and the baryon conservation equations $\nabla_\mu(\rho_0 u^\mu) = 0$ become (Shibata et al. (2011))

$$\partial_t \rho_* + \partial_j(\rho_* v^j) = 0, \quad (3.58)$$

$$\partial_t \tau + \partial_i(\alpha^2 \sqrt{\gamma} T^{0i} - \rho_* v^i) = -\alpha \sqrt{\gamma} T^{\mu\nu} \nabla_\nu n_\mu \quad (3.59)$$

$$\begin{aligned} &+ \alpha \tilde{S}_{\text{rad}}^\alpha n_\alpha, \\ \partial_t S_i + \partial_j(\alpha \sqrt{\gamma} T_i^j) &= \frac{1}{2} \alpha \sqrt{\gamma} T^{\mu\nu} \partial_i \gamma_{\mu\nu} \\ &- \alpha \tilde{S}_{\text{rad}}^\alpha \gamma_{i\alpha}. \end{aligned} \quad (3.60)$$

3.3.6 Composition Evolution

The fluid composition in our simulations is described by the electron fraction:

$$Y_e = \frac{n_p}{n_p + n_n}, \quad (3.61)$$

where n_p and n_n are the proton and neutron number densities, respectively (the net electron number density $n_{e^-} - n_{e^+} = n_p$, due to charge neutrality in the fluid). From lepton number conservation, we have:

$$\partial_t (\rho_* Y_e) + \partial_i \left(\rho_* Y_e v^i \right) = - \sum_{\nu} \text{sign}(\nu) \alpha \sqrt{\gamma} C_{(0)}^{\nu}, \quad (3.62)$$

where \sum_{ν} sums over neutrino species with $\text{sign}(\nu)$ set to 1 for ν_e , -1 for $\bar{\nu}_e$, and 0 for ν_x . Importantly for the electron fraction, evolving the neutrino number density N frees us from having to guess at the average neutrino energy when computing the coupling to the fluid. It also guarantees that the source term for the evolution of the electron fraction of the fluid is fully consistent with the evolution of the neutrino number density, thus conserving the total lepton number of the system (Foucart et al. (2016a)).

3.3.7 Equation of State (EOS)

We use three finite-temperature, composition dependent nuclear-theory based equations of state. Two of them are based on relativistic mean field models (Walecka (1974)) and one based on the single nucleus approximation for heavy nuclei (Lattimer & Swesty (1991)). They are:

1. DD2 (Hempel et al. (2012)): This EOS is based on nuclear statistical equilibrium with a finite volume correction coupled to a relativistic mean field theory for treating high-density nuclear matter. DD2 contains neutrons, protons, light nuclei such as deuterons, helions, tritons and alpha particles and heavy nuclei. DD2 does not satisfy the so-called flow-constraint (Hempel et al. (2017)).
2. LS220 (Lattimer & Swesty (1991)): is based on the single nucleus approximation for heavy nuclei where the thermal distribution of different nuclear species is replaced by a single representative heavy nucleus. LS220 contains neutrons, protons, alpha particles

and heavy nuclei. LS220 does not satisfy the constraints from Chiral effective field theory (Hempel et al. (2017)).

3. SFHo (Steiner et al. (2013)): This EOS, like DD2, also uses an RMF model, containing neutrons, protons, light nuclei such as deuterons, helions, tritons and alpha particles and heavy nuclei. However, SFHo uses a different RMF parameterization, specifically designed to match neutron star properties as inferred by observations. SFHo shows some minor deviations from Chiral effective field theory calculations (Hempel et al. (2017)).

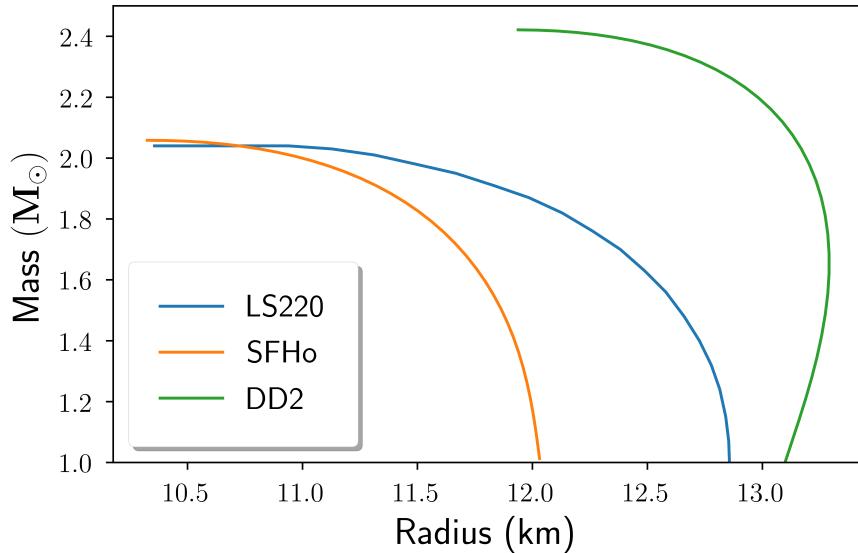


Figure 3.1: M-R curves for each of the equations of states used in this work.

While each EOS shows deviations from theoretical calculations, they all have a radius, maximum mass and tidal deformability that are compatible with observations (Demorest et al. (2010); Hempel et al. (2017)). For all three equations of state, we show mass-radius curves in Figure 3.1. These curves were computed by integrating the TOV equations for the EOS by using the table values at $T = 0.1$ MeV in beta equilibrium. From Figure 3.1, we can clearly see that for each EOS there is a maximum mass for the non-rotating isolated neutron star. Furthermore we can associate with each EOS an average density at this maximum mass as $\langle \rho \rangle := 3M_{max}/4\pi R_{max}^3$ and the ratio of this average density and the central density ρ_c of the star is an indication of

how “stiff” the EOS is. Higher values of $\langle \rho \rangle / \rho_c$ correspond to stiffer EOS, while lower values correspond to softer EOS (Bauswein et al. (2013)). In our case, from stiffest to softest, the EOS are ranked DD2, LS220 and SFHo. Softer EOS tend to make smaller stars at a fixed mass. For a fiducial mass of $1.4M_\odot$, the radii are 13.2 km, 12.7 km, 11.9 km for DD2, LS220 and SFHo respectively. We can also introduce a useful quantity called the compactness, which is defined by $C = M/R$. Softer EOS tend to make more compact stars. The maximum mass however is not a good indicator of whether the post-merger remnant will promptly collapse to a black hole (BH) after merger because the remnant will most likely be differentially rotating and held up by centrifugal and thermal forces against collapse. A more accurate estimate for the maximum mass at which the merger promptly collapses to a BH is given by Bauswein *et al.* (Bauswein et al. (2013)) who ran simulations of BNS mergers with a smoothed particle hydrodynamics code that employs the conformal flatness approximation of the Einstein field equations and includes a GW backreaction scheme to determine a threshold mass for prompt collapse across 12 different realistic EOS. Bauswein et al./ found the threshold masses for SFHo, LS220 and DD2 were $2.95M_\odot$, $3.05M_\odot$ and $3.35M_\odot$ respectively. Thus the SFHo and LS220 models are more likely to undergo prompt collapse for the higher end of the total masses considered in this paper ($\sim 2.9M_\odot$, see Sec. 3.3.8).

3.3.8 Initial Models

We extend our previous work (Foucart et al. (2016b)) and study the merger of unequal mass neutron star binaries with the neutrino M1 transport scheme introduced in (Foucart et al. (2016a)) and discussed above. We study mass ratios between .76 and 1 and masses between $1.2M_\odot$ and $1.56M_\odot$ which are both within the ranges of current binary neutron star observations (see e.g. (Lattimer (2012))). We focus here on the late stages of the coalescence, comprising the last 3 – 5 orbits (depending on EoS) before merger and we stop evolving any systems once they collapse. The parameters for the initial models and the type of post-merger remnants they create (before or as of 7.5-ms post-merger) are shown in Table 3.1. The specific grid setup for each model will

be discussed in the following section.

3.3.9 Grid Setup

Before the two neutron stars enter into contact, the pseudospectral grid on which we evolve Einstein’s equations takes advantage of the approximate spherical symmetry of the neighborhood of each star, and in the far-field region. The evolved spatial slice is decomposed into two small balls around the center of each neutron star, sets of spherical shells around each star, and distorted cubes to connect the three spherically symmetric regions. The inner ball is expanded into Zernike polynomials, the shells into Chebyshev polynomials (in radius) and spherical harmonics (in angle) and the distorted cubes in Chebyshev polynomials. The grid follows the centers of the neutron stars defined as the center of mass of the matter in the $x < 0$ and $x > 0$ half planes, through a simple rotation and scaling of the grid coordinates. We do not exploit the reflection symmetry $z \rightarrow -z$ in our runs.

We maintain this grid decomposition for the evolution of Einstein’s equations up to the point at which the maximum density on the grid increases beyond the low-level oscillations observed during the inspiral. This rise in the density signifies the transition from two well-separated neutron star cores to a single, more massive object. At that point we switch to a grid which is fully centered on the coordinate center of mass of the system. This grid is made of a ball at the origin of the coordinate system, surrounded by 59 spherical shells extending to the outer edge of the computational domain. Both before and after merger, the outer boundary is located at $40d_0$, with d_0 the initial separation of the binary, provided in Table 3.1.

The finite volume grid on which we evolve the general relativistic equations of hydrodynamics is very simple. Before the two neutron stars get into contact, it is composed of two cubes, each centered on a neutron star and composed of 96^3 cells. In the coordinate system comoving with the neutron star centers, the neutron stars expand as the binary inspirals. To avoid losing matter to the outer boundary of the finite volume grid, we expand the grid by 4.5% every time the flux of matter across the outer boundary exceeds $0.015M_\odot s^{-1}$. As the inspiral lasts less than 10 ms,

Model	EOS	M_1	M_2	q	C_1	C_2	$d_0(km)$	Collapse?
D144144	DD2	1.44	1.44	1.0	.161	.161	48.7	No
D12132	DD2	1.2	1.32	.91	.134	.147	48.7	No
D12144	DD2	1.2	1.44	.83	.134	.161	48.7	No
D12156	DD2	1.2	1.56	.77	.134	.173	48.7	No
L144144	LS220	1.44	1.44	1.0	.175	.175	44.3	(~ 2ms)
L12132	LS220	1.2	1.32	.91	.146	.161	44.3	No
L12144	LS220	1.2	1.44	.83	.146	.175	44.1	No
L12156	LS220	1.2	1.56	.77	.146	.191	44.3	(~ 4.5ms)
S144144	SFHO	1.44	1.44	1.0	.179	.179	44.3	(~ .5ms)
S12132	SFHO	1.2	1.32	.91	.148	.163	44.3	No
S12144	SFHO	1.2	1.44	.83	.148	.179	44.3	No
S12156	SFHO	1.2	1.56	.77	.148	.195	44.3	(~ 1.4ms)

Table 3.1: Parameters for the initial models presented in this paper. M_1 and M_2 are the gravitational masses of the two neutron stars, C_1 and C_2 are the compactness of the respective neutron stars and d_0 is the initial coordinate distance between the centers of stars. The last column gives the post-merger remnant as of 7.5-ms post-merger. If the binary collapsed before 7.5-ms a rough estimate for the time of collapse is shown in brackets.

Name	dx_{ins} (m)	dx_{mer} (m)
D12132	279	300
D12144	280	300
D12156	279	300
D144144	273	300
<hr/>		
S12132	251	300
S12144	250	300
S12156	238	300
S144144	238	300
<hr/>		
L12132	252	300
L12144	253	300
L12156	252	300
L144144	245	300

Table 3.2: Finite difference grid sizes for the initial models. All simulations were set so that they would have a 300m resolution during merger. For reference, the DD2 $1.44M_{\odot}$ star has a $\sim 10.48\text{km}$ radius in our grid coordinates.

this implies a mass loss well below $10^{-4}M_{\odot}$ before merger. As the two neutron stars approach each other, the two finite volume boxes will eventually intersect. During merger, we would like to follow the forming massive neutron star remnant, the tidal tails, the accretion disk, and any ejected material. We switch to a finite difference grid centered on the forming remnant with 3 levels of refinement. Each level has, at our standard resolution $200^2 \times 100$ cells, with the finest grid spacing listed in Table 3.2 and each coarser level increasing the grid spacing by a factor of 2. The lower number of cells in the vertical direction reflects the fact that the remnant is less extended in that direction, and thus that we do not need the finest grid to extend as far vertically as horizontally.

