



Theses and Dissertations

2019-08-01

Dynamical Compact Objects in Numerical Relativity

Hyun Lim
Brigham Young University

Follow this and additional works at: <https://scholarsarchive.byu.edu/etd>



Part of the [Physical Sciences and Mathematics Commons](#)

BYU ScholarsArchive Citation

Lim, Hyun, "Dynamical Compact Objects in Numerical Relativity" (2019). *Theses and Dissertations*. 7729.
<https://scholarsarchive.byu.edu/etd/7729>

This Dissertation is brought to you for free and open access by BYU ScholarsArchive. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of BYU ScholarsArchive. For more information, please contact ellen_amatangelo@byu.edu.

Dynamical Compact Objects in Numerical Relativity

Hyun Lim

A dissertation submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Eric W. Hirschmann, Advisor
Gus L. Hart
David W. Neilsen
Mark Kenneth Transtrum
Jean-François S. Van Huele

Department of Physics and Astronomy

Brigham Young University

Copyright © 2019 Hyun Lim

All Rights Reserved

ABSTRACT

Dynamical Compact Objects in Numerical Relativity

Hyun Lim

Department of Physics and Astronomy, BYU

Doctor of Philosophy

The work of this dissertation will study various aspects of the dynamics of compact objects using numerical simulations. We consider BH dynamics within two modified or alternative theories of gravity. Within a family of Einstein-Maxwell-Dilaton-Axion theories, we find that the GW waveforms from binary black hole (BBH) mergers differ from the standard GW waveform prediction of GR for especially large axion values. For more astrophysically realistic (i.e. smaller) values, the differences become negligible and undetectable. We establish the existence of a well-posed initial value problem for a second alternative theory of gravity (quadratic gravity) and demonstrate in spherical symmetry that a linear instability is effectively removed on consideration of the full nonlinear theory. We describe the key components and development of a code for studying BBH mergers for which the mass ratio of the binaries is not close to one. Such intermediate mass ratio inspirals (IMRIs) are much more difficult to simulate and present greater demands on resolution, distributed computing, accuracy and efficiency. To this end, we present a highly-scalable framework that combines a parallel octree-refined adaptive mesh with a wavelet adaptive multiresolution approach. We give results for IMRIs with mass ratios up to 100:1. We study the ejecta from BNS in Newtonian gravity. Using smoothed particle hydrodynamics we develop and present the highly scalable F1eCSPH code to simulate such mergers. As part of the ejecta analysis, we consider these mergers and their aftermath as prime candidates for heavy element creation and calculate r -process nucleosynthesis within the post-merger ejecta. Lastly we consider a non-standard, yet increasingly explored, interaction between a BH and a NS that serves as a toy model for primordial black holes (PBH) and their possible role as dark matter candidates. We present results from a study of such systems in which a small BH forms at the center of a NS. Evolving the spherically symmetric system in full GR, we follow the complete dynamics as the small BH consumes the NS from within. Using numerical simulations, we examine the time scale for the NS to collapse into the PBH and show that essentially nothing remains behind. As a result, and in contradiction to other claims in the literature, we conclude that this is an unlikely site for ejecta and nucleosynthesis, at least in spherical symmetry.

Keywords: black holes, neutron stars, gravitational waves, modified theories of gravity, smoothed particle hydrodynamics, wavelet representation, nucleosynthesis

ACKNOWLEDGMENTS

I am very fortunate to have met and worked with many amazing people during my graduate studies. This has affected me a lot in both direct and indirect ways. Because this document carries my name, I feel that I need to express my gratitude to them.

First of all, I am deeply grateful to my supervisor, Eric Hirschmann. His guidance and support have played an important role for myself on the road to becoming a good scientist. He encouraged me to make connections outside of campus including summer research positions and visiting fellowships. These have provided me numerous opportunities to advance my knowledge. I am also very indebted to David Neilsen. I would like to say that he is my other supervisor during my studies. I have learned a tremendously large amount from him. Without these two, I could not have come this far.

Thanks, too, for Wes Even and Oleg Korobkin who provided me various opportunities to work on SPH and different astrophysics projects at LANL. Thanks to Chris Fryer, Sam Jones, Irina Sagert, and Jonas Lippuner for their encouragement during my time at LANL.

I also would like thank to William East. He mentored me during my stay at the Perimeter Institute for Theoretical physics (PI). He provided a great deal of inspirations on many different topics in strong field astrophysics/gravity. Thanks, too, to Luis Lehner and Pablo Bosch. I would like to thank Milinda Fernando and Hari Sundar for their efforts on the Dendro project.

I would like to thank Jung-Han Kimn, who was my former advisor at South Dakota State University, for supporting and encouraging me when I chose to change my field of subjects from math to physics.

I would like to thank many friends who have inspired me and with whom I shared good fellowships over many years. These include, but are not limited to: Jonah Miller, Julien Loiseau, Daniel George, Aaron Held, Ignacio Reyes, Micheal Albergo, Raimon Luna, Yongchul Lee, Jungwook Min, Doohwan Choi, Siyeon Choi, Kyoyeon Cho, Jaeyoung Moon, and Yongwoo Lee.

Lastly but not least, I wish to thank my parents, Gi-taeg Lim and Seong-ae So for their unconditional support and encouragement to study in my field and in the US.

Table of Contents

Table of Contents	iv
List of Tables	viii
List of Figures	ix
1 Introduction and Background	1
1.1 Introduction	1
1.2 Background	5
1.3 Summary of Dissertation	9
1.4 Conventions	11
2 Black Hole Dynamics in Modified Theories of Gravity	13
2.1 Introduction	13
2.2 Einstein-Maxwell-Dilaton-Axion Theory	16
2.2.1 Introduction	16
2.2.2 Equations	16
2.2.3 The 3+1 Decomposition for EMDA	18
2.2.4 The BSSN Formalism	24
2.2.5 Black Hole Initial Data for EMDA	28
2.2.6 Results	30
2.3 Black Hole Dynamics in Quadratic Gravity	37
2.3.1 Quadratic Gravity in the Strong-Gravity Regime	38
2.3.2 Quadratic Gravity from the Point of View of Effective Field Theory	40
2.3.3 A Well-posed Initial Value Problem for Quadratic Gravity	41
2.3.3.1 Equations of Motion	43
2.3.3.2 Reduction to a Second Order System	44
2.3.3.3 Diagonalization and a Quasilinear Hyperbolic System	45
2.3.4 Results	47
2.3.4.1 Black Hole Stability in Spherical Symmetry	47
2.4 Conclusions	49
3 Intermediate Mass Ratio Inspirals Using a Wavelet Adaptive Mesh Refinement Scheme	51
3.1 Introduction	51
3.2 Wavelet Adaptive Multiresolution Implementation	53

3.3	Computational Framework	56
3.3.1	Preliminaries	56
3.3.2	Octree Partitioning	57
3.3.3	Octree Construction and Refinement	57
3.3.4	2 : 1 Balancing	58
3.3.5	Meshing	59
3.3.6	The <i>unzip</i> and <i>zip</i> Operations	65
3.3.7	The <i>re-mesh</i> and <i>inter-grid transfer</i> Operations	67
3.3.8	Symbolic Interface and Code Generation	67
3.3.9	Putting Everything Together	71
3.4	Other Numerical Methods	71
3.5	Results	72
3.5.1	Meshing Performance	73
3.5.2	Correctness of the Code	76
3.5.3	Comparison with the Einstein Toolkit	76
3.5.4	Parallel Scalability	79
3.5.5	Large Mass Ratios	81
3.6	Conclusions	82
4	Ejecta Morphology from Binary Neutron Star Mergers	84
4.1	Introduction	84
4.2	Numerical Method and Implementation	86
4.2.1	Smoothed Particle Hydrodynamics	86
4.2.1.1	Kernels	90
4.2.2	Equations of State	94
4.2.2.1	Analytic EOS	94
4.2.2.2	Tabulated EOS	97
4.2.3	Initial Data	97
4.2.3.1	The Tolman-Oppenheimer-Volkoff Equation	98
4.2.3.2	The Lane-Emden Equation	100
4.2.4	Time Integration Scheme	102
4.2.4.1	Adaptive timestep	104
4.2.5	Computing Gravitational Forces	106
4.2.5.1	The Fast Multipole Method	106
4.2.6	Gravitational Waves	109
4.3	Results	114
4.3.1	Sod Shock Tube	115
4.3.2	Sedov Blast Wave	115
4.3.3	Fluid Flow Test	116
4.3.4	Binary Neutron Stars	120
4.4	Conclusion	124
5	Exotic Primordial Black Hole and Neutron Star “Binaries”	125
5.1	Introduction	125
5.2	Horizon Penetrating Coordinates	127
5.3	Theoretical Model	130

5.3.1	Choice of Gauge	134
5.3.1.1	Maximal Slicing	135
5.3.1.2	Spherical-BSSN formulation with 1+log and Γ driver	136
5.3.2	Initial Data	138
5.4	Results	139
5.5	Conclusions	141
6	Conclusion	143
Appendix A At Introduction to the Mathematical Formulation of Numerical Relativity		147
A.1	The 3+1 Decomposition	147
A.1.1	Foliation of Spacetime	147
A.1.2	The Extrinsic Curvature	151
A.1.3	The Equations of Gauss, Codazzi, and Ricci	154
A.1.4	The Constraints and Evolution Equations	161
A.1.5	The ADM Equations	168
A.2	Conformal Transformations	172
A.2.1	Conformal Transformation of the Spatial Metric	173
A.2.2	Conformal Transformation of the Extrinsic Curvature	178
A.3	Reformulating: The Evolution Equations	179
A.3.1	The BSSN Equations	180
A.4	The Generalized Harmonic Formulation	190
A.5	The 3+1 Decomposition for Other Fields	193
A.5.1	Decomposition of the dilaton field, ϕ	193
A.5.2	Decomposition of the axion field, κ	196
A.5.3	Decomposition of the electromagnetic field equation	197
Appendix B Black Hole Initial data for EMDA		201
B.1	The Garfinkle-Horowitz-Strominger Black Hole	202
B.2	The Horne-Horowitz-Kaluza-Klein Black Hole	207
B.3	The Rasheed-Larsen Black Hole	211
B.4	The Kerr-Sen Black Hole	217
Appendix C Sparse Wavelet Representation via Iterated Interpolating Wavelets		223
C.1	Interpolating Bases	223
C.2	An Interpolating Wavelet Basis	226
C.2.1	Computing Wavelet Coefficients	229
C.3	The Sparse Wavelet Representation	230
Appendix D FleCSPH Implementation		232
D.1	FleCSI Framework	232
D.2	Domain Decomposition	233
D.3	Hierarchical Trees	235
D.4	Tree Generation	236
D.5	Tree Search	237

D.6	Distribution Strategies	237
D.7	I/O	243
Appendix E Artifact Evaluations		245
E.1	Toolkits/Softwares	245
E.1.1	HAD	245
E.1.1.1	Running the Application	247
E.1.2	Dendro-GR	248
E.1.2.1	Getting Started: Running <code>bssnSolver</code>	250
E.1.2.2	Generating Your Own Parameters	252
E.1.2.3	Symbolic Interface and Code Generation	253
E.1.2.4	Profiling the Code	254
E.1.2.5	Accuracy of Stencil Operators	255
E.1.2.6	Accuracy of Symbolic Interface and Code Generation	255
E.1.2.7	Constraint Equations	256
E.1.2.8	Single Black Hole	257
E.1.2.8.1	Boosted Single Black Hole	258
E.1.2.9	Binary Black Holes	260
E.1.2.9.1	Mass ratio $q = 1$	260
E.1.2.9.2	Mass ratio $q = 10$	262
E.1.2.9.3	Mass ratio $q = 100$	262
E.1.3	F1eCSPH	264
E.1.3.1	Running F1eCSPH Applications	265
E.1.3.2	Description of Parameters	266
E.1.4	PAMR/AMRD lib	267
E.1.4.1	Parameter Generation	268
E.2	Visualization of Results	269
E.2.1	DV/xvs	269
E.2.1.1	Getting DV/xvs and Installation	269
E.2.1.2	Visualizing <code>sdf</code> Files	270
E.2.2	Paraview	273
E.2.2.1	Visualize Data using a Script	278
E.2.2.2	<code>pvbatch</code> in Marylou	279
E.2.2.3	Remote Visualization	280
Bibliography		283

List of Tables

3.1	A comparison study for the 2:1 balancing approach used in [84] and the new balancing approach on a single core in Stampede2 (SKX node), with varying input octree sizes ranging from 3K to 6M octants.	73
3.2	The number of cores required to maintain 24K unknowns per core with different mass ratios using <code>MAXDEPTH 12</code> , a wavelet tolerance of 10^{-4} and a black hole separation distance of 32.	79
4.1	Proportionality constants with respect to radius for NS.	102
C.1	The coefficients for four point interpolation	226
E.1	The normed difference between the numerical and analytic derivatives evaluated at grid points for the function $f(x,y,z) = \sin(2\pi x) \sin(2\pi y) \sin(2\pi z)$. In both cases, the wavelet tolerance is 10^{-8} while the maximum refinement level (i.e. <code>maxDepth</code>) has been increased from 4 to 6. Note that when <code>maxDepth</code> increases, the number of grid points increases. Hence, the normed difference goes down significantly.	255
E.2	Violation of constraint equations with time for an equal mass ratio binary merger simulation done using OT. Note that $\mathcal{H}, M1, M2, M3$ denotes the Hamiltonian and 3 momentum component constraints that is being monitored through the evolution.	257

List of Figures

1.1	Spectrograms and waveforms for the gravitational-wave transient catalog from the LIGO O1 and O2 runs. Image credit: [8]	2
2.1	Comparison between the analytic and numerical results for the axion field. The expression for the axion comes from Eqn. B.166 We extract the profile at late times when the solution has settled down to a stationary configuration.	30
2.2	The L_2 norm of the Hamiltonian and momentum constraints plotted over time for a single BH evolution. We use Kerr-Sen initial data for this test. The feature at early time is apparently due to truncation error from the numerical integration but it will be stabilized and down to small value at late time.	31
2.3	Evolution of the axion associated with a single BH for different α_1 parameter values. We track the maximum value of the axion with an initial amplitude of $\kappa_0 = 10^{-7}$ during the evolution. Note that the system settles down to a stable configuration at later times. The resulting profiles are rescaled assuming a linear increase with respect to the coupling parameter value $\alpha_1 = 100$. From the result, linear scaling fits well.	32

2.4	This figure illustrates tests of mass conservation. Different combinations of initial data and parameters are used for these tests. KS-BH stands for the Kerr-Sen BH initial data found in Appendix B.4 and GHS-BH stands for the Garfinkle-Horowitz-Strominger BH found in Appendix B.1. Different combinations of coupling parameters α_0 and α_1 are used to check the effects of these parameters.	34
2.5	GW waveforms for different theories, namely standard GR, EMD, and EMDA. We use non-spinning equal mass binary merger for all these tests.	35
2.6	GW waveforms in the EMDA theory with different initial axion field values. We use non-spinning equal mass binary merger with fixed coupling parameters, $\alpha_0 = \alpha_1 = 1$, electric charge $q_e = 10^{-3}$, and initial dilaton amplitude $\phi_0 = 10^{-4}$	36
2.7	Mass conservation test of Schwarzschild type BH. Initial mass of BH is $M = 0.5$. There is small oscillatory behavior at early time but it settles to stable configuration at late time.	48
2.8	Classical stability ranges for Schwarzschild and non-Schwarzschild type BHs. BH masses are plotted as a function of horizon radius r_0 . The figure is adapted from [61]	49
3.1	(left) A example of the adaptive mesh created by DENDRO for the binary black-hole system. (right) the hierarchical wavelet grids generated for the binary black hole system.	53

3.2 For a given function $f : V \rightarrow \mathcal{R}$ let $V_i \subset V$ be the finite dimensional approximation of f (see Figure 3.2a). As the number of nodes increases (i.e. going from V_i to V_{i+1}) for each additional node introduced, we compute wavelet coefficients based on the absolute difference between $f(V_{i,k})$ and the interpolated value from the previous level $f(V_{i-1,:})$ (see Figure 3.2b). Figure 3.2c illustrates those nodes that violate the specified wavelet tolerance ϵ and are subsequently stored as the sparse/wavelet representation of function f (see Figure 3.2d). 55

3.3 Bucketing each point and reordering the buckets based on the SFC ordering at each level l with top-down traversal. Each color-coded point is represented by its x and y coordinates. From the MSD-Radix perspective, we start with the most-significant bit for both the x and y coordinates and progressively bucket (order) the points based on these. The bits are colored based on the points and turn black as they get used to (partially) order the points. 58

3.4 A 2D example of *octant local nodes* (in the center) and *shared octant nodes* (the rightmost figure) nodal representation (with $d = 2$, where d denotes the order of FD computations) of the adaptive quadtree shown in the leftmost figure. Note that in *octant local nodes* representation nodes are local to each octant and contain duplicate nodes. By removing all the duplicate and hanging nodes by the rule of nodal ownership we get the *shared octant nodes* representation. Note that the nodes are color coded based on the octant level. 60

3.5 For a given ordered octree τ and a set of keys (leftmost figure), TREESEARCH performs the traversal in a top-down order over the set of keys, while flagging k_2, k_4, k_5 at the level 1 split, k_3 at level 2 split, and k_1 at level 3 split. 61

- 3.6 An example of a hanging face and a hanging edge where in both cases octant (■) has a hanging face (left figure) and a hanging edge (right figure) with octant (■). Nodes on the hanging face/edge are mapped to the larger octant and the hanging nodal values are obtained via interpolation. Note that for illustrative purposes, the two octants are drawn separately, but are contiguous. 64
- 3.7 A simplistic example of octree to block decomposition and the *unzip* operation. The leftmost figure shows the considered adaptive octree with *shared octant nodes* and its block decomposition is shown in the middle. Note that the given octree is decomposed into four regular blocks of different sizes. The rightmost figure shows the decomposed blocks padded with values coming from neighboring octants with interpolation if needed. In order to perform the *unzip* operation both O2O and O2N mappings are used. 66
- 3.8 The left panel shows the BSSN formulation of the Einstein equations. These are tensor equations, with indices i, j, \dots taking the values 1, 2, 3. On the right we show the `DENDRO_sym` code for these equations. `DENDRO_sym` uses SymPy and other tools to generate optimized C++ code to evaluate the equations. Note that \mathcal{L}_β , D , ∂ denote Lie derivative, covariant derivative and partial derivative respectively, and we have excluded $\partial_i \Gamma^i$ from `DENDRO_sym` to save space. (See [118, 119] for more information about the equations and the differential operators.) 69

3.9 Comparison between `EINSTEIN TOOLKIT` and `DENDRO-GR` without factoring in adaptivity (i.e. both `EINSTEIN TOOLKIT` and `DENDRO-GR` support uniform grids.). For a fixed tolerance, we expand the domain for a 1 : 1 mass-ratio simulation such that both `EINSTEIN TOOLKIT` and `DENDRO-GR` have roughly the same number of dofs. We present both weak and strong scaling results using both codes. On the left table are results from `DENDRO-GR` and from `EINSTEIN TOOLKIT` on the right. In the middle, we plot a representative strong and weak scaling curve for each code. The `DENDRO-GR` scaling is plotted in green (lighter shade for weak) and blue for `EINSTEIN TOOLKIT`. The corresponding data entries are also marked in the tables. Note that the rows represent strong scaling and the diagonal entries represent weak scaling results and runtime is reported in seconds(*s*). 74

3.10 Comparison of `std::bsearch` with partial ordering operator `<` and comparison free `TREESearch` approach for performing, varying number of keys on 33M sorted complete octree using single core in `Stampede2 SKX` node. 75

3.11 Comparison between `ET` and `DENDRO-GR` for number of spatial points with increasing mass ratios. Note that these are not from complete simulations and the size of the problem as well as the time per RK-step is likely to increase, but it illustrates the rate of increase for both approaches. Parameters for the above experiment generated such that total mass of black holes equals to 1 and the separation distance is 32 for all cases and `MAXDEPTH` is set in a way that the spatial discretization $dx < \frac{\min(m_1, m_2)}{16}$ where m_1, m_2 denotes the individual masses of black holes. 75

3.12 Weak scaling results in ORNL’s Titan for $RK/(dof/p)$ (averaged over 10 steps) where RK, dof, p denotes the time for single RK step, degrees of freedom, and number of cores respectively, with derivative computation (`deriv`), right hand side (`rhs`) computation, unzip cost, wavelet computation (`wavelets`) and communication cost (`comm`) with the average of 1.41M unknowns per core where the number of cores ranging from 32 to 131,072 cores on 8,192 nodes where the largest problem having 206 Billion unknowns. Above results are generated with mass ratio $\mu = 10$ with `MAXDEPTH 18` and wavelet tolerance of 10^{-6} . Note that the unknowns per core have a slight variation since with WAMR we do not have explicit control over the grid size and WAMR decides the refinement region on the mesh based on the how wavelets behave during the time evolution. This is why we have reported normalized RK with dof/p metrics to report accurate weak scaling results. 77

3.13 Strong scaling results in ORNL’s Titan for a single RK step (averaged over 10 steps) with derivative computation (`deriv`), right hand side (`rhs`) computation, unzip cost and communication cost (`comm`) for a fixed problem size of 10.5B unknowns where the number of cores ranging from 4,096 to 65,536 cores on 4096 nodes. Note that for strong scaling results re-meshing is disabled in order to keep the problem size fixed. 78

3.14 An illustrative single core example to evaluate the overhead of zip/unzip operations to evaluate BSSN equations on a sequence of octree grids over 10 timesteps. The parameter α here denotes the ratio between the number of octants to the number of regular grid octants for MAXDEPTH 8. Hence moving towards the right direction on the x axis the octree grids converge towards a regular grid. RG and $RG + zip + unzip$ denotes the baseline performance for regular grid computations and regular grid computation with zip/unzip overhead respectively. The shaded $q = 1$ region denotes the α value for an equal mass ratio simulation until the merger event using MAXDEPTH 12 over 2×10^5 timesteps. For larger mass ratio runs the MAXDEPTH will be determined by the smaller black hole, hence for $q = 10$ with 15, α reached a maximum value of 1.4×10^{-8} over 6000 time steps and for $q = 100$ with MAXDEPTH 20, the maximum value of α reached 5.51×10^{-13} over 1000 timesteps. Note that the value of α can increase during the simulation. For the largest problem (see Figure 3.12) that was run on Titan with 131K cores for $q = 10$ case the α value was 4.8×10^{-7} . In the plot, we have marked the L_k values for $k = 3, 4, 5$ where L_k is the computed α ratio for an adaptive octree for which an equal number of octants spans across k levels. 80

3.15 Time step snapshots of the binary black hole problem of black hole mass ratios 1, 10 and 100 where we in the top row we plot the BSSN variable χ in the lower row we plot the WAMR grids for each case at that specific instance. 82

4.1 SPH kernel W and smoothing length h representation 88

4.2 Fast multipole method scheme. This figure presents each steps of scheme: Particle to Multipole (P2M), Multipole to Multipole (M2M), Multipole to Particles (M2P), Multipole to Local (M2L), Local to Local (L2L), and Particles to Particles (P2P). This schematic is inspired from [171] 106

4.3	Scheme of MAC for FMM	108
4.4	Sod shock tube with FleCSPH	116
4.5	Sedov Blast Wave with FleCSPH at respectively $t = 0.01$, $t = 0.03$, $t = 0.06$ and $t = 0.1$	117
4.6	The representation of dummy particle. The gray filled circle indicates dummy particles that consist a boundary and white circle represent fluid particles	118
4.7	Fluid flow simulation, the dam break, for different time, $t = 0$, $t = 0.4$, $t = 0.8$, and $t = 1$ seconds. The boundary conditions are based on dummy particles method	119
4.8	Efficiency on strong scaling	119
4.9	An equal-mass binary merger with 100,000 particles. The initial stars are generated by solving TOV equations (Sec. 4.2.3.1). The stars are sufficiently close that they merge via dynamical friction. The frames are at times (going from left to right) 9.8 ms, 12.0 ms, 16.0 ms, and 20.0 ms. Distances are in kilometers. The colormap shows density, integrated along the \hat{z} -axis in units of 10^{17} g/cm ² . The color bar is in a log scale. Images made with Splash [183].	120
4.10	Particle representation for equal mass binary merger with 40,000 particles. This simulation is done by FleCSPH. The initial data for the neutron stars is generated by solving the Lane-Emden equation (Sec. 4.2.3.2).	121
4.11	The morphology of the ejecta from the merger shown in figure 4.9. The time shown is 20 ms. Distances are in kilometers. The colormap shows density, integrated along the \hat{z} -axis in units of 10^{17} g/cm ² . The color bar is in a log scale. Images made with Splash [183].	122
4.12	Traces of the trajectories of particles in the ejecta. The color indicates time—bluer is earlier and red is later.	123

4.13	r-process residuals. We plot atomic abundance as a function of atomic mass. The green dots are observed solar abundances and the purple curve is as predicted by our ejecta.	123
5.1	Description for generation of ID. Small mass BH is created in the center of TOV star by imposing large ingoing velocity that occurs critical collapse .	139
5.2	Different collapse times with respect to different initial sizes of the BH. The mass of the TOV star is $1M_{\odot}$ and x is the normalized radius for the BH. We set the star's radius R_{star} to be one.	140
D.1	Morton and Hilbert space filling curves	234
D.2	Quadtree, space and data representation	236
D.3	Binaries tree for a 2 processes system. Exclusive, Shared and Ghosts particles resp. red, blue, green.	242
E.1	A single black hole boosted in the x -direction, with MAXDEPTH=12 and wavelet tolerance of 10^{-3} . Time is given in terms of the black hole mass, M .	259
E.2	L_2 norm of the Hamiltonian and momentum constraint violation as hyperbolic equations are evolved in time for equal mass ratio binary black hole configuration test run performed using DENDRO-GR. Note that the final spike in constraint violation happens during the merging event of the binary black holes.	260
E.3	This figure shows frames from the evolution of a black hole binary with an equal mass ratio, $q = 1$. Time is measured in terms of the total black hole mass M	261
E.4	This figure shows frames from the evolution of a black hole binary with mass ratio $q = 10$. Time is measured in terms of the total black hole mass M .	263
E.5	This figure shows frames from the evolution of a black hole binary with mass ratio $q = 100$. Time is given in terms of the total mass M	264

Chapter 1

Introduction and Background

1.1 Introduction

General Relativity (GR) is an extraordinarily successful theory. Not only has it radically modified our understanding of gravity, space and time, but it possesses enormous predictive power that has been confirmed in a number of rather dramatic ways. Among its predictions, GR describes exotic objects such as neutron stars and black holes as well as the dynamical, expanding, cosmological model of the universe that we call the Big Bang.

Significantly, and crucially for the current work, it also predicts the existence of gravitational waves (GW), the propagation of energy via the gravitational interaction. Indeed, it is fair to say that gravitational waves are among the most important physical predictions associated with strong, dynamical gravitational fields. As such, they are of great interest in both astronomy and astrophysics. Understanding their sources and the means of their production is crucial. While many sources of gravitational waves exist, among the strongest are binary systems near merger that are composed of compact objects such as black holes or neutron stars.

Compact objects are the endstates of stellar evolution, and are often referred to as stellar remnants. The nature of these remnants depends primarily on the mass of the original, progenitor star from which it was formed. All such objects have a large mass relative

to their size, thereby giving them a very large density as well. Examples of well-known compact objects are black holes (BHs) and neutron stars (NSs). In addition to these, there may exist other, possibly very dense, bodies such as quark stars, boson stars, or gravastars. These more exotic objects have not been detected.

In an extremely significant discovery, Advanced LIGO detected gravitational waves from a binary black hole (BBH) merger [1] on September 14, 2015. Since this original detection, there have been nine more GW detections from BBH mergers [2, 3, 4, 5, 6] and one from a binary neutron star (BNS) merger [7]. Fig 1.1 shows the waveforms from all confirmed detections during LIGO's O1 and O2 runs. As of this writing (9 July 2019), 18 additional candidate detections have been announced as part of the third observing (O3) run¹ which began 1 April 2019. This amounts to an average detection rate of more than one per week.

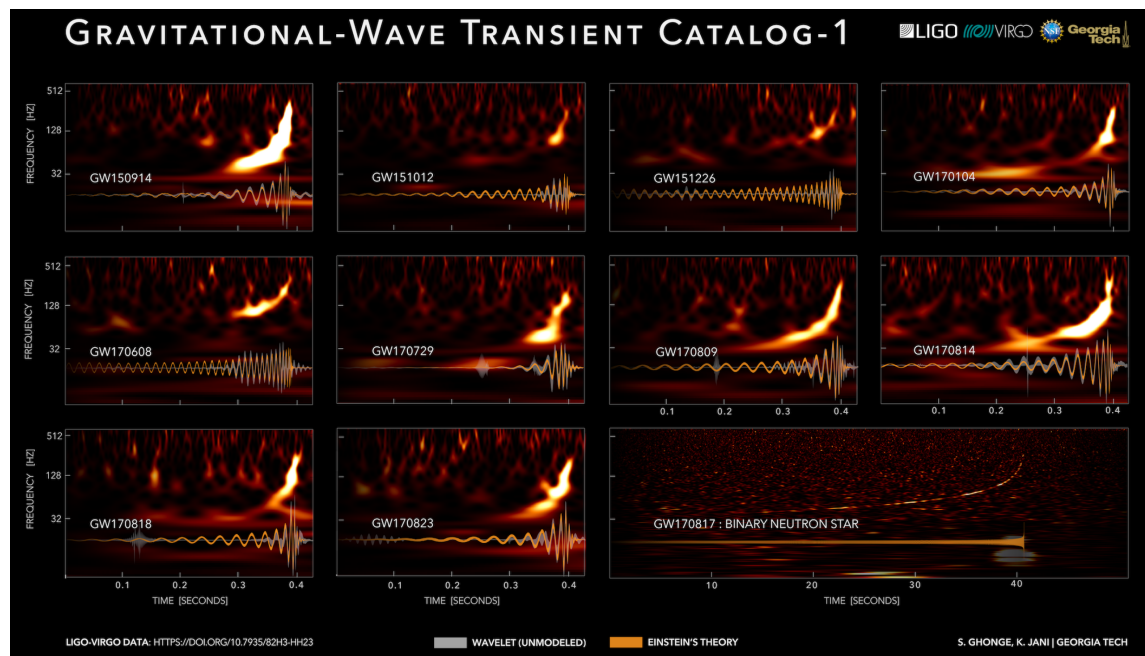


Figure 1.1 Spectrograms and waveforms for the gravitational-wave transient catalog from the LIGO O1 and O2 runs. Image credit: [8]

One aspect of the LIGO/VIRGO discoveries that is important to recognize is that the expected GW signals are so weak that even with the amazing sensitivity of these detectors it

¹<https://gracedb.ligo.org/latest/>

is necessary to extract the signals from the background noise. As it is much easier to extract a signal if we know what to look for, detailed and accurate calculations of the dynamics of these objects are needed in order to provide templates of expected GW signals. An understanding of these sources and their generated waveforms can be gained by solving the equations of general relativity. Such studies can provide valuable information on the design and tuning of future advanced GW detectors as well as the analysis of the received signals from those sources.

While recent detections have been direct measurements of GW, we have had evidence for them indirectly for a number of decades. In fact, in 1974, Taylor and Hulse discovered the first binary pulsar, referred to as *PSRB1913 + 16*, and thereby confirmed the general relativistic prediction of energy outflow (gravitational waves) from this system. A binary pulsar consists of two neutron stars orbiting around their center of mass. Such a system is expected in GR to decay over time because massive objects emit gravitational waves and lose energy. Exactly this prediction was observed in *PSRB1913 + 16* [9]. Today, several BNS systems are known. Although BH-NS binaries have not been detected to date, BH-NS binaries are expected to form through mechanisms analogous to BNS binaries. Detecting both will require templates of the GW signals these binaries can emit.

Due to the complexity of the two-body problem in general relativity, it is essentially impossible to obtain analytic solutions to the equations of GR for such binary systems. The Einstein equations are a system of ten coupled, nonlinear, partial differential equations (PDEs) in 3 (space) + 1 (time) dimensions. Exact solutions of the Einstein equations are only known in cases with high symmetry such as axisymmetry or for stationary systems. If we are interested in studying systems with astrophysical relevance, numerical approaches are required.

Many studies have considered and performed a variety of numerical simulations of binary systems in a variety of different environments. We have no hope of referencing all of the relevant work. However, we do mention a few review articles that may provide an entry point into some of this literature. The article [10] reviews initial data for numerical

relativity, while [11] surveys calculations of rotating stars in relativity, [12] provides an introduction to numerical hydrodynamics in special relativity, [13] presents some of the considerations needed to incorporate hydrodynamics and magnetohydrodynamics into general relativity, [14] considers issues in simulating BH-NS binary mergers, and [15] reviews BNS mergers.

Despite considerable progress in numerical relativity over several decades, a number of interesting problems are only now beginning to be addressed, including the structure of accretion disks around black holes, the amplification of magnetic field strength in hypermassive neutron stars, the structure and strength of magnetic fields in neutron stars after merger, and the variety of effects that can arise due to different theoretical equations of state (EOS) for neutron stars. Studying binary systems can help address a number of highly energetic and still somewhat mysterious systems in the universe.

Successful simulations of binary mergers in which gravitational wave emission can be correctly calculated require substantial resolution. However, obtaining such resolution still taxes the largest supercomputers in terms of time and memory. Various techniques have been developed over the years to obtain the required resolution while simultaneously attempting to shorten run times. Two such examples include adaptive mesh refinement (AMR) and parallel computing. AMR adapts the computational domain of a problem based on the local error of a computed solution and the fine features that may develop in a simulation. Additional resolution is added where numerical precision is most needed, and additional grid points are added to such regions dynamically. In the work to be described here, we will use a sparse wavelet representation to obtain this adaptivity.

Parallel computing entails partitioning the numerical work one needs to accomplish over many computer processors or cores. This can reduce computation time significantly. Message Passing Interface (MPI) and graphical processing units (GPU) are example toolkits for parallel computation. In the work described in this dissertation, we apply both AMR and parallel computation in various guises to the problem of simulating binary systems.