3.3.10 Ejecta Analysis

In a BNS merger, matter expelled at high velocity may ultimately become unbound from the central gravitational potential. Two indicators have been used to label matter unbound:

1. **$u_t < -1$** : For a stationary spacetime, u_t (the projection of the 4-velocity along the timelike Killing vector field) is a constant of motion for geodesics. Assuming the space is also asymptotically flat, the Lorentz factor W satisfies $W = -u_t$ at infinity. Therefore we may flag matter as unbound using the condition $u_t < -1$.
2. **$hu_t < -1$** : For a stationary relativistic fluid flow, the relativistic Bernoulli equation (Rezzolla & Zanotti (2013)) implies hu_t is constant along fluid worldlines. In an asymptotically flat spacetime, we would expect $W = -u_t$ (if the fluid particles follow geodesics). Since the relativistic enthalpy h is only defined up to a constant factor which can be set such that $h \leftarrow 1$ at spatial infinity. Therefore, we may flag matter as unbound using the condition $hu_t < -1$.

We have found previously (Foucart et al. (2016b)) that the second indicator $hu_t < -1$, the Bernoulli criterion, produces qualitatively more accurate results in SpEC, therefore all material labelled unbound in this paper uses this indicator. It is important to note that no indicator is exact. In fact, the Bernoulli criterion has been shown to result in as much as twice the ejected matter as the u_t condition (Kastaun & Galeazzi (2015)).

With the Bernoulli criterion, we flag matter that is still on the grid if it is at least $50M_\odot$ away from the center of the remnant and compute the ejected mass as follows

$$M_{ej}^{on}(t) = \int_{r>50M_\odot} \rho_0 W \mathcal{H}(-hu_t - 1) \sqrt{\gamma} d^3x, \quad (3.63)$$

where $\mathcal{H}(\cdot)$ is the Heaviside function. While the $50M_\odot$ threshold is arbitrary, we have found that it comfortably excludes any ejecta very close to the remnant which may not ultimately become unbound.

We also measure unbound material that leaves the computational grid (for each run, the grid is roughly $200M_{\odot} \times 200M_{\odot} \times 100M_{\odot}$) using

$$M_{\text{ej}}^{\text{off}}(t) = \int_0^t \int_S \rho_0 \mathcal{H}(-hu_t - 1) W(\alpha v^i - \beta^i) n_i dS dt'. \quad (3.64)$$

Here v^i is the fluid 3-velocity, W is the Lorentz factor, defined by $W := (1 - v^i v_i)^{-1/2}$, S is the grid boundary and $\rho_0 \mathcal{H}(-hu_t - 1) W(\alpha v^i - \beta^i) n_i$ represents the flux of unbound material leaving the boundary. At a time t , we estimate the full ejecta as

$$M_{\text{ej}}(t) = M_{\text{ej}}^{\text{on}} + M_{\text{ej}}^{\text{off}} \quad (3.65)$$

Finally, to compute average quantities for the ejected matter, we use the following definition of the mass-weighted average for a quantity X .

$$\langle X \rangle = \frac{1}{M_{\text{ej}}} \int X dM_{\text{ej}}, \quad (3.66)$$

3.3.11 Errors

Since we performed simulations only at one resolution, it is difficult to derive error estimates. However, Hotokezaka et al. (Hotokezaka et al. (2013)) find a $\sim 10\%$ relative error for their mass ejecta properties for unequal mass binaries and we have independently confirmed their results with our code using the same piecewise polytropes EOS they use at similar resolution. However in this work, we use tabulated EOS and have a slightly coarser resolution. Furthermore, in equal mass runs (see (Foucart et al. (2016b))) we have found that relative errors can be as high as $\sim 50\%$. Hotokezaka et al. (Hotokezaka et al. (2013)) find similar variations in the mass ejected by equal mass binaries, with some systems showing little sign of convergence even at higher resolution. We therefore conclude that our errors in the ejected mass may be as high as $\sim 50\%$ and set our error as

$$\Delta M_{\text{ej}} = .5M_{\text{ej}} + 10^{-4}M_{\odot}, \quad (3.67)$$

with the lower bound $10^{-4} M_{\odot}$ coming from the fact that we ignore outflows of this size during the regridding in the inspiral stage. Practically speaking, this is likely to be a significant overestimate of the error for the average simulation in our dataset and is more representative of the error for the worst simulations presented here. Our numerical setup also ignores magnetic fields. Over the short time scales considered here, magnetohydrodynamics (MHD) effects are not expected to affect the evolution of neutron star remnant, but could drive additional outflows from the disk (Kiuchi et al. (2014); Neilsen et al. (2014)). Over longer time scales, magnetic fields would be critical to the spin evolution of the remnant neutron star, angular momentum transport, heating in the disk, and possibly the formation of relativistic jets and magnetically-driven outflows. General relativistic MHD simulations of postmerger disks show that up to $\sim 40\%$ of the accretion disk (around $0.013 M_{\odot}$) can be ejected over 9 seconds (Fernandez et al. (2019)). We attempt to estimate the difference magnetic fields and longterm neutrino-winds would make on the ejecta estimate in Section 3.4.2.

3.4 Numerical Results

3.4.1 General Overview

Prior to merger, which is defined as the peak of the gravitational-wave amplitude, the compact objects inspiral around each other for 3-5 orbits, with the actual number varying between EOS. After around 3 milliseconds (ms), the cores start to merge. The properties of the merged system are then largely determined by the compactness and the mass-ratio of the pre-merger neutron stars.

With the SFHo equation of state, i.e. for the most compact neutron stars, a compact core forms rapidly. In the higher total mass models, the SFHo star collapses promptly to a BH, within a few ms. For the other equations of state (LS220, DD2), more strongly developed tidal features appear at merger. Figure ?? shows each of the $1.2 M_{\odot} + 1.44 M_{\odot}$ models across the three EOS at 3 ms post-merger.

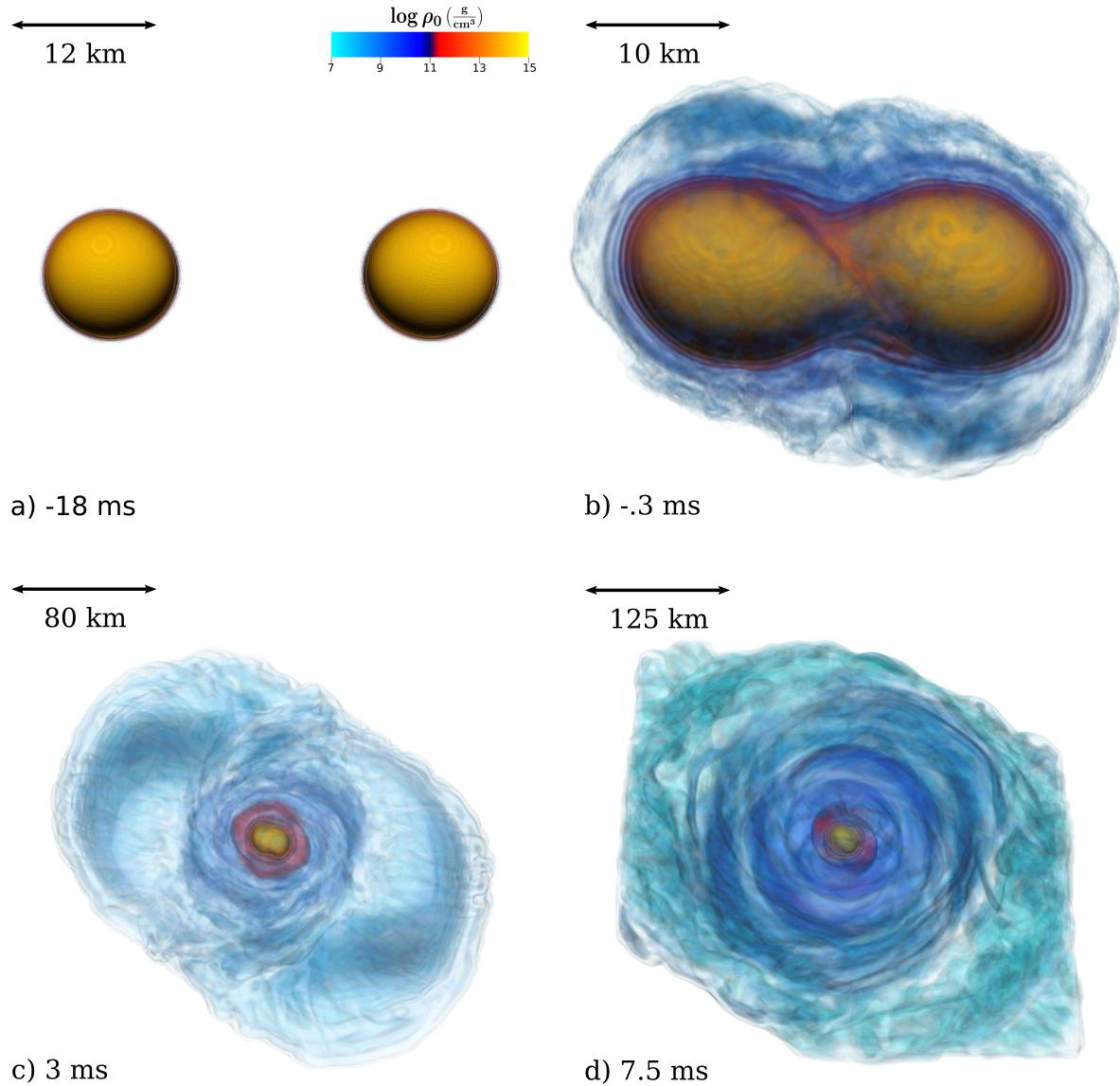


Figure 3.2: The evolution of rest mass density ρ_0 for the D144144 model. For the first ~ 18 ms the stars orbit each other before merging at ~ 0 ms. At around ~ 3 ms we can see a post-merger remnant almost fully formed with two tidal tails. At ~ 7.5 ms the post-merger remnant has largely settled and most of the unbound material has left the grid.

Finally, at around 7.5 ms after merger, the tidal tail has left the computational grid and the remnant is fully formed with an accretion torus. Figure ?? shows a snapshot of the density profile, temperature profile and electron fraction profile of the binaries at 7.5 ms for the mass-ratio

$q = 1.2$ initial models. As we can see the SFHo EOS has a more dense and hot core than the other two EOS. This is expected as the core bounce for SFHo is much more violent and it occurs deeper in the gravitational potential of the system due to the softness of the EOS. The outflow from the SFHO remnant is also more symmetric because outflows powered by core bounce tend to have a more quasi-spherical morphology than outflows from tidal tails. Stiff EOSs like DD2 undergo less shock heating, but tend to have larger tidal tails. As cold tidal tails are typically neutron-rich, the ejecta in the $z = 0$ plane also has a lower Y_e . The electron fraction is also noticeably higher in the disk region for SFHo. This could be due to the fact the SFHo remnant is much hotter and therefore neutrino irradiation effects cause a higher Y_e . This will have significant consequences for long-term evolution and the resulting kilonova signal, which tends to the blue side of the optical spectrum for $Y_e \geq 0.25$.

In the following two sections we will look at the matter and neutrino emission from the merger and merger remnant across EOS and mass-ratio. For both the matter and neutrino emission, two mechanisms are vital to explain our results. The first mechanism is tidal forces, which rip off matter as the binary gets closer, ejecting matter at angles close to the orbital plane. The second mechanism is the contraction and recoil of the cores, which produces shock waves that eject matter quasi-spherically. From these two mechanisms, we can make predictions on expected trends. For example, as the mass-ratio increases, the less massive star gets more tidally disrupted and thus on merger, the effect of the shock on ejection of matter is lessened, but the effect of tidal torque propelling ejecta off the grid is increased. Similarly, as the EOS softens, the compactness of the star increases and the effect of core-bounces on the ejection of matter increases. Therefore in general we might expect that changing the mass-ratio might have a small effect on the emission of SFHo and LS220, but a larger effect on the ejecta of DD2 (see e.g (Sekiguchi et al. (2016a))). However, since the total mass was not fixed in our study, these trends may be more complicated and we will discuss this in the following sections.

3.4.2 Matter outflow

General properties

We have compiled a table of the general properties of the ejecta up to either collapse or 7.5ms post-merger in Table 3.3. To help analyze the ejecta properties, we distinguish between two different regions in our grid: polar and equatorial. Polar ejecta is defined to be any matter within an angle $\theta < 30^\circ$ with respect to the z-axis and equatorial ejecta is defined as rest of the ejected matter. From Table 3.3 we first notice that for each EOS, the majority of the ejecta is equatorial, this is not surprising because tidal forces tend to rip off matter close to the orbital plane. Furthermore, the main contribution to dynamical polar ejecta is due to the shock from the core-bounce, which tends to eject matter in a quasi-spherical distribution. Since the equatorial segment of the quasi-spherical shocked ejecta will by definition come from a larger overall solid angle it will contain a higher percentage of this matter. Secondly, Table 3.3 shows that the LS220 and SFHo $1.44M_\odot + 1.44M_\odot$ models, which collapse promptly after the merger, have little to no ejecta. Third, the softest EOS, SFHo, has the most ejecta ($\sim 0.02M_\odot$), with DD2 in the middle and LS220 the least ejecta. Since LS220 is considered to be softer than DD2, it may be a bit surprising that it ejects less matter. However, LS220 contains a different set of particles and uses a different mathematical framework, so further studies varying stiffness, but keeping the mathematical underpinnings and composition of the EOS similar would be required to better investigate this discrepancy. Turning to the electron fraction and velocity, we see that $\langle Y_e \rangle$ and $\langle v_\infty \rangle$ are much higher in the polar regions and decrease roughly with mass ratio. $\langle Y_e \rangle$ tends to hover around ≈ 0.2 while $\langle v_\infty \rangle$ varies between $0.2c$ and $0.3c$. These trends will be touched on in the coming section, where we look at the Y_e and v_∞ distributions. There is also an interesting “bounce” trend in M_{ej} as binary asymmetry increases, e.g. there is a decrease in M_{ej} as we increase the mass ratio from S12132 to S12144. However M_{ej} increases as we increase the mass ratio further from S12144 to S12156. This was also seen for example in Sekiguchi et al. (Sekiguchi et al. (2016a)) for their SFHo models. However, Sekiguchi et al. found that for DD2 M_{ej} monotonically increased as the mass ratio became more unequal. The discrepancy between

model	Polar ($\theta < 30^\circ$)			Equatorial ($\theta > 30^\circ$)			Total		
	$M_{ej}(10^{-2}M_\odot)$	$\langle Y_e \rangle$	$\langle v_\infty \rangle$	$M_{ej}(10^{-2}M_\odot)$	$\langle Y_e \rangle$	$\langle v_\infty \rangle$	$M_{ej}(10^{-2}M_\odot)$	$\langle Y_e \rangle$	$\langle v_\infty \rangle$
D12132	0.054	0.392	0.392	0.406	0.186	0.258	0.460	0.210	0.273
D12144	0.016	0.309	0.327	0.319	0.157	0.198	0.335	0.164	0.204
D12156	0.010	0.342	0.365	0.463	0.179	0.161	0.473	0.182	0.165
D144144	0.036	0.351	0.385	0.324	0.201	0.254	0.360	0.216	0.267
L12132	0.011	0.331	0.377	0.083	0.188	0.204	0.094	0.205	0.224
L12144	0.026	0.292	0.345	0.357	0.186	0.185	0.384	0.194	0.196
L12156	0.004	0.302	0.336	0.230	0.190	0.131	0.234	0.192	0.135
L144144	0.000	0.344	0.362	0.012	0.213	0.255	0.012	0.217	0.258
S12132	0.114	0.358	0.346	1.461	0.214	0.221	1.574	0.224	0.230
S12144	0.061	0.317	0.319	0.778	0.197	0.212	0.839	0.206	0.220
S12156	0.097	0.297	0.301	1.704	0.198	0.175	1.802	0.204	0.181
S144144	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table 3.3: The first two wide-columns provide the mass (M_{ej}), average electron fraction ($\langle Y_e \rangle$) and average asymptotic velocity ($\langle v_\infty \rangle$) of the matter labelled unbound from time $t=0$ up to 7.5ms post-merger (or collapse) in the polar and equatorial regions. The last wide-column gives the mass (M_{ej}), average electron fraction ($\langle Y_e \rangle$) and average asymptotic velocity ($\langle v_\infty \rangle$) over all regions.

our results and those of Sekiguchi et al. could be due to the fact that Sekiguchi et al. fixed the total mass of their models or that the degree of binary asymmetry was less extreme in their work.

While there appear to be broad trends in mass, composition and velocity with mass-ratio, total-mass and EOS. There are still details in these trends that we do not yet understand. Larger sets of simulations with a broader ranges of parameters would help considerably.

Table 3.3 quantifies the unbound matter both on the grid (greater than $50M_\odot$ from the remnant) and off the grid. The material on the grid that is still labelled unbound at 7.5ms is most likely due to neutrino-driven wind. In Figure 3.5 we show the $1.2M_\odot + 1.32M_\odot$ models across EOS. The black contours encapsulate unbound matter, the white contours denote density and lastly the colormap denotes electron fraction, Y_e . We notice that there is still a small percentage of matter labelled unbound, with most being in the SFHo models. Also we again note the

neutron-rich core of SFHo and it's surrounding electron-rich disk, which is in contrast to the other EOS, with the DD2 and LS220 EOS having a less electron-rich disk. This could have a significant effect on long-term disk evolution.

Y_e and v_∞ distributions

The electron fraction and ejecta velocity are important in determining the attributes of the kilonova.

Figure 3.6 shows the velocity of the ejecta at infinity across EOS for both the equatorial and polar regions at 7.5ms post-merger. First we notice that polar ejecta have generally a larger velocity than the equatorial ejecta. This is most likely due to the polar ejecta originating primarily from shock waves produced at core-bounce, which produce faster moving ejecta than the more equatorial-plane bounded tidal ejecta. Furthermore, the faster-moving shock ejecta travelling in the equatorial-plane would collide with tidal-torque ejecta and slow it down. We also notice that overall, most of the ejecta is at the lower end of the velocity range, $v_\infty \sim 0.2c$ with very little ejecta at $v_\infty > 0.8c$ for any of the models pictured here. LS220 appears to eject matter with smaller velocities, however it is hard to definitively conclude this because the amount of ejecta from this EOS is so small. Thus, across EOS we cannot establish any robust differences in velocity distribution.

Figure 3.7 shows the electron fraction of ejected material for different EOS in both the equatorial and polar regions. Qualitatively, the shapes of the distributions are very similar, the equatorial ejecta vary over a broad range of electron fraction values whereas the polar ejecta, which make a much smaller fraction of the ejecta, take on much higher Y_e values on average. All EOS appear to share a similar broad distribution.

Figure 3.6 and 3.7 investigated EOS-dependence at fixed component-masses. We now consider varying the component masses, while keeping the EOS fixed. Figure 3.8 shows the distribution of v_∞ across ejecta for the SFHo EOS. As the mass ratio increases, it appears that the polar ejecta, becomes less broad and more compactly distributed around lower velocities,

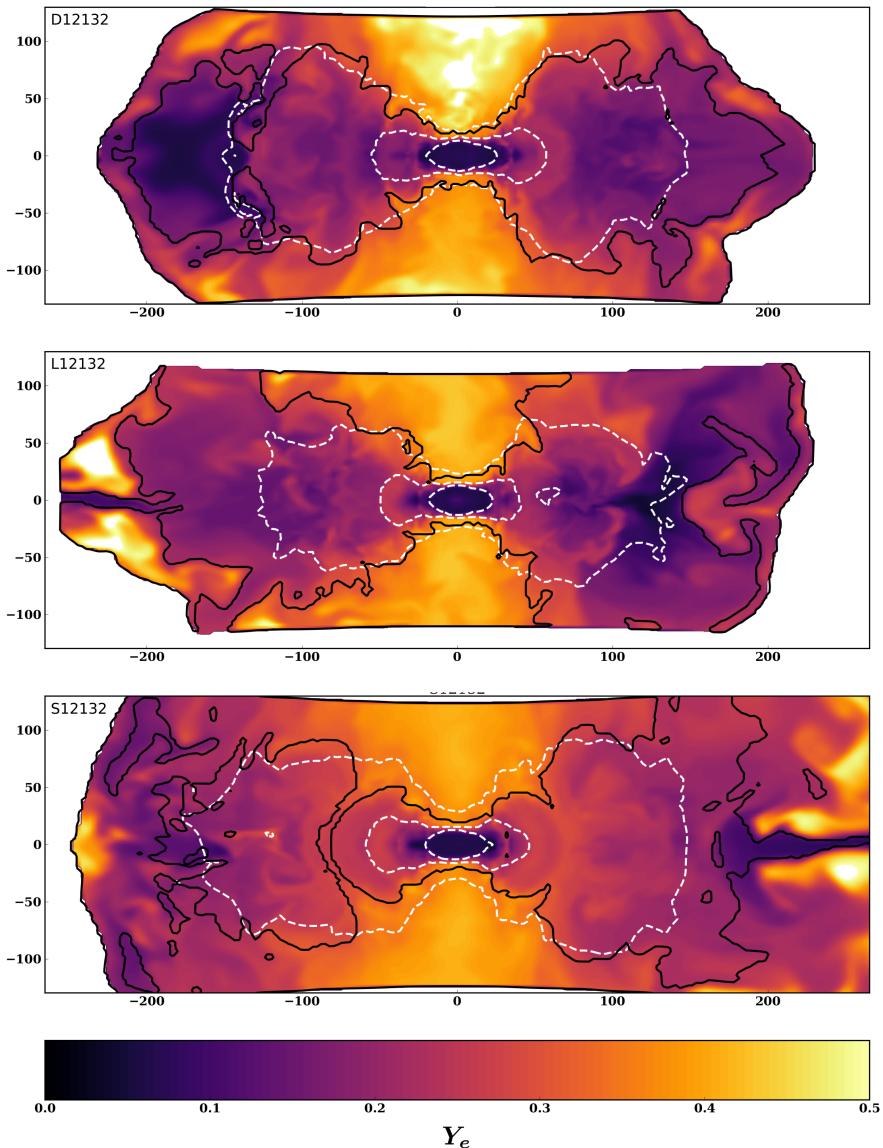


Figure 3.5: The electron fraction Y_e for the $1.2M_\odot + 1.32M_\odot$ models at 7.5ms post merger. The black contours encapsulate material that is flagged unbound. The white contours encapsulate material that is above densities $10^9, 10^{11}, 10^{13} g/cm^3$ respectively, with the higher density contours appearing closer to remnant core.

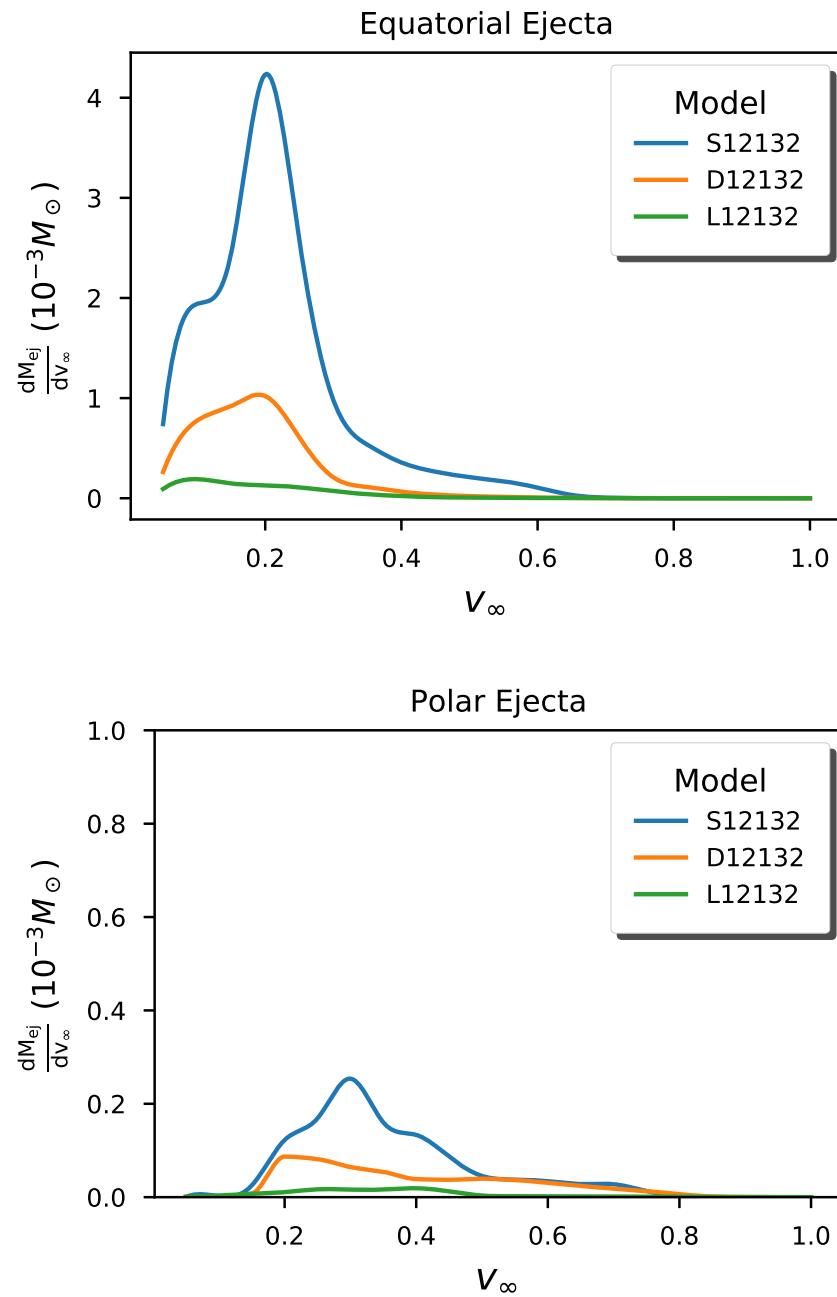


Figure 3.6: v_∞ distribution of the polar and equatorial ejecta across different EOS for the $1.2M_\odot + 1.32M_\odot$ models.