In the next section, we will review some of the basic theoretical underpinnings of the

work represented in this dissertation.

1.2 Background

As mentioned in the previous section, we would like to study the dynamics of compact objects such as BHs and NSs, particularly as they appear in merging binaries and thereby serve as sources for the production of gravitational radiation. These systems can be described by the Einstein equations. At its most basic level, the Einstein equations relate (curvature) tensors involving second derivatives of the metric tensor to the energy momentum tensor.²

The usual form of the Einstein equations are

$$G_{ab} = 8\pi T_{ab}, \quad (1.1)$$

where G_{ab} is the Einstein curvature tensor, and T_{ab} is the energy-momentum tensor. Since the energy-momentum tensor describes the matter distribution of a system, the vacuum Einstein equations are completely sufficient and appropriate to describe the dynamics of black hole spacetimes. The vacuum Einstein equations are then

$$G_{ab} = 0. \quad (1.2)$$

If we try to unpack this little bit, we recall that the Einstein curvature tensor G_{ab} is a symmetric tensor defined in terms of the Ricci curvature tensor R_{ab} , the metric tensor g_{ab} , and the Ricci curvature scalar R in the following combination:

$$G_{ab} = R_{ab} - \frac{1}{2}g_{ab}R. \quad (1.3)$$

Each of these curvature tensors are defined in terms of contractions of the Riemann curvature tensor, R_{abcd} . In particular, the Ricci tensor is formed from a single contraction

$$R_{bd} = g^{ac}R_{abcd}, \quad (1.4)$$

²In this section, all tensor objects are defined on a four dimensional, spacetime manifold with Lorentzian signature.

while the Ricci scalar results from a second contraction: $R = g^{bd}R_{bd}$. The Riemann curvature tensor is a combination of derivatives of the metric. It is most often written in the following form:

$$R^a{}_{bcd} = \partial_c \Gamma^a{}_{bd} - \partial_d \Gamma^a{}_{bc} + \Gamma^a{}_{ec} \Gamma^e{}_{bd} - \Gamma^a{}_{ed} \Gamma^e{}_{bc}, \quad (1.5)$$

where $\Gamma^a{}_{bc}$ signify the Christoffel symbols that are, themselves, written as

$$\Gamma^a{}_{bc} = \frac{1}{2} g^{ad} (\partial_b g_{dc} + \partial_c g_{bd} - \partial_d g_{bc}). \quad (1.6)$$

Referring to the Einstein equations, we note that these form a system of second order partial differential equations (PDEs). While not immediately obvious in this covariant notation, further analysis will show that they are a combined hyperbolic-elliptic system of PDEs. In particular, it can be shown that these equations can be rewritten in a form such that the dynamical degrees of freedom can be propagated via hyperbolic, or wave, equations, while a subset of the elliptic equations determine a set of constraints that any valid solution must obey. The numerical solution of the Einstein equations usually consists of solving the constraint equations for a particular set of initial data which describe some gravitationally valid configuration.

The description of the previous paragraph is a bit abstract, devoid of mathematical justification, and perhaps hard to understand. Fortunately, we have an analogy with the simpler system of Maxwell's electrodynamics. Recall electromagnetism as described by Maxwell's equations, rewritten perhaps in a slightly less familiar form:

$$\nabla \cdot \mathbf{E} = 4\pi\rho, \quad (1.7)$$

$$\nabla \cdot \mathbf{B} = 0, \quad (1.8)$$

$$\frac{\partial \mathbf{E}}{\partial t} = \nabla \times \mathbf{B} - 4\pi\mathbf{J}, \quad (1.9)$$

$$\frac{\partial \mathbf{B}}{\partial t} = -\nabla \times \mathbf{E}. \quad (1.10)$$

Of course, \mathbf{E} and \mathbf{B} are the electric and magnetic fields, ρ is the charge density, \mathbf{J} is the current density, and ∇ denotes a spatial derivative operator. (We can slip back into a

special relativistic, flat space formulation – it doesn't really matter for this discussion.) The important point that we want to make here and that provides our analogy to GR is that these equations can be thought of as splitting into two groups of equations. Note that the first group can be written as

$$C_E \equiv \nabla \cdot \mathbf{E} - 4\pi\rho = 0, \quad (1.11)$$

$$C_B \equiv \nabla \cdot \mathbf{B} = 0. \quad (1.12)$$

These equations contain only spatial derivatives of the electric and magnetic fields. These equations must hold at each instant of time independent of the subsequent evolution of the fields. Therefore, they constrain any possible configuration of the fields for all time, including the initial time. Hence they tell us what a good initial configuration of electric and magnetic fields might look like given an arrangement of charge and current densities. We refer to these as the constraint equations.

The second group of equations are, of course, the time evolution equations. These are again

$$\frac{\partial \mathbf{E}}{\partial t} = \nabla \times \mathbf{B} - 4\pi\mathbf{J}, \quad (1.13)$$

$$\frac{\partial \mathbf{B}}{\partial t} = -\nabla \times \mathbf{E}. \quad (1.14)$$

These equations describe how the electric and magnetic fields evolve forward in time once a good initial configuration, or good initial data, is specified.

This same splitting of the equations of electromagnetism into time evolution, or hyperbolic, equations and constraint, or elliptic, equations happens with regard to the Einstein equations. Appendix A describes some of the details surrounding this division of the Einstein equations into a set of constraint and evolution equations.

In order to study the evolution in time of a gravitating system, or any physical system for that matter, it is important that we are able to formulate the system in such a way that the evolution has a well posed initial value problem. Sometimes this is referred to as the Cauchy problem for a physical system. With regard to the Einstein equations, we need to be

able to describe the hyperbolic subset of the equations in such a way that we can prescribe valid initial data and the evolution of the gravitational field is guaranteed to exist, be unique, and to be stable (e.g. it does not blow up) arbitrarily far into the future. Setting up the Einstein equations in such a way is fundamental to being able to solve the system through numerical simulation. There are a variety of different formalisms relative to how this specific separation is carried out. In our work, we will concentrate on the 3 + 1 formalism (see Appendix A for a detailed description), in which one splits, or foliates, spacetime into three-dimensional spatial slices with each being parametrized by a quantity labeling the time. There are multiple ways to formulate such a 3 + 1 decomposition. Our work will use primarily the Baumgarte-Shapiro-Shibata-Nakamura (BSSN) [16, 17] formalism though we also considered the CCZ4 formalism [18].

For non-vacuum spacetimes, the energy-momentum tensor contributes matter source terms in the 3 + 1 field equations. The energy-momentum tensor includes all the sources of energy and momentum in our spacetime, excluding gravity. Therefore, it arises from all forms of matter: electromagnetic fields, neutrinos, scalar fields etc. To study the dynamics of matter sources, we also need to solve the evolution equations for possible matter sources. The evolution equations for matter sources will then be given by

$$\nabla_a T^{ab} = 0. \tag{1.15}$$

These conservation equations must be solved together with the evolution equations for the gravitational field. Some of the quantities in the energy-momentum tensor may require additional equations. For example, when considering the case of hydrodynamic matter, continuity equations and equations of state will be required to complete the fluid system.

In certain chapters of this dissertation, we will be interested in systems that will include gravitating fluids. Both relativistic and nonrelativistic fluids are used to model a variety of systems in high-energy astrophysics, such as neutron stars, accretion onto compact objects, supernovae, and gamma-ray burst outflows. Consequently, methods to solve the relevant hydrodynamic equations are a key scientific kernel in a large number of astrophysics

simulations and toolkits. Specifically, we will be interested in general relativity coupled to a perfect fluid.

There are a number of ways to describe this, but we will concentrate on one that allows us to use a conservative numerical scheme for the matter part. For this, we assume a energy-momentum tensor of the form

$$T_{ab} = [\rho_0(1 + \varepsilon) + P]u_a u_b + P g_{ab}, \quad (1.16)$$

where ρ_0 is defined as the rest energy density of the fluid, ε is the internal energy of the fluid, u^a is its 4-velocity and P is its pressure. We can include electromagnetic terms in the above energy-momentum tensor to generalize our study to that of magnetohydrodynamics (MHD):

$$T_{ab} = [\rho_0(1 + \varepsilon) + P]u_a u_b + P g_{ab} + F_{ac} F_b^c - \frac{1}{4} g_{ab} F_{cd} F^{cd}, \quad (1.17)$$

where the $F_{ac} F_b^c - \frac{1}{4} g_{ab} F_{cd} F^{cd}$ terms are the electromagnetic parts defined by the Maxwell tensor F_{ab} . On including this model for the matter, we can begin to simulate a variety of different astrophysical problems associated with compact objects, their dynamics and their gravitational wave emission.

Ultimately, solving the combined set of the Einstein equations and the relativistic magnetohydrodynamics equations numerically provides a route to laying the theoretical and computational groundwork for eventually doing multimessenger astronomy from binary compact objects.

1.3 Summary of Dissertation

In this dissertation, we will explore some of the dynamics of BHs and NSs. Our studies will include these objects both singly and in binaries. The dissertation is organized as follow: Chapter 2 describes BH dynamics in two modified theories of gravity, namely Einstein-Maxwell-Dilaton-Axion (EMDA) theory and quadratic gravity (QG). Chapter 3 discusses the motivation for and our efforts towards developing an infrastructure capable

of performing massively parallel simulations of intermediate mass ratio inspirals using a recently developed variety of mesh adaptivity. Chapter 4 changes tack a bit while presenting the development of and results from a code to simulate binary neutron stars and their ejecta using smoothed particle hydrodynamics. Chapter 5 considers a different version of a BHNS binary, namely a hypothesized interaction of a primordial black hole with a neutron star in which the BH sits at the center of a NS. Chapter 6 will provide a conclusion, tie some of our strands together and provide an outlook for possible future research directions.

Each chapter in this dissertation represents work that has been published or is currently in preparation to be submitted as journal articles. The work in Chap. 2 represents manuscripts that are being prepared for submission (both the EMDA [19] and QG [20] work). The work in Chap. 3 has been published in the SIAM Scientific Computing Journal [21]. The work in Chap. 4 represents one published article as well as a second that is in the process of submission. The published article was published in HPCS 2018 [22] and represents the computer science aspect of the research. A follow-up emphasizing the physics aspect is currently under internal review at LANL [23]. The work in Chap. 5 is being prepared for submission to a journal [24].

Of course each project described in this dissertation is a collaborative effort and various aspects of each project was worked on by, sometimes, a variety of collaborators. The EMDA project (Sec. 2.2) was done in collaboration with Eric Hirschmann (BYU) with input from Luis Lehner (Perimeter Institute), Steve Liebling (Long Island University), and Carlos Palenzuela (Universitat de Illes Balears). The quadratic gravity project (Sec. 2.3) is a collaboration with Aaron Held (Universität Heidelberg). The work described in Chap 3) originates in a collaboration with Milinda Fernando (University of Utah), Hari Sundar (University of Utah), David Neilsen (BYU), and Eric Hirschmann (BYU). The work on binary neutron star systems in Chap. 4 was done in collaboration with several people at LANL. These include Julien Loiseau, Oleg Korobkin, Jonah Miller, Irina Sargert, Wes Even, and Ben Bergen. The work in Chap. 5 was begun at Perimeter Institute in collaboration with William East (Perimeter Institute) and continued on my return to BYU.

1.4 Conventions

In this work, we follow some fairly standard conventions:

- We use geometrized units such that $c = G = 1$
- Lengths and times are given in units of the mass M :
 - Distance : $d = M \Rightarrow d = GM/c^2$
 - Time : $t = M \Rightarrow t = GM/c^3$
- The metric signature is mostly plus: $(-, +, +, +)$
- The standard Einstein summation convention will always hold unless explicitly stated otherwise:

$$\sum_{a=1}^N u^a g_a = u^a g_a$$

(where N is the number of dimensions) is used

- Indices a, b, c, \dots, h and o, p, q, \dots will be understood to run over spacetime indices, while i, j, k, \dots, n will run over spatial indices only (Sometimes referred to as the Fortran convention)
- The flat spacetime (or purely spatial) metric will be represented by η_{ab} (or η_{ij}) in any coordinate system. For example, we have $\eta_{ab} = \text{diag}(-1, 1, 1, 1)$ in Cartesian (inertial) coordinates
- In general, we refer to objects associated with
 - the spacetime manifold \mathcal{M} as g_{ab} , ${}^{(4)}\Gamma_{bc}^a$, ∇_a , ${}^{(4)}R_{ab}$, etc.
 - a spatial slice, Σ , as γ_{ij} , Γ_{jk}^i , D_i , R_{ij} , etc.
 - a conformally related space as $\bar{\gamma}_{ij}$, $\bar{\Gamma}_{jk}^i$, \bar{D}_i , \bar{R}_{ij} , etc.
- The symmetric and antisymmetric parts of a tensor are defined

$$T_{(ab)} \equiv \frac{1}{2}(T_{ab} + T_{ba}) \quad \text{and} \quad T_{[ab]} \equiv \frac{1}{2}(T_{ab} - T_{ba})$$

-
- Partial derivatives are designated as: $\partial_i g$ or $g_{,i}$
 - Covariant derivatives are designated as: $\nabla_a f_b$ or $f_{b;a}$
 - The Lie derivative of a tensor T^{ab} along the vector X^c is designated as: $\mathcal{L}_X T^{ab}$

Chapter 2

Black Hole Dynamics in Modified Theories of Gravity

2.1 Introduction

For over a century, Einstein's theory of general relativity (GR) has been considered as the consensus best theory of gravity. This consensus is not without reason. To date, all observations and experiments have lent increasing support to GR within the classical weak field regime. These include tests such as the perihelion of Mercury, the bending of starlight around the Sun, as well as the loss of orbital energy to gravitational waves in binary pulsar systems. As a result, one may reasonably ask why a lot of time and effort has been spent on studying possible alternative or modified theories of gravity.

One reason is that GR is nonrenormalizable in the ultraviolet limit (i.e. at short distances and high energies) and hence cannot be extended to a quantum theory [25]. Further, when GR is treated as a complete theory, it leads to predictions of singularities in two important scenarios. One is in the formation of a BH with a singularity at its center. Another relates to the singularity at the beginning of the universe, known as the Big Bang. These singularities are expected to be unphysical, yet they occur in situations that are important in understanding the universe. Therefore, they have been the focus of many investigations into

GR and possible modifications. Indeed, many alternative theories of gravity are motivated by the effort to resolve these singularities and the problems associated with them. Of course, there are other open questions in gravitational physics and cosmology including the nature of dark energy, dark matter and the possible unification of the forces. Even such ideas may ultimately be only approximations to more fundamental theories such as string theory [26].

A related reason to considering alternative and modified theories of gravity, and this is a main motivation for this work, is that GR may break down in the regime of very strong gravitational fields [27]. As already mentioned, GR has been tested in weak-field observations within our solar system and we know GR works well in this regime. However, in the highly nonlinear and strongly dynamical regime such as the merger of two black holes or two neutron stars, GR needs additional testing. Before 2015, it was almost impossible to test GR in the strong field limit. However, with the advent of gravitational wave astronomy, our interest in gravitational radiation shifts from the effort to detect this radiation to using observatories such as LIGO/VIRGO to begin to probe not only the physics and predictions of GR but to explore physics beyond the standard model of gravity which we might take GR to now be. Observing GWs can tell us more than just the existence of black holes in binary systems. We are now in the position of being able to use GW observations to test the fundamental nature of gravity and whether GR describes it entirely.

With the detection of several recent black hole merger events [1, 2, 3, 4, 5], there is already a considerable and growing literature [28, 29] which discusses the theoretical implications of GW observations. Therefore, accurate predictions of possible future GW signals are important to test alternative theories of GR and quantum gravity as well as to help in future detections and next generation detectors. These tests may possibly indicate that nature deviates from GR.

Since modified and alternative theories of gravity have been developed for a variety of reasons, one might expect that there exist a wide range of these theories. In general, we can categorize these theories by which fundamental aspects of GR they violate [29, 30]. One example is a class of theories that admit massive gravitons. Such theories allow for

a massive force carrier for gravity as opposed to a massless one as assumed in GR [31]. Other possibilities include alternatives which originate from higher dimensional variants such as string theory as well as theories which invoke a holographic type principle akin to the AdS/CFT conjecture or the gauge/gravity correspondence [32]. Suffice it to say, there are many ideas and alternative gravity theories built out of them. Each is interesting in its own right, but there are far too many to be able to explore them in detail in this work.

In this chapter, we restrict ourselves to studying black hole dynamics within two alternative theories. One is Einstein-Maxwell-Dilaton Axion (EMDA) theory and second is quadratic gravity. For EMDA theory, we study the behavior of different kinds of BHs in both single and binary form. We also calculate the GW signal from BBH mergers in this theory and compare with the standard GR result. For quadratic gravity, while our ultimate goal may be to do the same, because it is a less well developed and understood theory, for the present work we establish a way to obtain a well-posed initial value problem in higher derivative gravity and investigate a restricted version of nonlinear stability within this theory.

As mentioned at the end of Chapter 1, the work described here has been collaborative. The EMDA project was done with Eric Hirschmann with input from Luis Lehner, Steve Libeling, and Carlos Palenzuela. We used the long established and verified HAD as our computational infrastructure. This gave us the ability to include AMR, the BSSN formulation, as well as standard numerical finite difference algorithms. (See Appendix E.1.1 for detailed information about HAD.) My contribution to this work included deriving the equations of motion associated with EMDA theory, writing the code for this system into HAD for numerical simulations, and performing all the tests and simulations that are presented in this section.

The quadratic gravity (QG) project was a collaboration with Aaron Held. My contribution to this project included deriving and proving that a well-posed initial value formulation for QG exists. I also wrote the unigrid code to simulate the evolution system in spherical symmetry. As the unigrid code does not allow for sufficient resolution in many cases, I also

adapted the code for use with the PAMR [33] toolkit to allow for parallel AMR functionalities. (See Appendix E.1.4 for detailed information about PAMR.)

2.2 Einstein-Maxwell-Dilaton-Axion Theory

2.2.1 Introduction

In this section, we will study black hole systems within EMDA theory. This is extended work based on [34] which studies black hole dynamics in the Einstein-Maxwell-Dilaton (EMD) theory. This theory originates in the low energy limit of the bosonic sector of heterotic string theory. This theory allows for black holes that can have mass, angular momentum, charge and scalar hair together with scalar, vector, and tensor radiation channels. Further, its mathematical structure guarantees the definition of a well-posed initial value problem. Therefore, EMDA offers an interesting theoretical and computational model to explore possible deviations from the “standard model” GR provides.

2.2.2 Equations

We will begin by sketching a derivation of the equations of motion for EMDA. The action for Einstein-Maxwell-Dilaton-Axion (EMDA) is (in the Einstein frame)

$$S = \int d^4x \sqrt{-g} \left[R - 2(\nabla\phi)^2 - e^{-2\alpha_0\phi} F^2 - \frac{1}{2} e^{4\alpha_1\phi} (\nabla\kappa)^2 - \kappa F_{ab} (*F)^{ab} \right], \quad (2.1)$$

where ϕ is the dilaton field, κ is the axion field, and $*F$ indicates the dual of the Maxwell stress tensor. We include the parameters α_0 and α_1 in order to parametrize a broader family of theories. The strict low energy string theory corresponds to $\alpha_0 = \alpha_1 = 1$. Note that if the parameters are zero, we revert to general relativity with two real scalar fields and an otherwise uncoupled $U(1)$ gauge field. From this action, we can derive the equations of motion.

On varying the action with respect to ϕ we get its equation of motion

$$\nabla_a \nabla^a \phi = -\frac{1}{2} \alpha_0 e^{-2\alpha_0 \phi} F^2 + \frac{1}{2} \alpha_0 e^{4\alpha_1 \phi} (\nabla \kappa)^2. \quad (2.2)$$

On varying the action with respect to the axion, κ , we obtain

$$\nabla_a \left(e^{4\alpha_1 \phi} \nabla^a \kappa \right) = F_{ab} (*F)^{ab}, \quad (2.3)$$

while varying with respect to the Maxwell field produces

$$\nabla_a \left(e^{-2\alpha_0 \phi} F^{ab} \right) = -(*F)^{ab} \nabla_a \kappa. \quad (2.4)$$

Finally, on varying with respect to the metric, we get the full Einstein equation:

$$\begin{aligned} R_{ab} - \frac{1}{2} g_{ab} R &= 2 \nabla_a \phi \nabla_b \phi - g_{ab} \nabla_a \phi \nabla^a \phi + 2 e^{-2\alpha_0 \phi} \left(F_{ac} F_b^c - \frac{1}{4} g_{ab} F_{cd} F^{cd} \right) \\ &+ \frac{1}{2} e^{4\alpha_0 \phi} \left(\nabla_a \kappa \nabla_b \kappa - \frac{1}{2} g_{ab} \nabla_c \kappa \nabla^c \kappa \right). \end{aligned} \quad (2.5)$$

Note that the final term in the action, the so-called Chern-Simons term,

$$\mathcal{L}^{CS} = -\sqrt{-g} \kappa F_{ab} (*F)^{ab}, \quad (2.6)$$

does not contain the metric explicitly:

$$\begin{aligned} \mathcal{L}^{CS} &= -\sqrt{-g} \kappa F_{ab} (*F)^{ab} \\ &= -\frac{1}{2} \sqrt{-g} \epsilon^{abcd} \kappa F_{ab} F_{cd} = -[abcd] \kappa F_{ab} F_{cd}, \end{aligned}$$

where $[abcd]$ is the totally antisymmetric alternating symbol and takes values of -1 , 0 or 1 . As a result, on variation of the Lagrangian density, this term does not contribute to the evolution equations for the gravitational field. As a result, this term is often referred to as the topological term.

With the covariant form of the equations in hand, we are now in a position where we need to decompose these equations into $3+1$, or ADM (and eventually BSSN), form.

2.2.3 The 3+1 Decomposition for EMDA

As mentioned in the introduction, we need to make a split of the spacetime between spacelike and timelike parts. What Einstein put together, we must separate in order to be able to define a good evolution scheme to which we can apply numerical techniques. To this end, we follow a decades old prescription to effect this split [35]. (See Appendix A for a more detailed discussion of the 3+1 decomposition and its meaning.)

To begin, we foliate the spacetime into a family of three dimensional spacelike hypersurfaces we will call Σ and which will be parametrized by a time function t . We define the usual 3+1 decomposition quantities

$$\begin{aligned} g_{ab} &= \gamma_{ab} - n_a n_b, \\ n_a &= (-\alpha, 0, 0, 0), \\ n^a &= \left(\frac{1}{\alpha}, -\frac{\vec{\beta}}{\alpha} \right), \\ K_{ab} &= -\gamma_a^c \gamma_b^d \nabla_c n_d, \end{aligned}$$

where γ_{ab} is the metric induced on the 3D spacelike hypersurface Σ , n_a is the timelike, normal vector to those same hypersurfaces and which therefore satisfies $n_a n^a = -1$ and K_{ab} is the extrinsic curvature of the hypersurfaces.

The point of these definitions is to be able to define a timelike direction such that physical quantities that live and breath in the three dimensional space with their three spatial directions can be thought of as evolving and developing forward in time. Hence, by using these quantities, we are effectively projecting out the spatial directions from the time direction. In the problem of EMDA this includes projecting out an evolution equation or equation of motion for the dilaton (Eqn. 2.2), the axion (Eqn. 2.3), the Maxwell fields (Eqn. 2.4), and the metric (Eqn. 2.5). Along the way, additional, related projections will also provide us constraints that will need to be satisfied. In the end, we are able to obtain an initial value problem that will allow us to study the time evolution of the combined system.

Decomposition of the dilaton field, ϕ

We will start with the equation of motion for the dilaton, ϕ

$$\nabla_a \nabla^a \phi = -\frac{1}{2} \alpha_0 e^{-2\alpha_0 \phi} F^2 + \frac{1}{2} \alpha_0 e^{4\alpha_1 \phi} (\nabla \kappa)^2. \quad (2.7)$$

We will project the equation into and normal to the spacelike hypersurface, Σ . Using this, we will define a new variable $\Pi = -n^a \nabla_a \phi$ which makes our system of equation into first order in time

Appendix A.5.1 describes detailed derivation for decomposition of the dilaton field, ϕ .

After all these algebraic effort, we have

$$n^a \nabla_a \phi = \frac{1}{\alpha} (\partial_t \phi - \beta^i \partial_i \phi) = -\Pi, \quad (2.8)$$

$$\begin{aligned} n^b \nabla_b \Pi &= \frac{1}{\alpha} (\partial_t \Pi - \beta^i \partial_i \Pi) \\ &= -\frac{1}{\alpha} D_a (\alpha D^a \phi) + K \Pi \\ &\quad - \alpha_0 e^{-2\alpha_0 \phi} [(\perp B)^2 - (\perp E)^2] + \frac{1}{2} \alpha_0 e^{4\alpha_1 \phi} (D^a \kappa D_a \kappa - (\Xi)^2). \end{aligned} \quad (2.9)$$

Decomposition of equations for the axion field, κ

We now consider the equation of motion for the axion field, κ .

$$\nabla_a \left(e^{4\alpha_1 \phi} \nabla^a \kappa \right) = F_{ab} (*F)^{ab}. \quad (2.10)$$

It is among the new pieces in our theory on extending EMD. Significantly, it is sourced by an invariant of the Maxwell field as well as the dilaton. Rewriting the equation to reflect this, we have

$$\nabla_a \nabla^a \kappa = -4\alpha_1 \nabla_a \phi \nabla^a \kappa + e^{-4\alpha_1 \phi} F_{ab} (*F)^{ab}. \quad (2.11)$$

We follow a similar manner as pervious decomposition of the dilaton equations. Appendix A.5.2 shows detailed works for decomposition of the axion field, κ . After all these efforts, we have

$$n^b \nabla_b \Xi = -\frac{1}{\alpha} D_a (\alpha D^a \kappa) + K \Xi + 4\alpha_1 D^a \kappa D_a \phi + 4\alpha_1 \Pi \Xi + 4e^{-4\alpha_1 \phi} \perp B^c \perp E_c, \quad (2.12)$$

or, in terms of derivatives with respect to the time coordinate

$$\partial_t \Xi = \beta^a \partial_a \Xi - D_a (\alpha D^a \kappa) + \alpha K \Xi + 4\alpha \alpha_1 D^a \kappa D_a \phi + 4\alpha \alpha_1 \Pi \Xi + 4\alpha e^{-4\alpha_1 \phi} \perp B^c \perp E_c. \quad (2.13)$$

Decomposition of the electromagnetic field equation

We will next decompose the electromagnetic field equation.

$$\nabla_a \left(e^{-2\alpha_0 \phi} F^{ab} \right) = -(*F)^{ab} \nabla_a \kappa, \quad (2.14)$$

or we rewrite the equations into more relevant form for the decomposition.

$$\nabla_a F^{ab} = 2\alpha_0 (\nabla_a \phi) F^{ab} - e^{2\alpha_0 \phi} (\nabla_a \kappa) (*F)^{ab}. \quad (2.15)$$

Again, we follow analogous manner in previous dilaton and axion cases. Appendix A.5.3 shows detailed discussion for decomposition of the electromagnetic fields. With works, we have

$$D_a \perp F^{ab} = 2\alpha_0 \Pi \perp E^a + e^{2\alpha_0 \phi} \Xi \perp B^a \quad (2.16)$$

$$n^d \nabla_d \perp E_c = -K_{ac} \perp E^a - \perp F_c^a D_a \ln \alpha - K \perp E_c. \quad (2.17)$$

Of course, in EM, these equations are only half the Maxwell equations; namely the sourced EM equations. They are supplemented with the mathematical identity $\nabla_{[a} F_{bc]} = 0$, or the homogeneous Maxwell equations. Thus, we need to decompose this identity in an analogous way. In particular, we write the equation as

$$\nabla_a (*F)^{ab} = 0, \quad (2.18)$$

and project it again against n_b and γ_{bc} . We describe this procedure in the Appendix A.5.3.

With works, we have

$$D_a \perp (*F)^{ab} = 0, \quad (2.19)$$

$$n^d \nabla_d \perp B_c = -K_{ac} \perp B^a + \perp (*F)_c^a D_a \ln \alpha - K \perp B_c. \quad (2.20)$$

Eqns. 2.17 and 2.20 provide the evolution equations for the electromagnetic fields. The other two equations are the constraint equations and can be thought of as being purely spatial equations. It has been found to be useful to incorporate the constraints into the time evolution equations to ensure that the constraints are satisfied in numerical simulations. There are a number of ways to do this including at the level of numerical algorithms. We choose an alternative procedure that begins at the level of the equations themselves and involves adding versions of the constraints into the evolutions equations in such a way that the violation of the constraints becomes a sort of damping term that gets driven to zero in the course of the evolution. This process is known as constraint damping. We briefly describe our approach.

We can incorporate constraint damping by including new scalar fields into the electromagnetic equations. The purpose for adding these constraints can be understood easily by considering the usual Maxwell equations. Consider first the equations for the magnetic field in flat space and the usual notation of vector calculus:

$$\nabla \cdot \mathbf{B} = 0, \quad (2.21)$$

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \times \mathbf{E} = 0. \quad (2.22)$$

Of course, the magnetic field must satisfy divergence free condition (Eqn. 2.21). As mentioned earlier, this condition can be thought of as a constraint on the initial configuration of magnetic fields because Eqn. 2.22 will then ensure that the magnetic field is divergence free at any time during the evolution. However, this analytic property of the system will not necessarily be preserved precisely by most numerical schemes. This lack of consistency may lead to the corruption of numerical solutions in the creation of creating magnetic monopoles across the grid. To address this problem, we follow an idea similar to the so called generalized Lagrange multiplier method [36]. The main idea is to introduce additional dynamical variables into the original system. This will create a new system of differential equations that will have the same solution of the original system provided the initial solution satisfies the constraints of the original system. However, if the initial

solution does not satisfy the constraints then deviations will decay or are rapidly carried off the computational domain. Using this idea, we modify the magnetic field equations to be

$$\frac{\partial \Phi}{\partial t} + \nabla \cdot \mathbf{B} = -\eta \Phi, \quad (2.23)$$

$$\frac{\partial \mathbf{B}}{\partial t} - \nabla \times \mathbf{E} + \nabla \Phi = 0, \quad (2.24)$$

where Φ is a new scalar dynamical variable and η is a positive constant. From these equation, we can see that Φ satisfies the telegraph equation

$$-\frac{\partial^2 \Phi}{\partial t^2} - \eta \frac{\partial \Phi}{\partial t} + \nabla^2 \Phi = 0. \quad (2.25)$$

Therefore, Φ is transported away by decaying waves provided $\eta > 0$. For real positive η , the evolution of Φ is towards zero and strict satisfaction of the constraint. Thus, Eqn. 2.23 shows that in such a state the magnetic field will satisfy the divergence free condition. A similar argument can also be applied to the electric field.

Based on this idea, we apply constraint damping terms in our 3 + 1 formulation of the equations for EMDA. In particular, we posit as replacements for the usual constraint equations of electromagnetism, the following two equations

$$\nabla_a (F^{ab} + g^{ab} \Psi) - 2\alpha_0 (\nabla_a \phi) F^{ab} + e^{2\alpha_0 \phi} (\nabla_a \kappa) (*F)^{ab} = \eta_1 n^b \Psi, \quad (2.26)$$

$$\nabla_a ((*F)^{ab} + g^{ab} \Phi) = \eta_2 n^b \Phi, \quad (2.27)$$

where η_1 and η_2 are positive real constants. From the electromagnetic equations, we define the charge and current to be

$$J_{em}^i = -2\alpha_0 (\nabla_a \phi) F^{ab} + e^{2\alpha_0 \phi} (\nabla_a \kappa) (*F)^{ab},$$

$$q_e = J_{em}^i n_i,$$

where the negative sign in the current is merely a sign convention.

We follow our earlier techniques for decomposing this equation. Again, appendix A.5.3

describes detailed derivations of this work. After all works, we have

$$\begin{aligned} n^a \nabla_a \perp E^b &= -\frac{1}{\alpha} D_a (\alpha \perp F^{ab}) - \perp E^a K_a^b + \perp E^b K - D^b \Psi - 2\alpha_0 [\Pi \perp E^b - \varepsilon^{abc} D_a \phi \perp B_c] \\ &\quad - e^{2\alpha_0 \phi} [\Xi \perp B^b + D_a K \perp (*F^{ab})], \end{aligned} \quad (2.28)$$

$$n^a \nabla_a \perp B^b = \frac{1}{\alpha} D_a (\alpha \varepsilon^{abc} \perp E_c) - \perp B^a K_a^b + \perp B^b K + D^b \Phi. \quad (2.29)$$

In terms of time derivatives we can write these as

$$\begin{aligned} \partial_t \perp E^b &= \beta^i \partial_i \perp E^b - \perp E^a \partial_a \beta^b - D_a (\alpha \varepsilon^{abc} \perp B_c) + \alpha \perp E^b K - \alpha D^b \Psi \\ &\quad - 2\alpha \alpha_0 [\Pi \perp E^b - \varepsilon^{abc} D_a \phi \perp B_c] - \alpha e^{2\alpha_0 \phi} [\Xi \perp B^b + D_a K \varepsilon^{abc} \perp E_c], \end{aligned} \quad (2.30)$$

$$\partial_t \perp B^b = \beta^i \partial_i \perp B^b - \perp B^a \partial_a \beta^b + D_a (\alpha \varepsilon^{abc} \perp E_c) + \alpha \perp B^b K + \alpha D^b \Phi. \quad (2.31)$$

Decomposition of the Einstein field equation

The last step in our procedure to derive all equations of motion for the system is the decomposition of the Einstein equation:

$$\begin{aligned} R_{ab} - \frac{1}{2} g_{ab} R &= 2\nabla_a \phi \nabla_b \phi - g_{ab} \nabla_a \phi \nabla^a \phi + 2e^{-2\alpha_0 \phi} (F_{ac} F_b^c - \frac{1}{4} g_{ab} F^2) \\ &\quad + \frac{1}{2} e^{4\alpha_1 \phi} (\nabla_a \kappa \nabla_b \kappa - \frac{1}{2} g_{ab} \nabla_a \kappa \nabla^a \kappa). \end{aligned} \quad (2.32)$$

We will consider the RHS of the Einstein equation first. In our units, the energy-momentum tensor T_{ab} is

$$T_{ab} = 2\nabla_a \phi \nabla_b \phi - g_{ab} \nabla_a \phi \nabla^a \phi + 2e^{-2\alpha_0 \phi} (F_{ac} F_b^c - \frac{1}{4} g_{ab} F^2) + \frac{1}{2} e^{4\alpha_1 \phi} (\nabla_a \kappa \nabla_b \kappa - \frac{1}{2} g_{ab} \nabla_a \kappa \nabla^a \kappa) \quad (2.33)$$

Using this, we can define the projected stress tensor, $\perp T_{ab}$, the momentum flux density, J^i , and the energy density ρ . On decomposing the energy-momentum tensor along the normal components, we get

$$\begin{aligned} \rho &= T_{ab} n^a n^b \\ &= \Pi^2 + D_a \phi D^a \phi + e^{-2\alpha_0 \phi} [(\perp B)^2 + (\perp E)^2] + \frac{1}{4} e^{4\alpha_1 \phi} (\Xi^2 + D_a \kappa D^a \kappa). \end{aligned} \quad (2.34)$$

Here, we use the fact that $g_{ab}n^an^b = (h_{ab} - n_an_b)n^an^b = -1$ together with previous definitions and decompositions that relate ϕ , κ , and F_{ab} . The momentum flux density becomes

$$\begin{aligned} J^i &= -\gamma^{ia}T_{ab}n^b \\ &= 2D^i\phi\Pi + \frac{1}{2}e^{4\alpha_1\phi}D^i\kappa\Xi + 2e^{-2\alpha_0\phi}\epsilon^{ief}\perp B_f\perp E_e, \end{aligned} \quad (2.35)$$

and finally, the spatial projection of the stress-energy tensor is

$$\begin{aligned} \perp T_{ab} &= D_a\phi D_b\phi - h_{ab}(D_c\phi D^c\phi - \Pi) \\ &\quad + 2e^{-2\alpha_0\phi}[\perp F_{ac}\perp F_b^c + \perp E_a\perp E_b - \frac{1}{2}h_{ab}((\perp B)^2 + \perp E)^2] \\ &\quad + \frac{1}{2}e^{4\alpha_1\phi}(D_a\kappa D_b\kappa - \frac{1}{2}h_{ab}(D_c\kappa D^c\kappa - \Xi^2)). \end{aligned} \quad (2.36)$$

Contracting $\gamma_a^c\gamma_b^d g_{cd}$ turns the 4-metric g_{cd} into the spatial 3-metric γ_{ab} .

The usual decomposition of the Einstein tensor is used for the LHS of the equation. (It will give constraint equations and evolution equations of extrinsic curvature and induced metric. See Appendix A for the details of this decomposition.) With all of this done, we have our basic equations split into time and spatial pieces. We can identify relevant evolution equations together with constraint equations that our system must obey. However, long experience has taught us that even with these equations, we do not necessarily have a well-posed initial value problem and a strongly hyperbolic system of equations. The next section is intended to provide us that.

2.2.4 The BSSN Formalism

In the previous section, we went to great lengths to derive the relevant 3 + 1 set of equations for EMDA. They purport to model an extension of general relativity to include additional fields that may describe aspects of our universe. These equations can be thought of as describing various fields interacting on a dynamical spacetime. Hence, these equations must be augmented with the Einstein equation in 3+1 form. The usual ADM decomposition which we have used has been shown to be weakly hyperbolic and prone to instabilities [37]

in numerical simulation. It has been shown that the so-called BSSN formulation of the Einstein equations is strongly hyperbolic and has better stability properties than the ADM formulation. We refer to reader Appendix A for details about the BSSN formalism. For now, we present the entire set of equations in BSSN form:

$$\begin{aligned}
\partial_t \phi &= \beta^a \partial_a \phi - \alpha \Pi, \\
\partial_t \kappa &= \beta^a \partial_a \kappa - \alpha \Xi, \\
\partial_t \Pi &= \beta^a \partial_a \Pi - D_a(\alpha D^a \phi) + \alpha K \Pi - \alpha_0 \alpha e^{-2\alpha_0 \phi} [(\perp B)^2 - (\perp E)^2] \\
&\quad + \frac{1}{2} \alpha_0 \alpha e^{4\alpha_1 \phi} (D^a \kappa D_a \kappa - (\Xi)^2), \\
\partial_t \Xi &= \beta^a \partial_a \Xi - D_a(\alpha D^a \kappa) + \alpha K \Xi + 4\alpha \alpha_1 D^a \kappa D_a \phi + 4\alpha \alpha_1 \Pi \Xi + 4\alpha e^{-4\alpha_1 \phi} \perp B^c \perp E_c, \\
\partial_t \Psi &= \beta^a \partial_a \Psi - \alpha D_a \perp E^a + 2\alpha \alpha_0 (D_c \phi) \perp E^c + \alpha e^{2\alpha_0 \phi} (D_c \kappa) \perp B^c - \alpha \eta_1 \Psi, \\
\partial_t \Phi &= \beta^a \partial_a \Phi + \alpha D_a \perp B^a - \alpha \eta_2 \Phi, \\
\partial_t \perp E^b &= \beta^i \partial_i \perp E^b - \perp E^a \partial_a \beta^b - D_a(\alpha \varepsilon^{abc} \perp B_c) + \alpha \perp E^b K - \alpha D^b \Psi \\
&\quad - 2\alpha \alpha_0 [\Pi \perp E^b - \varepsilon^{abc} D_a \phi \perp B_c] \\
&\quad - \alpha e^{2\alpha_0 \phi} [\Xi \perp B^b + D_a K \varepsilon^{abc} \perp E_c], \\
\partial_t \perp B^b &= \beta^i \partial_i \perp B^b - \perp B^a \partial_a \beta^b + D_a(\alpha \varepsilon^{abc} \perp E_c) + \alpha \perp B^b K + \alpha D^b \Phi.
\end{aligned} \tag{2.37}$$

In order to reach the BSSN formulation, we first consider a conformal transformation of the 3-metric of the form

$$\bar{\gamma}_{ij} = e^{-4\xi} \gamma_{ij}, \tag{2.38}$$

where ξ is the conformal factor such that the conformal metric has unit determinant. Thus, $\gamma = \exp(12\xi)$ because $\bar{\gamma} = 1$. We also define a conformally rescaled traceless extrinsic curvature

$$\bar{A}_{ij} = e^{-4\xi} \left(K_{ij} - \frac{1}{3} \gamma_{ij} K \right). \tag{2.39}$$

Finally, we define the conformal connection functions

$$\bar{\Gamma}^i = \bar{\gamma}^{jk} \bar{\Gamma}_{jk}^i = -\partial_j \bar{\gamma}^{ij}, \tag{2.40}$$

where the $\bar{\Gamma}_{jk}^i$ are the Christoffel symbols built out of the conformal 3-metric such that

$$\bar{\Gamma}_{ij}^k = \Gamma_{ij}^k - 2(\delta_i^k \partial_j \xi + \delta_j^k \partial_i \xi - \gamma_{ij} \gamma^{kl} \cdot \partial_l \xi) \quad (2.41)$$

In order to write down the previous ADM equations in BSSN form, we need to be reminded of a few relations. Let ψ be an arbitrary scalar function. By definition,

$$D_i \psi = \partial_i \psi. \quad (2.42)$$

This is true because there are no connection coefficients in the derivative of the scalar field. Therefore, we can easily rewrite the 3D covariant derivative as the 3D partial derivative (gradient) of a scalar field. For a more realistic case, consider a term such as $D^a(\alpha D_a \phi)$. Expressed in terms of the BSSN variables we have just defined, we have

$$\begin{aligned} D^a(\alpha D_a \phi) &= \gamma^{ij} \nabla_i(\alpha D_j \phi) = \gamma^{ij} [\partial_i(\alpha \partial_j \phi) - \Gamma_{ij}^k \alpha \partial_k \phi] \\ &= \alpha \chi [\bar{\gamma}^{ij} \partial_i \partial_j \phi + \bar{\gamma}^{ij} (\partial_i \ln \alpha) \partial_j \phi - \bar{\Gamma}^i \partial_i \phi - \frac{1}{2} \bar{\gamma}^{ij} \partial_i \phi \partial_j \ln \chi], \end{aligned} \quad (2.43)$$

where we use $\bar{\gamma}_{ij} \bar{\gamma}^{ij} = 3$, rearrange some dummy indices, and define $\chi = e^{-4\xi}$ for purposes of convenience. From the latter, note that we have the relations:

$$\bar{\gamma}_{ij} = \chi \gamma_{ij}, \quad (2.44)$$

$$\bar{\gamma}^{ij} = \frac{1}{\chi} \gamma^{ij}, \quad (2.45)$$

$$\xi = -\frac{1}{4} \ln \chi. \quad (2.46)$$

Other terms can also be written in a similar way. Consider the divergence of an arbitrary spatial vector field, f^a :

$$\begin{aligned} D_a f^a &= \partial_a f^a + {}^{(3)}\Gamma_{ac}^a f^c = \partial_a f^a + \partial_c \ln \sqrt{\gamma} f^c \\ &= \frac{1}{\sqrt{\gamma}} \partial_a (\sqrt{\gamma} f^a) = \frac{1}{\sqrt{e^{12\xi}}} \partial_a (\sqrt{e^{12\xi}} f^a) \\ &= \frac{1}{\sqrt{\chi^{-6}}} \partial_a (\sqrt{\chi^{-6}} f^a) = \chi^{3/2} \partial_a (\chi^{-3/2} f^a) \\ &= \partial_a f^a - \frac{3}{2} \chi^{3/2} \chi^{-5/2} \partial_a \chi f^a \\ &= \partial_a f^a - \frac{3}{2\chi} f^a \partial_a \chi, \end{aligned} \quad (2.47)$$

where ${}^{(3)}\Gamma_{bc}^a$ are the connection coefficients built from the original induced metric in 3 spatial dimensions and we use the property ${}^{(3)}\Gamma_{ac}^a = \partial_c \ln \sqrt{\bar{\gamma}}$ just as $\Gamma_{ac}^a = \partial_c \ln \sqrt{-g}$ in the full 4D case. Using this, we can write

$$D_a \perp E^a = \partial_a \perp E^a - \frac{3}{2\chi} \perp E^a \partial_a \chi, \quad (2.48)$$

$$D_a \perp B^a = \partial_a \perp E^a - \frac{3}{2\chi} \perp B^a \partial_a \chi. \quad (2.49)$$

In general, consider the general covariant vector V_a . In this case, $D_a V_b$ implies the full projection on both the derivative and the vector such that

$$D_a V_b = \perp(\nabla_a V_b) = \gamma_a^c \nabla_c (\gamma_b^d V_d). \quad (2.50)$$

To write this down in BSSN form i.e. using the conformal transformation, we need to evaluate the covariant derivative and change the connection coefficients from Γ to $\bar{\Gamma}$. We must also change the 3-metric γ to $\bar{\gamma}$ using the transformation properties that we defined above. For higher rank tensors such as W_{ab} , we follow the same procedure but include additional connection coefficients (For W_{ab} , we will have two Γ 's so we will need two $\bar{\Gamma}$'s.) Rewriting the equations for the dilaton, the axion, and the EM equations in BSSN form, we get

$$\partial_t \phi = \beta^a \partial_a \phi - \alpha \Pi, \quad (2.51)$$

$$\partial_t \kappa = \beta^a \partial_a \kappa - \alpha \Xi, \quad (2.52)$$

$$\begin{aligned} \partial_t \Pi = & \beta^a \partial_a \Pi - \alpha \chi [\bar{\gamma}^{ij} \partial_i \partial_j \phi + \bar{\gamma}^{ij} (\partial_i \ln \alpha) \partial_j \phi - \bar{\Gamma}^i \partial_i \phi - \frac{1}{2} \bar{\gamma}^{ij} \partial_i \phi \partial_j \ln \chi] + \alpha K \Pi \\ & - \alpha_0 \alpha e^{-2\alpha_0 \phi} [(\perp B)^2 - (\perp E)^2] + \frac{1}{2} \alpha_0 \alpha e^{4\alpha_1 \phi} (\partial^a \kappa \partial_a \kappa - (\Xi)^2), \end{aligned} \quad (2.53)$$

$$\begin{aligned} \partial_t \Xi = & \beta^a \partial_a \Xi - \alpha \chi [\bar{\gamma}^{ij} \partial_i \partial_j \kappa + \bar{\gamma}^{ij} (\partial_i \ln \alpha) \partial_j \kappa - \bar{\Gamma}^i \partial_i \kappa - \frac{1}{2} \bar{\gamma}^{ij} \partial_i \kappa \partial_j \ln \chi] \\ & + \alpha K \Xi + 4\alpha \alpha_1 \partial^a \kappa \partial_a \phi + 4\alpha \alpha_1 \Pi \Xi + 4\alpha e^{-4\alpha_1 \phi} \perp B^c \perp E_c, \end{aligned} \quad (2.54)$$

$$\begin{aligned} \partial_t \Psi = & \beta^a \partial_a \Psi - \alpha \left(\partial_a \perp E^a - \frac{3}{2\chi} \perp E^a \partial_a \chi \right) + 2\alpha \alpha_0 (\partial_c \phi) \perp E^c \\ & + \alpha e^{2\alpha_0 \phi} (\partial_c \kappa) \perp B^c - \alpha \eta_1 \Psi, \end{aligned} \quad (2.55)$$

$$\partial_t \Phi = \beta^a \partial_a \Phi + \alpha \left(\partial_a \perp B^a - \frac{3}{2\chi} \perp B^a \partial_a \chi \right) - \alpha \eta_2 \Phi, \quad (2.56)$$

$$\begin{aligned} \partial_t \perp E^a &= \beta^i \partial_i \perp E^a - \perp E^a \partial_b \beta^a \\ &\quad - \chi^{1/2} \varepsilon^{abc} \left\{ \alpha [\partial_b \bar{\gamma}_{cd} \perp B^d + \bar{\gamma}_{cd} \partial_b \perp B^d] + \bar{\gamma}_{cd} \perp B^d \left[\partial_b \alpha - \alpha \frac{\partial_b \chi}{\chi} \right] \right\} \\ &\quad + \alpha K \perp E^a - \alpha \chi \bar{\gamma}^{ab} \partial_b \Psi - 2\alpha \alpha_0 [\Pi \perp E^a - \varepsilon^{abc} \chi^{1/2} \partial_b \phi \bar{\gamma}_{cd} \perp B^d] \\ &\quad - \alpha e^{2\alpha_0 \phi} [\Xi \perp B^a + \varepsilon^{abc} \chi^{1/2} \partial_b K \bar{\gamma}_{cd} E^d], \end{aligned} \quad (2.57)$$

$$\begin{aligned} \partial_t \perp B^a &= \beta^i \partial_i \perp B^a - \perp B^a \partial_b \beta^a \\ &\quad - \chi^{1/2} \varepsilon^{abc} \left\{ \alpha [\partial_b \bar{\gamma}_{cd} \perp E^d + \bar{\gamma}_{cd} \partial_b \perp E^d] + \bar{\gamma}_{cd} \perp E^d \left[\partial_b \alpha - \alpha \frac{\partial_b \chi}{\chi} \right] \right\} \\ &\quad + \alpha K \perp B^a + \alpha \chi \bar{\gamma}^{ab} \partial_b \Phi. \end{aligned} \quad (2.58)$$

The evolution equations for the variables $\bar{\gamma}_{ij}$, $\bar{\Gamma}^i$, \bar{A}_{ij} , ξ , and K can be obtained from the decomposition of the Einstein tensor (again, see Appendix A for details)

$$\partial_t \bar{\gamma}_{ij} = \beta^k \partial_k \bar{\gamma}_{ij} + \bar{\gamma}_{kj} \partial_i \beta^k - \frac{2}{3} \bar{\gamma}_{ij} \partial_k \beta^k - 2\alpha \bar{A}_{ij}, \quad (2.59)$$

$$\partial_t \xi = \beta^k \partial_k \xi + \frac{1}{6} \partial_k \beta^k - \frac{1}{6} \alpha K, \quad (2.60)$$

$$\begin{aligned} \partial_t \bar{A}_{ij} &= \beta^k \partial_k \bar{A}_{ij} + \bar{A}_{kj} \partial_i \beta^k - \frac{2}{3} \bar{A}_{ij} \partial_k \beta^k + e^{-4\xi} \left[-D_i D_j \alpha + \alpha^{(3)} R_{ij} - 8\pi G \alpha \perp T_{ij} \right]^{TF} \\ &\quad + \alpha \left(K \bar{A}_{ij} - 2\bar{A}_{ik} \bar{A}_j^k \right), \end{aligned} \quad (2.61)$$

$$\partial_t K = \beta^k \partial_k K - D_i D^i \alpha + \alpha \left(\bar{A}_{ij} \bar{A}^{ij} + \frac{1}{3} K^2 \right) + 4\pi G \alpha (\rho + \perp T), \quad (2.62)$$

$$\begin{aligned} \partial_t \bar{\Gamma}^i &= \beta^j \partial_j \bar{\Gamma}^i - \bar{\Gamma}^j \partial_j \beta^i + \frac{2}{3} \bar{\Gamma}^i \partial_j \beta^j + \bar{\gamma}^{jk} \partial_j \partial_k \beta^i + \frac{1}{3} \bar{\gamma}^{ij} \partial_j \partial_k \beta^k \\ &\quad - 2\bar{A}^{ij} \partial_j \alpha + 2\alpha \left(\bar{\Gamma}_{jk}^i \bar{A}^{jk} + 6\bar{A}^{jk} - \frac{2}{3} \bar{\gamma}^{jj} \partial_j K - 8\pi G e^{4\xi} J^i \right). \end{aligned} \quad (2.63)$$

where TF means the trace free part, the quantities ρ , J^i , and $\perp T_{ij}$ were obtained previously and recall that $\perp T \equiv \perp T_i^i$

2.2.5 Black Hole Initial Data for EMDA

Having established the equations and formalism within which we will work, it is necessary to consider appropriate initial conditions which will represent the physical system that we

would like to explore. It therefore becomes necessary to consider black hole solutions as initial data within EMDA.

There are, of course, well known black hole solutions within Einstein's gravity. These include the Kerr-Newman solution which represents the most general black hole solution with three parameters, namely mass, charge and angular momentum. We should, of course, also acknowledge that it is also the external, vacuum solution of a massive, charged, rotating source of matter. As our focus is black hole solutions, we will focus on interpreting it as such. However, we also want to generalize such black hole solutions to EMDA. Such a solution has been found [38, 39]. It is known as the Kerr-Sen [38] solution and represents a massive, charged (under a $U(1)$ gauge field), rotating, black hole that is also endowed with both dilaton and axion fields. While much is known about this black hole, to our knowledge it remains unknown whether a single black hole is stable and whether in a dynamical process of say, the inspiral of two such black holes, the final state remains a Kerr-Sen black hole.

In addition to the Kerr-Sen black hole, there is also a study [39] which obtains a class of axisymmetric solutions to EMDA. Furthermore, there are several additional classes of BHs in EMD theories: [40] finds charged BHs in low energy string theory, [41] describes rotating dilatonic BHs, and [42, 43] presents rotating BHs in Kaluza-Klein theory which can be thought of as a subset of EMD.

In this work, we use different known analytic BHs in the literatures [38, 40, 41, 42, 43] as our initial data for single BH tests. These types of BH initial data admit additional components such as dilaton and axion as scalar hairs into BHs which are the standard GR do not allow to have these scalar hairs. We treat these additional scalar hairs as a constant at the initial BH and perform following dynamics. We also adapt to specific gauge or coordinate system that is useful for numerical simulations.

Furthermore, one of the main interests of our research is obtaining the GW waveform of a BBH merger within this theory. To construct initial data for BBH merger, we simply use puncture initial data to create BBH systems then adding additional constant EM, dilaton,

and axion fields. Detailed information and derivations for both single and binary BH initial data is described in the Appendix B.

2.2.6 Results

In this section, we present the results of some of our numerical simulations. To perform these, we use the HAD infrastructure (See Appendix E.1.1 for details of HAD). HAD is a longstanding toolkit (over a decade) used within the numerical relativity community for solving the Einstein equations coupled to a variety of matter systems. It incorporates Berger-Oliger style block AMR [44]. The HAD code has been well tested and provides us a basic, trusted computational environment within which to develop the numerical approach to solving the equations of EMDA.

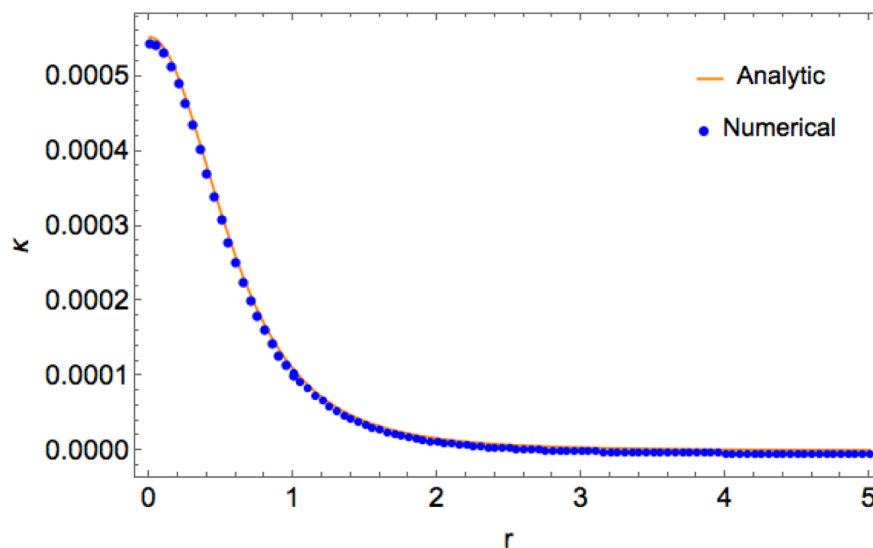


Figure 2.1 Comparison between the analytic and numerical results for the axion field. The expression for the axion comes from Eqn. B.166 We extract the profile at late times when the solution has settled down to a stationary configuration.

As tests of our implementation of EMDA, we first provide a test of a single BH evolved for checking dynamical stability (or a possible unstable mode) of BH in EMDA as well as an additional scalar field, axion, in this theory. These tests will provide an insight how BH properties in EMDA can be understood as comparing with the standard GR.

Before presenting single BH results, we validate our code implementation first. Fig 2.1 shows a comparison plot between analytic and numerical results for the axion field. The analytic axion field equation is adapted from Eqn. B.166. We extract the radial profile of the axion field at late times when the BH settles down to some stationary solution. This result shows that both analytic and numerical results match well.

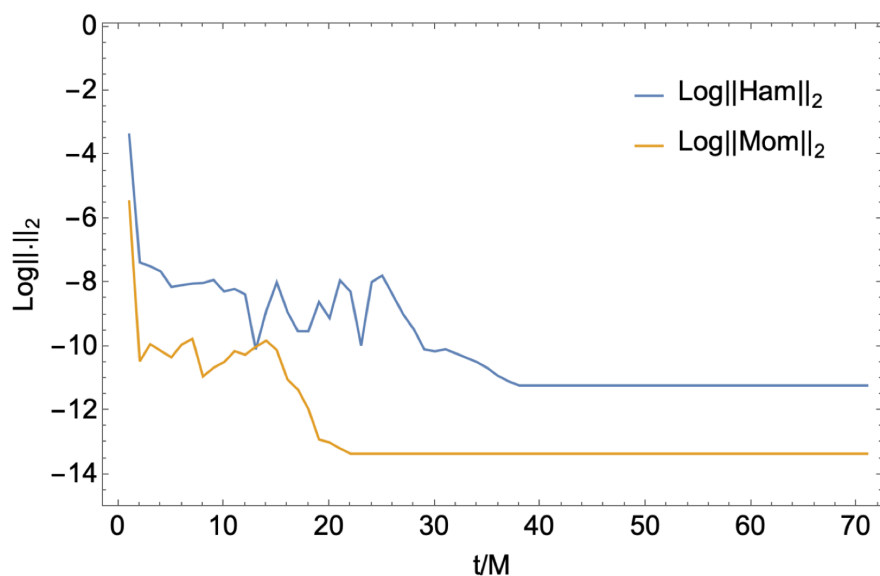


Figure 2.2 The L_2 norm of the Hamiltonian and momentum constraints plotted over time for a single BH evolution. We use Kerr-Sen initial data for this test. The feature at early time is apparently due to truncation error from the numerical integration but it will be stabilized and down to small value at late time.

We also monitor the Hamiltonian and momentum constraints as a check on the validity of the numerical integration. Analytically, both Hamiltonian and momentum constraints should be zero during the evolution. However, because of truncation errors and the approximations inherent in our finite difference approach, we expect that there will be violations of these constraints across our numerical grid. As a confirmation of the stability and correctness of our code, we expect these violations to remain small and bounded throughout the simulation. If we observe large violations during the evolution, this suggests that our implementation is incorrect or unstable. Fig 2.2 shows the L_2 norm of the Hamiltonian and momentum constraints over time for a single BH evolution. For this case, our initial data is a single

Kerr-Sen BH. At early times, there are somewhat larger constraint violations, but these becomes damped out over the course of the evolution. This gives us confidence in the validity of our code. We have not done thorough convergence testing but should as a more stringent test of our code. Nonetheless, the two tests that we have done suggest that our implementation provides a believable evolution of our physical system that we will use for exploring GW production.

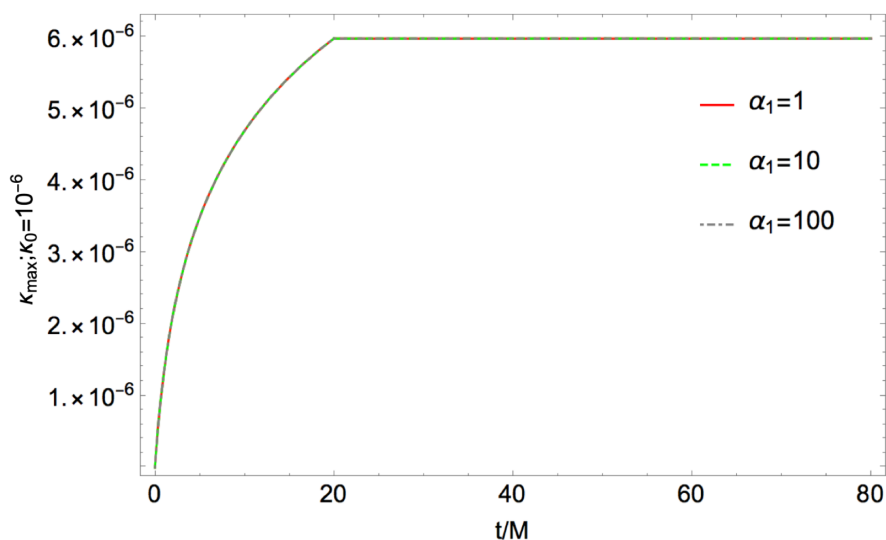


Figure 2.3 Evolution of the axion associated with a single BH for different α_1 parameter values. We track the maximum value of the axion with an initial amplitude of $\kappa_0 = 10^{-7}$ during the evolution. Note that the system settles down to a stable configuration at later times. The resulting profiles are rescaled assuming a linear increase with respect to the coupling parameter value $\alpha_1 = 100$. From the result, linear scaling fits well.

We now present aspects of the behavior of a single EMDA BH. We are focused in particular on the behavior of the axion field as it is the new piece in our EMDA theory. Fig 2.3 shows the evolution of the maximum of the axion field. We use Kerr-Sen initial data (See Appendix B.4) for these simulations with an initial amplitude of $\kappa_0 = 10^{-7}$. We initially choose small initial amplitude of axion to check how this additional field affects behavior of single BH over EMD BH and its stability. As the purpose of this test is to examine the behavior of the axion, we fix all other parameters such as the dilaton coupling constant $\alpha_0 = 1$, the initial dilaton amplitude $\phi_0 = 10^{-7}$, and the electric charge $q_e = 10^{-4}$.

One can see that the maximum value of the axion grows with time and then levels off at a value that remains unchanged for roughly a dynamical crossing time. The axion field thus settles into a stationary state. To test this across the family of couplings or theories, we vary the axion coupling parameter α_1 to check whether the stationary solution depends on this parameter α_1 . For comparison purposes, we linearly rescale the solutions with respect to the value $\alpha_1 = 100$. We test with $\alpha_1 = \{1, 10, 100\}$. The results show that linear scaling fits well. Thus, the axion field settle into a stationary state for $\alpha_1 = \{1, 10, 100\}$.

One interesting aspect from this result is our full nonlinear dynamics shows some linear regime result. This is somewhat surprise but here note that our initial value of axion is small, $\kappa_0 = 10^{-7}$, thus initial BH with small additional axion field might be interpreted as linear perturbation of EMD BH. Although this might not be interesting regime to test, it is suffice to say that EMDA BH shows stable behavior. There might be possibility that linear scaling will fail if we increase the value of α_1 or having large enough initial axion amplitude value i.e. enter full nonlinear regime for EMDA. However, stability for linear scaling fail case remains unknown.

Another check of the code and the stability of the solutions is whether the mass of the BH remains constant over time. As there are different BHs in both EMD and EMDA theories, knowing whether these are in fact the stable endstates of dynamical collapse is first investigated by knowing whether these black holes are stable to small perturbations such as numerical truncation error. Fig. 2.4 shows the result of just such a test; namely mass conservation for different initial data. For this test, we use KS (Appendix B.4) and GHS (Appendix B.1) BHs as initial data. In addition, we use different combinations of the coupling parameters α_0 and α_1 , again to vary across theory space. The initial mass of the BH is always taken to be $M = 0.5$ and we fix other initial values to be constant. We take the dilaton field as $\phi_0 = 10^{-7}$, the axion field to be $\kappa_0 = 10^{-7}$, and the electric charge as $q_e = 10^{-3}$. Note that at the initial time, there is some oscillatory behavior due to numerical noise in initial data but mass is stabilized at later time for all test cases. This indicates that the code conserves the mass of the BHs. This suggests that indeed the mass of the BH is

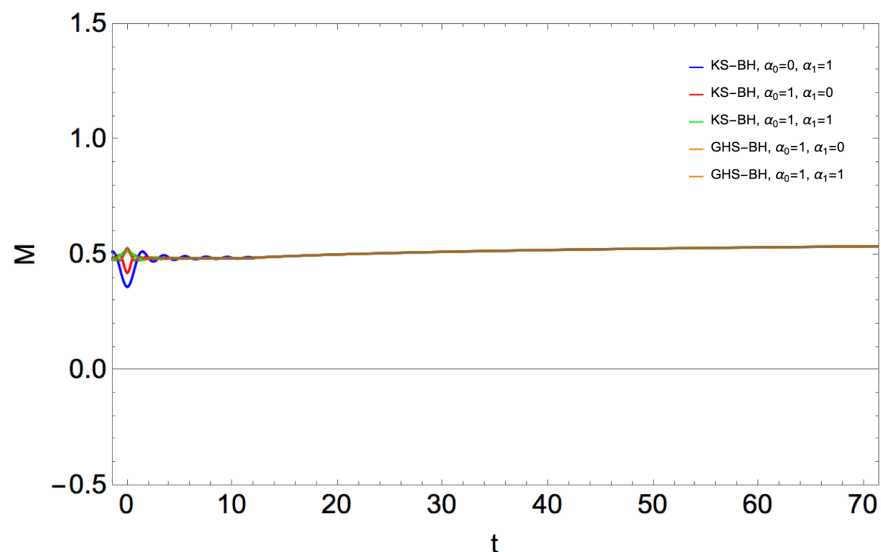


Figure 2.4 This figure illustrates tests of mass conservation. Different combinations of initial data and parameters are used for these tests. KS-BH stands for the Kerr-Sen BH initial data found in Appendix B.4 and GHS-BH stands for the Garfinkle-Horowitz-Strominger BH found in Appendix B.1. Different combinations of coupling parameters α_0 and α_1 are used to check the effects of these parameters.

conserved during short time evolutions and that the Kerr-Sen black hole of EMDA is, at least for short times, stable.

These numerical simulations of single BHs suggest that we are able to confirm the dynamical stability of stationary BH configurations in EMDA theory using numerical simulations. This further suggests that BHs in EMDA have similar stability properties to those in standard GR. In addition, this implies that additional solutions exist in the parameter space of the couplings α_0 and α_1 that are not now known analytically. These are generalizations of Kerr-Sen BHs (with dilaton and axion fields) to arbitrary parameter values.

Furthermore, dynamics of an additional scalar channel shows stable behavior which has been known a little with regard to their stability properties. This information will be used to understand dilaton and axion as a actual phenomenological particles but current understanding of these types of weak interacting particles are quite limited.

We turn now to BBH systems. Based on our single BH simulations, we expect that we

can construct initial data for BBH systems from individually stable Kerr-Sen black holes and their generalizations. We expect these BBH systems to emit gravitational radiation and eventually to merge. Indeed, one of the main aims of this work is generating GW waveforms and comparing them with standard GR to determine what, if any, deviations may be present and detectable by LIGO/VIRGO or future detectors. We use BBH initial data that is described in Appendix. ??.

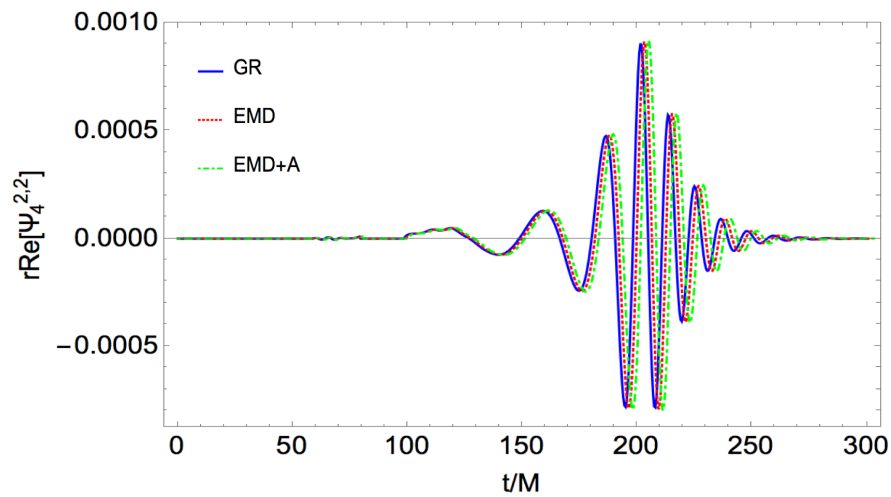


Figure 2.5 GW waveforms for different theories, namely standard GR, EMD, and EMDA. We use non-spinning equal mass binary merger for all these tests.