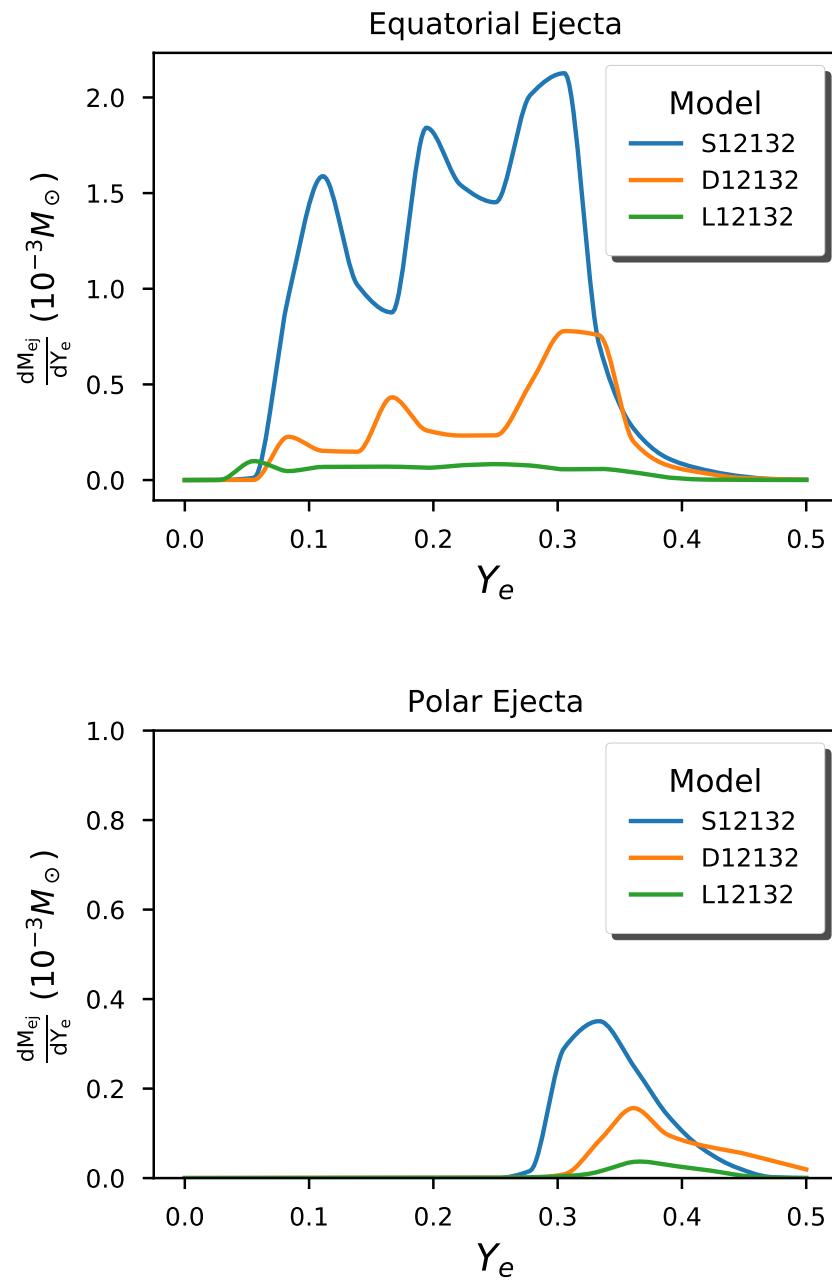


Figure 3.7: Y_e distribution of the polar and equatorial ejecta across different EOS for the $1.2M_\odot + 1.32M_\odot$ models .

but the equatorial ejecta seems more robust and exhibits very little change with mass-ratio. The lack of change in the equatorial ejecta is due to not fixing the total mass of the models. That is, since total mass enhances the impact at merger, but binary asymmetry lessens it, we get a cancellation and no significant change.

Finally, Figure 3.9 shows the distribution of electron fraction across ejecta for the SFHo EOS. We see that as mass asymmetry increases, there is impact on the distributions in terms of lowering the average Y_e , but the distributions still remain broad, with higher values in the polar regions.

Long-term Emission

Analysis of the kilonova emission associated with GW170817 showed that the ejecta mass was on the order of $\sim 0.05M_\odot$ (Metzger (2017b); Shibata et al. (2017)). This is above the estimates from dynamical ejecta seen in Table 3.3, but this is because we ignored other forms of ejecta which become more important at longer evolution times. At much longer times, when magnetic and neutrino driven winds increasingly dominate, a large fraction of the disk mass can be ejected. In this section we attempt to speculate on the size of such long-term ejecta mass. Our estimates rely on the following definition of disk mass:

$$M_{\text{disk}} = \int_{\rho < 10^{13} \text{ g/cm}^3} \rho_0 W \sqrt{\gamma} d^3 x, \quad (3.68)$$

which has been used in previous works, see (Radice et al. (2018); Shibata et al. (2017)). This definition is used for two reasons. Firstly, based on visualizations of the remnant, such as Fig. 3.5 we can clearly see that the 10^{13} g/cm^3 density contour encapsulates only the core of the remnant. Secondly, it was shown that for densities approximately below 10^{13} g/cm^3 , the matter is rotationally bound with roughly a $r^{-3/2}$ fall-off (Hanauske et al. (2017)). Unfortunately M_{disk} defined by Eq. (3.68) drifts to larger values over time and indeed it has not converged yet at 7.5ms. Nonetheless, we note similar evolution of this quantity over time for all EOS and mass-ratios, so using estimates at 7.5ms will give us a crude estimate on the lower bound of the disk mass.

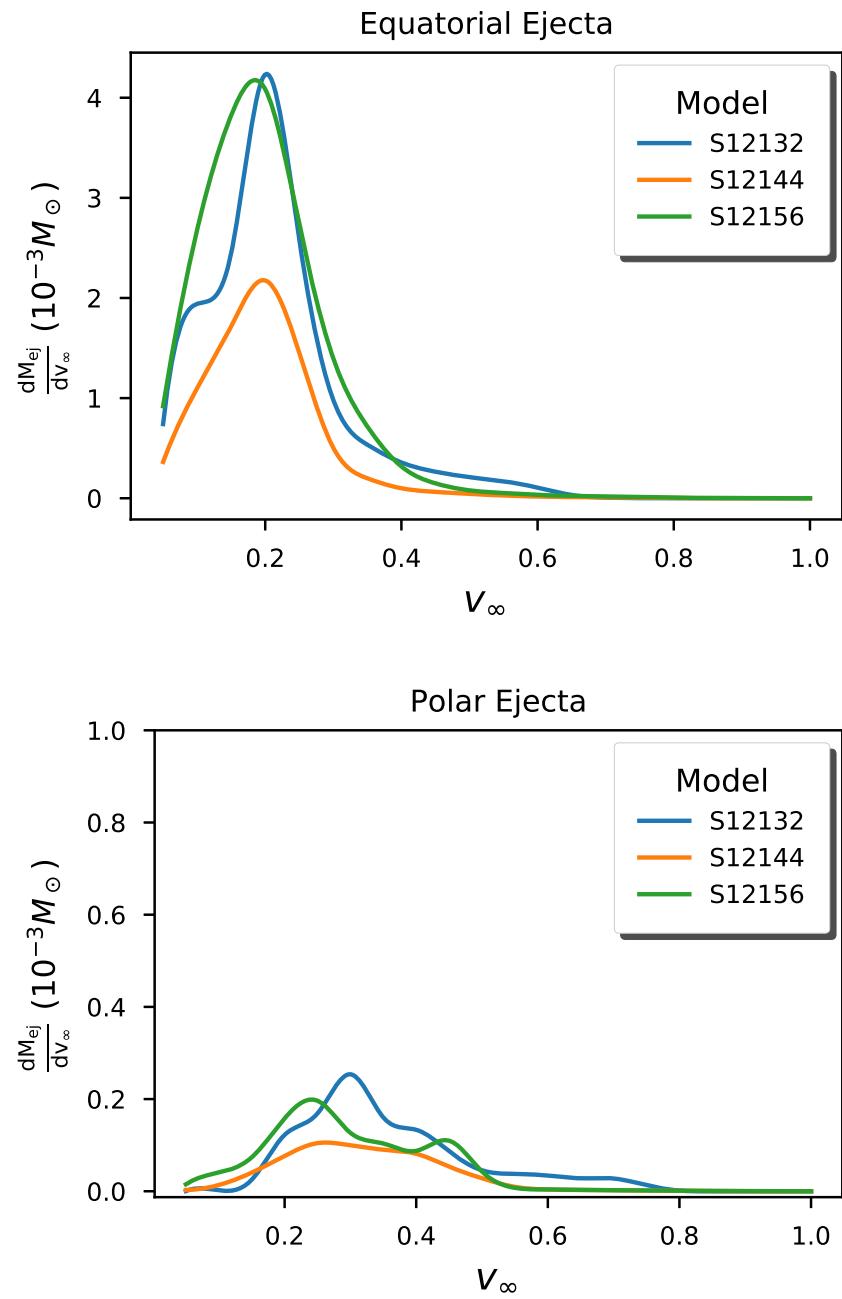


Figure 3.8: v_∞ distribution of the polar and equatorial ejecta across different mass ratios of the SFHo EOS.

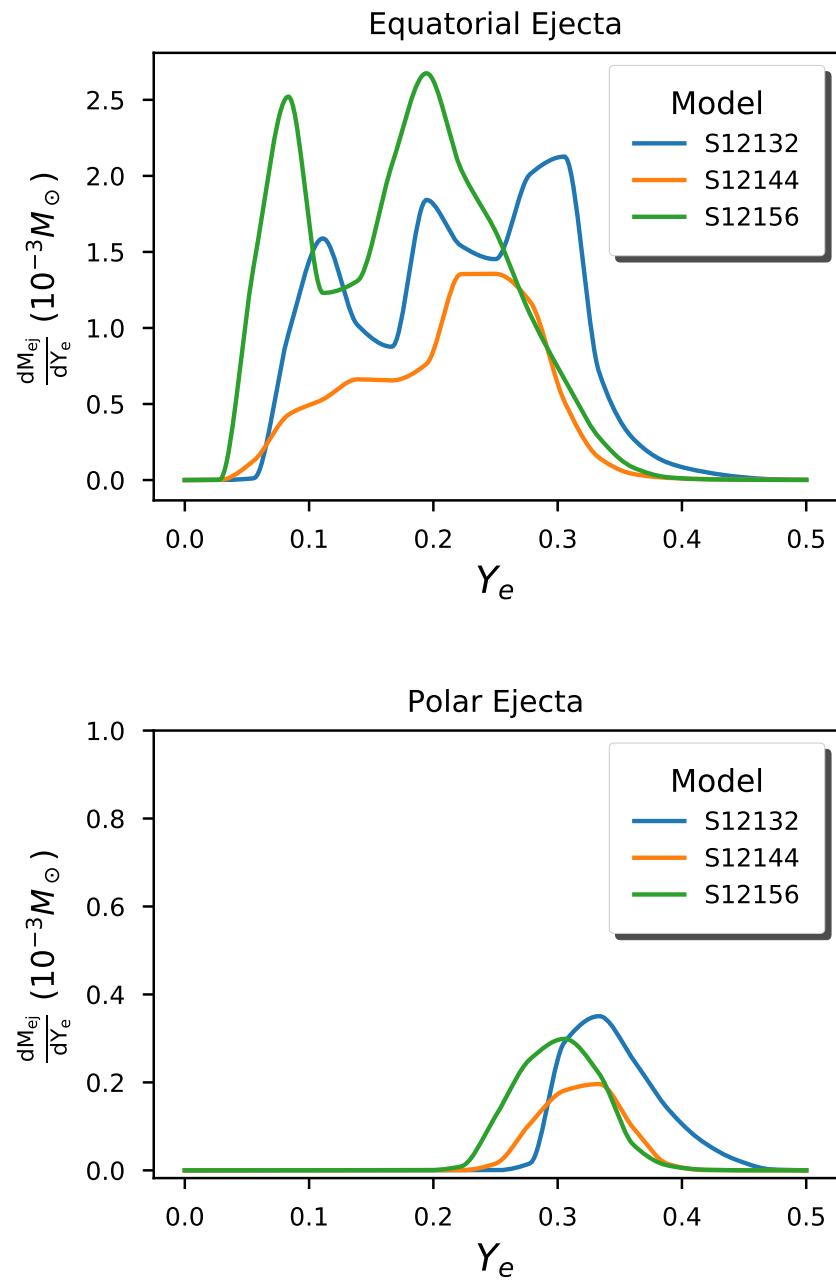


Figure 3.9: Y_e distribution of the polar and equatorial ejecta across different mass ratios of the SFHo EOS.