We test differences between GW waveforms for different theories, namely standard GR, EMD, and EMDA. Fig. 2.5 shows GW waveforms for a binary in each of these theories. We use non-spinning equal mass binaries for our tests. Mass of each BH is 0.5 and separation between two BHs is 6. And, the number of orbit for this simulation is roughly 1. For EMD, we choose the dilaton coupling parameter $\alpha_0 = 1$, the initial dilaton amplitude $\phi_0 = 10^{-7}$ with electric charge $q_e = 10^{-3}$. For EMDA, we keep the same parameter values as with EMD and choose the axion coupling parameter $\alpha_1 = 1$ with an initial axion amplitude of $\kappa_0 = 10^{-7}$. In this case, the differences in the GW waveforms, for these small amplitude scalar fields, between those theories are essentially indistinguishable and would not be measurable by LIGO/VIRGO.

As we are looking for regions of the parameter space that may hold larger deviations

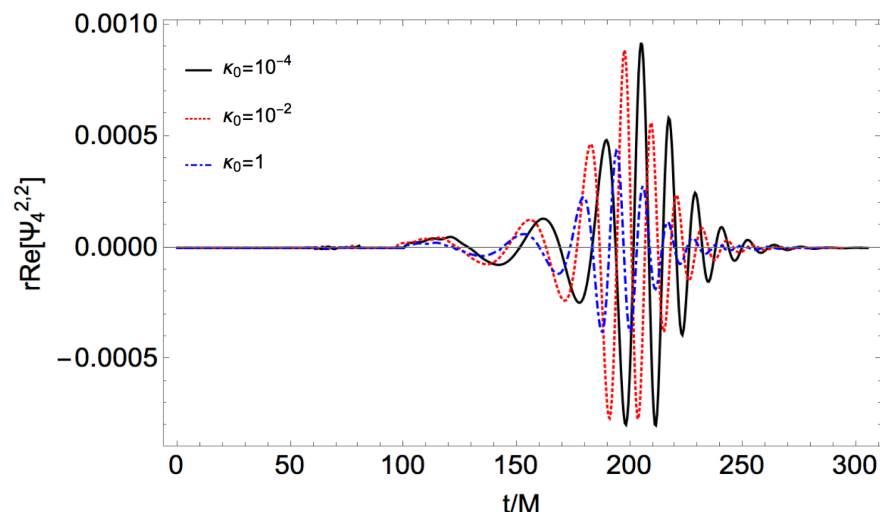


Figure 2.6 GW waveforms in the EMDA theory with different initial axion field values. We use non-spinning equal mass binary merger with fixed coupling parameters, $\alpha_0 = \alpha_1 = 1$, electric charge $q_e = 10^{-3}$, and initial dilaton amplitude $\phi_0 = 10^{-4}$

from the GR prediction, we might ought to look at changing more dramatically those pieces of our system that are distinct from GR. A more dramatic comparison that can be made is by examining the effects of the strength of the axion field on BBH mergers. Fig. 2.6 shows GW waveforms in the EMDA theory for varying initial axion field values. Again, we use non-spinning equal mass binaries with the same parameter values as in Fig 2.5 except for initial axion amplitude values. Three different initial axion amplitudes $\kappa_0 = 10^{-4}, 10^{-2}, 1$ are chosen. The cases with larger initial axion amplitudes merge slightly earlier. More noticeable, however, is that the GW amplitude decreases with increasing κ_0 . Presumably, in the dynamical process that is merger, increasing amounts of the mass of the binaries is being radiated away in non-gravitational wave channels. However, this assertion would need to be checked by examining the power radiated in scalar channels, and, in particular, the axion radiation.

Thus, if LIGO/VIRGO or other possible future GW detectors can distinguish signals between scalar radiation channels and GW quadrupole moment radiation, the current results may provide an interesting testbed in which to check GR in the strong field regime

and deviation from the GR prediction. However, finding relevant astrophysical is quite challengeable since our understanding of the beyond Standard Model physics is quite limited. For example, in this work, the axion is understood by additional scalar field coupled to the GR but axion is also considered as ultralight particle to resolve the strong CP problem in QCD. There is no a such thing that fills cap between axion as scalar field and axion as an actual particle. We expect that this GW waveform will provide some information to particle physics community also. To sum up, a preliminary investigation in here does not reveal any dramatic effects in GW waveform but we may see certain deviation if we consider extremal cases for binary mergers.

From our GW waveform result, scalar radiations from EMD and EMDA might affect BBH behavior during their evolution. Therefore, extracting these scalar field in addition to GW waveform will provide additional insight of BBH behavior and how this will change GW waveforms. Furthermore, we can obtain additional information by comparing quasi-normal mode frequencies between EMDA and standard GR and this will show the role of dilaton and/or axion in terms of its effect on the dynamics and the formation of final BH.

2.3 Black Hole Dynamics in Quadratic Gravity

In this section, we discuss a somewhat different modified theory of gravity, namely quadratic gravity. Where EMDA theory originates in low energy string theory and can be seen to postulate additional fields describing the gravitational interaction, quadratic gravity is an example of a higher derivative theory in which gravity continues to be modeled by curvature, but with an action principle that includes higher derivative terms, i.e. higher curvature (energy) effects at small distance scales. Some of our aims for this work is implementing a well-posed initial value problem within this theory and evaluating possible BH solutions and examining their stability properties within quadratic gravity.

2.3.1 Quadratic Gravity in the Strong-Gravity Regime

GR is built upon diffeomorphism invariance which consists of the invariance of the form of physical laws under arbitrary differentiable coordinate transformations. This is manifest in its formulation in terms of the Einstein-Hilbert action

$$S_{\text{EH}} = \frac{1}{16\pi G_N} \int d^4x \sqrt{-g} (R - 2\Lambda), \quad (2.64)$$

where G_N and Λ denote the Newtonian coupling and cosmological constant, respectively. (Note that we reinsert G_N for the remainder of this and the next section.) In some sense, GR, as built from this action principle is a rather simple theory in that it possesses only two free parameters and yet purports to describe all gravitational phenomena. However, if we set aside this notion of simplicity (*à la Occam's razor*), there is no fundamental reason within classical gravity that would explain the absence of higher-order, diffeomorphism-invariant curvature terms in defining a theory of gravity. Indeed, were we to consider the possibility of higher order curvature effects as contributing to a theory of gravity, we might ask what such might look like and how they would affect the standard model of GR. One possibility would be to consider the next (quadratic) order in curvature invariants. One example, and the one we will consider here, is what is known as Quadratic Gravity (QG) and is given by

$$S_{\text{gravity}} = S_{\text{EH}} + \int d^4x \sqrt{-g} \left(\alpha R_{ab} R^{ab} + \beta R^2 \right). \quad (2.65)$$

In spacetimes topologically equivalent to flat space the Gauss-Bonnet topological invariant ensures that this is the most general action at quadratic order. Stelle has shown that, in contrast to GR, QG is perturbatively renormalizable as a quantum field theory [45]. He and subsequent authors have also discussed the appearance of additional massive ghost-like modes, one scalar and one spin-2, which spoil the unitarity of the corresponding quantized theory of gravity. Further, the theory can be motivated as a generic infrared limit of an effective field theory (EFT) treatment of the quantization of gravity, see e.g. [46] for a pedagogical review. Significantly, all subsequent (unresummed local) curvature invariants are suppressed by powers of the Planck scale.

One might be inclined to argue for the uniqueness of GR by virtue of the fact that it admits a well-posed initial value problem and hence a well-defined evolutionary dynamical system. However, as Noakes has shown in [47], the dynamics of quadratic gravity can, in fact, be formulated as a well-posed initial value problem. Because we regard these concepts of such crucial importance for all that follows in this chapter, we will review them.

Experiments on solar-system scales only test the weak-gravity regime: the existing constraints on higher-order modifications of GR are therefore extremely weak. Submillimeter-tests using pendulums constrain Yukawa-like corrections to the Newtonian potential that arise from higher-derivative terms like $\alpha R_{ab}R^{ab}$ and βR^2 . But, the constraints are as weak as $\alpha, \beta < 10^{60} \sim 10^{70}$ [48, 49, 50]. Further, they do not allow us to distinguish between the two couplings.

The main motivation for our current efforts is to study the stability of dynamical BHs and eventually generate gravitational wave templates from QG in order to compare these with observational data from binary mergers in order to constraint the QG-couplings α and β , (See also [51, 52] for a similar study in $f(R)$ -gravity.) A simple comparison of typical curvature scales illustrates by how much studying the strong-gravity regime could improve these bounds. For that, we compare the curvature at the surface of the earth horizon with that of a solar-mass BH, by use of the Kretschmann scalar $K = R^{abcd}R_{abcd} \sim M^2/r^6$, i.e.,

$$\frac{K_{\oplus\text{-surface}}}{K_{\odot\text{-horizon}}} = \frac{K_{\oplus\text{-surface}}}{K_{\oplus\text{-horizon}}} \frac{K_{\oplus\text{-horizon}}}{K_{\odot\text{-horizon}}} \approx 10^{-32} . \quad (2.66)$$

To compute this, we assume that the Schwarzschild type BH. From this test, we have the constraints are roughly $\alpha, \beta < 10^{30}$ which are still weak constraint but improved compare with solar-system scales. In this scenario, this bound might not be suitable for the most sensitive detection region of the LIGO/VIRGO (mass range around $30 - 300M_{\odot}$). However, for extreme case such as extreme mass ratio inspiral about a 10^6M_{\odot} , it might be possible to constraint the higher order parameters α and β

2.3.2 Quadratic Gravity from the Point of View of Effective Field Theory

Field-theoretic attempts to quantize gravity have difficulties retaining renormalizability. Without higher-derivative terms, General Relativity is a non-renormalizable theory. The underlying reason can be associated with the negative canonical dimension of the Newtonian coupling, i.e., $[G_N] = -2$. The quantization of gravity is not just an academic problem. If G_N was of order one at accessible energy scales, then the effects of delocalized quantum matter would directly imply significant quantum fluctuations in the associated gravitational field. It is only the smallness of the gravitational force, i.e., $G_N \sim 10^{-38}$ GeV, which hides these effects from experimental observation. In fact, this is one particularly insightful way of obtaining the Planck scale. The dimensionless Newtonian coupling $g_N(\mu) = G_N \times \mu^2$ scales with a quadratic power-law in the characteristic energy scale μ . Since it is dimensionless couplings which determine field-theoretic cross sections, gravitational effects become important whenever $g_N(\mu) \approx 1$, i.e., at a mass scale of $M_{\text{Planck}} = 10^{19}$ GeV, i.e. the Planck scale.

The most agnostic approach to the quantization of gravity is to look at quantum gravity as an effective field theory (EFT). In correspondence to, for instance, the Standard Model of particle physics, this assumes that *all* operators allowed by symmetry are present at M_{Planck} . Neglecting the index structure we can schematically denote those (dimensionful) couplings and the corresponding curvature invariants as $C_N \times R^N$. Since the underlying theory is not known, *all* EFT-couplings are assumed to be of order one at the Planck-scale, i.e., $C_N(\mu = M_{\text{Planck}}) \approx 1$. The EFT draws its predictive power from canonical scaling. Since in four dimensions the dimension of curvature is $[R] = 2$, the associated couplings have dimension $[C_N] = 2N - 4$. The corresponding dimensionless couplings $c_N(\mu) = C_N/\mu^{2N-4}$ scale like $c_N \sim \mu^{4-2N}$. Hence they are suppressed by

$$c_N(M_{\text{exp}}) \approx \left(\frac{M_{\text{exp}}}{M_{\text{Planck}}} \right)^{(2N-4)}. \quad (2.67)$$

In an EFT one would thus conclude that all couplings $c_{N>2}$ are suppressed by increasing

powers of the enormously large Planck scale. This does not apply for the quadratic couplings $c_{N=2}$, which constitute so-called marginal couplings of the EFT. These are *not* power-law suppressed and are hence expected to be of similar order as at the Planck-scale.

The EFT for quantum gravity is valid below the Planck scale. It might be possible to embed it into a non-perturbatively renormalizable quantum field theory of gravity solely defined by diffeomorphism symmetry and a corresponding asymptotically safe fixed point. This theory of Asymptotically Safe Gravity (ASG) was conjectured by Steven Weinberg in 1976 [53]. It is supported by the mounting evidence around the corresponding Reuter fixed-point [54]. Since the Reuter fixed-point is fully interacting, *all* higher-order couplings will be present at Planckian energies. Below the Planck scale the EFT-description applies and one, again, expects the associated low-energy effective theory to be governed by the four couplings of QG, i.e., G_N , Λ , α and β . It can be added here, that current approximations of the Reuter fixed-point indicate that there are only three so-called relevant couplings. ASG restores the predictivity of a quantum field theory of gravity precisely by non-perturbative relations, which express *all* other (irrelevant) couplings in terms of the three relevant ones. Hence, one of the two QG couplings could follow as a prediction from ASG, e.g., $\alpha = \alpha(\beta, G_N, \Lambda)$.

2.3.3 A Well-posed Initial Value Problem for Quadratic Gravity

In this section, we return to a geometrized unit system ($G_N = c = 1$). To understand BH behaviors within QG, we need to evolve our system of equations in time. Due to the complexity of the problem, solving the equations analytically is almost certainly impossible. We need to solve the system numerically and therefore we need to construct the initial value, or Cauchy, problem. In particular, we must have a well-posed problem in order to obtain stable numerical integrations. The concept of a well-posed problem is understood to satisfy the following three conditions

- a solution exists (existence)

- the solution is unique (uniqueness)
- the solution depends continuously on initial and boundary data (stability)

A well-posed initial value problem is therefore crucial to having a long term stable numerical simulation.

As an example, consider the following PDE:

$$\frac{\partial^2 u}{\partial t^2} - \frac{\partial^2 u}{\partial x^2} = 0, \quad (2.68)$$

together with initial and boundary conditions

$$x \in [0, 1],$$

$$\text{ID} : u = 0, \quad \partial_t u = \frac{\sin(2\pi n x)}{(2\pi n)^P}, \quad P \geq 1,$$

$$\text{BC} : u = 0 \text{ at } x = 0, 1.$$

A solution of Eqn. 2.68 can be obtained straightforwardly as

$$u(x, t) = \frac{\sin(2\pi n x) \sin(2\pi n t)}{(2\pi n)^{P+1}}. \quad (2.69)$$

Thus, as n approaches ∞ the initial data approaches zero, and the solution, $u(x, t)$, approaches zero as well. This is an example of a well-posed system.

Now consider the nearly identical PDE with the same initial and boundary conditions as previously

$$\frac{\partial^2 u}{\partial t^2} + \frac{\partial^2 u}{\partial x^2} = 0, \quad x \in [0, 1],$$

$$\text{ID} : u = 0, \quad \partial_t u = \frac{\sin(2\pi n x)}{(2\pi n)^P}, \quad P \geq 1,$$

$$\text{BC} : u = 0 \text{ at } x = 0, 1.$$

In this case, the solution is

$$u(x, t) = \frac{\sin(2\pi n x) \sinh(2\pi n t)}{(2\pi n)^{P+1}}.$$

Note that we have only changed a sign in the PDE. However, as n approaches ∞ , the initial data approaches zero, but the solution $u(x, t)$ approaches ∞ . In other words, small perturbations at the initial time will produce an arbitrarily large solution in given finite time. This is an example of an ill-posed problem.

A number of articles in the literature discuss the well-posedness of GR [37, 55] within different formulations [56] and in terms of their hyperbolicity [57]. Here, we adapt a theorem from [47, 58] and apply this to QG. To orient ourselves, we first note that in this work, second-order quasi-linear hyperbolic systems will play an important role; the generic form for which is given by

$$g^{ab}(x, t, u_i) \frac{\partial u_q}{\partial x^a \partial x^b} = f_q(x, t, u_i, \partial_j u_i), \quad (2.70)$$

where u_q with $q = 1 \dots n$ is a vector constructed from the unknowns, g^{ab} is an inverse metric (which can have matrix form), and $f_q(u_i, \partial_j u_i)$ contains all lower order terms including possibly nonlinear terms. Our relevant theorem [47, 58] states

Theorem 1 *A given quasi-linear, diagonal, second order hyperbolic system with constraints on the initial data possesses a well-posed initial value problem.*

In subsequent sections, we will derive a well-posed initial value formulation for QG based on Theorem 1 .

2.3.3.1 Equations of Motion

The equations of motion (eom) of QG are given by

$$H_{ab} = \frac{1}{16\pi} G_{ab} + E_{ab} = \frac{1}{2} T_{ab}, \quad (2.71)$$

where $G_{ab} = R_{ab} - 1/2 R g_{ab}$ is the usual Einstein tensor, which is supplemented by its quadratic-order counter-part

$$\begin{aligned} E_{ab} = & (\alpha - 2\beta) \nabla_a \nabla_b R - \alpha \square R_{ab} - \left(\frac{1}{2} \alpha - 2\beta \right) g_{ab} \square R + 2\alpha R^{cd} R_{acbd} \\ & - 2\beta R R_{ab} - \frac{1}{2} g_{ab} (\alpha R_{cd} R^{cd} - \beta R^2). \end{aligned} \quad (2.72)$$

QG propagates, besides the graviton, an additional massive scalar mode associated with the Ricci-scalar R and a massive spin-2 mode corresponding to the traceless part of the Ricci-tensor $\tilde{R}_{ab} = R_{ab} - \frac{1}{4}g_{ab}R$ [59]. Knowing this, it is useful to split the equations of motion into a trace and a traceless part, i.e.,

$$\square R = \frac{1}{32\pi(3\beta - 2\alpha)}R, \quad (2.73)$$

$$\begin{aligned} \alpha\square\tilde{R}_{ab} = & -\frac{1}{16\pi}\tilde{R}_{ab} - (2\beta - \alpha)\left(\frac{1}{128\pi(3\beta - 2\alpha)}g_{ab} - \nabla_a\nabla_b\right)R \\ & - (\alpha - 2\beta)R\tilde{R}_{ab} - 2\alpha\left(R_{acbd} - \frac{1}{4}g_{ab}\tilde{R}_{cd}\right)\tilde{R}^{cd}. \end{aligned} \quad (2.74)$$

As anticipated, these equations are fourth order in derivatives. Following [47], there are two additional essential steps needed in order to cast the equations of motion into a well-posed IVP. These are: (i) We must employ harmonic coordinates in order to treat g_{ab} , \tilde{R}_{ab} , and R as independent variables and which reduces the system to having only second-order derivatives and (ii) We use a differentiation procedure to diagonalize the resulting equations and put them in quasilinear form.

2.3.3.2 Reduction to a Second Order System

Our aim now is to reduce our system from its fourth order current form to a system that is second order in derivatives. To this end, we collect terms in an after-the-fact ‘‘obvious’’ way. Note that the Ricci tensor in a general coordinate system is given by

$$R_{ab} = -\frac{1}{2}g^{cd}\partial_c\partial_d g_{ab} + Q_{ab}(g, \partial g) + \frac{1}{2}\left(g_{ad}\partial_b F^c + g_{bc}\partial_a F^d\right), \quad (2.75)$$

where

$$Q^{ab}(g, \partial g) = g^{cd}\left(\Gamma^a_{ce}\partial_d g^{be} + \Gamma^b_{ce}\partial_d g^{ae} - 2\Gamma^e_{cd}\partial_e g^{ba}\right), \quad (2.76)$$

and

$$F^a = g^{cd}\Gamma^a_{cd} = \frac{1}{\sqrt{-g}}\frac{\partial}{\partial x^b}\left(\sqrt{-g}g^{ab}\right). \quad (2.77)$$

Note that this collection of terms groups them in such a way that the terms involving F^a can be isolated and ultimately simplified on recourse to harmonic coordinates. Indeed, harmonic coordinates are defined by $F^a = 0$. This choice is advantageous exactly because it allows us to reduce the expression of the Ricci tensor into one that takes a strictly quasilinear form. Therefore we will take that the last term in Eq. (2.75) vanishes in harmonic coordinates. Hence,

$$-\frac{1}{2}g^{cd}\partial_c\partial_d g_{ab} + Q_{ab}(g, \partial g) = R_{ab} = \tilde{R}_{ab} + \frac{1}{4}g_{ab}R. \quad (2.78)$$

Combining this into the equations of motion and treating g_{ab} , \tilde{R}_{ab} , and R as independent variables reduces the system to second order. Notice that Eq. (2.78) and therefore all following equations only hold in harmonic coordinates. The second order set of equations reads

$$\frac{1}{2}g^{cd}\partial_c\partial_d g_{ab} = Q_{ab}(g, \partial g) - \tilde{R}_{ab} - \frac{1}{4}g_{ab}R, \quad (2.79)$$

$$\square R = \frac{1}{36\pi(3\beta - 2\alpha)}R, \quad (2.80)$$

$$\begin{aligned} \alpha\square\tilde{R}_{ab} &= (2\beta - \alpha)\nabla_a\nabla_b R - \frac{1}{16\pi}\tilde{R}_{ab} - \frac{2\beta - \alpha}{128\pi(3\beta - 2\alpha)}g_{ab}R \\ &+ (\alpha - 2\beta)\left[g^{cd}g^{mn}g_{mn,cd} - 2Q(g, \partial g)\right]\tilde{R}_{ab} \\ &+ \frac{\alpha}{4}g_{ab}\left[g^{cd}g_{mn,cd} - 2Q_{mn}(g, \partial g)\right]g^{om}g^{pn}\tilde{R}_{op} \\ &- \alpha\left[g_{cd,ab} + g_{ab,cd} - g_{cb,ad} - g_{ad,cb} + 4g_{mn}\Gamma^m_{c[b}\Gamma^n_{d]a}\right]\tilde{R}^{cd}. \end{aligned} \quad (2.81)$$

Here we have written $2\beta R$ and $R_{abcd} - \frac{1}{4}g_{ab}\tilde{R}_{\rho\sigma}$ in Eq. (2.74) in terms of the metric (using again harmonic coordinates). This set of equations is now only of second order, but the last equation is still not quasilinear (because of terms like $\nabla_a\nabla_b R$ and, $g_{cd,ab}$).

2.3.3.3 Diagonalization and a Quasilinear Hyperbolic System

We can diagonalize the equations to a quasilinear system by introducing additional variables $V_a = \partial_a R$ and $h_{abc} = g_{ab,c}$ and adding derivatives of the first two (already quasilinear)

equations to the system. We obtain

$$-\frac{1}{2}g^{cd}\partial_c\partial_d g_{ab} = -Q_{ab}(g, \partial g) + \tilde{R}_{ab} + \frac{1}{4}g_{ab}R, \quad (2.82)$$

$$g^{ab}\partial_a\partial_b R = \frac{1}{36\pi(3\beta - 2\alpha)}R, \quad (2.83)$$

$$-\frac{1}{2}g^{de}\partial_d\partial_e h_{abc} = \frac{1}{2}g^{de}{}_{,c}h_{abd,e} - Q_{ab,c}(g, \partial g, h) + \tilde{R}_{ab,c} + \frac{1}{4}g_{ab,c}R + \frac{1}{4}g_{ab}V_c, \quad (2.84)$$

$$g^{ab}\partial_a\partial_b V_c = \frac{1}{32\pi(3\beta - 2\alpha)}V_c \quad (2.85)$$

$$\begin{aligned} \alpha\Box\tilde{R}_{ab} &= (2\beta - \alpha)\nabla_a V_b - \frac{1}{16\pi}\tilde{R}_{ab} - \frac{2\beta - \alpha}{128\pi(3\beta - 2\alpha)}g_{ab}R \\ &+ (\alpha - 2\beta)\left[g^{cd}g^{mn}h_{mnc,d} - 2Q(g, \partial g)\right]\tilde{R}_{ab} \\ &+ \frac{\alpha}{4}g_{ab}\left[g^{cd}h_{mnc,d} - 2Q_{mn}(g, \partial g)\right]\tilde{R}{}^{op} \\ &- \alpha\left[h_{cda,b} + h_{abc,d} - h_{cba,d} - h_{adc,b} + 4g_{mn}\Gamma^m{}_{c[b}\Gamma^n{}_{d]a}\right]\tilde{R}{}^{cd}. \end{aligned} \quad (2.86)$$

These final forms are quasi-linear hyperbolic equations (see Eqn. 2.70) as we desired. In order to see this, first consider the vector u_q constituted by all unknowns

$$u_q = (g_{ab}, R, V_a, h_{abc}, \tilde{R}_{ab}), \quad (2.87)$$

The LHS of Eqns. 2.82 - 2.85 already satisfy the form $g^{ab}\partial_a\partial_b u_q$ (which is the LHS of Eqn. 2.70). The Laplacian in Eqn. 2.86 can be written as

$$\Box\tilde{R}_{ab} = g^{cd}\partial_c\partial_d\tilde{R}_{ab} + H_{ab}(\tilde{R}_{ab}, \partial_c\tilde{R}_{ab}, \Gamma^a{}_{bc}), \quad (2.88)$$

where the term $H_{ab}(\tilde{R}_{ab}, \partial_c\tilde{R}_{ab}, \Gamma^a{}_{bc})$ is a linear combination of its arguments. The Christoffel symbols $\Gamma^a{}_{bc}$ depend on the inverse metric and its first derivative. Moving the H_{ab} term into the RHS of Eqn. 2.86 makes the LHS of Eqn. 2.86 also satisfy the form $g^{ab}\partial_a\partial_b u_q$. The RHS of Eqns. 2.82 - 2.86 do not contain any second or mixed derivatives so these terms satisfy the RHS of Eqn. 2.70 $f_q(x, t, u_i, \partial_j u_i)$. Therefore, the system of equations Eqns. 2.82 - 2.86 satisfy the form in Eqn. 2.70 so these are well-posed by Theorem 1.

2.3.4 Results

Many quantum theories including string theory, asymptotic safety, and generic EFTs predict the presence of higher order curvature operators. In QG, these higher order derivative terms are likewise present. Stelle [45] shows that, in linearized theory, the Ricci scalar R and the trace-free Ricci tensor \tilde{R}_{ab} give rise to a massive spin-0 and a massive spin-2 degree of freedom in addition to the usual massless spin-2 graviton. While these massive modes guarantee renormalizability, the kinetic pieces for the massive spin-2 mode and the graviton have opposite signs. This results in a linearly unstable Hamiltonian and is known as the so-called the Ostrogradsky instability.

We have shown that the full nonlinear theory admits a well-posed initial value problem. We might expect that numerically, the resulting nonlinear evolution is stable. Such a conclusion is further supported by the stability of cosmological perturbations [60] as well as the perturbative stability of BH solutions [61]. It would therefore be interesting to understand the interplay in stability between the linear and nonlinear cases and its possible impact on the the Ostrogradsky instability. As a preliminary effort, we first consider a simple spherically symmetry case in order to explore the nonlinear stability of BHs in QG via numerical simulation.

2.3.4.1 Black Hole Stability in Spherical Symmetry

For this preliminary study, we will focus on spherically symmetric systems. We assume a spherically symmetric spacetime in polar areal coordinates

$$ds^2 = -\alpha(r,t)^2 dt^2 + a(r,t)^2 dr^2 + r^2 d\Omega^2. \quad (2.89)$$

The equations of motion, Eqns. 2.82 - 2.86, in spherically symmetric case are obtained by using xAct¹. The script can be found in <https://github.com/hlim88/quadGrav/tree/master/script>. We use the Schwarzschild BH as initial data. We evolve it and check mass conservation of the Schwarzschild BH.

¹<http://www.xact.es/>

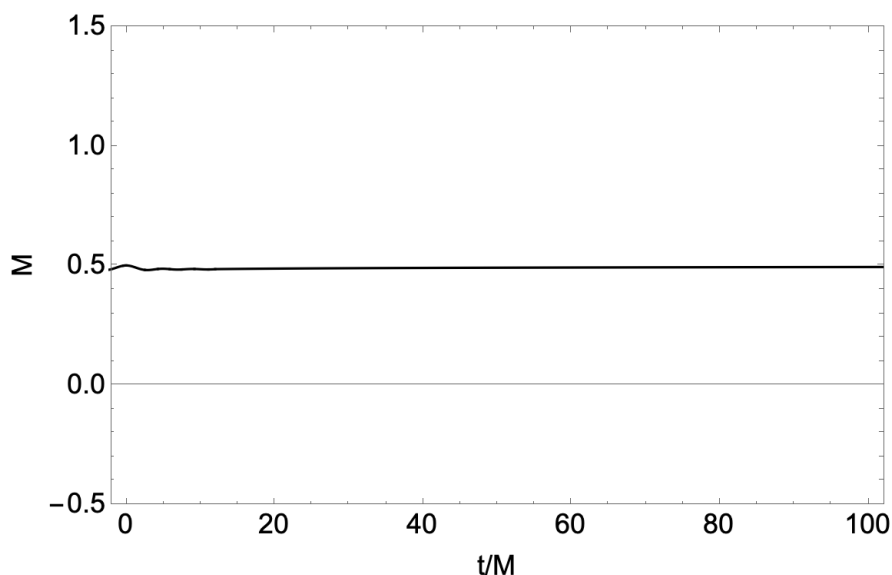


Figure 2.7 Mass conservation test of Schwarzschild type BH. Initial mass of BH is $M = 0.5$. There is small oscillatory behavior at early time but it settles to stable configuration at late time.

Fig. 2.7 shows that mass of the Schwarzschild BH. The initial mass of the BH is $M = 0.5$ and we evolve it for roughly a dynamical crossing time. A small perturbation will be noticed at early times but it quickly settles down to a stable configuration remains so for the duration of the simulation. This result suggests a Schwarzschild black hole is numerically stable in the fully nonlinear regime within a spherically symmetric ansatz. This is, of course, not unexpected because the Schwarzschild solution is itself an analytic solution to quadratic gravity. Nevertheless, it is important to stress that this stability is not, in fact, possible within the linearized version of this theory due to the Ostrogradski instability. Thus, within the spherically symmetric sector of the full, nonlinear theory, we restore stability of a BH solution. One might therefore imagine that having added higher order curvature terms to the standard GR theory has not disturbed some of these BH stability properties.

Another interesting aspect of QG is the existence of non-Schwarzschild families of BH solutions which GR does not admit in static, spherically symmetric spacetimes. It would be interesting to test some of these other possible solutions [62]. Note that there is a perturbative stability analysis of BH in QG in [61].

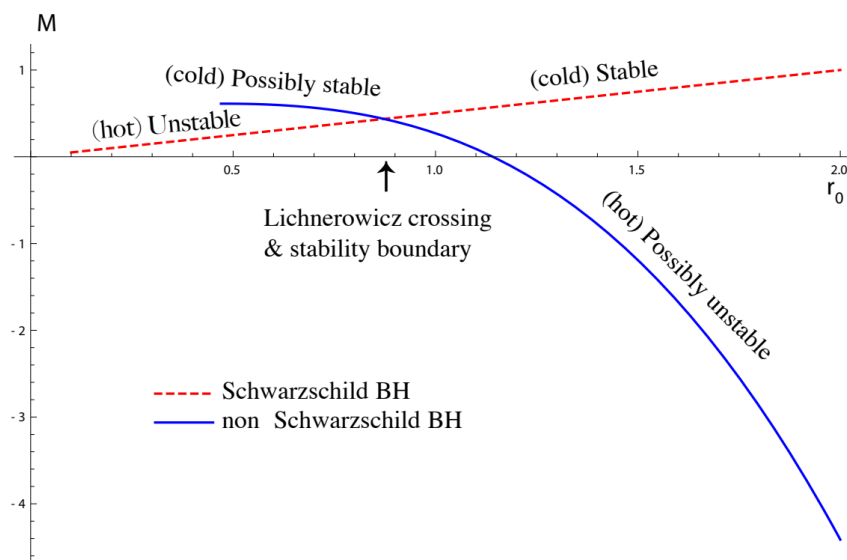


Figure 2.8 Classical stability ranges for Schwarzschild and non-Schwarzschild type BHs. BH masses are plotted as a function of horizon radius r_0 . The figure is adapted from [61]

Fig. 2.8 shows the phase structure of stable and unstable regimes for Schwarzschild and non-Schwarzschild BH families. BH masses are plotted as a function of horizon radius r_0 with a crossing point $r_0^{Cross} \approx 0.876$. This suggests that there exists a threshold for the unstable modes in linear perturbation away from the Schwarzschild and non-Schwarzschild BH families near the Lichnerowicz crossing point. We show a stable evolution of the Schwarzschild BH in the full nonlinear dynamical regime so we can explore further possible (un)stable branches in dynamics and for physical conditions/implications.

2.4 Conclusions

In this chapter, we have explored BH behaviors in two different theories, EMDA and quadratic gravity. For EMDA, we have examined various BHs in order to examine their stability in the presence of different scalar and EM radiation channels. We also performed BBH mergers to obtain GW signature within this theory. From our simulations, we have found several stable BH configurations dynamically using different analytic BH as our

initial data. To our knowledge this is the first demonstration of the existence of different families of BHs that are generalizations of the Kerr-Sen BH across a family of theories different from both GR and low energy string theory. This expands the range of possible alternative/modified theories of gravity that do admit stable BH dynamics and provide possibly more interesting compact objects with beyond-GR physics. Furthermore, our comparison of GW waveforms suggests that deviations from GR will not be significant if we use similar initial parameter as standard GR. However, there might be dramatic effects in GW waveform if we consider certain extreme cases such as large axion field values. It remains possible that such could be revealed by future detection studies and provide a different energy regime for high energy theory. However, the likelihood of the astrophysical relevance of these more extreme cases remains highly questionable.