We compute the disk masses at 7.5-ms post-merger except for the models that collapse (e.g. S12156, S144144, L12156, L144144) where we compute the disk mass right before collapse. The masses are summarized in Fig. 3.10. We first note that the stiffer EOS produce larger disks. Moreover, disk mass increases as the degree of binary asymmetry increases.

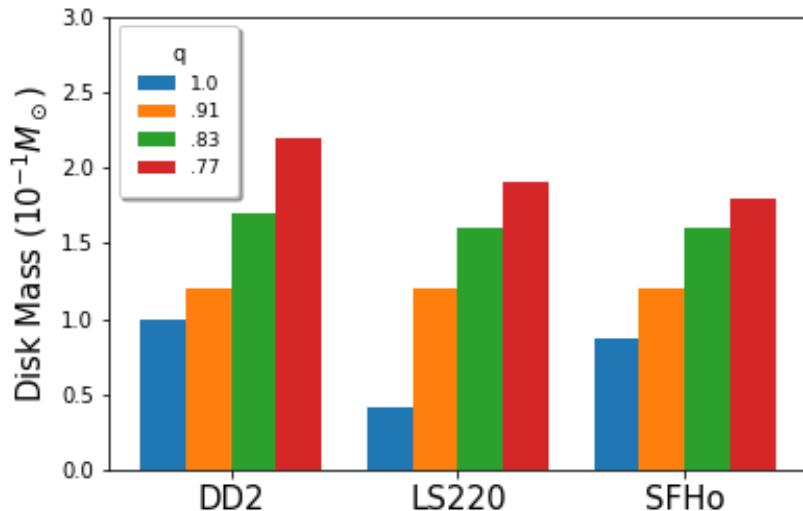


Figure 3.10: Disk masses at 7.5-ms post-merger or just before collapse if remnant is BH.

For ejecta from unmagnetized neutrino-driven winds, Qian et al. and Dessart et al. (Qian & Woosley (1996); Dessart et al. (2009)) find ejecta rate around $\sim 10^{-3} - 10^{-4} M_{\odot} s^{-1}$. Compared to an estimated 40% of the accretion disk ejected from magnetically driven winds (see for example (Fernandez et al. (2019))), which for the biggest disk would be $\sim .01 - .09 M_{\odot}$, the neutrino winds are negligible. Thus after longterm emission is taken into account, the possible estimated ejecta from GW170817 is very much in line with with expected ejections from simulations.

3.4.3 Neutrino Emission

Finally we turn to the neutrino emission of our models. Neutrinos play an important role in the post-merger evolution, not only cooling the central massive regions, but also transporting energy into the surface layers of the disk, providing a long-term outflow. On top of this, neutrinos

deposit net lepton number through weak interactions which can drastically change the electron fraction of outflows and thus affect the kilonova emission. Furthermore, they can shuffle large amounts of thermal energy into the polar regions which are electron-rich through $\nu\bar{\nu}$ annihilation, potentially producing jet-like emissions.

As seen in the bottom portion of Fig. 3.11, which shows the electron anti-neutrino energy density for the D144144 model at 3ms post-merger, the main emission regions are the hot, dense parts of the remnant; the central core, and the shocked tidal arms. In these dense parts of the remnant the neutrinos are advected with the flow. The only escape route for the neutrinos trapped in the dense part of the remnant is via the polar regions of the remnant neutrinosphere which show a greater neutrino energy density in the top portion of Fig. 3.11 than the equatorial parts. Due to the more compact and hotter remnants, we would expect that the SFHo models might show greater neutrino luminosity than the other EOS and this is confirmed in Figure 3.12 for the electron anti-neutrino luminosity. Although we do not plot them, the other species show exactly the same type of curve with SFHo having a much higher luminosity in all three species evolved in the transport scheme.

As the mass ratio changes, we find only mild changes in the neutrino luminosity as shown in Fig. 3.13. The heavy neutrino luminosity has the strongest reliance on the mass ratio, showing a sizable increase as the mass ratio becomes bigger. This is in contrast to Sekiguchi et al. (Sekiguchi et al. (2016b)) which finds a decrease in neutrino luminosity as the binary asymmetry increases. However, Sekiguchi et al. fix the total mass at $2.7M_{\odot}$ and since we only fix the lower mass in the binary, more compact mergers will result which means higher temperatures and higher neutrino emission. Although these more compact mergers tend to higher neutrino luminosity, it is cancelled out by the effect of the mass ratio increasing, which as Sekiguchi showed, tends to decrease the luminosity. Hence a cancellation effect might occur and this would explain the weak dependence on mass-ratio.

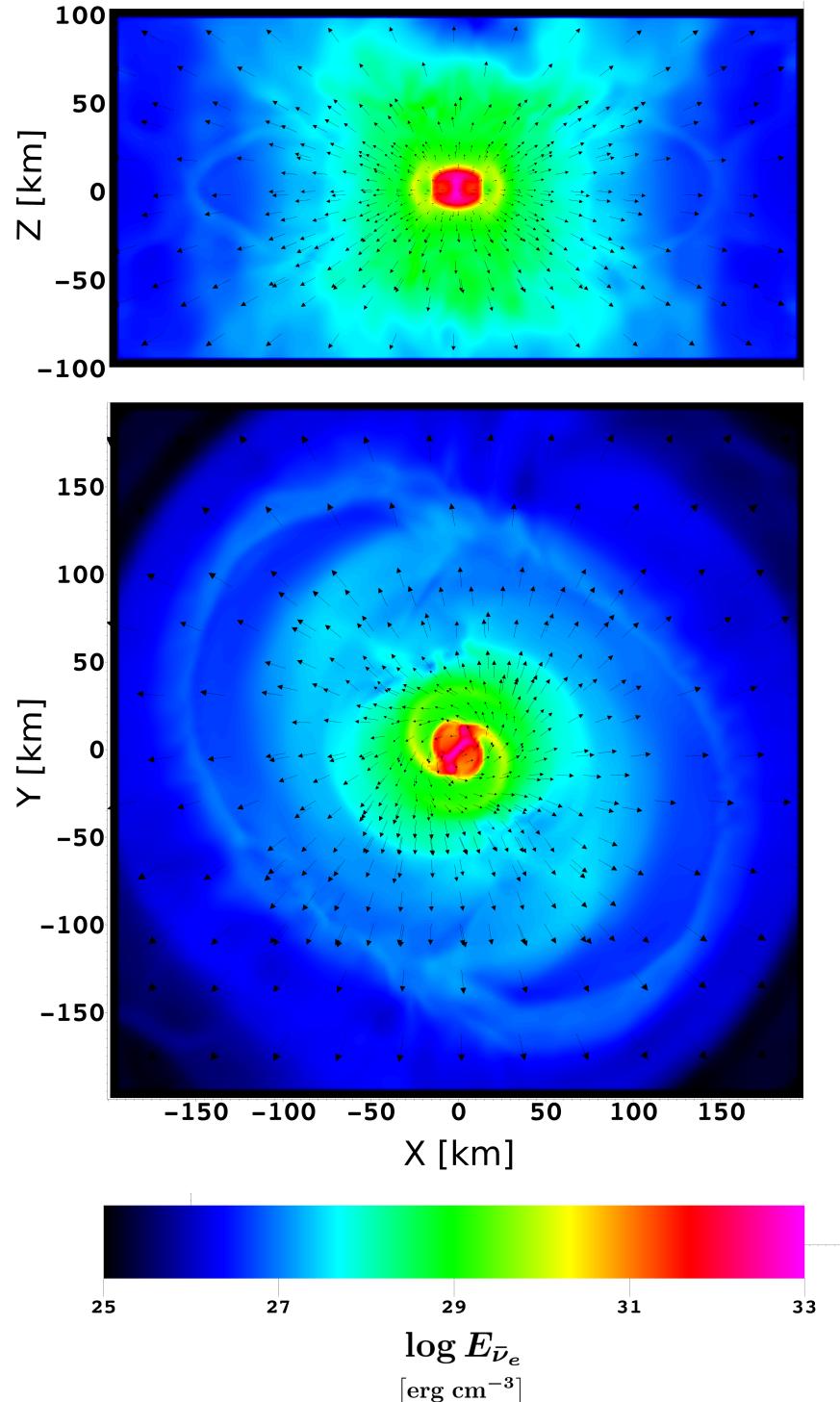


Figure 3.11: Anti-electron neutrino energy density (first moment) for the $1.44M_\odot + 1.44M_\odot$ DD2 model at 3ms post-merger. **top portion:** A xz-slice of the anti-electron neutrino energy density $E_{\bar{\nu}_e}$ with the arrows showing the effective neutrino transport velocity $v_\nu^i = \alpha \frac{E_{\bar{\nu}_e}}{F_{\bar{\nu}_e}^i} - \beta^i$. We see the largest energy density near the polar regions of the remnant, where the density is less and the neutrinos are free to stream. **bottom portion:** A xy-slice of the anti-electron neutrino energy density $E_{\bar{\nu}_e}$ with the arrows showing the effective neutrino transport velocity. We see the neutrinos advecting with the fluid in the dense regions near the core.

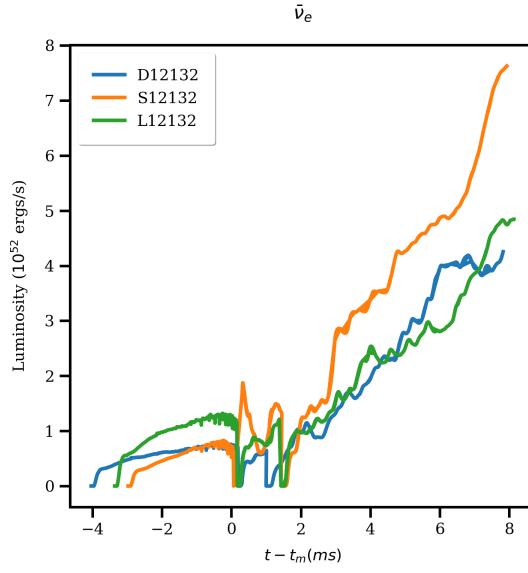


Figure 3.12: Anti-electron neutrino luminosity for the 12132 models up to around 7.5 post-merger. We see that for the softer SFHo EOS, there is a larger luminosity. Similar trends hold for the other neutrino species.

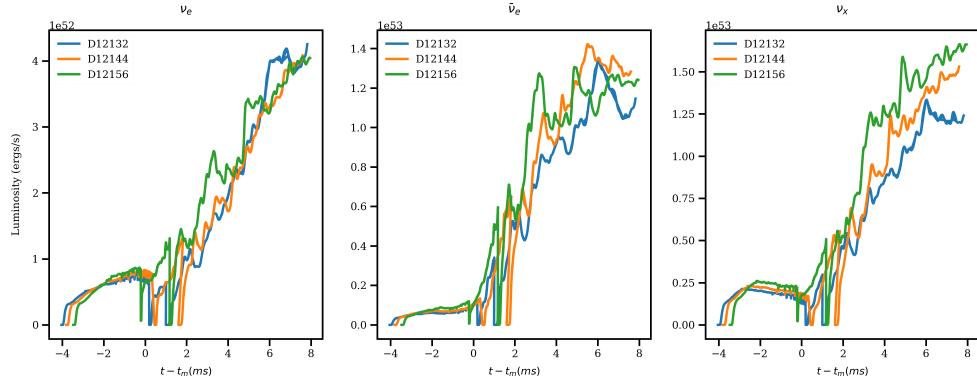


Figure 3.13: Neutrino luminosity for DD2 runs up to around 7.5ms across neutrino species.