For quadratic gravity, we have established that well-posed initial value problem using generalized harmonic coordinates exists and can be constructed. This allows us to make the system of equations a quasilinear hyperbolic system. We claim that this method should work for other theories including massive gravity etc. Such will allow standard numerical techniques to be used and eventually provide a means of probing the nonlinear behavior of, say, BHs in these theories. In a move in that directions, we have examined the spherically symmetric case in numerical evolution and established the nonlinear stability (for short times) of the standard Schwarzschild BH in QG. This is an important nonlinear counterpoint to the known linear instability that arises due to the Ostrogradsky theorem. Indeed, it suggests that we might have nonlinear stability in QG. Also we observe the possibility of non-Schwarzschild type solutions in QG (which are not solutions in GR) and show that our nonlinear results do not rule out such new solutions in spherical symmetry. The question of their nonlinear stability, however, remains open.

To sum up, it is safe to say that studying alternative or modified theories of gravity using BH dynamics numerically provides an interesting testbed to check different theories beyond GR and to provide insight into gravitational physics beyond our current standard model, peering even, perhaps, into quantum gravity.

Chapter 3

Intermediate Mass Ratio Inspirals

Using a Wavelet Adaptive Mesh

Refinement Scheme

3.1 Introduction

In 2015, shortly after beginning its first observing run, the Laser Interferometer Gravitational-Wave Observatory (LIGO) [63, 64] made the first direct detection of gravitational waves from the merger of two black holes [65]. Since that time, gravitational waves from nine other binary black hole mergers [2, 3, 4, 5, 6] have been detected by LIGO and the European Virgo detectors [66, 67]. In August 2017, LIGO and Virgo detected gravitational waves from the merger of a neutron star binary [7]. This latter detection was particularly exciting because electromagnetic radiation from the resulting gamma-ray burst was detected by the Fermi Gamma-Ray Burst Monitor [68] and INTEGRAL [69], as well as by several other observatories [70]. The combination of gravitational and electromagnetic observations of binary mergers will give new insight into the physics of black holes (BHs) and neutron stars [71, 72, 73]. As the sensitivity of the LIGO detectors improves, gravitational wave detections will increase in frequency and open a new era of gravitational wave astronomy.

Gravitational waves carry the imprint of their origins within the complicated pattern of their waveform. The information therein can be untangled through a careful comparison of the gravitational wave signal with a library of possible waveforms constructed using approximate methods and results from numerical simulations. Indeed, waveform information from numerical relativity is particularly important for certain binary black hole configurations. Examples include binary black holes with arbitrary spin configurations [74], binaries with orbital eccentricity, and binaries for which the black holes have very different masses [75, 76]. In this chapter, we use q to denote the mass ratio of a binary with $q \equiv m_1/m_2$, where $m_1 \geq m_2$. At this time, very few large mass-ratio BH binaries ($q \gg 1$) have been studied in numerical relativity. This is to be compared to studies with nearly equal mass ($q \approx 1$) [77, 78] which have been simulated much more extensively. Codes developed for $q \approx 1$ binaries are accurate and well tuned, so the problem is well-understood and numerical results are confidently used in the LIGO data-analysis pipeline. However, configurations with large q remain largely beyond the capabilities of current techniques in numerical relativity. Examples include Intermediate Mass-Ratio Inspiral (IMRI) binaries and are characterized roughly by $q \simeq 100$. It is estimated that about 5% of the detections in LIGO might come from these IMRIs [79, 80].

For an IMRI, the size of the smaller black hole adds an extra length-scale to the problem, compared to the $q \approx 1$ case. The need to resolve this new scale, together with the large range of other important length scales, makes this a very challenging computational problem. It requires a highly adaptive and efficient computational algorithm tuned to handle binaries with large mass ratios. While BH evolutions with $q = 100$ have been performed [81, 82, 83] previously, they did not include full merger events or simulated unrealistic head-on collisions.

The central goal of our effort is to create a general purpose framework to study the evolution of spacetimes with black holes or neutron stars, including binary black holes with large mass ratios up to $q \simeq 100$.

Here we present our portable, highly-scalable, extensible, and easy-to-use framework

for general relativity (GR) simulations that will be forward-compatible with next-generation heterogeneous clusters.

We build on our octree-based adaptive mesh refinement (AMR) framework Dendro [84, 85] to support Wavelet Adaptive Multiresolution (WAMR) [86, 87, 88]. The fast wavelet transform can be used to create a sparse representation of functions that retains sharp features and an *a priori* error bound.

In Sec. 3.2, we introduce our wavelet adaptive multiresolution scheme. Sec. 3.3 provides detailed descriptions for the implementation of the DENDRO-GR code. In Sec. 3.5, we present our results. We show the efficiency of our toolkit as well as our preliminary IMRI simulations.

This work described in this chapter has been collaborative work with Milinda Fernando, Hari Sundar, David Neilsen, and Eric Hirschmann. My contribution to it includes implementing the calculation of the Weyl scalar, Ψ_4 , that is required to extract GW waveforms. In addition, I generated all the data used in the comparisons with [EINSTEIN TOOLKIT](#) in Sec. 3.5.3. I also performed all the basic tests to verify stable evolutions of both single and binary BHs.

3.2 Wavelet Adaptive Multiresolution Implementation

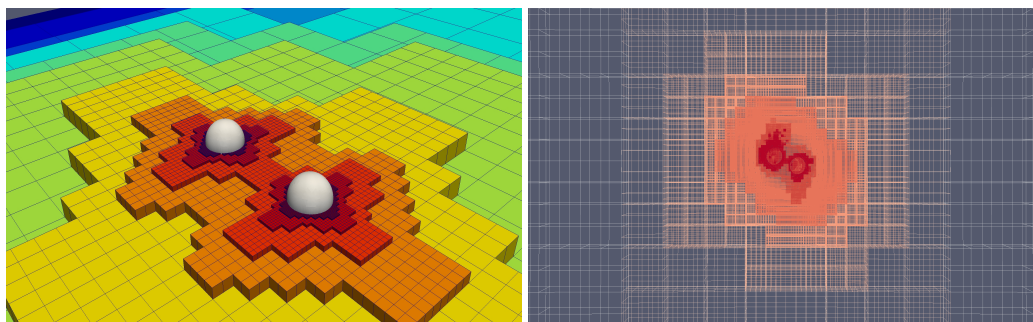


Figure 3.1 (left) A example of the adaptive mesh created by DENDRO for the binary black-hole system. (right) the hierarchical wavelet grids generated for the binary black hole system.

In this section, we describe a detailed implementation of the wavelet adaptive multires-

olution (WAMR) scheme that is used in our code. (See Appendix C for a more detailed derivation and discussion of the sparse wavelet representation through iterated interpolating wavelets.) WAMR uses a basis of interpolating wavelets to create a sparse, quasi-structured grid that naturally adapts to the features of the solution [86, 87, 88]. This grid adaptivity is realized by expanding functions using the fast wavelet transform [89], and thresholding (i.e. based on a user-specified tolerance $\epsilon > 0$) the solution to create a sparse representation that retains small-scale features [90]. In WAMR, we start with the coarsest representation (V_0) of a given function f . Then we compute wavelet coefficients based on the interpolation error as a result of the interpolation of f from the coarser representation to the next finer representation (V_1). Hence the wavelet coefficient denotes the interpolation error that occurs when f is constructed from the immediate coarser level. The sparse representation of the function f is computed based on the user-specified tolerance and removing the spatial points whose wavelet coefficients are within the specified threshold (see Fig. 3.2).

Wavelet basis functions are localized (i.e. have compact support) both spatially and with respect to scale. In comparison, spectral bases are infinitely differentiable, but have global support; basis functions used in finite difference or finite element methods have small compact support, but poor continuity properties. As an example, in Figure 3.1 we show a binary black hole spacetime generated with WAMR using Dendro-GR.

Wavelets encode solution features at different scales very efficiently, a characteristic that leads to many applications in data and image compression [91]. Studies of WAMR have shown the method to be significantly more efficient in terms of computational cost when compared with traditional numerical schemes [92]. The wavelet amplitudes also provide a direct measure of the local approximation error and serve as a refinement criterion. We work simultaneously with both the point and wavelet representations [93, 94, 95, 96, 97, 98]. This gives wavelet methods some of the same advantages as the discontinuous Galerkin method [99, 100], including exponential convergence. Combining the sparse grid generated by the truncated wavelet expansion with DENDRO yields a wavelet adaptive multiresolution method that enables a promising improvement for simulating the mergers of compact object

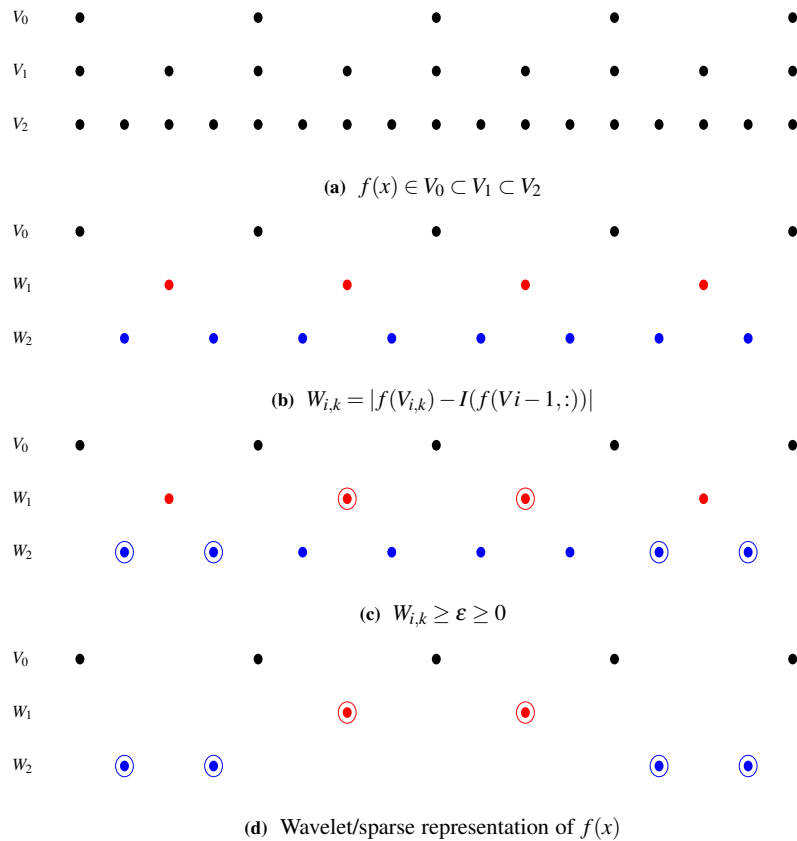


Figure 3.2 For a given function $f : V \rightarrow \mathcal{R}$ let $V_i \subset V$ be the finite dimensional approximation of f (see Figure 3.2a). As the number of nodes increases (i.e. going from V_i to V_{i+1}) for each additional node introduced, we compute wavelet coefficients based on the absolute difference between $f(V_{i,k})$ and the interpolated value from the previous level $f(V_{i-1,:})$ (see Figure 3.2b). Figure 3.2c illustrates those nodes that violate the specified wavelet tolerance ε and are subsequently stored as the sparse/wavelet representation of function f (see Figure 3.2d).

binaries.

3.3 Computational Framework

We now give an overview of the approach that DENDRO takes to obtain excellent scalability in the context of the WAMR method. Our parallel WAMR framework is based on adaptive spatial octrees [101, 102] where the adaptivity is determined by the hierarchical computation of wavelet coefficients and a user-specified tolerance. The construction of adaptive octrees is similar to other octree-codes such that every element at level l gets replaced by eight finer (smaller) elements (i.e. level $l + 1$) if the computed wavelet coefficient is larger than the user-specified tolerance. The main steps in building the parallel octree-WAMR framework are partitioning, construction and enforcement of constraints on the relative sizes of neighboring octants, and meshing. By meshing, we refer to the process of building required data structures to perform numerical computations on a topological octree.

3.3.1 Preliminaries

Octree based spatial subdivisions are fairly common in computational science applications [85, 103, 104, 105, 106], due to their simplicity and scalability. Here we present some basic concepts and notation related to the octrees used in this chapter. A distributed octree \mathcal{T} consists of p subtrees $\tau_i, i = 1, \dots, p$, where p is the number of processes and $\mathcal{T} = \cup \tau_i$. For an octant e , $F(e)$ denotes the faces, $E(e)$ denotes the edges and $V(e)$ denotes the vertices of e . The neighbors of e are given by $N(e) = N_F(e) \cup N_E(e) \cup N_V(e)$, where $N_F(e)$ denotes the octants that share only a face, $N_E(e)$ denotes the octants that share only an edge and $N_V(e)$ denotes the octants that share only a vertex with e . If τ_i spans the sub-domain $\omega_i \subset \Omega$, the boundary octant set, $bdy(\tau_i)$, consists of those octants that share faces, edges and vertices with $\partial\omega_i$. Correspondingly, the set of interior octants is given by $int(\tau_i) = \tau_i \setminus bdy(\tau_i)$. Finally, the octree τ is said to be 2 : 1 balanced if and only if for any $e_k \in \tau$ where $level(e_k) = l_k, \forall e \in N(e_k)$ then $level(e) = \max(l_k \pm 1, 0)$. In this work, we

enforce a 2 : 1 balance constraint on our octrees.

3.3.2 Octree Partitioning

The problem size or the local number of octants varies significantly during WAMR based octree construction (in WAMR we start with the coarsest representation and add grid points until all the wavelet coefficients are within the specified tolerance), balancing, meshing and during the simulation as well.

This necessitates efficient partitioning of the octree to make it load balanced. We use a space-filling curve (SFC) [107] based partitioning scheme [102], specifically the Hilbert-curve. An SFC specifies a surjective mapping from the one-dimensional space to higher dimensional space. This can be used to enforce an SFC based ordering operator on higher dimensional space. The Hilbert ordering maps higher dimensional data (octants) to a 1D curve which makes the process of partitioning trivial. The key challenge is to order the octants or regions according to the SFC, usually performed using an ordering function and sorting algorithm. This approach is easily parallelized using efficient parallel sorting algorithms such as SampleSort [108] and BitonicSort [109] which is the approach used by several state-of-the-art packages [84, 103, 110]. We use a comparison-free SFC sorting algorithm TREESORT, based on the radix sort. In TREESORT, we start with the *root* octant and hierarchically split each dimension while bucketing points for each octant and reordering the buckets, based on the specified SFC ordering (see Fig. 3.3). This is performed recursively on depth-first traversal until we reach all the leaf nodes (see Algorithm 3.1). Additional details on our partitioning algorithm can be found in [102].

3.3.3 Octree Construction and Refinement

The octree construction is based on expanding user-specified functions (e.g. initial conditions for a hyperbolic differential equation) via the wavelet transformation and truncating the expansion (i.e. stopping the refinement at that level) when the coefficients are smaller

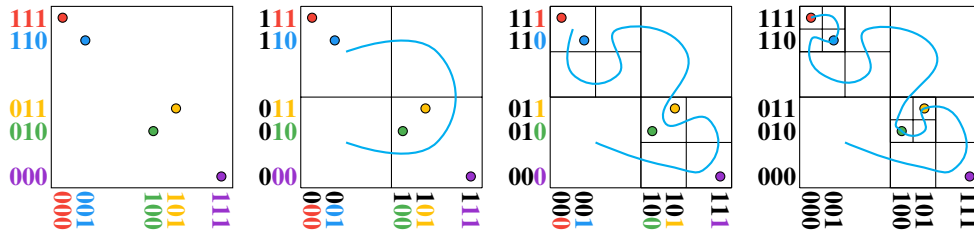


Figure 3.3 Bucketing each point and reordering the buckets based on the SFC ordering at each level l with top-down traversal. Each color-coded point is represented by its x and y coordinates. From the MSD-Radix perspective, we start with the most-significant bit for both the x and y coordinates and progressively bucket (order) the points based on these. The bits are colored based on the points and turn black as they get used to (partially) order the points.

than a user-specified tolerance $\varepsilon > 0$. Intuitively, the wavelet coefficient measures the failure of the field to be interpolated from the coarser level. In distributed memory, all processes start from the root and refine until at least p^2 octants are produced, where p denotes the number of processors. These are equally partitioned across all processes. Subsequent refinements happen in an element-local fashion, and are embarrassingly parallel. A re-partition is performed at the end of construction to load-balance.

3.3.4 2 : 1 Balancing

Following the octree construction, we enforce a 2 : 1 balance condition. This makes subsequent operations (Secs. sec:meshing, 3.3.6 and 3.3.7) simpler without affecting the adaptive properties. Our balancing algorithm [110, 111] is an updated version of the algorithm presented in [84]. The octree is divided into small blocks, that are independently balanced by preemptively generating all balancing octants [111], followed by ripple propagation for balancing across the blocks. The ripple propagation inter-block balancing approach performs poorly at large levels of parallelism, and we instead extend the generation algorithm [111]. The basic idea is to generate all balancing octants for a given octant and then to remove duplicates. While this approach can generate up to eight times the number of total octants, it is very simple and highly parallel. We ensure that the overall algorithm works efficiently, by relying on our TREESORT algorithm to sort and remove duplicates

ALGORITHM 3.1TREESORT

Input: A list of points or regions W , the starting level l_1 and the ending level l_2

Output: W is reordered according to the SFC.

```

1:  $counts[] \leftarrow 0$  ▷  $|counts| = 2^d$ , 8 for 3D
2: for  $w \in W$  do
3:   increment  $counts[child\_num(w)]$ 
4: end for
5:  $counts[] \leftarrow R_h(counts)$  ▷ Permute counts using SFC ordering
6:  $offsets[] \leftarrow scan(counts)$ 
7: for  $w \in W$  do
8:    $i \leftarrow child\_num(w)$ 
9:   append  $w$  to  $W_i$  at  $offsets[i]$ 
10:  increment  $offset[i]$ 
11: end for
12: if  $l_1 > l_2$  then
13:   for  $i := 1 : 2^d$  do
14:    TREESORT( $W_i, l_1 - 1, l_2$ ) ▷ local sort
15:   end for
16: end if
17: return  $W$ 

```

periodically, ensuring that the number of octants generated remains small.

3.3.5 Meshing

By meshing or mesh generation we simply refer to the construction of data structures required to perform numerical computations on topological octree data. We use 4th order Finite Differences (FD) with 5-point stencils for $\partial_i, \partial_{ij}^2$, and 7-point stencils for $\partial_i, i, j \in [1, 2, 3]$ with upwind/downwind and Kriess-Oliger derivatives. We use a Runge-Kutta time integrator with the method of lines to solve the BSSN equations. In this section, we present the data structure choices that we have made and how everything comes together to perform

numerical computations on adaptive octree data to evolve the BSSN equations in time.

Embedding nodal information on an octree: In order to perform FD computations on an octree, we need to embed spatial/nodal points for each octant. Assuming that we want to perform d^{th} order FD computations, we uniformly place $(d+1) \times (d+1) \times (d+1)$ points for each octant. In our simulations, we have used $d = 4$ since we are performing 4^{th} order FD computations, but the meshing algorithms presented in the paper are valid for any integer value of d . The nodes obtained by uniform node placement are referred to as *octant local nodes* (V_D). Octants that share a face or an edge will have duplicate nodal points in V_D , and we need to remove the duplicate nodes from V_D to get *shared octant nodes* (V_S) for several reasons. (1) V_S has a lower memory footprint compared to V_D . (2) A function representation, on a V_D , can be discontinuous due to node duplications, unless function values of duplicate nodes are synchronized. Due to the above reasons, we use V_S as our prime nodal representation (also referred as the *zipped* representation) of the octree. Since the octrees are generated with WAMR, finding the duplicate and hanging nodes (see Figure 3.6) from V_D becomes non-trivial and requires an octant level neighborhood data structure which is referred to as octant to octant (O2O) mapping. Also, we need an additional mapping to map the octants to the V_S representation which is referred to as octant to nodal (O2N) mapping.

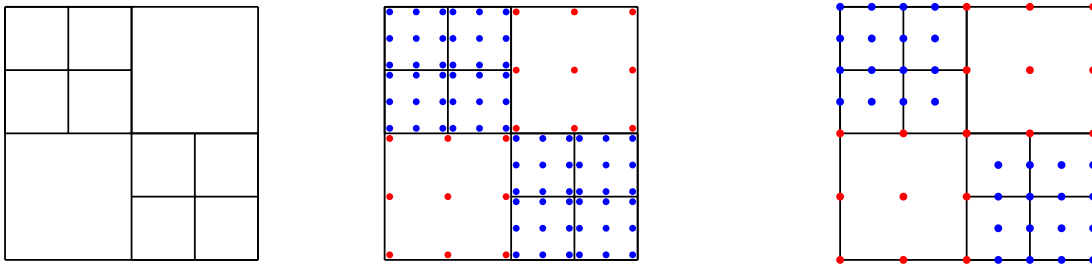


Figure 3.4 A 2D example of *octant local nodes* (in the center) and *shared octant nodes* (the rightmost figure) nodal representation (with $d = 2$, where d denotes the order of FD computations) of the adaptive quadtree shown in the leftmost figure. Note that in *octant local nodes* representation nodes are local to each octant and contain duplicate nodes. By removing all the duplicate and hanging nodes by the rule of nodal ownership we get the *shared octant nodes* representation. Note that the nodes are color coded based on the octant level.

We now describe the methods for building these maps. Note that for the mesh generation,

we assume the input is a complete, ordered and 2:1 balanced octree.

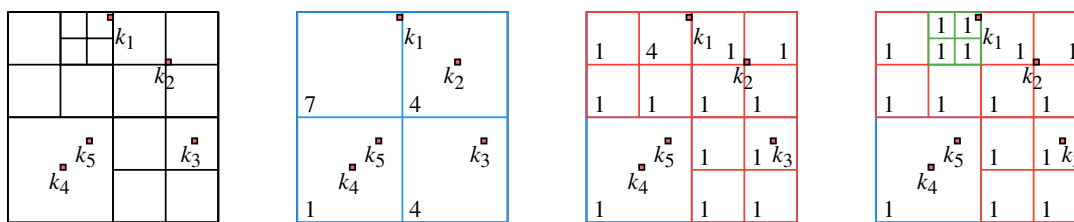


Figure 3.5 For a given ordered octree τ and a set of keys (leftmost figure), `TREESEARCH` performs the traversal in a top-down order over the set of keys, while flagging k_2, k_4, k_5 at the level 1 split, k_3 at level 2 split, and k_1 at level 3 split.

TREESEARCH: amortized search operations on octrees: The common approach for building the maps `O2O` and `O2N` is to generate keys corresponding to the location of neighboring octants and to perform a parallel binary search on the octree and build the maps [84, 103]. Assuming the number of keys we need to search is $\mathcal{O}(n)$, where n is the number of octants, the cost of performing binary searches for all the keys is $\mathcal{O}(n \log(n))$. The $\log(n)$ term corresponding to the binary search is inefficient due to poor memory access and can end up being very expensive for large n . We present an alternative `TREESEARCH`, with better memory access for performing search operations on an ordered octree. To the best of our knowledge this algorithm is new. The approach used in `TREESEARCH` is influenced by radix sort, where we traverse the set of search keys and the octree in the space filling curve (SFC) induced ordering. As shown in Figure 3.5, we start at level 1, split and calculate bucket counts $|b|$ generated by the split while reordering the keys in the same traversal order dictated by the SFC. $|b| = 1$ suggests that octant $e \in \tau$. At this point, b is an ancestor of all keys $k \in b$, and we have found the index for the octant. In contrast with the other approaches, `TREESEARCH` performs a serial traversal over the set of keys and the elements of the ordered octree leading to better memory and cache performance (see Algorithm 3.2). Although the complexity for this approach is still $\mathcal{O}(n \log(n))$, it can be thought of as performing $\log n$ streaming sweeps over the $\mathcal{O}(n)$ octants, leading to better performance.

Ghost/Halo octants: Since the octree is partitioned into disjoint subtrees owned by different processes, we need access to a layer of octants belonging to other processes. These

ALGORITHM 3.2TREESearch: Searching in octrees

Input: ordered octree τ on domain Γ , list keys $\mathcal{K} \in \Gamma$,

Output: list keys $\mathcal{K} \in \Gamma$ with flagged search results.

```

1:  $oct\_counts[] \leftarrow 0$ 
2:  $key\_counts[] \leftarrow 0$  ▷  $|oct\_counts| = |key\_counts| = 2^d$ , 8 for 3D
3: for  $e \in \tau$  do
4:   increment  $oct\_counts[child\_num(e)]$ 
5: end for
6: for  $k \in \mathcal{K}$  do
7:   increment  $key\_counts[child\_num(k)]$ 
8:    $k.result \leftarrow \emptyset$ 
9: end for
10:  $oct\_counts[], key\_counts[] \leftarrow R_h(oct\_counts, key\_counts)$  ▷ Permute counts using SFC ordering
11:  $offsets\_oct[] \leftarrow scan(oct\_counts)$ 
12:  $offsets\_key[] \leftarrow scan(key\_counts)$ 
13: for  $k \in \mathcal{K}$  do
14:    $i \leftarrow child\_num(k)$ 
15:   append  $k$  to  $\mathcal{K}_i$  at  $offsets\_key[i]$ 
16:   increment  $offset\_key[i]$ 
17: end for
18: for  $i := 1 : 2^d$  do
19:   if  $oct\_counts > 1$  then
20:     TREESearch( $\mathcal{K}_i, \tau_i$ )
21:   else
22:     for  $k \in \mathcal{K}_i$  do
23:        $k.result \leftarrow offset\_oct[i]$ 
24:     end for
25:   end if
26: end for
27: return  $\mathcal{K}$ 

```


are commonly known as the *halo* or a *ghost* layer. The computation of ghost octants can be reduced to a distributed search problem where each octant in each partition generates a set of keys that can be searched for to determine the ghost layer. Note that after the ghost-exchange all search operations are local to each partition/process.

Octant to octant map (O2O): Once the ghost layer has been received, we compute the O2O map by performing local searches using TREESEARCH for the neighbors (along the x, y and z axes) of all octants and storing their indices. Therefore for a given octant $e \in \tau$, $O2O(e) = \{e_1, \dots, e_8\}$ will return 8 neighbor octants of e . The algorithm for the O2O map construction is listed in Algorithm 3.3.

ALGORITHM 3.3BUILD OCTANT TO OCTANT: compute O2O

Input: an ordered 2:1 balanced distributed octree \mathcal{T} on Γ , $comm, p, p_r$ of current task in $comm$.

Output: compute O2O

```

1:  $\hat{\tau}_{p_r} \leftarrow \text{COMPUTEGHOSTOCTANTS}(\mathcal{T}, comm, p, p_r)$ 
2:  $O2O \leftarrow \emptyset$ 
3:  $keys[] \leftarrow \text{compute } \mathcal{K}(\hat{\tau}_{p_r})$ 
4: TREESEARCH( $\hat{\tau}_{p_r}, keys$ )
5: for  $key \in keys$  do
6:   if  $key$  is found then
7:      $O2O[key.owner][key.neighbor] = key.result$ 
8:   end if
9: end for
10: return  $O2O$ 

```

Octant to nodal map (O2N): Since we use the *shared octant nodes* (V_S) to store all the simulation variables, we need a mapping between the underlying octree and V_S . The O2N map simply specifies the subset v of *shared octant nodes* (where $|v| = (d+1)^3$) of a given octant $e \in \tau$. When computing O2N we initially start with octant to the V_D map which is trivially constructed by definition of V_D . In order to remove duplicate nodes (see Figure 3.4), we need to define a globally consistent rule of nodal ownership. The

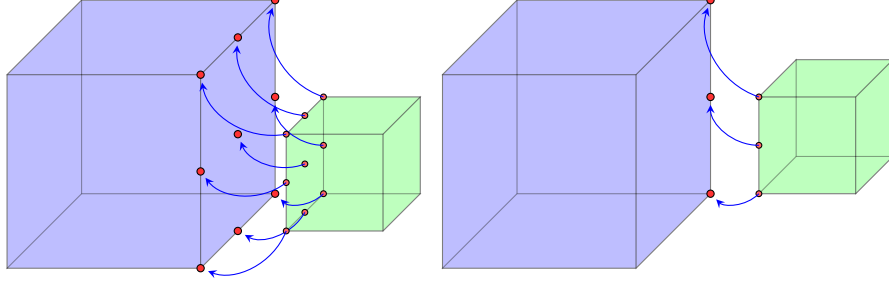


Figure 3.6 An example of a hanging face and a hanging edge where in both cases octant (■) has a hanging face (left figure) and a hanging edge (right figure) with octant (■). Nodes on the hanging face/edge are mapped to the larger octant and the hanging nodal values are obtained via interpolation. Note that for illustrative purposes, the two octants are drawn separately, but are contiguous.

ALGORITHM 3.4BUILD OCTANT TO NODAL: Octant to nodal map generation - O2N

Input: an ordered 2:1 balanced distributed octree \mathcal{T} on Γ , $comm$, p , p_r of current task in $comm$.

Output: compute O2N

- 1: $\hat{\tau}_{p_r} \leftarrow \text{COMPUTE GHOST OCTANTS}(\mathcal{T}, comm, p, p_r)$
- 2: $\text{O2N} \leftarrow \text{initialize}(\mathcal{V}_L)$ ▷ O2N initialized with *octant local nodes*
- 3: $\mathcal{V}_S \leftarrow \mathcal{V}_L \setminus \mathcal{V}_D$
- 4: **for** $e \in \hat{\tau}_k$ **do**
- 5: **for** $v \in N_d(e)$ **do** ▷ $N_d(e)$ denotes $(d+1)^3$ nodes of e
- 6: $owner_idx \leftarrow \text{compute } \mathcal{O}(v)$
- 7: $\text{O2N} \leftarrow owner_idx$
- 8: **end for**
- 9: **end for**
- 10: **return** O2N

ownership of nodes which lie on a hanging face or edge (see Fig. 3.6) will belong to the coarser octant (since they can be interpolated from coarser level) while ties and non-hanging nodal ownership are determined by the SFC ordering of octants. Duplicate nodes are removed from the *octant local nodes* (V_D) to obtain the *shared octant nodes* (V_S) while modifying octant to V_D map to generate octant to V_S map (O2N). The overview of computing O2N map is presented in Algorithm 3.4. By the assumption that the octree is 2:1 balanced, the owner nodes (see Figure 3.6) of hanging nodes cannot be hanging, which simplifies the construction of the O2N mapping.

3.3.6 The *unzip* and *zip* Operations

All simulation variables are stored in their most compact or *zipped* representation, i.e., without any duplication. Due to the use of 2:1 balanced adaptive octrees, performing FD computation on the octree is non-trivial. In order to overcome the above, we use the *unzip* representation (a representation in between *shared octant nodes* and *octant local nodes*). Any given adaptive octree τ_k can be decomposed into a set of regular grid blocks of different sizes—basically a set of octants that are all at the same level of refinement. Due to the memory allocation and performance, we enforce block sizes to be powers of two. In order to perform stencil operations on these blocks, we need information from neighboring blocks, similar to the ghost layer which is required by the distributed case. In the context of blocks, we refer to this layer as the *padding*. During meshing, we compute and save the octree-to-block decomposition, i.e., which octants are grouped together as a block. The computation of octree-to-block decomposition primarily involves a top-down traversal over the local octants and stopping when all elements in the block are at the same level. In order to convert the *zipped* to the *unzipped* representation, we copy the data from the *zipped* representation to the blocks with padding region. This involves copying the data within the block, and copying—potentially with interpolation—from neighboring octants. Nodes on the block boundary which are hanging need to be interpolated during the copy. The 2:1 balance

condition guarantees that at most a single interpolation is performed for any given octant.

The stencil and other update operations are only performed on the block internal while the padding region is read-only. At the end of the update, the simulation variables are *zipped* back, i.e., injected back to the *zipped* representation. This step does not involve any interpolations or communication and is very fast. Note that several key operations such as the RK update and inter-process communications operate using the *zip* representation, and are extremely efficient which is depicted in strong and weak scaling results (see Figures 3.12 and 3.13). There are several additional advantages for the *unzipped* representation. 1). The *unzipped* representation decouples the octree adaptivity from the FD computations. 2). The block representation enables code portability and enables to perform architecture specific optimizations.

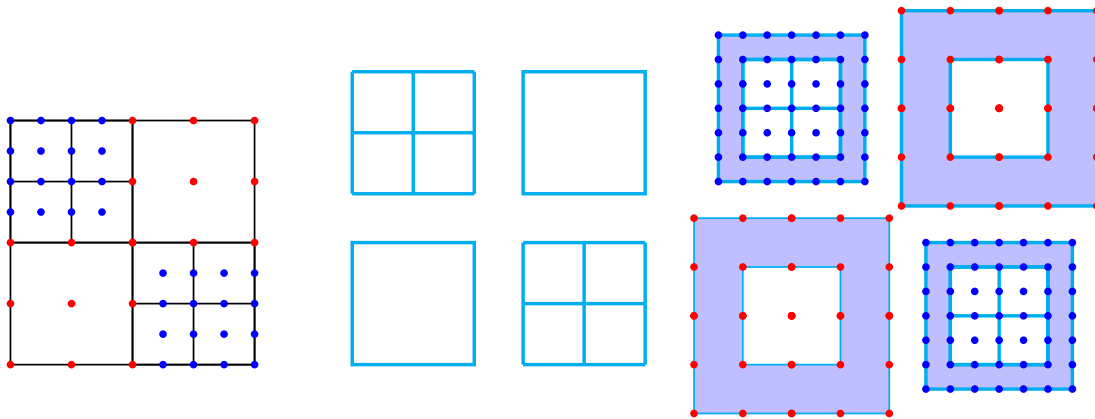


Figure 3.7 A simplistic example of octree to block decomposition and the *unzip* operation. The leftmost figure shows the considered adaptive octree with *shared octant nodes* and its block decomposition is shown in the middle. Note that the given octree is decomposed into four regular blocks of different sizes. The rightmost figure shows the decomposed blocks padded with values coming from neighboring octants with interpolation if needed. In order to perform the *unzip* operation both O2O and O2N mappings are used.

Although, several similar approaches [112, 113] exist for *zip* and *unzip* operations, these approaches rely on structured or block structured adaptivity. In contrast to existing approaches we have designed efficient scalable data structures to perform *zip* and *unzip* operations on fully adaptive 2 : 1 balanced grids.