Next we take a look at the angular distribution of the neutrino emission. Figure 3.14 shows the neutrino flux density as a function of its angle with respect to the equatorial plane at 7.5 ms after merger. From this figure, we can clearly see that most of the neutrinos are emitted in the polar directions. Once a disk forms, the neutrinos are mostly confined with a core of 40 degrees around the poles, with an amplitude peak at 30 - 40 degrees from the poles becoming

more visible at later times. This peak is probably due to neutrinos beamed from the shocked tidal arms, which become less optically thick as time passes. The confinement of the neutrinos to the polar directions stems from the fact that neutrinos escape through the low-density regions above and below the disk and are confined by the optically thick accretion disk. As the EOS changes, the distributions stay qualitatively the same. However, we notice that the SFHo model has overall higher production of neutrinos across all angles. This is simply due to it's higher temperatures. Likewise, across mass-ratio there is little impact on the morphology, with the plots looking almost identical to Figure 3.14 and thus we do not show them here.

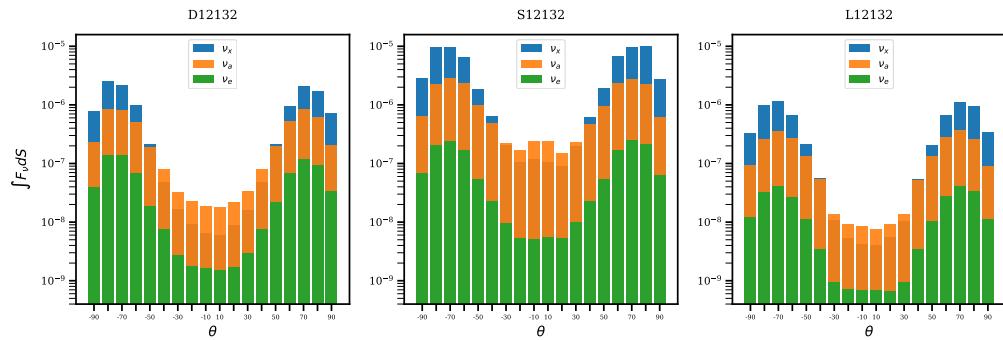


Figure 3.14: Neutrino flux density moment as a function of angle for different EOS runs at 7.5ms. The angle is defined with respect to the equatorial plane with 0° being the equator, and $\pm 90^\circ$ being the North and South poles. The different neutrino species are color coded.

While neutrino irradiation can also effect the amount of ejected matter, this has the greatest effect over a longer time period than considered here. We do see some effects of neutrino irradiation in the electron fraction distributions up to 10ms however. Figure 3.15 shows the electron fraction distributions at 7.5-ms and 10-ms post-merger. As we can see, there is an increase in Y_e as the simulation proceeds. The reason for this is that in the presence of strong electron neutrino luminosity, the neutrino capture processes are activated, increasing the overall electron fraction.

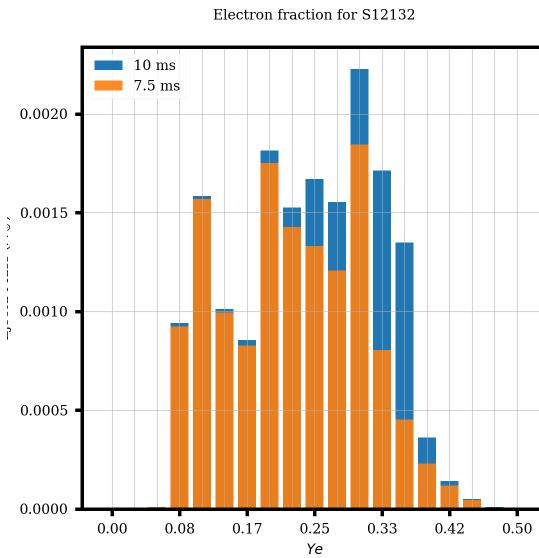


Figure 3.15: Electron fraction distribution for the off-grid ejecta of the $1.2M_{\odot} + 1.32M_{\odot}$ SFHo model at 7.5 ms and 10 ms post-merger. We can clearly see that more matter is being ejected at higher Y_e fractions due to neutrino irradiation.

Furthermore, as seen in the previous sections, the SFHo disk is at a very high Y_e . This is most-likely due to the greater temperatures and thus the increased neutrino irradiation. This can have a big effect on the Y_e distribution of the long-term ejecta and thus on the kilonova signal.

3.5 Conclusion and Future Work

We presented a first set of SpEC simulations of unequal mass neutron star mergers using a nuclear-theory based equations of state, a two-moment grey neutrino transport scheme with an improved energy estimate based on evolving the number density. This set of simulations varied EOS, total mass and mass-ratio in order to determine any robust trends between these parameters and emission characteristics. We found the following

- For the models that promptly collapsed, namely L144144 and S144144, there was little to no ejecta and the disk masses were much smaller. Presumably the lack of ejecta in these cases also has to do with them being equal mass.

- The softest models, those of the SFHo EOS, had the most ejecta and SFHo was the only EOS with $M_{ej} \sim 10^{-2} M_{\odot}$. For all EOS the ejecta was mostly confined to the equatorial plane.
- The Y_e distributions were very broad ($\sim 0.06 - 0.48$) and quite similar across EOS and mass-ratio. $\langle Y_e \rangle$ is around ~ 0.2 , but increases over time as the ejected matter is irradiated by neutrinos. We see a decrease in the average Y_e with increasing binary asymmetry, but there is some scatter, so the relationship might be more complicated.
- The asymptotic velocity distributions were also very broad ($\sim 0.05c - 0.7c$) and mostly distributed in the lower ranges of velocities with the average being within the range $0.2 - 0.3c$ across EOS and mass-ratio. We see a decrease in the average asymptotic velocity with mass-ratio.
- The disk masses of the models at 7.5 ms post-merger appear to increase with mass-ratio and stiffness of the EOS. SFHo appears to have a neutron-poor disk with a much higher Y_e than the other EOS. This could be significant in long-term studies of the models.
- Neutrino emission was dominated by the SFHo EOS, which showed higher luminosities. We found no dependence on the mass-ratio with luminosity however, except in the case of the heavy neutrino group. Most of the neutrino emission was in the polar region and we did not find any dependence on its morphology with mass-ratio or EOS.

The results presented here are limited by two important assumptions in our simulations. Firstly, the absence of magnetic fields. Over the short time scales considered here, magnetohydrodynamics effects are not expected to affect the evolution of neutron star remnant, but could drive additional outflows from the disk (Kiuchi et al. (2014); Neilsen et al. (2014)). Over longer time scales, magnetic fields would be critical to the spin evolution of the remnant neutron star, angular momentum transport, heating in the disk, and possibly the formation of relativistic jets and magnetically-driven outflows. The second limitation is the relatively small numerical grid

on which we evolve the equations of general relativistic hydrodynamics. The grid limits our ability to measure the mass and properties of the outflows accurately. We expect to address these issues in the future. However, we do not expect these assumptions to significantly affect the main results of this work, i.e. the properties of the ejecta and the trends across EOS and mass-ratio in the first 7.5 ms after merger.

In the future, we hope to expand this set of simulations, both in terms of mass-ratio, EOS, eccentricity, spins, evolution time as well as evolve the collapsed runs to study the accretion disk evolution and the effect of collapse on ejected matter. On top of this, improving the neutrino scheme will be important. While we are a long way off from solving the full Boltzmann transport equations, there are many other possible improvements. Eliminating the gray approximation, using more accurate closure schemes such as in (Foucart (2018)), or including neglected effects such as $\nu\bar{\nu}$ annihilation and inelastic scatterings are all ways to improve the accuracy of the scheme. Another long-term goal is add MHD into the simulations in order to see the role it plays in the emission characteristics.

Chapter 4

Conclusions & Future Work

4.1 Conclusions

It is an exciting time in gravitational-wave astrophysics and numerical relativity. With the first detection of gravitational waves from a binary neutron star and the coincident detection of electromagnetic counterparts across the EM spectrum, a lot can be discovered about these extreme systems. With many more detections on the horizon, we will need to be prepared on the theoretical side. To help with the ongoing theoretical effort of properly modelling binary compact object systems and their emissions, this thesis has attempted to make progress along two different directions.

Firstly, we have sought to improve the numerical techniques used to solve the partial differential equations that arise in binary neutron star simulations. To this effect, Chapter 2 presented a completely new scheme based on discontinuous Galerkin methods, and we tested it to solve for constant density star initial data, which contains phase transitions similar to that in neutron stars. We also tested the code on multi-black hole puncture initial data, which contains multiple non-smooth points. All tests showed promising results. Currently, work is being done to extend this code by Nils Fischer and Prof. Harald Pfeiffer at the Albert Einstein Institute (AEI) in Potsdam, Germany by porting it into the task based parallel code SpECTRE (Kidder et al. (2016)), which is currently being developed at Cornell, Caltech and AEI to solve

hyperbolic problems in numerical relativity on future exascale supercomputers using a task-based parallelism framework. The final goal will be to run our implementation on binary neutron star initial data with realistic microphysics and solve down to accuracies previously unobtainable.

Secondly, we wanted to increase the knowledge-base surrounding emissions from BNS mergers and their connection to binary parameters. Chapter 3 presented 12 state-of-art general relativistic radiation-hydrodynamics simulations of binary neutron star mergers with varying realistic EOS and mass-ratios. With these simulations we established that previous results using different codes were qualitatively correct, even though our neutrino schemes (amongst other things) were not identical. Talk is under way to use these simulations for new studies. Firstly, the subset of simulations which collapsed to a black hole were not evolved past this point and it would be interesting to see what emissions arose afterwards and to study the properties and evolution of the resulting accretion disks, which have been shown to produce a significant amount of ejecta when MHD is introduced (Fernandez et al. (2019)). Secondly, we initially ran each simulation with tracer particles that followed the fluid flow, but stopped this because it was slowing down the code too much to get a report out on time for this thesis. Tracer particles would allow us to better study the properties of the ejecta, by not only tracing out the fluid worldlines of the ejecta, but also by using the tracer data to seed light-curve simulations and nuclear reaction networks. All of this was done for neutron star - black-hole binaries in (Fernández et al. (2016)) by some of our collaborators. In terms of other source parameters, We also neglected in our study the eccentricity of the binary and the spins of the neutron stars, which can produce significantly more ejecta. In the future, the addition of magneto-hydrodynamics, eccentricity, spins and more realistic neutrino transport schemes will definitely be on the agenda.

In conclusion, while this thesis has helped push us closer to the goal of simulating and understanding binary neutron star mergers, a great deal of work still needs to be done. The next few decades should be a very interesting period for both numerical relativity and gravitational wave astrophysics, so stay tuned.

Bibliography

2019, The Spectral Einstein Code, <http://www.black-holes.org/SpEC.html>

Abbott, B. P., et al. 2016a, Physical Review X, 6, 041014

—. 2016b, Phys. Rev. Lett., 116, 241102

—. 2017, Astrophys. J., 848, L13

Abbott, B. P., et al. 2017, "Astrophys. J. Lett.", 848, L13

Abbott, B. P., et al. 2017a, Physical Review Letters, 119, 161101

—. 2017b, Phys. Rev. Lett., 119, 161101

—. 2017c, Astrophys. J., 848, L12

—. 2017d, Astrophys. J., 851, L16

Acernese, F., et al. 2015, Classical and Quantum Gravity, 32, 024001

Affeldt, C., et al. 2014, Classical and quantum gravity, 31, 224002

Ajello, M., et al. 2018, The Astrophysical Journal, 861, 85

Ansorg, M. 2007, Class. Quantum Grav., 24, S1

Ansorg, M., Brügmann, B., & Tichy, W. 2004, Physical Review D, 70, 064011

Antonietti, P. F., & Houston, P. 2011, Journal of Scientific Computing, 46, 124

- Arnold, D. N., Brezzi, F., Cockburn, B., & Marini, L. D. 2002, SIAM Journal on Numerical Analysis, 39, 1749
- Arnowitt, R., Deser, S., & Misner, C. W. 1959, Physical Review, 116, 1322
- Baiotti, L., & Rezzolla, L. 2016, ArXiv e-prints
- Baiotti, L., & Rezzolla, L. 2016, arXiv preprint arXiv:1607.03540
- Balay, S., Buschelman, K., Gropp, W. D., Kaushik, D., McInnes, L. C., & Smith, B. F. 2001, PETSc home page, <http://www.mcs.anl.gov/petsc>
- Baumgarte, T. W., O'Murchadha, N., & Pfeiffer, H. P. 2007, Phys. Rev. D, 75, 044009
- Baumgarte, T. W., & Shapiro, S. L. 1998, Physical Review D, 59, 024007
- . 2003, The Astrophysical Journal, 585, 921
- . 2010, Numerical relativity: solving Einstein's equations on the computer (Cambridge University Press)
- Bauswein, A., Baumgarte, T., & Janka, H.-T. 2013, Physical review letters, 111, 131101
- Bauswein, A., Janka, H.-T., & Oechslin, R. 2010, Physical Review D, 82, 084043
- Bauswein, A., & Stergioulas, N. 2019, arXiv preprint arXiv:1901.06969
- Bauswein, A., Stergioulas, N., & Janka, H.-T. 2014, Physical Review D, 90, 023002
- Bell, H. E. 1965, The American Mathematical Monthly, 72, 292
- Bi, C., Wang, C., & Lin, Y. 2015, Computer Methods in Applied Mechanics and Engineering, 297, 140
- Bona, C., Ledvinka, T., Palenzuela, C., & Žáček, M. 2003, Physical Review D, 67, 104005
- Bowen, J. M., & York Jr, J. W. 1980, Physical Review D, 21, 2047