3.3.7 The *re-mesh* and *inter-grid transfer* Operations

As the BHs orbit around each other, we need to remesh so that maximum refinement occurs around the singularities. We do not enforce maximum refinement at singularities artificially, this is automatically performed by WAMR due to the fact that, BSSN variables might not even be C^0 continuous at BH locations. The *unzip* representation at the end of the time-step is used to determine the wavelet coefficients for each block based on a user-specified threshold. This allows us to *tag* each octant with *refine*, *coarsen* or *no change*. This is used to remesh, followed by a repartition to ensure load-balance. Once the *re-mesh* operation is performed we transfer the solution from the old mesh to the newly generated mesh using interpolations as needed. We refer to this as *inter-grid transfer* and this is done via interpolations or injections at the block level.

3.3.8 Symbolic Interface and Code Generation

The Einstein equations are a set of non-linear, coupled, partial differential equations. Upon discretization, one can end up with 24 or more equations with thousands of terms. Sustainability, code optimizations and keeping it relevant for architectural changes are additional difficulties. To address these issues, we have developed a symbolic interface for DENDRO-GR. Note that there are several significant attempts such as Kranc[114] and NRPy [115] on symbolic code generation for computational relativity due to the complexity of the BSSN equations. We leverage symbolic Python (SymPy) as the backend for this along with the Python package `cog` to embed Python code within our application-level C++ code. The `DENDRO_sym` package allows us to write the discretized versions of the equations similar to how they are written mathematically and enable improved usability for DENDRO-GR users. An example for the BSSN equations are shown in Figure 3.8, with the equations on the left and the corresponding Python code on the right.

There are several advantages to using a symbolic interface like `DENDRO_sym` for the application-specific equations. First, it improves the portability of the code by separating

the high-level description of the equations from the low-level optimizations, which can be handled by architecture-specific code generators. We support avx2 code generators and are working on developing a CUDA generator as well. Since these are applied at a block level, it is straightforward to schedule these blocks across cores or GPUs. Note that the auto-generated code consists of several derivative terms that are spatially dependent as well as other point-wise update operations. We perform common subexpression elimination (CSE) [116, 117] to minimize the number of operations. Additionally, we auto-vectorize the pointwise operations and have specialized implementations based on the stencil-structure for the derivative terms.

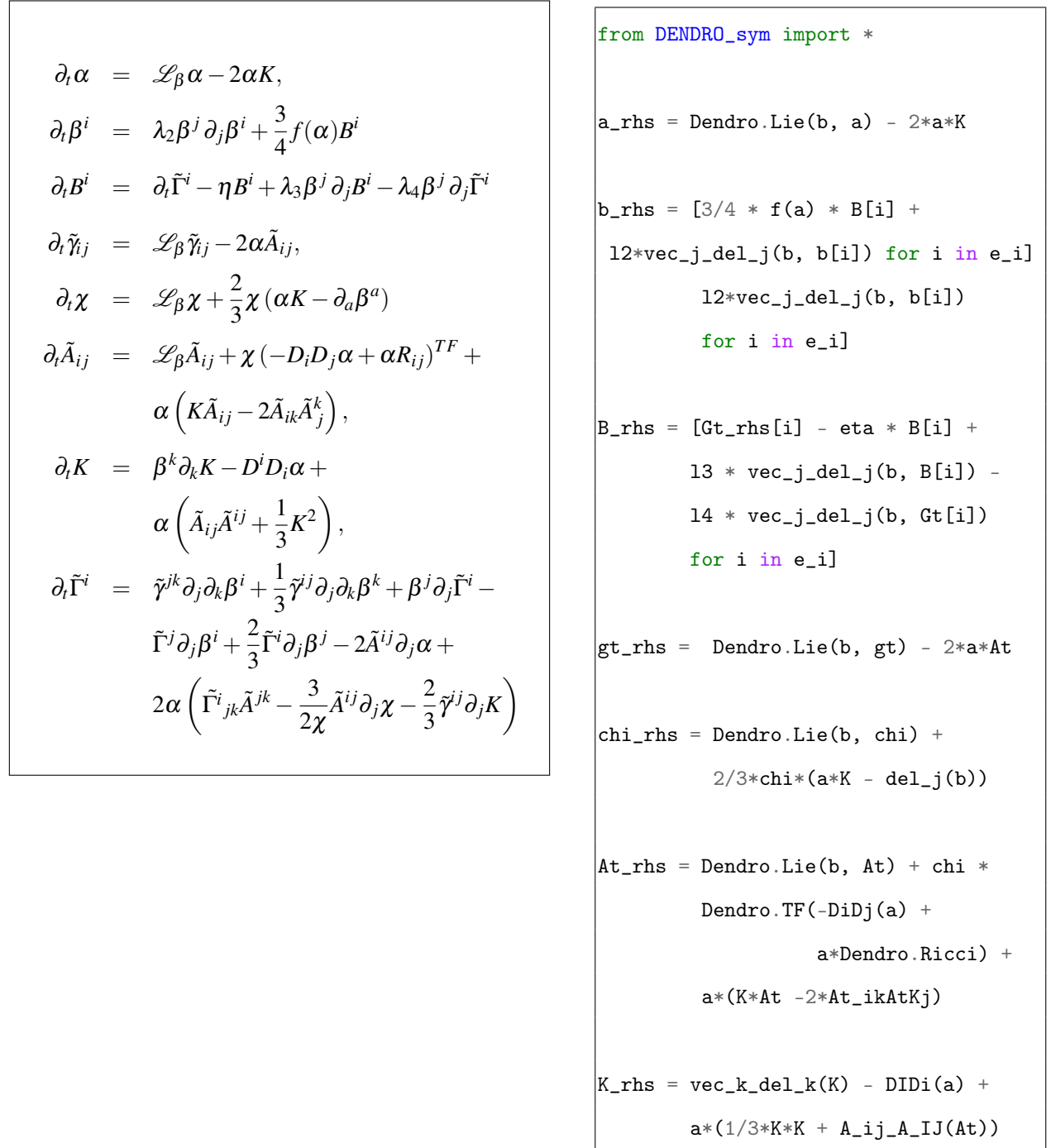


Figure 3.8 The left panel shows the BSSN formulation of the Einstein equations. These are tensor equations, with indices i, j, \dots taking the values 1, 2, 3. On the right we show the `DENDRO_sym` code for these equations. `DENDRO_sym` uses `SymPy` and other tools to generate optimized C++ code to evaluate the equations. Note that \mathcal{L}_β , D , ∂ denote Lie derivative, covariant derivative and partial derivative respectively, and we have excluded $\partial_t \Gamma^i$ from `DENDRO_sym` to save space. (See [118, 119] for more information about the equations and the differential operators.)

ALGORITHM 3.5 Overview of our approach

```

1:  $M \leftarrow$  initialize mesh ▷ §3.3.5
2:  $u \leftarrow$  initialize variables ( $M$ )
3: while  $t < T$  do
4:   for  $r = 1 : 3$  do ▷ RK stages
5:      $B, \hat{u} \leftarrow$  Unzip( $M, u$ ) ▷ §3.3.6
6:     for  $b \in B$  do
7:       Compute derivatives ▷ Machine generated code §3.3.8
8:       Compute  $\hat{u}_{rhs}(b)$  ▷ Machine generated code §3.3.8
9:     end for
10:     $u_{rhs} \leftarrow$  Zip( $M, B, \hat{u}_{rhs}$ ) ▷ §3.3.6
11:    RK update
12:  end for
13:   $t \leftarrow t + dt$ 
14:  if need remesh  $M$  then ▷ §3.3.7
15:     $M' \leftarrow$  remesh( $M$ )
16:     $u' \leftarrow$  Intergid_Transfer( $M, M', u$ ) ▷ §3.3.7
17:  end if
18: end while

```


3.3.9 Putting Everything Together

We use an RK time stepper to perform the time evolution. The algorithmic choices we have made in DENDRO-GR support arbitrary d^{th} order RK time integration. The initial octree is constructed based on the WAMR method until the generated grid converges to capture specified initial data, which is followed by the 2 : 1 octree balancing and mesh generation phase which result in all the distributed data structures that are needed to perform ghost/halo exchange, *unzip* and *zip* operations. A given RK stage is computed by performing *unzip* operations with overlapped exchange of the ghost layer for the evolution variables, computation of the derivatives and right-hand-sides (rhs) using the code generated by the symbolic framework for all local blocks and finally performing *zip* operations to get the computed *zipped* rhs variables. The RK update is then performed on the *zipped* variables. After a specified number of timesteps, we compute the wavelet coefficient for the current solution represented on the grid, and perform *remesh* and *inter-grid transfer* operations if the underlying octree grid needs to be changed. Note that wavelet computation for the grid is a local to each process while *remesh* and *inter-grid transfer* need interprocess communication. A complete outline of our approach for simulating binary BH mergers demonstrating how the various components come together is listed in Algorithm 3.5

3.4 Other Numerical Methods

There is an extensive literature on solving the BSSN equations in general relativity, and some general reviews include [119, 120, 121]. In this section we briefly outline our particular choices for solving the BSSN equations. We write the BSSN equations in terms of the conformal factor χ [122]. We use the parameterization of the “1 + log” slicing condition and the Γ -driver shift used in [123]. Spatial derivatives are calculated using finite difference operators that are $O(h^4)$ in the grid spacing, h , with upwind derivatives for Lie derivative terms [124]. We calculate derivatives for the Ricci tensor and enforce

the algebraic constraints as described in [125]. Outgoing radiative boundary conditions are applied to each BSSN function. The BSSN equations are integrated in time using an explicit RK scheme. The solution at each point is integrated with a single global time step, that is set by the smallest grid spacing and the Courant condition [126]. While we support 3rd and 4th order RK, the tests in this work were done using 3rd order RK with a Courant factor of $\lambda = 0.1$. Kreiss-Oliger dissipation is added [119, 127] to the solution to eliminate high-frequency noise that might be generated near the black hole singularities.

3.5 Results

In this section we perform a thorough evaluation of our code, including detailed comparisons with the [EINSTEIN TOOLKIT](#). We first describe the machines used for these experiments followed by results demonstrating the improvements to DENDRO and comparisons with [EINSTEIN TOOLKIT](#). Finally, we push our code to the limit of extreme adaptability to demonstrate its capability, using cases that are currently—to the best of our knowledge—beyond the capability of [EINSTEIN TOOLKIT](#).

Experimental Setup: The large scalability experiments reported in this chapter were performed on [Titan](#) and [Stampede2](#). [Titan](#) is a Cray XK7 supercomputer at Oak Ridge National Laboratory (ORNL) with a total of 18,688 nodes, each consisting of a single 16-core AMD Opteron 6200 series processor, with a total of 299,008 cores. Each node has 32GB of memory. It has a Gemini interconnect and 600TB of memory across all nodes. [Stampede2](#) is the flagship supercomputer at the Texas Advanced Computing Center (TACC), University of Texas at Austin. It has 1,736 Intel Xeon Platinum 8160 (SKX) compute nodes with 2×24 cores and 192GB of RAM per node. Stampede2 has a 100Gb/sec Intel Omni-Path (OPA) interconnect in a fat tree topology. We used the SKX nodes for the experiments reported in this work.

Implementation Details: The DENDRO-GR framework is written in C++ using MPI. The

symbolic interface and code generation module uses symbolic Python (SymPy). In the comparisons with the [EINSTEIN TOOLKIT](#), we have used Cactus v4.2.3 and the Tesla release of the Einstein Toolkit. We integrated the BSSN equations with third-order RK for all comparisons in this chapter. The constraint analysis, apparent horizon finder, and output were turned off for these runs.

unbalanced octants	balanced octants	2:1 balance ([84]) (s)	2:1 balance (DENDRO-GR) (s)
3K	5K	0.0087	0.0043
33K	59K	0.0908	0.0541
338K	553K	0.7951	0.5461
3M	5M	7.7938	6.9313
6M	11M	16.1828	14.7374

Table 3.1 A comparison study for the 2:1 balancing approach used in [84] and the new balancing approach on a single core in [Stampede2](#) (SKX node), with varying input octree sizes ranging from 3K to 6M octants.

3.5.1 Meshing Performance

In this section, we briefly present results for the improved scalability and performance of the proposed balancing and meshing algorithms. In Table 3.1, we list the improvement in enforcing 2:1 balancing by using TREESORT instead of the ripple propagation used in [84]. We present only single core results, as the algorithmic changes are for the sequential portions of the algorithms. In Table 3.1 we demonstrate significant savings for a range of problem sizes.

Similarly, significant savings are also obtained by the use of TREESEARCH compared to the use of binary searches for the various search operations needed for meshing. In this experiment, we searched for k keys in an array of size $n = 33M$ octants. This size was chosen based on the average grain size we used in our experiments. Note that for meshing, $k = \mathcal{O}(n)$, and therefore we plot results up to $k = n$. Having a large array that is being searched in (large n), results in the first few steps of the binary search resulting in cache misses affecting overall performance. TREESEARCH utilizes the deep memory hierarchy in

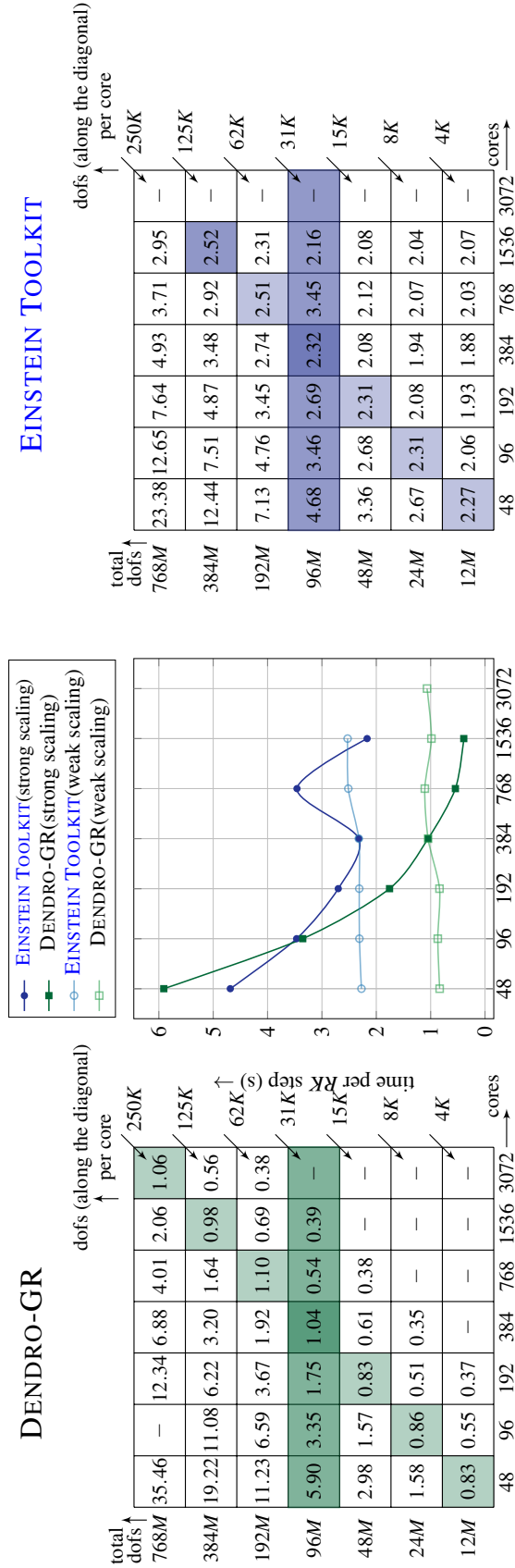


Figure 3.9 Comparison between **EINSTEIN TOOLKIT** and **DENDRO-GR** without factoring in adaptivity (i.e. both **EINSTEIN TOOLKIT** and **DENDRO-GR** support uniform grids.). For a fixed tolerance, we expand the domain for a 1 : 1 mass-ratio simulation such that both **EINSTEIN TOOLKIT** and **DENDRO-GR** have roughly the same number of dofs. We present both weak and strong scaling results using both codes. On the left table are results from **DENDRO-GR** and from **EINSTEIN TOOLKIT** on the right. In the middle, we plot a representative strong and weak scaling curve for each code. The **DENDRO-GR** scaling is plotted in green (lighter shade for weak) and blue for **EINSTEIN TOOLKIT**. The corresponding data entries are also marked in the tables. Note that the rows represent strong scaling and the diagonal entries represent weak scaling results and runtime is reported in seconds(*s*).

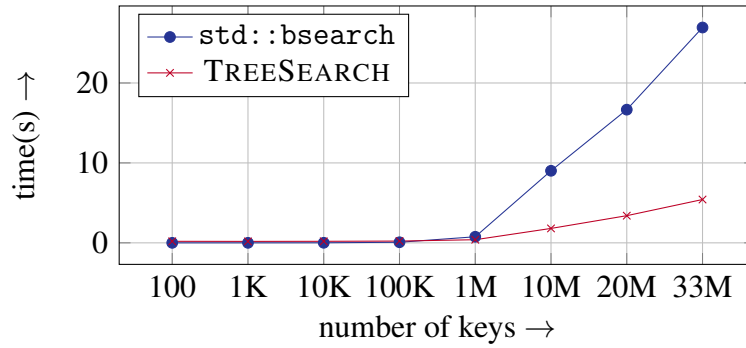


Figure 3.10 Comparison of `std::bsearch` with partial ordering operator `<` and comparison free `TREESEARCH` approach for performing, varying number of keys on $33M$ sorted complete octree using single core in [Stampede2](#) SKX node.

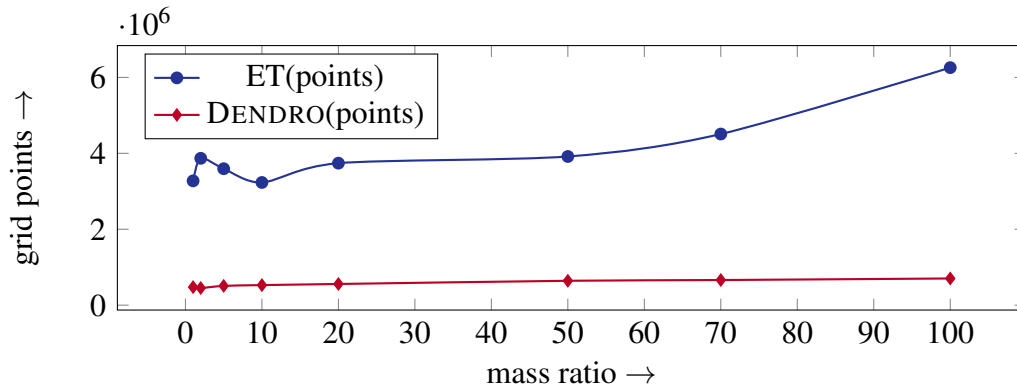


Figure 3.11 Comparison between ET and DENDRO-GR for number of spatial points with increasing mass ratios. Note that these are not from complete simulations and the size of the problem as well as the time per RK-step is likely to increase, but it illustrates the rate of increase for both approaches. Parameters for the above experiment generated such that total mass of black holes equals to 1 and the separation distance is 32 for all cases and `MAXDEPTH` is set in a way that the spatial discretization $dx < \frac{\min(m_1, m_2)}{16}$ where m_1, m_2 denotes the individual masses of black holes.

a more effective fashion, but because of the additional work involved in sorting, requires a minimum number of keys to be searched for before it is cheaper. This can be seen in our results plotted in Figure 3.10, where `TREESEARCH` scales better than `std::bsearch`, and is faster for $k > 1M$. Given the number of keys being searched for during meshing, `TREESEARCH` improves the overall meshing performance and scalability significantly.

3.5.2 Correctness of the Code

We performed a number of tests to assess the correctness of our code, including simulations of static (Appendix E.1.2.8) and boosted black holes (Appendix E.1.2.8.1), as well as comparisons to other codes. To verify the automatic generation of the computer code for the BSSN equations, we evaluated these equations using arbitrary analytic functions of order unity over a grid of points, and compared the results to a known solution. The L2-norms of the error are equivalent to machine zero, given the limitations of finite-precision arithmetic. Additional results on the correctness are presented in Appendix E.1.2.

3.5.3 Comparison with the Einstein Toolkit

We compare our code with the BSSN formulation implemented in ET. The AMR driver for Cactus, CARPET [128], only supports block adaptivity. Therefore, we would expect DENDRO-GR to require fewer degrees of freedom (dof) for a given simulation and consequently be faster. Although ET uses vectorization for improved performance [129], DENDRO-GR outperforms ET. Note that for all comparison studies with [EINSTEIN TOOLKIT](#), we have used the non-vectorized version of the DENDRO-GR generated code. To highlight the improvements over ET, we perform two independent experiments. First, we compare the performance of both codes without adaptivity, and then in a separate experiment we show that DENDRO-GR provides a more efficient adaptivity and scaling than CARPET.

Uniform Grid Tests: In Figure 3.9, we present a comparison between DENDRO-GR and ET. This uses a regular grid for ET, and serves to highlight the efficiency of the DENDRO-GR code including the overhead of adaptivity (i.e., *zip* and *unzip*). These runs were performed on [Stampede2](#), and we show strong and weak scaling for both codes. Although both codes demonstrate good scaling, the performance (as measured by the time for one complete timestep) is better for DENDRO-GR at higher core counts. [EINSTEIN TOOLKIT](#) has better performance for large number of unknowns per core at low core



Figure 3.12 Weak scaling results in ORNL’s Titan for $RK/(dof/p)$ (averaged over 10 steps) where RK, dof, p denotes the time for single RK step, degrees of freedom, and number of cores respectively, with derivative computation (deriv), right hand side (rhs) computation, unzip cost, wavelet computation (wavelets) and communication cost (comm) with the average of 1.41M unknowns per core where the number of cores ranging from 32 to 131,072 cores on 8,192 nodes where the largest problem having 206 Billion unknowns. Above results are generated with mass ratio $\mu = 10$ with MAXDEPTH 18 and wavelet tolerance of 10^{-6} . Note that the unknowns per core have a slight variation since with WAMR we do not have explicit control over the grid size and WAMR decides the refinement region on the mesh based on the how wavelets behave during the time evolution. This is why we have reported normalized RK with dof/p metrics to report accurate weak scaling results.

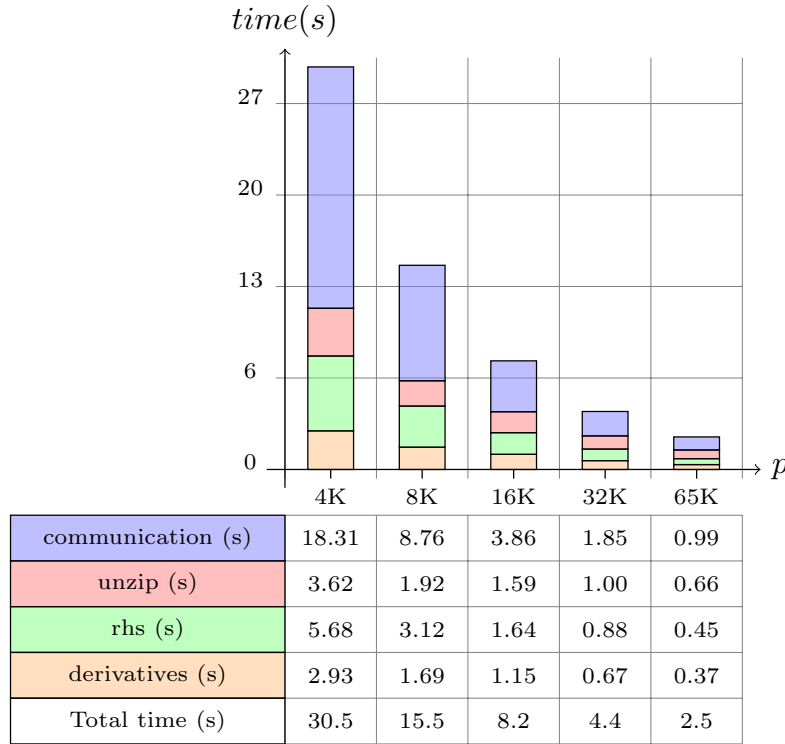


Figure 3.13 Strong scaling results in ORNL’s *Titan* for a single RK step (averaged over 10 steps) with derivative computation (*deriv*), right hand side (*rhs*) computation, *unzip* cost and communication cost (*comm*) for a fixed problem size of $10.5B$ unknowns where the number of cores ranging from 4,096 to 65,536 cores on 4096 nodes. Note that for strong scaling results re-meshing is disabled in order to keep the problem size fixed.

counts. This effectively captures the overhead of *zip* and *unzip* compared to an efficient and mature code. For cases with higher core counts as well as smaller number of unknowns per core, DENDRO-GR performs better. This is largely due to better cache utilization due to blocking, that largely compensates for the overhead of *zip/unzip*. Additionally, DENDRO-GR demonstrates better strong and weak scaling. The plot in Figure 3.9 highlights the strong and weak scaling of both codes for a representative grain and problem size. DENDRO-GR scales well far beyond the 3072 cores, as shown in Figure 3.12 and discussed in Sec. 3.5.4.

Octree Adaptivity vs. Block Adaptivity: As motivated earlier, we wish to perform simulations of binary black hole mergers with large mass-ratios, $q \simeq 100$. Large mass-ratios require extensive refinement, increasing the number of spatial degrees of freedom. For an example let’s assume an equal mass binary requires a certain resolution for the BSSN equations, about 100 points per BH in each dimension. For a BH 100 times smaller, we need

mass ratio q	1	2	5	10	20	50	70	100
cores	27	31	64	79	89	90	99	96

Table 3.2 The number of cores required to maintain 24K unknowns per core with different mass ratios using MAXDEPTH 12, a wavelet tolerance of 10^{-4} and a black hole separation distance of 32.

a resolution equivalent to 1/10,000 times the total mass and $100\times$ more time steps. The consequent high computational cost is the primary reason that current catalogs of numerical waveforms contain templates with a maximum mass ratio of $q < 10$ [130, 131, 132].

The efficient adaptivity of DENDRO is a big advantage over the block adaptivity of the ET. To assess the effect of adaptivity on code performance, we compared both codes for increasing mass-ratios from $q = 1$ –100, measuring the *dofs*, as shown in Figure 3.11. As the mass ratio increases, the number of *dofs* for ET increases much more rapidly than for DENDRO-GR. While these results capture the differences at the beginning of the simulation¹, they are representative of the full simulation in terms of comparing the two codes. Because of better adaptivity and better scalability, DENDRO-GR keeps the cost of a single RK-step fairly flat as we scale up from $q = 1$ –100 mass-ratio. In Table 3.2 we present the number of cores needed to maintain a 24K unknowns per core with increasing mass ratio using the DENDRO-GR framework. Finally, to illustrate the advantages of octree-adaptivity over block-adaptivity, even with the overhead of *zip/unzip*, we plot the reduction in runtime for 10 timesteps for different levels of adaptivity in Figure 3.14. Based on these results, it is clear that *zip/unzip* together with adaptivity will be a benefit for simulations of large mass ratio configurations and that the overhead is about 20% even for a full regular grid simulation. These results further support our findings in Figure 3.9.

3.5.4 Parallel Scalability

Scalability is a key requirement for large-scale codes such as DENDRO. In the context of our target application this could be in order to run simulations of black hole mergers with large

¹It would be very expensive to run full simulations with ET due to its scalability for large q .

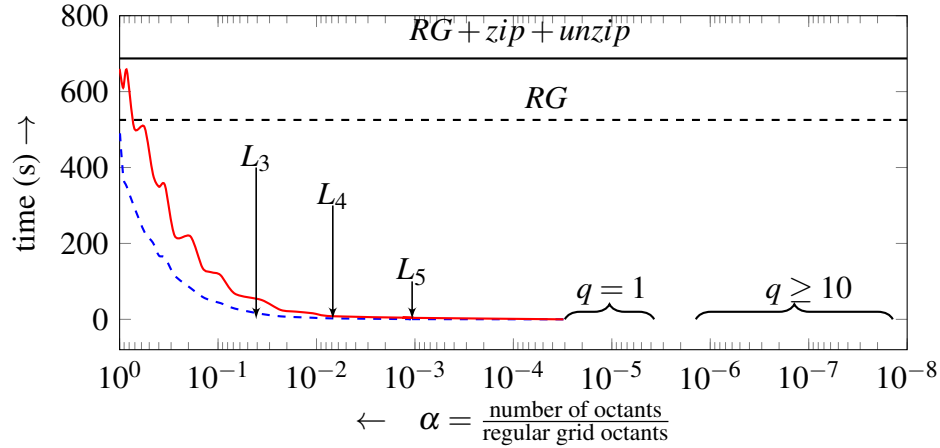


Figure 3.14 An illustrative single core example to evaluate the overhead of zip/unzip operations to evaluate BSSN equations on a sequence of octree grids over 10 timesteps. The parameter α here denotes the ratio between the number of octants to the number of regular grid octants for MAXDEPTH 8. Hence moving towards the right direction on the x axis the octree grids converge towards a regular grid. RG and $RG + zip + unzip$ denotes the baseline performance for regular grid computations and regular grid computation with zip/unzip overhead respectively. The shaded $q = 1$ region denotes the α value for an equal mass ratio simulation until the merger event using MAXDEPTH 12 over 2×10^5 timesteps. For larger mass ratio runs the MAXDEPTH will be determined by the smaller black hole, hence for $q = 10$ with 15, α reached a maximum value of 1.4×10^{-8} over 6000 time steps and for $q = 100$ with MAXDEPTH 20, the maximum value of α reached 5.51×10^{-13} over 1000 timesteps. Note that the value of α can increase during the simulation. For the largest problem (see Figure 3.12) that was run on Titan with 131K cores for $q = 10$ case the α value was 4.8×10^{-7} . In the plot, we have marked the L_k values for $k = 3, 4, 5$ where L_k is the computed α ratio for an adaptive octree for which an equal number of octants spans across k levels.

mass ratios, or to be able to run these simulations much faster. These cases correspond to weak and strong scaling respectively. As demonstrated in Sec. 3.5.3, DENDRO-GR scales well in both cases. In this section, we demonstrate the ability of DENDRO-GR to scale to much larger core-counts for weak scaling. In Figure 3.12, we demonstrate weak scalability to 131,072 cores on [Titan](#) using 1.5×10^6 dofs/core for a largest problem of 206×10^9 dofs. These correspond to a simulation of $q = 10$, with increasing levels of refinement. There is a slight fluctuation in the number of unknowns per core at each problem size. Therefore, we report the average time for a single RK-step/ dof/p . We can see that the average time per RK-step/ dof/p remains less than $10\mu s$. We note that the unzip costs are high at the lower core-counts and stabilize at higher core counts. This is due to a larger grain size at the smaller core-counts. The communication for a smaller number of processes is lower due to architectural reasons, as the majority of the communication happens primarily within the same node. This is common for most codes, and stabilizes for $p > 512$ as the communication starts to get dominated by inter-node communication rather than intra-node communication.

In Fig. 3.13, we present the strong scaling results for DENDRO-GR to perform a single RK step (i.e. averaged over 10 steps) for a fixed problem size of $10.5B$ unknowns in [Titan](#) up to 4096 nodes. This experiment is for a BBH merger with a mass ratio 10 with a wavelet tolerance of 10^{-6} . Note that for the strong scaling experiment we have disabled the *re-mesh* and *inter-grid* transfer operations in order to keep the problem size constant.

3.5.5 Large Mass Ratios

Finally, we present some representative images from simulations at mass ratios of $q = 1, 10$, and 100 in Figure 3.15. Here we plot one of the evolved variables, χ , (related to the conformal factor on the 3D spatial slice) along with the adaptively refined mesh. We can observe the increased refinement needed to handle the increased mass ratio.

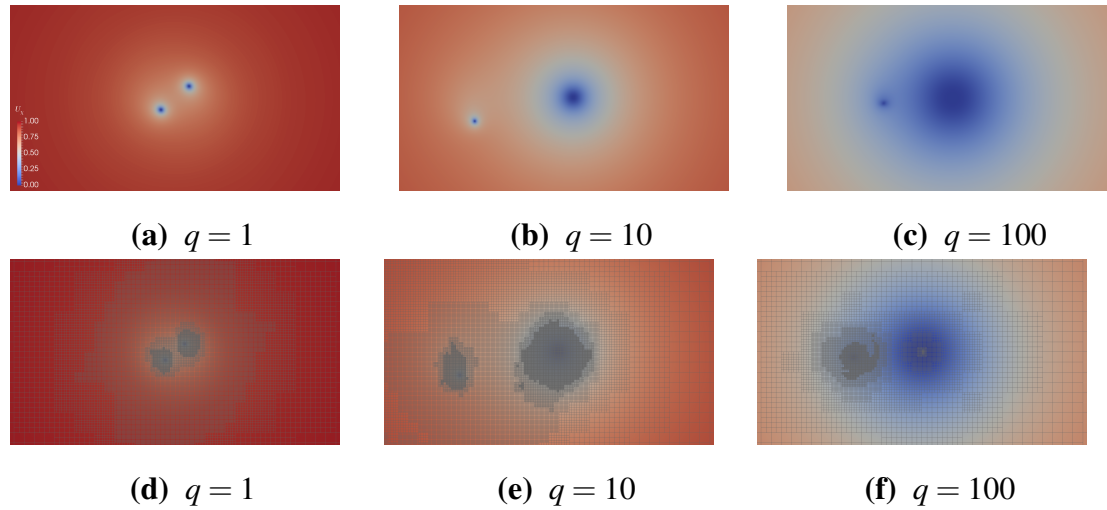


Figure 3.15 Time step snapshots of the binary black hole problem of black hole mass ratios 1, 10 and 100 where we in the top row we plot the BSSN variable χ in the lower row we plot the WAMR grids for each case at that specific instance.

3.6 Conclusions

In this chapter, we have presented a highly-adaptive and highly scalable framework for relativistic simulations. By combining a parallel octree-refined adaptive mesh with wavelet adaptive multiresolution and a physics module to solve the Einstein equations of general relativity in the BSSN formulation, we were able to perform simulations of IMRIs of binary black holes with mass ratios on the order of 100:1. In designing our framework and methods, we have focused on ensuring portability and extensibility by the use of automatic code generation from symbolic notation. This enables portability, since new architectures can be supported by adding new generators rather than rewriting the application. The code is extensible as new applications can be created by using the symbolic interface without having to focus on writing scalable code. Since both of these are achieved using Python, this reduces the cost of extending and porting the framework, especially by non-specialist programmers. We also made improvements to fundamental algorithms required for generating octree-based meshes, specifically an improved 2:1 balancing algorithm and a scalable search algorithm fundamental to constructing meshing data-structures. Finally, we performed extensive comparisons with the current state-of-the-art codes and demonstrated

excellent weak and strong scalability.