- Brandt, S., & Brügmann, B. 1997, *Physical Review Letters*, 78, 3606
- Briggs, W., Henson, V., & McCormick, S. 2000, *A Multigrid Tutorial: Second Edition* (Society for Industrial and Applied Mathematics)
- Brown, J. D., et al. 2012, *Phys. Rev. D*, 85, 084004
- Bugner, M., Dietrich, T., Bernuzzi, S., Weyhausen, A., & Brügmann, B. 2016, *Phys. Rev. D*, 94, 084004
- Buis, P. E., & Dyksen, W. R. 1996, *ACM Transactions on Mathematical Software (TOMS)*, 22, 18
- Burrows, A., Reddy, S., & Thompson, T. A. 2006, *Nuc. Phys. A*, 777, 356
- Burstedde, C., Wilcox, L. C., & Ghattas, O. 2011, *SIAM Journal on Scientific Computing*, 33, 1103
- Caudill, M., Cook, G. B., Grigsby, J. D., & Pfeiffer, H. P. 2006, *Physical Review D*, 74, 064011
- Chornock, R., et al. 2017, "Astrophys. J. Lett.", 848, L19
- Cockburn, B. 1998, in *An introduction to the Discontinuous Galerkin method for convection-dominated problems*; in *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations: Lectures given at the 2nd Session of the Centro Internazionale Matematico Estivo (C.I.M.E.) held in Cetraro, Italy, June 23–28, 1997*, ed. A. Quarteroni, *Lecture Notes in Physics* (Berlin, New York: Springer), 150–268
- Cockburn, B. 2001, *J. Comput. Appl. Math.*, 128, 187
- Cockburn, B., Karniadakis, G. E., & Shu, C.-W. 2000, in *Discontinuous Galerkin Methods: Theory, Computation and Applications*, ed. B. Cockburn, G. E. Karniadakis, & C.-W. Shu, *Lecture Notes in Computational Science and Engineering* (Berlin, New York: Springer), 3–50

- Cockburn, B., & Shu, C.-W. 1998, *J. Comput. Phys.*, 141, 199
- Collaboration, L. S., et al. 2017, *Nature*, 551, 85
- Cook, G. 2000, *Living Rev. Rel.*, 3, 5
- Cook, G. B. 1994, *Phys. Rev. D*, 50, 5025
- Cook, G. B., & Pfeiffer, H. P. 2004, *Physical Review D*, 70, 104016
- Cordero-Carrión, I., Cerdá-Durán, P., Dimmelmeier, H., Jaramillo, J. L., Novak, J., & Gourgoulhon, E. 2009, *Physical Review D*, 79, 024017
- Coulter, D. A., et al. 2017, *Science*, 358, 1556
- Cowperthwaite, P. S., et al. 2017, *Astrophys. J.*, 848, L17
- Davies, M., Benz, W., Piran, T., & Thielemann, F. 1994, arXiv preprint astro-ph/9401032
- Deaton, M. B., et al. 2013, *The Astrophysical Journal*, 776, 47
- Del Pozzo, W., Li, T. G. F., Agathos, M., Van Den Broeck, C., & Vitale, S. 2013, *Phys. Rev. Lett.*, 111, 071101
- Demorest, P., Pennucci, T., Ransom, S., Roberts, M., & Hessels, J. 2010, *Nature*, 467, 1081
- Dennison, K. A., Baumgarte, T. W., & Pfeiffer, H. P. 2006, *Phys. Rev. D*, 74, 064016
- Dessart, L., Ott, C. D., Burrows, A., Rosswog, S., & Livne, E. 2009, *Astrophys. J.*, 690, 1681
- Di Pietro, D. A., & Ern, A. 2011, Mathematical aspects of discontinuous Galerkin methods, Vol. 69 (Springer Science & Business Media)
- Elman, H. C., Ernst, O. G., & O'leary, D. P. 2001, *SIAM Journal on scientific computing*, 23, 1291
- Endeve, E., Hauck, C. D., Xing, Y., & Mezzacappa, A. 2015, *J. Comput. Phys.*, 287, 151

- Evans, P. A., et al. 2017, "Science", 358, 1565
- Faber, J. A., & Rasio, F. A. 2012, *Living Rev. Relativity*, 15
- Fabien, M. S., Knepley, M. G., Mills, R. T., & Riviere, B. M. 2019, *SIAM Journal on Scientific Computing*, 41, C73
- Feng, Y., Straka, M., Di Matteo, T., & Croft, R. 2015
- Fernández, R., Foucart, F., Kasen, D., Lippuner, J., Desai, D., & Roberts, L. F. 2016, arXiv preprint arXiv:1612.04829
- Fernández, R., & Metzger, B. D. 2013, *Mon. Not. Roy. Astr. Soc.*, 435, 502
- Fernandez, R., Tchekhovskoy, A., Quataert, E., Foucart, F., & Kasen, D. 2019, *Bulletin of the American Physical Society*
- Fick, P. W. 2014, arXiv preprint
- Field, S. E., Hesthaven, J. S., & Lau, S. R. 2009, *Class. Quantum Grav.*, 26, 165010
- Field, S. E., Hesthaven, J. S., Lau, S. R., & Mroue, A. H. 2010, *Phys. Rev. D*, 82, 104051
- Fishman, G., & Johnson, W. 1989, *Proc. GRO Science Workshop*
- Fishman, G. J., & Meegan, C. A. 1995, *Annual Review of Astronomy and Astrophysics*, 33, 415
- Foucart, F. 2018, *Mon. Not. Roy. Astr. Soc.*, 475, 4186
- Foucart, F., Kidder, L. E., Pfeiffer, H. P., & Teukolsky, S. A. 2008, *Physical Review D*, 77, 124051
- Foucart, F., O'Connor, E., Roberts, L., Kidder, L. E., Pfeiffer, H. P., & Scheel, M. A. 2016, *Phys. Rev.*, D94, 123016
- Foucart, F., O'Connor, E., Roberts, L., Kidder, L. E., Pfeiffer, H. P., & Scheel, M. A. 2016a, *Physical Review D*, 94, 123016

- Foucart, F., et al. 2014, *Physical Review D*, 90, 024026
- . 2015a, arXiv preprint arXiv:1510.06398
- . 2015b, arXiv preprint arXiv:1502.04146
- Foucart, F., et al. 2015, *Phys. Rev. D*, 91, 124021
- Foucart, F., et al. 2016b, *Phys. Rev.*, D93, 044019
- Foucart, F., et al. 2016, *Phys. Rev. D*, 93, 044019
- Freiburghaus, C., Rosswog, S., & Thielemann, F.-K. 1999, *The Astrophysical Journal Letters*, 525, L121
- Ghysels, P., Kłosiewicz, P., & Vanroose, W. 2012, *Numerical Linear Algebra with Applications*, 19, 253
- Gourgoulhon, E., Grandclement, P., Taniguchi, K., Marck, J.-A., & Bonazzola, S. 2001, *Physical Review D*, 63, 064029
- Haas, R., et al. 2016, *Phys. Rev. D*, D93, 124062
- Hanauske, M., Takami, K., Bovard, L., Rezzolla, L., Font, J. A., Galeazzi, F., & Stöcker, H. 2017, *Physical Review D*, 96, 043004
- Hansbo, P., & Larson, M. G. 2011, *Computer Methods in Applied Mechanics and Engineering*, 200, 3026
- Harten, A., Lax, P. D., & van Leer, B. 1983, *SIAM Review*, 25, 35
- Hempel, M., Fischer, T., Schaffner-Bielich, J., & Liebendörfer, M. 2012, *Astrophys.J.*, 748, 70
- Hempel, M., Oertel, M., Typel, S., & Klähn, T. 2017, in *Proceedings of the 14th International Symposium on Nuclei in the Cosmos (NIC2016)*, 010802

- Henriksson, K., Foucart, F., Kidder, L. E., & Teukolsky, S. A. 2014, arXiv preprint arXiv:1409.7159
- . 2016, *Class. Quantum Grav.*, 33, 105009
- Hesthaven, J., & Warburton, T. 2008, *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*, Texts in Applied Mathematics (Springer)
- Hinderer, T., Lackey, B. D., Lang, R. N., & Read, J. S. 2010, *Physical Review D*, 81, 123016
- Hjorth, J. 2013, *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 371, 20120275
- Hotokezaka, K., Kiuchi, K., Kyutoku, K., Okawa, H., Sekiguchi, Y., Shibata, M., & Taniguchi, K. 2013, *Phys. Rev. D*, 87, 024001
- Houston, P., Perugia, I., & Schötzau, D. 2005a, *Comput. Methods Appl. Mech. Eng.*, 194, 499
- Houston, P., Robson, J., & Süli, E. 2005b, *IMA journal of numerical analysis*, 25, 726
- Houston, P., Schötzau, D., & Wihler, T. 2005c, *J. Sci. Comput.*, 22, 347
- Houston, P., Schötzau, D., & Wihler, T. P. 2007, *Mathematical Models and Methods in Applied Sciences*, 17, 33
- Houston, P., Süli, E., & Wihler, T. P. 2008, *IMA journal of numerical analysis*, 28, 245
- Hulse, R. A., & Taylor, J. H. 1975, *Astrophys. J.*, 195, L51
- Janka, H.-T., Eberl, T., Ruffert, M., & Fryer, C. L. 1999, *Astrophys. J.*, 527, L39
- Jiang, G.-S., & Shu, C.-W. 1996, *J. Comput. Phys.*, 126, 202
- Kasen, D., Metzger, B., Barnes, J., Quataert, E., & Ramirez-Ruiz, E. 2017, *Nature*, 551, 80
- Kasliwal, M. M., et al. 2017, *Science*, 358, 1559

- Kastaun, W., & Galeazzi, F. 2015, Phys. Rev. D, 91, 064027
- Kidder, L. E., et al. 2016, arXiv preprint arXiv:1609.00098
- Kidder, L. E., et al. 2017, J. Comput. Phys., 335, 84
- Kilpatrick, C. D., et al. 2017, "Science", 358, 1583
- Kiuchi, K., Kyutoku, K., Sekiguchi, Y., Shibata, M., & Wada, T. 2014, Phys. Rev. D, 90, 041502
- Klebesadel, R. W., Strong, I. B., & Olson, R. A. 1973, The Astrophysical Journal, 182, L85
- Kouveliotou, C., Meegan, C. A., Fishman, G. J., Bhat, N. P., Briggs, M. S., Koshut, T. M., Paciesas, W. S., & Pendleton, G. N. 1993, The Astrophysical Journal, 413, L101
- Kozdon, J. E., & Wilcox, L. C. 2018, Journal of Scientific Computing, 76, 1742
- Kozdon, J. E., Wilcox, L. C., Hagstrom, T., & Banks, J. W. 2019, Journal of Computational Physics
- Kronbichler, M., & Wall, W. A. 2018, SIAM Journal on Scientific Computing, 40, A3423
- Kulkarni, S. R. 2005, ArXiv Astrophysics e-prints
- Kyutoku, K. 2013, The Black Hole-Neutron Star Binary Merger in Full General Relativity: Dependence on Neutron Star Equations of State (Springer Science & Business Media)
- Kyutoku, K., Ioka, K., Okawa, H., Shibata, M., & Taniguchi, K. 2015, arXiv preprint arXiv:1502.05402
- Lackey, B. D., & Wade, L. 2015, Phys. Rev. D, 91, 043002
- Lattimer, J. M. 2012, Ann.Rev.Nucl.Part.Sci., 62, 485
- Lattimer, J. M., & Schramm, D. N. 1974, The Astrophysical Journal, 192, L145
- Lattimer, J. M., & Schramm, D. N. 1976, Astrophys. J., 210, 549