In the short time that LIGO and Virgo have been searching for gravitational waves, we have already learned exciting things about neutron stars [133, 134], the production of heavy elements (such as gold) [135], and the population of black holes in the universe [136].

The full scientific impact of multi-messenger astronomy is only realized when the observations are informed by sophisticated computer models of the underlying astrophysical phenomena. DENDRO provides the ability to run these models in a scalable way, with local adaptivity criteria using WAMR. While AMR codes with block-adaptivity typically lose performance as the number of adaptive levels increases, DENDRO achieves impressive scalability on a real application even with many levels of refinement. The combination of scalability and adaptivity will allow us to study the gravitational radiation from IMRIs without simplifying approximations in direct numerical simulations.

Chapter 4

Ejecta Morphology from Binary Neutron Star Mergers

4.1 Introduction

Recently, Advanced LIGO has detected gravitational waves (GWs) from a binary neutron star merger (BNS) [7]. BNS mergers have several interesting aspects. The problems that we are interested in concern high resolution studies of winds and ejecta from binary mergers of neutron stars and a companion which could be a black hole (BH), neutron star (NS), or white dwarf (WD). The ejecta from such mergers provide an ideal environment for r -process nucleosynthesis as the heaviest r -process nuclei may be formed in binary mergers rather than supernova explosions. Understanding the ejecta from binary mergers is particularly interesting because r -process nucleosynthesis can power isotropic electromagnetic emission that may be observable for several days after the merger. This emission, called a macronova or kilonova [137, 138, 139, 140, 141, 142], would provide an important electromagnetic counterpart to gravitational wave detections, and help to isolate gravitational wave sources and provide important complementary information about these mergers and the resulting gamma ray burst. (For possible observations of a macronova, see [143, 144].) The dynamics of the ejecta is complicated with different properties and compositions depending on the

process by which it was expelled and its interactions with the hot merger remnant. The low density of the ejecta, the small-scale structure, the neutrino interactions, and the fact that it is not localized makes the ejecta particularly difficult to resolve in a convergent manner.

In this work, we study BNS merger systems. We solve the equations of hydrodynamics within Newtonian gravity using smoothed particle hydrodynamics (SPH) with an extremely large number of particles. We present a new open source toolkit [FleCSPH](#). FleCSPH is a complement of the [FleCSI](#)¹ framework, focusing on tree data structures with support for binary, quad and octrees. The framework provides parallel, distributed and accelerated tree construction and search in the context of multi-physics problems. FleCSI is a compile-time configurable framework designed to support multi-physics applications and is developed and maintained by the Los Alamos National Laboratory. FleCSI provides domain scientists with a set of data structures and tools to target parallel and distributed architectures on current and future supercomputers, including the ongoing 2020 target to support the first Exascale supercomputers. In this regard, high performance computing (HPC) faces some increasingly difficult challenges. For example, we need to provide software systems that harness the full power of these machines composed of complex hybrid architectures, next generation runtimes, accelerators, and complex memory hierarchies. In addition, these systems must promote performance portability, programmer productivity and provide abstractions that are suitable for domain scientists.

Our work on FleCSPH is based on an SPH that emphasizes different hybrid architectures as well as runtimes in HPC. This method can be efficiently solved using binary, quad and octrees while providing irregularities in terms of computation and communication.

This project described in this chapter has been a collaborative effort with many people at LANL. These include Julien Loiseau, Oleg Korobkin, Jonah Miller, Irina Sargert, Wes Even, and Ben Bergen. My contributions to this work include being a founder and one of the main developers of the FleCSPH code. The FleCSPH code project was created during my 2017 summer graduate internship with another summer student, Julien Loiseau, who

¹Visit <https://github.com/laristra/flecsi> to learn more about FleCSI

is a computer scientist. My major task was to implement and develop all the necessary physics and mathematics components for FleCSPH. I also implemented several different toolkits including visualization and data format transfer for the code. (See Appendix D for more details and a discussion on how to implement FleCSPH.) I also tested and generated all the results including the Sod shock, the Sedov blast wave, and the BNS system; not only validating the code but also obtaining interesting results.

4.2 Numerical Method and Implementation

In this section, we present our numerical technique and the details of its implementation. This section also includes a detailed discussion about our toolkit and its basic structure for solving binary neutron star systems. For detailed computer science implementation aspects, see Appendix D.

4.2.1 Smoothed Particle Hydrodynamics

Smoothed Particle Hydrodynamics (SPH) is a numerical and computational method used for simulating fluid flow. It works by dividing the fluid into a set of discrete elements, called particles. These particles have a spatial distance, known as the smoothing length, over which their properties are smoothed by a kernel function. SPH was originally suggested by Gingold and Monaghan [145] and Lucy [146] initially for astrophysics problems. There are already a number of numerical methods handling problems in ideal fluid dynamics. However, SPH has several advantages for astrophysical and cosmological simulations.

First, SPH allows us to exactly conserve mass, energy momentum, and angular momentum in a straightforward way; indeed by construction. As a result, simulations are independent of the numerical resolution. For example, the mass transfer between two stars and the orbital dynamics of two stars are extremely sensitive to the conservation of angular momentum. Eulerian methods, which are grid-based methods, have challenges in accurately simulating this problem. Second, natural adaptivity comes essentially for free

with SPH. The scale of the smoothing which is set by the smoothing length h varies as the particles move with the flow without any additional infrastructure such as adaptive meshes as in Eulerian methods. Another advantage is that SPH is designed to ignore vacuum regions thus greatly reducing computational costs as vacuum regions do not need to be evolved. Furthermore, SPH is Galilean invariant and so is independent of the chosen computing frame. If simulations are performed in inappropriate reference frames with Eulerian schemes, serious artifacts such as spurious inspiral and merger for binary systems can occur. In contrast, it does not matter in which reference frame the calculation is performed for SPH.

Another benefit of using SPH is that physical properties attached to SPH particles are simply carried along as the particles move. This property is helpful to post-process the nucleosynthesis from a particle trajectory without any need for additional tracer particles. High velocities with respect to the computational grid can cause inaccurate simulations otherwise. In addition, this property makes advection effectively automatic so SPH is free of any errors from advection alone.

The particle structure of SPH allows for a natural computational structure to solve the N-body problem. SPH can be straightforwardly combined with highly flexible and accurate gravity (Newtonian) solvers such as the tree method. Many successful SPH methods with tree algorithms exist to solve a variety of different astrophysical problems. More details about these implementations can be found in the literature [147, 148, 149, 150, 151, 152, 153, 154].

To formulate and simulate different astrophysical and cosmological problems, different sets of equations are chosen with appropriate physical assumptions. In the work described in this chapter, we will solve the hydrodynamical equations with Newtonian gravity.

The method, as illustrated in Fig. 4.1, computes the evolution of physical quantities for every particle and its neighbors in the radius of its smoothing length h . The particles within this radius are then valued according to their distance using a smoothing function W , called the kernel. The fundamental SPH formulation for any physical quantity A is then to

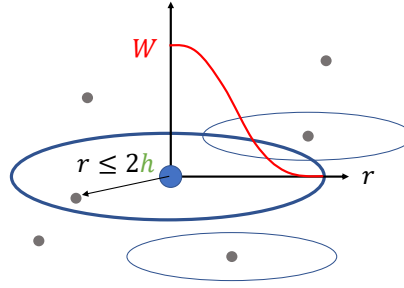


Figure 4.1 SPH kernel W and smoothing length h representation

compute a weighted average over a subset of all the neighbors of the b th particle using

$$A(\vec{r}) \simeq \sum_b \frac{m_b}{\rho_b} A(\vec{r}_b) W(|\vec{r} - \vec{r}_b|, h), \quad (4.1)$$

where m_b is the mass, ρ_b is the density, and \vec{r}_b is the position of the b th particle, respectively. This represents the so-called summation interpolant and its accuracy is dictated by the number of particles and the choice of the kernel.

In our work, our problem is to simply solve the Lagrangian conservation equations (Euler equations) for mass, energy, and momentum for an ideal fluid

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \vec{v}, \quad (4.2)$$

$$\frac{du}{dt} = \frac{P}{\rho^2} \frac{d\rho}{dt}, \quad (4.3)$$

$$\frac{d\vec{v}}{dt} = -\frac{\nabla P}{\rho}, \quad (4.4)$$

where ρ is the density, $d/dt = \partial_t + \vec{v} \cdot \nabla$ is the convective derivative, \vec{v} is the fluid velocity, u is the specific thermal energy, and P is the pressure.

As with other numerical methods, in order to solve this problem within the SPH formulation, different discrete approximations to the equations can be used. Of course, these choices can result in a variety of different accuracies. We use the simplest SPH formulation – sometimes referred to as vanilla ice SPH. This formulation is still used in many astrophysical simulations and is fully conservative. Detailed derivations of the vanilla ice SPH formulation can be found in the literature [154, 155].

By using the volume element $V_b = m_b/\rho_b$, we can reformulate the vanilla ice SPH

scheme so that

$$\rho_a = \sum_b m_b W_{ab}(h_a), \quad (4.5)$$

$$\frac{du_a}{dt} = \frac{P_a}{\rho_a^2} \sum_b m_b \vec{v}_{ab} \cdot \nabla_a W_{ab}, \quad (4.6)$$

$$\frac{d\vec{v}_a}{dt} = - \sum_b m_b \left(\frac{P_a}{\rho_a^2} + \frac{P_b}{\rho_b^2} \right) \nabla_a W_{ab}, \quad (4.7)$$

where $W_{ab} = W(|\vec{r}_a - \vec{r}_b|, h)$ is the smoothing kernel. These equations allow for the emergence of discontinuities from smooth initial data. At discontinuities or the formation of shocks, the entropy will increase. In addition, dissipation will occur behind shock-fronts. The SPH formulation here is inviscid so we will need to be able to handle this dissipation in the vicinity of shock forming regions. There are a number of ways to address this problem. The most widespread approach is to add artificial viscosity (or artificial dissipation). Within the SPH formulation we add new terms to the internal energy and acceleration equations of the form:

$$\left(\frac{du_a}{dt} \right)_{art} = \frac{1}{2} \sum_b m_b \Pi_{ab} \vec{v}_{ab} \cdot \nabla_a W_{ab}, \quad (4.8)$$

$$\left(\frac{d\vec{v}_a}{dt} \right)_{art} = - \sum_b m_b \Pi_{ab} \nabla_a W_{ab}, \quad (4.9)$$

where Π_{ab} is the artificial viscosity tensor. As long as Π_{ab} is symmetric, the conservation of energy, linear and angular momentum continues to be assured. This is because of the form of the equations and the antisymmetry of the gradient of the kernel with respect to the exchange of a and b indices. Note that the artificial viscosity tensor, Π_{ab} , can be defined in different ways, but the most commonly used form is given by [156]:

$$\Pi_{ab} = \begin{cases} \frac{-\alpha \bar{c}_{s,ab} \mu_{ab} + \beta \mu_{ab}^2}{\bar{\rho}_{ab}} & \text{for } \vec{r}_{ab} \cdot \vec{v}_{ab} < 0 \\ 0 & \text{otherwise.} \end{cases} \quad (4.10)$$

The condition $\vec{r}_{ab} \cdot \vec{v}_{ab} < 0$ detects whether particles are approaching. It will only be in this case that artificial viscosity becomes active. We define μ_{ab} as

$$\mu_{ab} = \frac{\bar{h}_{ab} \vec{r}_{ab} \cdot \vec{v}_{ab}}{r_{ab}^2 + \epsilon \bar{h}_{ab}^2}, \quad (4.11)$$

where we use the notation that $r_{ab} = r_a - r_b$, $\vec{v}_{ab} = \vec{v}_a - \vec{v}_b$, $\bar{h}_{ab} = (h_a + h_b)/2$, and $\bar{c}_{s,ab} = (c_{s,a} + c_{s,b})/2$. Note that the term $\varepsilon \bar{h}_{ab}^2$ appears in the denominator of Eqn. 4.11 in order to prevent it from diverging. The sound speed, c_s , can be obtained by $c_s = \sqrt{\frac{\partial p}{\partial \rho}}$. In this artificial viscosity term, Π_{ab} , we can see a term that is linear in the velocity difference which produces shear and bulk viscosity. The quadratic term in the velocity difference is required to control high Mach number shocks. The values of ε , α , and β are often set specifically with regard to the targeted problem. However, numerical experiments have suggested the following values for these parameters: $\varepsilon = 0.01$, $\alpha = 1.0$, and $\beta = 2.0$. In general, we can express the equations for internal energy and acceleration with these new additional terms

$$\frac{du_a}{dt} = \sum_b m_b \left(\frac{P_a}{\rho_a^2} + \frac{\Pi_{ab}}{2} \right) \vec{v}_{ab} \cdot \nabla_a W_{ab}, \quad (4.12)$$

$$\frac{d\vec{v}_a}{dt} = - \sum_b m_b \left(\frac{P_a}{\rho_a^2} + \frac{P_b}{\rho_b^2} + \Pi_{ab} \right) \nabla_a W_{ab}. \quad (4.13)$$

These combined expressions continue to conserve energy, linear and angular momentum by construction. Additional discussion of the artificial viscosity can be found in [155].

4.2.1.1 Kernels

Choosing a sufficiently good smoothing, or kernel, function is a key ingredient in computing certain physical quantities accurately and effectively. Different smoothing kernels within the SPH method have been explored in the literature [157, 158]. Various requirements and properties for smoothing kernels have been discussed but here we present some of the major requirements and properties that are relevant to this work. First, all of the below listed kernels satisfy the following normalization condition:

$$\iiint_{S_D(h)} W(\vec{r}, h) d^D \vec{r} = 1, \quad (4.14)$$

where the integration is performed over the D -dimensional volume of a sphere $S_D(h)$ of radius h .

Second, all kernels will have compact support, i.e. $W(\vec{r}_a - \vec{r}_b, h) = 0$ for $|\vec{r}_a - \vec{r}_b| > \kappa h$. The region of the support is defined by the smoothing length h and a constant factor κ . Furthermore, κ determines the spread in the specified smoothing function.

Third, all kernels will satisfy a positivity condition, namely $W(\vec{r}_a - \vec{r}_b, h) \geq 0$ for all points, \vec{r}_b , within the support domain of the particle point \vec{r}_a . In addition, the kernel should decrease monotonically with respect to the relative distance and possess a smooth derivative.

Furthermore, all kernels should satisfy the symmetric property $W(\vec{r}_a - \vec{r}_b, h) \equiv W(|\vec{r}_a - \vec{r}_b|, h)$ and should reduce to a Dirac delta function as the smoothing length h approaches zero, i.e.

$$\lim_{h \rightarrow 0} W(\vec{r}_a - \vec{r}_b, h) = \delta(\vec{r}_a - \vec{r}_b). \quad (4.15)$$

Any function having the above properties can be considered as a smoothing kernel for SPH. All the currently implemented kernels in our approach possess spherical symmetry which makes it easy to impose exact conservation of linear momentum. For such kernels, the gradients can be computed as follows:

$$\nabla_a W_{ab} \equiv \nabla_a W(|\vec{r}_a - \vec{r}_b|, h_a) \quad (4.16)$$

$$= \frac{dW}{dr} \vec{\epsilon}_{ab}, \quad (4.17)$$

where $\vec{\epsilon}_{ab} \equiv \vec{r}_{ab}/|\vec{r}_{ab}|$ is a unit vector in the direction from particle b to particle a . It is therefore sufficient to implement dW/dr for each kernel. In the formulae below, we define $q \equiv |\vec{r}|/h$.

Cubic spline

The simplest (but *not* the best) is Monaghan's cubic spline. This is given by:

$$W(\vec{r}, h) = \frac{\sigma_D}{h^D} \begin{cases} 1 - 6q^2 + 6q^3 & \text{if } 0 \leq q \leq 1/2, \\ 2(1 - q)^3 & \text{if } 1/2 \leq q \leq 1, \\ 0 & \text{otherwise,} \end{cases} \quad (4.18)$$

where $q = r/h$, D is the number of dimensions and σ_D is a normalization constant:

$$\sigma_D = \left\{ \frac{4}{3}, \frac{40}{7\pi}, \frac{8}{\pi} \right\} \text{ in 1D, 2D and 3D resp.} \quad (4.19)$$

The radial derivative of the cubic spline is:

$$\frac{dW}{dr} = \frac{\sigma_D}{h^{D+1}} \begin{cases} -6q(2-3q) & \text{if } 0 \leq q \leq 1/2, \\ -6(1-q)^2 & \text{if } 1/2 \leq q \leq 1, \\ 0 & \text{otherwise.} \end{cases} \quad (4.20)$$

Gaussian

The Gaussian kernel is given by:

$$W(\vec{r}, h) = \frac{\sigma_D}{h^D} \begin{cases} e^{-q^2} & \text{if } 0 \leq q \leq 3, \\ 0 & \text{if } q > 3, \end{cases} \quad (4.21)$$

where σ_D is :

$$\sigma_D = \left\{ \frac{1}{\pi^{1/2}}, \frac{1}{\pi}, \frac{1}{\pi^{3/2}} \right\} \text{ in 1D, 2D and 3D resp.} \quad (4.22)$$

The radial derivative of the Gaussian kernel is:

$$\frac{dW}{dr} = \frac{\sigma_D}{h^{D+1}} \begin{cases} -2qe^{-q^2} & \text{if } 0 \leq q \leq 3, \\ 0 & \text{if } q > 3. \end{cases} \quad (4.23)$$

Quintic spline

The Quintic spline kernel is given by:

$$W(\vec{r}, h) = \frac{\sigma_D}{h^D} \begin{cases} [(3-q)^5 - 6(2-q)^5 + 15(1-q)^5] & \text{if } 0 \leq q \leq 1, \\ [(3-q)^5 - 6(2-q)^5] & \text{if } 1 \leq q \leq 2, \\ (3-q)^5 & \text{if } 2 \leq q \leq 3, \\ 0 & \text{if } q > 3, \end{cases} \quad (4.24)$$

where σ_D is:

$$\sigma_D = \left\{ \frac{1}{120}, \frac{7}{478\pi}, \frac{3}{359\pi} \right\} \text{ in 1D, 2D and 3D resp.} \quad (4.25)$$

The radial derivative of the quintic spline is:

$$\frac{dW}{dr} = \frac{\sigma_D}{h^{D+1}} \begin{cases} [-5(3-q)^4 + 30(2-q)^4 - 75(1-q)^4] & \text{if } 0 \leq q \leq 1, \\ [-5(3-q)^4 + 30(2-q)^4] & \text{if } 1 \leq q \leq 2, \\ -5(3-q)^4 & \text{if } 2 \leq q \leq 3, \\ 0 & \text{if } q > 3. \end{cases} \quad (4.26)$$

Wendland C2

The Wendland C2 kernel for 2D and 3D is

$$W(\vec{r}, h) = \frac{\sigma_D}{h^D} \begin{cases} (1 - \frac{q}{2})^4 (2q + 1) & \text{if } 0 \leq q \leq 2, \\ 0 & \text{if } q > 2, \end{cases} \quad (4.27)$$

where σ_D is:

$$\sigma_D = \left\{ \frac{7}{4\pi}, \frac{21}{16\pi} \right\} \text{ in 2D and 3D resp.} \quad (4.28)$$

For 1D:

$$W(\vec{r}, h) = \frac{\sigma_D}{h} \begin{cases} (1 - \frac{q}{2})^3 (1.5q + 1) & \text{if } 0 \leq q \leq 2, \\ 0 & \text{if } q > 2, \end{cases} \quad (4.29)$$

where σ_D is:

$$\sigma_D = \frac{5}{8} \text{ in 1D.} \quad (4.30)$$

These are examples of smoothing kernels that we use in this work. Certainly, there are a variety of different kernels usable for improving the performance of the SPH method and which generalize the requirements for constructing smoothing kernels [158]. Our SPH resolution scheme and its routines are presented in algorithm 4.1.

The main downside for the implementation of this method is the requirement that every particle is subject to the SPH loop algorithm. The particles must be grouped locally in order to perform the computation of Eqns. 4.5, 4.12, and 4.13. A communication step is also needed before and after Eqn. 4.5 to get the local physical data and to be able to compute Eqn. 4.12 and Eqn. 4.13. The tree data structure allows us to perform an $O(N \log(N))$

ALGORITHM 4.1 SPH loop algorithm

- 1: **while** not last step **do**
- 2: Compute density for each particle (4.5)
- 3: Compute pressure using EOS
- 4: Compute acceleration from pressure forces (4.13)
- 5: Compute change of internal energy for acceleration (4.12)
- 6: Advance particles after integration
- 7: **end while**

neighbor search but also adds a domain decomposition and distribution layer. A more comprehensive discussion of our implementation can be found in Sec. D

4.2.2 Equations of State

An equation of state (EOS) relates various thermodynamic quantities to each other. In neutron star physics (and more generally in astrophysics), the two relevant equations of state relate pressure P and the specific internal energy u to the density ρ , electron fraction Y_e , and temperature T :

$$P = P(\rho, Y_e, T), \quad (4.31)$$

$$u = u(\rho, Y_e, T). \quad (4.32)$$

Together with the Euler equations Eqns. 4.2, 4.3, and 4.4, these equations of state complete the full system of equations.

4.2.2.1 Analytic EOS

Here we discuss some of the various analytic equations of state which are relevant for binary neutron star mergers.

By far the simplest equation of state that completes the Euler equations (4.2), (4.3), and (4.4) is the ideal gas equation of state

$$P_{\text{id}}(\rho, u) = (\Gamma - 1)\rho u, \quad (4.33)$$

where Γ is the adiabatic index of the gas (typically $5/3$ for a monatomic gas). The ideal gas equation of state is valid for low-temperature, low-pressure regimes. It can, and should, be combined with the more sophisticated analytic equations of state to account for the thermal component of the pressure. At low pressures, we use equation (4.33) combined with other equations of state to fully describe this regime.

A polytropic process obeys the relationship

$$P = K\rho^{(n+1)/n} := K\rho^\gamma, \quad (4.34)$$

where P is the pressure of the gas, ρ is the density, n is a free parameter—the so-called polytropic index—and K is a constant of proportionality.

If constant entropy is assumed, many thermodynamic situations can be approximated as polytropes or as piecewise functions made up of polytropes—so-called piecewise polytropes. In the case of a neutron star, a reasonable polytropic equation of state can be achieved by assuming a (relativistic or non-relativistic) degenerate Fermi gas of neutrons and calculating $P(\rho)$. For a non-relativistic degenerate neutron gas, we have

$$K_0 = \frac{(3\pi^2)^{2/3} \hbar^2}{5m_n^{8/3}}, \quad (4.35)$$

where m_n is the neutron mass, \hbar is the reduced Planck constant, and

$$\gamma_0 = \Gamma = 5/3. \quad (4.36)$$

For the relativistic case we have

$$\gamma_1 = 5/2. \quad (4.37)$$

In our implementation of the polytropic equation of state, we stitch the relativistic and

non-relativistic polytropes smoothly together to attain the following piecewise polytrope:

$$P_{\text{ppt}}(\rho) = \begin{cases} K_0 \rho^{\gamma_0} & \text{if } \rho \leq \rho_0 \\ \left(\frac{K_0 \rho_0^{\gamma_0}}{\rho_0^{\gamma_1}} \right) \rho^{\gamma_1} & \text{if } \rho > \rho_0, \end{cases} \quad (4.38)$$

where $\rho_0 = 5 \times 10^{14} \text{ g/cm}^3$ is a reasonable transition point between the relativistic and non-relativistic regimes.

Finally, we combine the piecewise polytrope with the thermal equation of state described in Eqn 4.33 to obtain an equation of state valid at both low and high densities:

$$P(\rho, u) = P_{\text{ppt}}(\rho) + P_{\text{id}}(\rho, u). \quad (4.39)$$

Another interesting problem using SPH is binary white dwarfs (BWD). Simulations of these systems are useful for studying possible progenitors of type Ia supernovae. Here, we use a zero temperature white dwarf (ZTWD) EOS as a variation of the self consistent field technique [159, 160]. In ZTWD EOS, the electron degeneracy pressure P varies with the mass density ρ according to the relation

$$P = A \left[x(2x^2 - 3)(x^2 + 1)^{1/2} + 3 \sinh^{-1} x \right], \quad (4.40)$$

where the dimensionless parameter

$$x \equiv \left(\frac{\rho}{B} \right)^{1/3}, \quad (4.41)$$

and the constants A and B are

$$A \equiv \frac{\pi m_e^4 c^5}{3h^3} = 6.00288 \times 10^{22} \text{ dynes cm}^{-2}, \quad (4.42)$$

$$\frac{B}{\mu_e} \equiv \frac{8\pi m_p}{3} \left(\frac{m_e c}{h} \right)^3 = 9.81011 \times 10^5 \text{ g cm}^{-3}. \quad (4.43)$$

Using ZTWD EOS, we can also compute the maximum mass of an isolated non-rotating WD [161]. Furthermore, we can derive analytic approximations for the mass-radius relationship of not only WDs [162] but also degenerate stars composed of pure helium [163].

4.2.2.2 Tabulated EOS

For more realistic descriptions of thermodynamical relations for nuclear matter, an EOS built from a microphysical finite temperature model for nuclear matter is required. Such microphysical EOS are too complicated to obtain in analytic form. So, we read thermodynamical quantities for different models from tabulated forms with interpolation.

In our simulations, we use the Lattimer-Swesty EOS [164], which is a good description of hot, non-relativistic neutron matter. There are also different tabulated EOS such as the Shen EOS [165] which provides inhomogeneous matter composition. Tables for these EOS can be obtained from [166].

In this work, the SPH formulation we are using does not evolve temperature. Rather, we track specific internal energy. We therefore require the following alternate relationships:

$$P = P(\rho, Y_e, u), \quad (4.44)$$

$$T = T(\rho, Y_e, u), \quad (4.45)$$

which we must derive from our original table.

To achieve this, we “invert” the relationship (Eqn. 4.32) at runtime: at iteration i and for particle p with density ρ_p^i , electron fraction $(Y_e)_p^i$, and specific internal energy u_p^i , we use a root-finding algorithm such as Newton-Raphson to solve for the temperature T_p^i such that

$$u_p^i = T(\rho_p^i, (Y_e)_p^i, T_p^i). \quad (4.46)$$

After finding T_p^i , we can simply use Eqn. 4.31 to calculate the pressure.

4.2.3 Initial Data

Initial particle configurations are constructed using various methods, including regular cubic lattices, random particle distributions, or sequences of spherical shells. It is well-known that generating good initial data is crucial and necessary for the SPH method to work well [167]. In this section, we concentrate on two ways to generate an initial star for our

NS simulations. In the following subsections, we describe both single and binary systems, the Tolman-Oppenheimer-Volkoff Equation and the Lane-Emden equation, both used to generate NS initial data.

4.2.3.1 The Tolman-Oppenheimer-Volkoff Equation

The Tolman-Oppenheimer-Volkoff (TOV) equations describe the structure of a spherically symmetric body of isotropic material. In this section, we constrain ourselves to the Newtonian case. This gives the simple stellar structure equations which have the form

$$\frac{\partial P}{\partial r} = -\frac{Gm(r)\rho(r)}{r^2}, \quad (4.47)$$

$$\frac{\partial m}{\partial r} = 4\pi r^2 \rho(r), \quad (4.48)$$

where $m(r)$ is the mass contained in a spherical shell of radius r , ρ is the density of the object, and P is the pressure. To find the pressure, we match these with an EOS

$$f(\rho) = P, \quad (4.49)$$

so we can find the density ρ by inverting this function such that $\rho = f^{-1}(P)$. In this work, one choice is to assume a piecewise polytropic EOS (Eqn. 4.35) as an analytic model. Another choice which will include more microphysics is to use a more sophisticated tabulated EOS. This will provide for more realistic simulations. To solve these Eqns. 4.47 - 4.48, we use conventional finite differencing together with a 4th order Runge-Kutta time integrator.

If we use a tabulated EOS, the internal energy is generally included. However, in our case, we must derive the internal energy to complete the system. From the first law of thermodynamics, we have

$$dU = TdS - PdV, \quad (4.50)$$

where U is internal energy, S is entropy, T is temperature, and V is volume. We want to express this in terms of the specific internal energy u which is energy per unit mass. We

assume baryon mass conservation so we can express the first law of thermodynamics in terms of u and ρ so that

$$du = Tds + \frac{P}{\rho^2}d\rho, \quad (4.51)$$

where s is specific entropy. A star without a heat source in thermodynamical equilibrium is isothermal, i.e. $T = \text{constant}$ so that $dT = 0$. Using that, we can integrate a thermodynamical potential which depends on T . Consider the specific Helmholtz free energy, f , to be $f = u - Ts$. Its differential is

$$df = du - sdT - Tds = Tds + \frac{P}{\rho^2}d\rho - sdT - Tds = \frac{P}{\rho^2}d\rho. \quad (4.52)$$

From a thermodynamical relation, we obtain

$$\left(\frac{\partial f}{\partial \rho}\right)_T = \frac{P}{\rho^2}. \quad (4.53)$$

On considering a polytropic EOS, $P = K\rho^\gamma$, we have

$$\left(\frac{\partial f}{\partial \rho}\right)_T = K\rho^{\gamma-2}. \quad (4.54)$$

On integrating this with respect to ρ , we get the form for the specific Helmholtz free energy

$$f(\rho, T) = \frac{K}{\gamma-1}\rho^{\gamma-1} + c(T), \quad (4.55)$$

where $c(T)$ is an arbitrary ‘‘constant’’ of integration which depends on the local temperature.

Using the relation, $f = u - Ts$, we finally derive the specific internal energy

$$\begin{aligned} u(T, \rho) &= f + Ts = \frac{K}{\gamma-1}\rho^{\gamma-1} + c(T) - T \left(\frac{\partial f}{\partial T}\right)_\rho \\ &= \frac{K}{\gamma-1}\rho^{\gamma-1} + c(T) - T \frac{dc}{dT}. \end{aligned} \quad (4.56)$$

If we define $u_{th}(T) = c(T) - T \frac{dc}{dT}$ as the thermal part of the specific energy (which depends only on temperature) together with a polytropic EOS, $P = K\rho^\gamma$, we then have

$$u(T, \rho) = \frac{1}{1-\gamma} \frac{P}{\rho} + u_{th}(T). \quad (4.57)$$

If there is no thermal specific energy initially i.e. $u_{th}(T) = 0$, we obtain the usual ideal gas EOS $P = (\gamma - 1)\rho u$.

On solving for the TOV star, we need to modify the initial data file that is appropriate for use in the SPH code. To do that, we need to calculate the radial positions of the particles. We assume that each particle has the same mass so that mass of an individual particle is

$$m_{part} = \frac{m(R)}{n_{part}}, \quad (4.58)$$

where R is the radius of the star and n_{part} is the number of particles. The sum of the masses of all particles within radius r_i less than r is given by

$$m_{sum}(r) = \sum_i m_{part} \text{ for } r_i \leq r, \quad (4.59)$$

where i labels a radial grid point. This should be approximately the same as $m(r)$ which is given by

$$m_{sum}(r) \approx m(r) = \int r^2 \rho dr. \quad (4.60)$$

To calculate this, we need the probability of finding the particle at radius r . This is given by

$$\frac{dm}{dr} = 4\pi r^2 \rho, \quad (4.61)$$

with appropriate normalization. We use a rejection sampling method to satisfy this. We then assume θ and φ coordinates for the particles from the radially uniform distribution. After these, we convert from spherical coordinates to Cartesian coordinates via the usual coordinate transformation.

To define the smoothing length h , a root finding algorithm is used to demand that each particle has 10 neighbors within a distance $2h$. (This choice is our convention for the radius of support for most smoothing kernels in this work.) We also set particle velocities to zero. We set particle internal energies to be $u_i = u(r_i)$ for particle i at position r_i .

4.2.3.2 The Lane-Emden Equation

Another way to obtain the initial data for a star is by solving the Lane-Emden equation. On doing this, we obtain the density as a function of radius $\rho(\vec{r})$, with polytropic index $n = 1$.

We pick particles with variable mass, such that the mass of a particle is computed as:

$$m_i = \frac{\rho(\vec{r}_i)}{n_r} \text{ with } n_r = \frac{3N}{4\pi R^3}. \quad (4.62)$$

The smoothing length is set to a constant and is uniform for all particles:

$$h = \frac{1}{2} \sqrt{\frac{3N_N}{4\pi n}}. \quad (4.63)$$

Here we choose N_N , the average number of neighbors, to be 100.

The Lane-Emden equation describes equilibrium configurations of a Newtonian star. The equation determines the density of the star as a function of radius.

Considering the star as a polytropic fluid, the Lane-Emden equation takes the form of a linear differential equation with nonlinear source:

$$\frac{d^2\theta}{d\xi^2} + \frac{2}{\xi} \frac{d\theta}{d\xi} + \theta^n = 0, \quad (4.64)$$

where ξ and θ are two dimensionless variables which represent rescaled radius and density respectively. There exist exact solutions to the Lane-Emden equation for polytropic indices $n = 0.5, 1$ and 2 . In our work we will use a polytropic index of 1 which corresponds to a NS.

For $n = 1$ the exact solution of equation 4.64 is:

$$\theta(\xi) = \frac{\sin(\xi)}{\xi}. \quad (4.65)$$

We note that for $\xi_1 = \pi$, the first value of ξ is $\theta(\xi) = 0$. The variable $\theta(\xi)$ is as mentioned previously, the rescaled density, also defined as:

$$\theta(\xi) = \left(\frac{\rho(\xi)}{\rho_c} \right)^{\frac{1}{n}} = \frac{\rho(\xi)}{\rho_c}, \quad (4.66)$$

where ρ is the star's density and ρ_c is its central density. The variable ξ is defined as:

$$\xi = \sqrt{\frac{4\pi G}{K(n+1)} \rho_c^{(n-1)/n}} \times r = \sqrt{\frac{2\pi G}{K}} \times r \text{ (for } n = 1),$$

where K is the proportionality constant coming from the polytropic relation.