- Lattimer, J. M., & Swesty, F. D. 1991, Nuclear Physics A, 535, 331
- Lau, S. R., Lovelace, G., & Pfeiffer, H. P. 2011, Phys. Rev. D, 84, 084023
- Lau, S. R., Pfeiffer, H. P., & Hesthaven, J. S. 2009, Commun. Comput. Phys., 6, 1063
- Lee, W. H., & Kluzniak, W. L. 1999, Astroph. J., 526, 178
- Lehner, L., Liebling, S. L., Palenzuela, C., Caballero, O., O'Connor, E., Anderson, M., & Neilsen, D. 2016, arXiv preprint arXiv:1603.00501
- Levan, A., Crowther, P., de Grijs, R., Langer, N., Xu, D., & Yoon, S.-C. 2016, Space Science Reviews, 202, 33
- Li, H.-B., Huang, T.-Z., Zhang, Y., Liu, X.-P., & Gu, T.-X. 2011, Applied Mathematics and Computation, 218, 260
- Li, L.-X., & Paczynski, B. 1998, Astrophys. J., 507, L59
- LIGO, Virgo, et al. 2018, arXiv preprint arXiv:1811.12907
- Lindblom, L., Scheel, M. A., Kidder, L. E., Owen, R., & Rinne, O. 2006, Class. Quantum Grav., 23, S447
- Liu, X.-D., Osher, S., & Chan, T. 1994, J. Comput. Phys., 115, 200
- Lovadina, C., & Marini, L. 2009, J. Sci. Comput., 40, 340
- Lovelace, G., Owen, R., Pfeiffer, H. P., & Chu, T. 2008, Phys. Rev. D78, 084017
- McCully, C., et al. 2017, "Astrophys. J. Lett.", 848, L32
- Melenk, J. M., & Wohlmuth, B. I. 2001, Advances in Computational Mathematics, 15, 311
- Mengaldo, G., De Grazia, D., Moxey, D., Vincent, P., & Sherwin, S. 2015, Journal of Computational Physics, 299, 56

- Metzger, B. D. 2016, arXiv:1610.09381
- Metzger, B. D. 2017a, *Living Reviews in Relativity*, 20, 3
- . 2017b
- Metzger, B. D., & Fernández, R. 2014, *Mon. Not. Roy. Astr. Soc.*, 441, 3444
- Metzger, B. D., et al. 2010, *Mon. Not. Roy. Astr. Soc.*, 406, 2650
- Miller, J. M., & Schnetter, E. 2017, *Class. Quantum Grav.*, 34, 015003
- Mitchell, W. F., & McClain, M. A. 2011, in *Recent advances in computational and applied mathematics* (Springer), 227–258
- Mochkovitch, R., Hernanz, M., Isern, J., & Martin, X. 1993, *Nature*, 361, 236
- Mocz, P., Vogelsberger, M., Sijacki, D., Pakmor, R., & Hernquist, L. 2014, *Mon. Not. Roy. Astr. Soc.*, 437, 397
- Mooley, K. P., et al. 2018, *Nature*, 561, 355
- Muralikrishnan, S., Bui-Thanh, T., & Shadid, J. N. 2019, arXiv preprint
- Nagakura, H., Hotokezaka, K., Sekiguchi, Y., Shibata, M., & Ioka, K. 2014, *Astrophys.J.*, 784, L28
- Neilsen, D., Liebling, S. L., Anderson, M., Lehner, L., O'Connor, E., & Palenzuela, C. 2014, *Physical Review D*, 89, 104029
- Nicholl, M., et al. 2017, "Astrophys. J. Lett.", 848, L18
- Notay, Y. 2000, *SIAM Journal on Scientific Computing*, 22, 1444
- Oechslin, R., & Janka, H.-T. 2006, *Monthly Notices of the Royal Astronomical Society*, 368, 1489

- Palenzuela, C., Liebling, S. L., Neilsen, D., Lehner, L., Caballero, O., O'Connor, E., & Anderson, M. 2015, arXiv preprint arXiv:1505.01607
- Palenzuela, C., Liebling, S. L., Neilsen, D., Lehner, L., Caballero, O. L., O'Connor, E., & Anderson, M. 2015, Phys. Rev. D, 92, 044045
- Pfeiffer, H. P. 2005, J. Hyperbol. Differ. Eq., 2, 497
- Pfeiffer, H. P., Brown, D. A., Kidder, L. E., Lindblom, L., Lovelace, G., & Scheel, M. A. 2007, Classical and Quantum Gravity, 24, S59
- Pfeiffer, H. P., Kidder, L. E., Scheel, M. A., & Teukolsky, S. A. 2003a, Computer physics communications, 152, 253
- . 2003b, Comput. Phys. Commun., 152, 253
- Pfeiffer, H. P., & York, J. W. 2003, Phys. Rev. D, 67, 044022
- Pfeiffer, H. P., & York Jr., J. W. 2005, Phys. Rev. Lett., 95, 091101
- Pian, E., et al. 2017, "Nature", 551, 67
- Pretorius, F. 2005a, Phys. Rev. Lett., 95, 121101
- . 2005b, Classical and Quantum Gravity, 22, 425
- Punturo, M., et al. 2010, Class. Quantum Grav., 27, 084007
- Qian, Y.-Z., & Woosley, S. 1996, The Astrophysical Journal, 471, 331
- Radice, D., Galeazzi, F., Lippuner, J., Roberts, L. F., Ott, C. D., & Rezzolla, L. 2016, arXiv preprint arXiv:1601.02426
- Radice, D., Perego, A., Hotokezaka, K., Fromm, S. A., Bernuzzi, S., & Roberts, L. F. 2018, The Astrophysical Journal, 869, 130

- Radice, D., Perego, A., & Zappa, F. 2017, arXiv preprint arXiv:1711.03647
- Radice, D., & Rezzolla, L. 2011, Phys. Rev. D, 84, 024010
- Raithel, C. A. 2019, arXiv preprint arXiv:1904.10002
- Read, J. S. 2008, Neutron stars in compact binary systems: From the equation of state to gravitational radiation (ProQuest)
- Read, J. S., Lackey, B. D., Owen, B. J., & Friedman, J. L. 2009, Phys. Rev. D, 79, 124032
- Reed, W., & Hill, T. 1973, Triangular mesh methods for the neutron transport equation, Conference: National topical meeting on mathematical models and computational techniques for analysis of nuclear systems, Ann Arbor, Michigan, USA, 8 Apr 1973 LA-UR-73-479, CONF-730414-2
- Rezzolla, L., & Zanotti, O. 2013, Relativistic hydrodynamics (Oxford University Press)
- Rosswog, S., Liebendoerfer, M., Thielemann, F., Davies, M., Benz, W., et al. 1999, Astron. Astrophys., 341, 499
- Rosswog, S., & Liebendörfer, M. 2003, Mon. Not. Roy. Astr. Soc., 342, 673
- Rosswog, S., Liebendörfer, M., Thielemann, F.-K., Davies, M., Benz, W., & Piran, T. 1998, Arxiv preprint astro-ph/9811367
- Ruffert, M., Janka, H.-T., & Schaefer, G. 1996, Astron. Astrophys., 311, 532
- Ruffert, M., Janka, H.-T., Takahashi, K., & Schaefer, G. 1996, arXiv preprint astro-ph/9606181
- Saad, Y. 2003, Iterative methods for sparse linear systems (Siam)
- Sachdev, S., et al. 2019, arXiv preprint arXiv:1901.08580
- Sampath, R. S., Adavani, S. S., Sundar, H., Lashuk, I., & Biros, G. 2008, in Proceedings of the 2008 ACM/IEEE conference on Supercomputing, IEEE Press, 18

- Scales, J. A. 1989, *Geophysical Journal International*, 97, 179
- Schaal, K., Bauer, A., Chandrashekhar, P., Pakmor, R., Klingenberg, C., & Springel, V. 2015, *Monthly Notices of the Royal Astronomical Society*, 453, 4278
- Schötzau, D., Schwab, C., Wihler, T., & Witz, M. 2014, *Exponential convergence of hp-DGFEM for elliptic problems in polyhedral domains* (Springer)
- Schötzau, D., & Zhu, L. 2009, *Applied Numerical Mathematics*, 59, 2236
- Sekiguchi, Y., Kiuchi, K., Kyutoku, K., & Shibata, M. 2011, *Phys. Rev. Lett.*, 107, 051102
- . 2015, *Physical Review D*, 91, 064059
- Sekiguchi, Y., Kiuchi, K., Kyutoku, K., Shibata, M., & Taniguchi, K. 2016a, arXiv preprint arXiv:1603.01918
- . 2016b, *Phys. Rev.*, D93, 124046
- Shang, Y. 2009, *Computers & Mathematics with Applications*, 57, 1369
- Sherwin, S., Kirby, R., Peiró, J., Taylor, R., & Zienkiewicz, O. 2006, *International journal for numerical methods in engineering*, 65, 752
- Shibata, M. 2015, *Numerical Relativity, 100 Years of General Relativity* (World Scientific Publishing Company)
- Shibata, M., Kiuchi, K., Sekiguchi, Y., & Suwa, Y. 2011, *Progress of Theoretical Physics*, 125, 1255
- Shibata, M., Kiuchi, K., Sekiguchi, Y.-i., & Suwa, Y. 2011, *Progress of Theoretical Physics*, 125, 1255
- Shibata, M., Kyutoku, K., Fujibayashi, S., Tanaka, M., Hotokezaka, K., Sekiguchi, Y., & Kiuchi, K. 2017, *GW170817: Modeling based on numerical relativity and its implications*, Tech. rep.

- Shibata, M., & Nakamura, T. 1995, Physical Review D, 52, 5428
- Shibata, M., & Sekiguchi, Y. 2012, Progress of theoretical physics, 127, 535
- Shibata, M., & Taniguchi, K. 2011, Living Rev. Relativity, 14
- Shibata, M., & Uryū, K. 2000, Physical Review D, 61, 064001
- . 2006, Physical Review D, 74, 121503
- Smartt, S. J., et al. 2017, "Nature", 551, 75
- Soares-Santos, M., et al. 2017, "Astrophys. J. Lett.", 848, L16
- Somiya, K. 2012, Classical and Quantum Gravity, 29, 124007
- Sperhake, U. 2014, arXiv preprint arXiv:1411.3997
- Stamm, B., & Wihler, T. 2010, Mathematics of Computation, 79, 2117
- Steiner, A. W., Lattimer, J. M., & Brown, E. F. 2013, ApJ Letters, 765, L5
- Stiller, J. 2017, in Spectral and High Order Methods for Partial Differential Equations ICOSA-HOM 2016 (Springer), 189–201
- Sundar, H., Biros, G., Burstedde, C., Rudi, J., Ghattas, O., & Stadler, G. 2012, in Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis, IEEE Computer Society Press, 43
- Symbalisty, E., & Schramm, D. N. 1982, Astrophysical Letters, 22, 143
- Szilágyi, B. 2014, International Journal of Modern Physics D, 23, 1430014
- Tacik, N., et al. 2015, Physical Review D, 92, 124012
- Takami, K., Rezzolla, L., & Baiotti, L. 2014, Physical review letters, 113, 091104

- Takami, K., Rezzolla, L., & Baiotti, L. 2015, *Phys. Rev. D*, 91, 064001
- Tanvir, N. R., et al. 2017, "Astrophys. J. Lett.", 848, L27
- Teukolsky, S. A. 2016, *J. Comput. Phys.*, 312, 333
- The LIGO Scientific Collaboration et al. 2018a, ArXiv e-prints
- . 2018b, ArXiv e-prints
- Thorne, K. S. 1981, *Mon. Not. Roy. Astr. Soc.*, 194, 439
- Tsokaros, A., Mundim, B. C., Galeazzi, F., Rezzolla, L., & Uryū, K. 2016, *Physical Review D*, 94, 044049
- Villar, V. A., et al. 2017, *Astrophys. J.*, 851, L21
- Walecka, J. 1974, *Annals of Physics*, 83, 491
- Wanajo, S., Sekiguchi, Y., Nishimura, N., Kiuchi, K., Kyutoku, K., & Shibata, M. 2014, *Astrophys.J.Lett.*, 789, L39
- Weisberg, J. M., Taylor, J. H., & Fowler, L. A. 1981, *Scientific American*, 245, 74
- York Jr, J. W. 1999, *Physical review letters*, 82, 1350
- Yu, Y.-W., Liu, L.-D., & Dai, Z.-G. 2018, *The Astrophysical Journal*, 861, 114
- Zanotti, O., Dumbser, M., Hidalgo, A., & Balsara, D. 2014, in *Astronomical Society of the Pacific Conference Series*, Vol. 488, 8th International Conference of Numerical Modeling of Space Plasma Flows (ASTRONUM 2013), ed. N. V. Pogorelov, E. Audit, & G. P. Zank, 285–290
- Zanotti, O., Fambri, F., & Dumbser, M. 2015, *Monthly Notices of the Royal Astronomical Society*, 452, 3010

Zhu, L., Giani, S., Houston, P., & Schötzau, D. 2011, Mathematical Models and Methods in Applied Sciences, 21, 267

Zumbusch, G. 2009, Class. Quantum Grav., 26, 175011