	NS_1	NS_2	NS_3	NS_4
Radius (cm)	$R = G = M = 1$	1500000	1400000	960000
K	0.636619	95598.00	83576.48	39156.94

Table 4.1 Proportionality constants with respect to radius for NS.

From the previous equations we can write the stellar radius R as:

$$R = \sqrt{\frac{K(n+1)}{4\pi G}} \rho_c^{(1-n)/2} \xi_1 = \sqrt{\frac{K}{2\pi G}} \times \xi_1. \quad (4.67)$$

(We note that for $n = 1$ the radius does not depend on the central density.) If, for example, we use dimensionless units as $G = R = M = 1$ (for other results we use cgs with $G = 6.674 \times 10^{-8} \text{cm}^3 \text{g}^{-1} \text{s}^{-2}$), we can compute K as:

$$K = \frac{R^2 2\pi G}{\xi_1^2}. \quad (4.68)$$

We then deduce the density function of r as :

$$\rho(\xi) = \frac{\sin(A_1 r)}{A_1 r} \rho_c \quad \text{with } A_1 = \sqrt{\frac{2\pi G}{K}}.$$

As the total Mass M , radius R and the gravitational constant G are known input parameters we can finally compute the central density as [168]:

$$\rho_c = \frac{MA_1^3}{4\pi(\sin(A_1 R) - A_1 R \cos(A_1 R))}.$$

Then we normalize the results to fit $R = M = G = 1$: $K' = K/(R^2 G)$, $m'_i = m_i/M$, $h'_i = h_i/R$, $\vec{x}'_i = \vec{x}_i/R$

4.2.4 Time Integration Scheme

For the time integrator, we use a leapfrog algorithm, specifically its "kick-drift-kick" variation (see Algorithm 4.2). This algorithm belongs to the family of symplectic integrators and, in the absence of gravity, conserves energy exactly.

Leapfrog time integrators are efficient for Hamiltonian systems, specifically for particle simulations. The name comes from the fact that particle velocities are updated at half-steps while the positions are updated at integer steps, so that the two effectively “leap” over each other. A pair of updates from timestep n to $n + 1$ has the following form:

$$v^{n+1/2} = v^{n-1/2} + a(r^n)\Delta t, \quad (4.69)$$

$$r^{n+1} = r^n + v^{n+1/2}\Delta t. \quad (4.70)$$

Note that in this simple form the accelerations $a(r^n)$ are computed synchronously with positions and are assumed independent of the velocities v^n . This formulation is time-symmetric and reversible up to roundoff.

An equivalent "kick-drift-kick" formulation was shown to be stable for variable time steps:

$$v^{n+1/2} = v^n + a(r^n)\Delta t/2, \quad \text{"kick"} \quad (4.71)$$

$$r^{n+1} = r^n + v^{n+1/2}\Delta t, \quad \text{"drift"} \quad (4.72)$$

$$v^{n+1} = v^n + a(r^{n+1})\Delta t/2, \quad \text{"kick"} \quad (4.73)$$

When applied to the basic SPH formulation (4.12-4.13), this method reads:

$$v^{n+1/2} = v^n + a[r^n, \Pi_{ab}(v^{n-1/2})]\Delta t/2, \quad (4.74)$$

$$u^{n+1/2} = u^n + \frac{du}{dt} [r^n, v^n, \Pi_{ab}(v^{n-1/2})] \Delta t/2, \quad (4.75)$$

$$r^{n+1} = r^n + v^{n+1/2}\Delta t, \quad (4.76)$$

$$v^{n+1} = v^{n+1/2} + a[r^{n+1}, \Pi_{ab}(v^{n+1/2})] \Delta t/2, \quad (4.77)$$

$$u^{n+1} = u^{n+1/2} + \frac{du}{dt} [r^{n+1}, v^{n+1}, \Pi_{ab}(v^{n+1/2})] \Delta t/2. \quad (4.78)$$

where we have highlighted variable dependencies between different time levels.

In this formulation, the internal energy is stored and updated synchronously with the velocities rather than with the coordinates. This is done to achieve exact energy

conservation:

$$E^{n+1/2} - E^{n-1/2} = \sum_a m_a \left[u_a^{n+1/2} - u_a^{n-1/2} \right] + \sum_a \frac{1}{2} m_a \left[\left(\bar{v}_a^{n+1/2} \right)^2 - \left(\bar{v}_a^{n-1/2} \right)^2 \right] \quad (4.79)$$

$$= \sum_a m_a \left(\frac{du_a^n}{dt} + \bar{v}_a^n \cdot \frac{d\bar{v}_a^n}{dt} \right), \quad \text{where } \bar{v}_a^n := \frac{\bar{v}_a^{n-1/2} + \bar{v}_a^{n+1/2}}{2}. \quad (4.80)$$

If we substitute the corresponding time derivatives from (4.12-4.13), this expression vanishes:

$$E^{n+1/2} - E^{n-1/2} = - \sum_{a,b} m_a m_b \left(\frac{P_a}{\rho_a^2} \bar{v}_b^n + \frac{P_b}{\rho_b^2} \bar{v}_a^n + \Pi_{ab} \frac{\bar{v}_a^n + \bar{v}_b^n}{2} \right) \cdot \nabla_a W_{ab}^n = 0. \quad (4.81)$$

For this expression to vanish, the viscosity tensors Π_{ab} in Eqns. 4.74 and 4.75 above (or in Eqns. 4.77 and 4.77) should be identical. Even though they might depend on the values of the internal energy at the previous half-step, $u^{n-1/2}$, it does not affect tensor symmetry and energy conservation at half-steps. Similarly, the pressure in Eqn. 4.81 depends on the density at the current integer step and the internal energy at the previous half-step, but it does not violate energy conservation as long as the same value of the pressure is used to compute accelerations and time derivatives of internal energy. At the same time, when computing derivatives of internal energy using expression Eqn. 4.12), the dot products, $\bar{v}_{ab}^n \cdot \nabla_a W_{ab}^n$, must be computed with velocities \bar{v}_a^n at integer timesteps, as in Eqn. 4.80.

4.2.4.1 Adaptive timestep

The timestep can be adaptive and determined by $\Delta t = \lambda_{\text{CFL}} \min(\Delta t_1, \Delta t_2)$, where $\lambda_{\text{CFL}} \approx 0.1$ is a Courant factor limit, and the timescales $\Delta t_1, \Delta t_2$ are [169]:

$$\Delta t_1 = \min_a \left(\frac{h_a}{c_a(1 + 1.2\alpha) + 1.2 \beta \max_b \mu_{ab}} \right), \quad (4.82)$$

$$\Delta t_2 = \min_a \sqrt{\frac{h_a}{|\dot{\bar{v}}_a|}}, \quad (4.83)$$

and where c_a is the sound speed, α and β are viscosity parameters, and μ_{ab} is the viscosity function as defined in Eqn. 4.10.

ALGORITHM 4.2 Leapfrog time integration

- 1: read initial state: $\{r, v, u\} \leftarrow \{r^0, v^0, u^0\}$
- 2: compute $\rho(r)$, $P(\rho, u)$, $cs(\rho, P, u)$
- 3: save $v^{1/2}$: $v_{12} = v$
- 4: compute acceleration: $a(P, \rho, v_{12})$
- 5: compute $Dudt(P, \rho, v, v_{12})$
- 6: **while** iterations **do**
- 7: select timestep $dt \leftarrow \Delta t$
- 8: $v += a*dt/2$ ▷ velocity kick: $v^{n+1/2} = v^n + a^n \frac{\Delta t}{2}$
- 9: $u += Dudt*dt/2$ ▷ internal energy kick: $u^{n+1/2} = u^n + \left(\frac{du}{dt}\right)^n \frac{\Delta t}{2}$
- 10: update neighbors
- 11: $r += v*dt$ ▷ drift: $r^{n+1} = r^n + v^{n+1/2} \Delta t$
- 12: compute $\rho(r)$, $P(\rho, u)$, $cs(\rho, P, u)$ ▷ uses $u^{n+1/2}$!
- 13: save $v^{n+1/2}$: $v_{12} = v$
- 14: compute acceleration: $a(P, \rho, v_{12})$ ▷ $a^{n+1} = a[P^{n+1}, \rho^{n+1}, \Pi_{ab}(v^{n+1/2})]$
- 15: $v += a*dt/2$ ▷ second velocity kick: $v^{n+1} = v^{n+1/2} + a^{n+1} \frac{\Delta t}{2}$
- 16: update neighbors
- 17: compute $Dudt(P, \rho, v, v_{12})$ ▷ $\left(\frac{du}{dt}\right)^{n+1} = \dot{u}[P^{n+1}, \rho^{n+1}, v^{n+1}, \Pi_{ab}(v^{n+1/2})]$
- 18: $u += Dudt*dt/2$ ▷ second internal energy kick: $u^{n+1} = u^{n+1/2} + \left(\frac{du}{dt}\right)^{n+1} \frac{\Delta t}{2}$
- 19: **end while**

4.2.5 Computing Gravitational Forces

The gravitational force acting on a particle \vec{F}_a^{Grav} is, of course, due to attraction from other particles and is described by the standard Newton's law of gravity

$$\vec{F}_a^{Grav} = \sum_b G \frac{m_a m_b}{(|\vec{r}_a - \vec{r}_b|)^3} \vec{r}_{ab}. \quad (4.84)$$

with G being the Newton's gravitational constant. To avoid $O(N^2)$ computational complexity, we use the Fast Multipole Method (FMM) [170], as described below.

4.2.5.1 The Fast Multipole Method

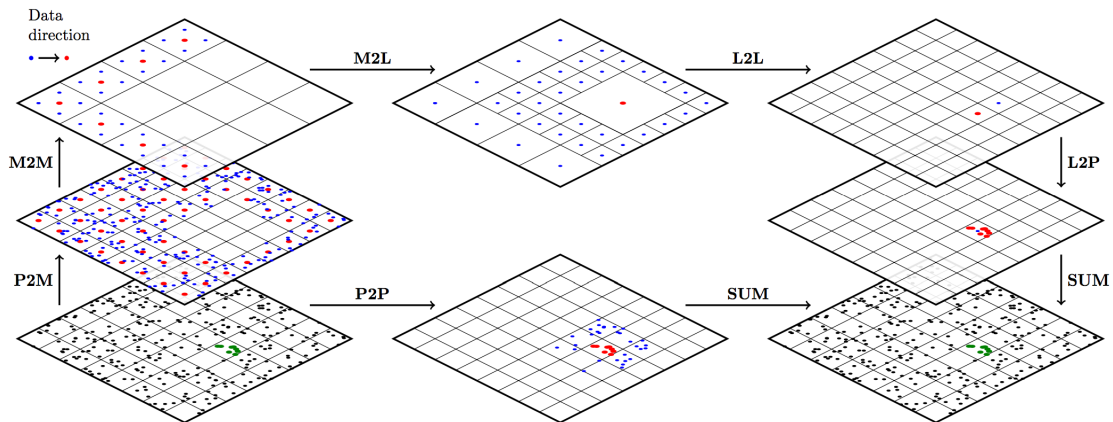


Figure 4.2 Fast multipole method scheme. This figure presents each steps of scheme: Particle to Multipole (P2M), Multipole to Multipole (M2M), Multipole to Particles (M2P), Multipole to Local (M2L), Local to Local (L2L), and Particles to Particles (P2P). This schematic is inspired from [171]

See Algorithm 4.3 for the basic outline of the method. FMM aims to compute the gravitational force in as accurate a way as direct summation. Fig. 4.2 shows details of the scheme for FMM, from left bottom to right bottom for a group of particles. By way of clarification, we provide some terminology here.

Particles: These are the bodies on which we need to compute the gravitational force with respect to the other particles. In Fig. 4.2, green particles are separated for which we are computing the gravitation from the other blue particles

Multipoles: These represent the center of mass (COM) of a group particles. In this case, we re-group the mass and barycenter of sub-particles. There are several different levels of multipoles: particles' multipole and multipoles' multipole.

Locals: These are the COM for the reduction on the particles concerned in this work. They have the same behavior as the multipoles, but the information goes down to particles instead of going up.

In order to compute the gravitational force for a group particles in the domain, we proceed according to the following steps

Particles to Particles (P2P): For the particles that are closed, we use the direct $O(N^2)$ algorithm. This portion will increase the size with greater desired accuracy.

Particles to Multipoles (P2M): This gathers the data from all sub-particles to calculate the COM and the multipoles. This will become the first layer of the tree, namely the leaves.

Multipoles to Multipoles (M2M): This gathers the data from multipoles at a higher level of the tree from the leaves to the root.

Multipoles to Local (M2L): This computes the gravitational portion of all the distant multipoles on the local.

Local to Local (L2L): Going down in the tree and spreading the components to sub-locals.

Local to Particles (L2P): Compute the application of the local for all sub particles when a leaf of the tree is reached.

Summation (SUM): Two interactions can be summed up to compute the gravitational force applied to the particle at the end of computations for both P2P and L2P.

These steps are repeated for every group of particles. The P2M-M2M steps are done just once before the FMM for all groups of particles. In this algorithm, the *macangle* is the angle of the Multipole Acceptance Criterion (MAC) for the choice between P2P or M2L. We use an angle between the local COM and the edge of distant multipoles. If the angle fits the criterion, we use the current multipole otherwise we go lower in the tree to consider smaller multipoles. If the criterion is never satisfied, this implies that the particles are too

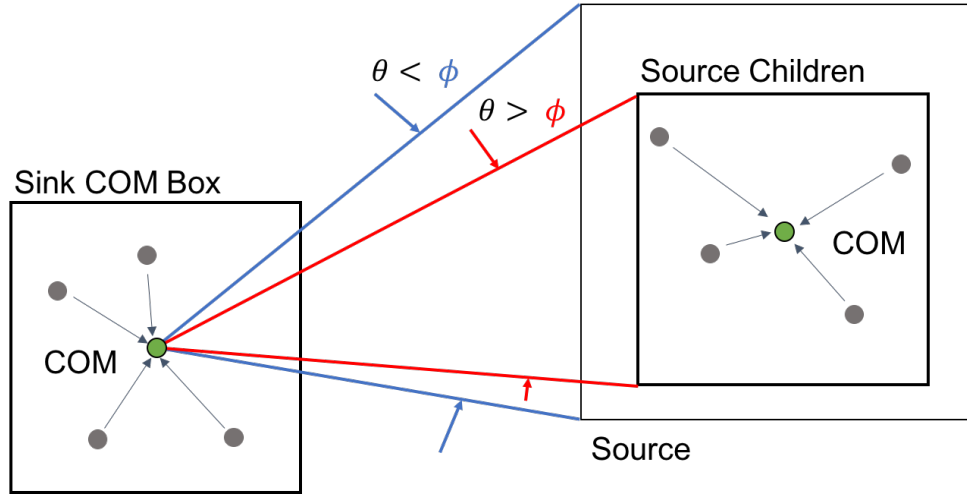


Figure 4.3 Scheme of MAC for FMM

close and consider P2P. Fig. 4.3 shows a version of the MAC scheme of

For the P2P step, we use the classical gravitational computation (like Eqn. 4.84). However, for the interaction with distant multipoles, we use the Taylor series defined below. The acceleration is, for the center of mass c , the sum of all the contributions from nearby particles as well as from the distant cells as given by:

$$\vec{f}_c(\vec{r}_c) = - \sum_p \frac{m_p \cdot (\vec{r}_c - \vec{r}_p)}{|\vec{r}_c - \vec{r}_p|^3}. \quad (4.85)$$

In this express, p labels the particle inside this cell and cl labels the cells that are accepted with the MAC. Note that we only consider the gravitational acceleration. We don't take into account the mass associated with the center of mass, c . Note we use $G = 1$.

The acceleration at a point due to the center of mass can be found by the Taylor series:

$$\vec{f}(\vec{r}) \approx \vec{f}_c(\vec{r}_c) + \left\| \frac{\partial \vec{f}_c}{\partial \vec{r}_c} \right\| \cdot (\vec{r} - \vec{r}_c) + \frac{1}{2} (\vec{r} - \vec{r}_c)^\top \cdot \left\| \frac{\partial \vec{f}_c}{\partial \vec{r}_c \partial \vec{r}_c} \right\| \cdot (\vec{r} - \vec{r}_c). \quad (4.86)$$

The Jacobian matrix, $\left\| \frac{\partial \vec{f}_c}{\partial \vec{r}} \right\|$, is then:

$$- \sum_p \frac{m_p}{|\vec{r}_c - \vec{r}_p|^3} \begin{bmatrix} 1 - \frac{3(x_c - x_p)(x_c - x_p)}{|\vec{r}_c - \vec{r}_p|^2} & -\frac{3(y_c - y_p)(x_c - x_p)}{|\vec{r}_c - \vec{r}_p|^2} & -\frac{3(z_c - z_p)(x_c - x_p)}{|\vec{r}_c - \vec{r}_p|^2} \\ -\frac{3(x_c - x_p)(y_c - y_p)}{|\vec{r}_c - \vec{r}_p|^2} & 1 - \frac{3(y_c - y_p)(y_c - y_p)}{|\vec{r}_c - \vec{r}_p|^2} & -\frac{3(z_c - z_p)(y_c - y_p)}{|\vec{r}_c - \vec{r}_p|^2} \\ -\frac{3(x_c - x_p)(z_c - z_p)}{|\vec{r}_c - \vec{r}_p|^2} & -\frac{3(y_c - y_p)(z_c - z_p)}{|\vec{r}_c - \vec{r}_p|^2} & 1 - \frac{3(z_c - z_p)(z_c - z_p)}{|\vec{r}_c - \vec{r}_p|^2} \end{bmatrix}, \quad (4.87)$$

and the derivative is

$$\left\| \frac{\partial f_c^i}{\partial r_c^j} \right\| = - \sum_p \frac{m_p}{|\vec{r}_c - \vec{r}_p|^3} \left[\delta_{ij} - \frac{3 \cdot (r_c^i - r_p^i)(r_c^j - r_p^j)}{|\vec{r}_c - \vec{r}_p|^2} \right], \quad (4.88)$$

where δ_{ij} is the Kronecker delta.

The Hessian matrix $\left\| \frac{\partial^2 f_c}{\partial \vec{r}_c \partial \vec{r}_c} \right\|$ can be calculated as:

$$\left\| \frac{\partial^2 f_c^x}{\partial r_c^i \partial r_c^j} \right\| = - \sum_p \frac{3m_p}{|\vec{r}_c - \vec{r}_p|^5} \begin{bmatrix} \frac{5(x_c - x_p)^3}{|\vec{r}_c - \vec{r}_p|^2} - 3(x_c - x_p) & \frac{5(x_c - x_p)^2(y_c - y_p)}{|\vec{r}_c - \vec{r}_p|^2} - 3(y_c - y_p) & \frac{5(x_c - x_p)^2(z_c - z_p)}{|\vec{r}_c - \vec{r}_p|^2} - 3(z_c - z_p) \\ \frac{5(x_c - x_p)^2(y_c - y_p)}{|\vec{r}_c - \vec{r}_p|^2} - 3(y_c - y_p) & \frac{5(x_c - x_p)(y_c - y_p)^2}{|\vec{r}_c - \vec{r}_p|^2} - 3(x_c - x_p) & \frac{5(x_c - x_p)(y_c - y_p)(z_c - z_p)}{|\vec{r}_c - \vec{r}_p|^2} \\ \frac{5(x_c - x_p)^2(z_c - z_p)}{|\vec{r}_c - \vec{r}_p|^2} - 3(z_c - z_p) & \frac{5(x_c - x_p)(y_c - y_p)(z_c - z_p)}{|\vec{r}_c - \vec{r}_p|^2} & \frac{5(x_c - x_p)(z_c - z_p)^2}{|\vec{r}_c - \vec{r}_p|^2} - 3(x_c - x_p) \end{bmatrix}, \quad (4.89)$$

$$\left\| \frac{\partial^2 f_c^y}{\partial r_c^i \partial r_c^j} \right\| = - \sum_p \frac{3m_p}{|\vec{r}_c - \vec{r}_p|^5} \begin{bmatrix} \frac{5(x_c - x_p)^2(y_c - y_p)}{|\vec{r}_c - \vec{r}_p|^2} - 3(y_c - y_p) & \frac{5(x_c - x_p)(y_c - y_p)^2}{|\vec{r}_c - \vec{r}_p|^2} - 3(x_c - x_p) & \frac{5(x_c - x_p)(y_c - y_p)(z_c - z_p)}{|\vec{r}_c - \vec{r}_p|^2} \\ \frac{5(x_c - x_p)(y_c - y_p)^2}{|\vec{r}_c - \vec{r}_p|^2} - 3(x_c - x_p) & \frac{5(y_c - y_p)^3}{|\vec{r}_c - \vec{r}_p|^2} - 3(y_c - y_p) & \frac{5(y_c - y_p)^2(z_c - z_p)}{|\vec{r}_c - \vec{r}_p|^2} - 3(z_c - z_p) \\ \frac{5(x_c - x_p)(y_c - y_p)(z_c - z_p)}{|\vec{r}_c - \vec{r}_p|^2} & \frac{5(y_c - y_p)^2(z_c - z_p)}{|\vec{r}_c - \vec{r}_p|^2} - 3(z_c - z_p) & \frac{5(y_c - y_p)(z_c - z_p)^2}{|\vec{r}_c - \vec{r}_p|^2} - 3(y_c - y_p) \end{bmatrix}, \quad (4.90)$$

$$\left\| \frac{\partial^2 f_c^z}{\partial r_c^i \partial r_c^j} \right\| = - \sum_p \frac{3m_p}{|\vec{r}_c - \vec{r}_p|^5} \begin{bmatrix} \frac{5(x_c - x_p)^2(z_c - z_p)}{|\vec{r}_c - \vec{r}_p|^2} - 3(z_c - z_p) & \frac{5(x_c - x_p)(y_c - y_p)(z_c - z_p)}{|\vec{r}_c - \vec{r}_p|^2} & \frac{5(x_c - x_p)(z_c - z_p)^2}{|\vec{r}_c - \vec{r}_p|^2} - 3(x_c - x_p) \\ \frac{5(x_c - x_p)(y_c - y_p)(z_c - z_p)}{|\vec{r}_c - \vec{r}_p|^2} & \frac{5(y_c - y_p)^2(z_c - z_p)}{|\vec{r}_c - \vec{r}_p|^2} - 3(z_c - z_p) & \frac{5(y_c - y_p)(z_c - z_p)^2}{|\vec{r}_c - \vec{r}_p|^2} - 3(y_c - y_p) \\ \frac{5(x_c - x_p)(z_c - z_p)^2}{|\vec{r}_c - \vec{r}_p|^2} - 3(x_c - x_p) & \frac{5(y_c - y_p)(z_c - z_p)^2}{|\vec{r}_c - \vec{r}_p|^2} - 3(y_c - y_p) & \frac{5(z_c - z_p)^3}{|\vec{r}_c - \vec{r}_p|^2} - 3(z_c - z_p) \end{bmatrix}, \quad (4.91)$$

$$\left\| \frac{\partial^2 f_c^i}{\partial r_c^j \partial r_c^k} \right\| = - \sum_p \frac{3m_p}{|\vec{r}_c - \vec{r}_p|^5} \left[\frac{5(r_c^i - r_p^i)(r_c^j - r_p^j)(r_c^k - r_p^k)}{|\vec{r}_c - \vec{r}_p|^2} - \frac{3}{w} \left(\delta_{ij}(r_c^k - r_p^k) + \delta_{jk}(r_c^i - r_p^i) + \delta_{ik}(r_c^j - r_p^j) \right) \right]. \quad (4.92)$$

where w in the last line is a pure number from the SPH calculation which ensures that the denominator does not vanish. It is defined by $w = \delta_{ij} + \delta_{jk} + \delta_{ki} + \epsilon_{ijk}$ with ϵ_{ijk} the 3D Levi-Civita symbol. We the add the Levi-Civita symbol to avoid getting zero in the denominator. Here again, Latin indices i , j , and k indicate spatial components.

Eqns. 4.88 and 4.92 are used during the M2L step. Eqn. 4.86 is used during the tree traversal and when we apply the gravitational calculations to locals then particles in L2L and L2P.

4.2.6 Gravitational Waves

Our ultimate interest is obtaining GW waveform from BNS mergers using this method. For example, consider two orbiting bodies, such as neutron stars, of masses m_1 and m_2 . Though neutron stars are extended bodies, we assume they are point masses for the time being. In the center-of-mass frame of the bodies, we can write the equations of motion as [172]:

$$\mu \ddot{\vec{r}} = -\vec{\nabla} U_{\text{eff}}(\vec{r}), \quad (4.93)$$

ALGORITHM 4.3 Gravitation computation

```

1: procedure TREE_TRAVERSAL_GRAV(branch sink)
2:   if sink.mass <  $M_{cellmax}$  then                                ▷ Another choice criterion can be use
3:      $\vec{f}_c \leftarrow \vec{0}$ 
4:      $\frac{\partial \vec{f}_c}{\partial \vec{r}} \leftarrow \vec{0}$ 
5:     TREE_TRAVERSAL_C2C(sink, tree.root,  $\vec{f}_c$ ,  $\frac{\partial \vec{f}_c}{\partial \vec{r}}$ ) ▷ Compute  $\vec{f}_c$  and  $\frac{\partial \vec{f}_c}{\partial \vec{r}}$  using
      MAC
6:     SINK_TRAVERSAL_C2P(sink,  $\vec{f}_c$ ,  $\frac{\partial \vec{f}_c}{\partial \vec{r}}$ ) ▷ Expand to the particles below
7:   else
8:     for All children c of sink do
9:       TREE_TRAVERSAL_GRAV(c)
10:    end for
11:  end if
12: end procedure
13:
14: function MAC(branch sink, branch source, double macangle)
15:    $d_{max} \leftarrow source.radius \times 2$ 
16:    $dist \leftarrow distance(sink.position, source.position)$ 
17:   return  $d_{max}/dist < macangle$ 
18: end function

```



```

19: procedure TREE_TRAVERSAL_C2C(branch sink, branch source, acceleration  $\vec{f}_c$ )
20:   if MAC(sink, source, macangle) then
21:      $\vec{f}_c \leftarrow \vec{f}_c + \left( -\frac{\text{source.mass} \times (\text{sink.position} - \text{source.position})}{|\text{sink.position} - \text{source.position}|^3} \right)$ 
22:      $\frac{\partial \vec{f}_c}{\partial \vec{r}} \leftarrow \dots$ 
23:   else
24:     if source.is_leaf() then
25:       for All particles p of source do
26:          $\vec{f}_c \leftarrow \vec{f}_c + \left( -\frac{p.mass \times (\text{sink.position} - p.position)}{|\text{sink.position} - p.position|^3} \right)$ 
27:          $\frac{\partial \vec{f}_c}{\partial \vec{r}} \leftarrow \dots$ 
28:       end for
29:     else
30:       for All children c of source do
31:         TREE_TRAVERSAL_C2C(sink, c,  $\vec{f}_c$ )
32:       end for
33:     end if
34:   end if
35: end procedure

```

```

36: procedure TREE_TRAVERSAL_C2P(branch current, acceleration  $\vec{f}_c$ )
37:   if current.is_leaf() then
38:     for All particle p of current do
39:        $p.\text{grav} \leftarrow \vec{f}_c + \frac{\delta \vec{f}_c}{\delta \text{current.position}} \cdot (p.\text{position} - \text{current.position}) + \dots$ 
40:     end for
41:   else
42:     for All children c of current do
43:       TREE_TRAVERSAL_C2P(c,  $\vec{f}_c$ ,  $\frac{\partial \vec{f}_c}{\partial \vec{r}_c}$ )
44:     end for
45:   end if
46: end procedure

```

where \vec{r} is the separation vector,

$$\mu := \frac{m_1 m_2}{m_1 + m_2}, \quad (4.94)$$

is the reduced mass, and

$$U_{\text{eff}}(\vec{r}) = -\frac{Gm_1 m_2}{r} + \frac{L_z^2(\vec{r})}{2\mu r^2}, \quad (4.95)$$

is the effective potential energy for some angular momentum vector

$$\vec{L} = \vec{r} \times m\vec{v},$$

which we have chosen to align purely along the \hat{z} -axis of our coordinate system.

For large radii, the gravitational attraction dominates Eqn 4.95. However, for small radii, the repulsive angular momentum term dominates. This effect is called the angular momentum barrier. It implies that, at least in Newtonian gravity, binary systems do not merge.

Of course, neutron stars are general relativistic systems and in general relativity, binary systems *do* merge. In the relativistic case, angular momentum is carried away via gravitational waves and the angular momentum barrier is reduced.

Because our code is Newtonian, in order to study binary mergers, we must account for

dissipation of angular momentum via gravitational waves. We incorporate this gravitational wave radiation reaction via post-Newtonian (PN) theory. To implement PN theory, we use a similar technique in [173] which is based on [174, 175] that develop a framework for extracting gravitational wave signatures from SPH-type simulations. For a detailed review of post-Newtonian theory in the context of compact binary coalescence, see [176].

The relativistic 2-body problem can be solved with numerical relativity, or by approximate analytical methods such as the PN expansion or via a perturbation-based self-force approach ($\frac{m_1}{m_2} \ll 1$). The former is a GR approximation, valid for weak gravitational field and slow matter motion, so it applies to compact object binary mergers with low mutual gravity, and breaks down when the stars get too close together.

The n th order PN approximation corresponds to $\left(\frac{v}{c}\right)^{2n}$ or $\left(\frac{Gm}{rc^2}\right)^n$, often abbreviated as c^{-2n} , with the zeroth order corresponding to Newtonian gravity.

The NS of radius R_1 with respect to the center of mass (that we take to be at the zero coordinate) has an acceleration in polar coordinates

$$a_1 = \begin{bmatrix} \ddot{R}_1 - R_1 \Omega^2 \\ 2\dot{R}_1 \Omega + R_1 \dot{\Omega} \end{bmatrix}_{r,\theta},$$

with the Kepler 3rd law generalized as

$$\Omega = \sqrt{\frac{Gm}{r^3}} \left[1 - \frac{Gm}{rc^2} \frac{3-\eta}{2} + \left(\frac{Gm}{rc^2}\right)^2 \frac{15+47\eta+3\eta^2}{8} \right],$$

plus terms of order $\mathcal{O}(c^{-6})$. The orbital frequency increase is given by

$$\dot{\Omega} = \frac{96}{5} \frac{Gm\eta}{r^3} \left(\frac{Gm}{rc^2}\right)^{5/2} + \mathcal{O}(c^{-6}), \quad (4.96)$$

while the secular decrease in the orbital separation becomes

$$\dot{R}_1 = -\frac{64}{5} \frac{G^3 m^2 m_2 \eta}{r^3 c^5} + \mathcal{O}(c^{-6}), \quad (4.97)$$

These are the first PN terms due to GW radiation reaction, taken from [177] at 2.5PN order, with r in harmonic coordinates. Note that \ddot{R} is zero at this order.

We use the following definitions: the total mass is $m = m_1 + m_2$, the dimensionless reduced mass is $\eta = \frac{m_1 m_2}{m^2} = \frac{\mu}{m}$, and the orbital separation is $r = R_1 + R_2$ and is calculated by finding the center of mass position of each star.

Then we rest the Keplerian terms from the acceleration:

$$a_1 = \left[\begin{array}{c} \frac{G^2 m m_2}{c^2 r^3} (3 - \eta) - \frac{G^3 m^2 m_2}{c^4 r^4} \left(6 + \frac{41}{4} \eta + \eta^2 \right) \\ - \frac{32 G^{7/2} m^{5/2} m_2 \eta}{5 c^5 r^{9/2}} \end{array} \right]_{r,\theta}, \quad (4.98)$$

plus terms of order 6. Map it to the set of SPH particles constituting the given NS as $a_{1,r}^{\text{part}} = a_{1,r}$, $a_{1,\theta}^{\text{part}} = \frac{r}{R_1} a_{1,\theta}$, and add it to the momentum SPH equation after passing it to cartesian coordinates through the rotation endomorphism

$$\begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}.$$

We input initial velocities including Keplerian velocities to the initial model

$$v_1 = \left[\begin{array}{c} \dot{R}_1 \\ R_1 \Omega \end{array} \right]_{r,\theta}, \quad \text{with } R_1 = \frac{m_2}{m} r, \quad (4.99)$$

Another option is to take the acceleration from the computed positions and velocities of each star center of mass

$$\vec{a}_i = -\Omega^2 \vec{R}_i - \frac{32 (Gm)^3 \eta}{5 c^5 r^4} \vec{v}_i + \mathcal{O}(c^{-6}), \quad (4.100)$$

with Ω^2 given without the Keplerian contribution, plus terms of order 6

$$\Omega^2 = \frac{Gm}{r^3} \left[-\frac{Gm}{rc^2} (3 - \eta) + \left(\frac{Gm}{rc^2} \right)^2 \left(6 + \frac{41}{4} \eta + \eta^2 \right) \right].$$

4.3 Results

As described previously, SPH can handle different types of hydrodynamical and astrophysical problems. We now present several application tests to validate our toolkit. All the tests below have been performed on the ROMEO Supercomputer of the University of Reims

Champagne-Ardenne and the Darwin Supercomputer at Los Alamos National Laboratory. The ROMEO cluster is composed of 130 nodes each with $2 \times E5-2650v2$ 8 cores CPUs with 32GB RAM memory and $2 \times K20Xm$ GPUs. The nodes are linked using InfiniBand QDR fat-tree topology. The Darwin cluster is a heterogeneous supercomputer. We use 23 specific nodes in our testing. Each node is Hashwell machine with $20.26GHz$ cores on 2 sockets and 128GB of memory. The nodes are connected by a 56 GB/s FDR infiniband network with a latency of $0.7\mu s$.

4.3.1 Sod Shock Tube

The Sod shock tube is a test consisting of a one-dimensional Riemann problem with the following initial parameters [178].

$$(\rho, v, p)_{t=0} = \begin{cases} (1.0, 0.0, 1.0) & \text{if } 0 < x \leq 0.5 \\ (0.125, 0.0, 0.1) & \text{if } 0.5 < x < 1.0 \end{cases}. \quad (4.101)$$

In our code, we use the same initial data as in [178] with an ideal gas EOS such as:

$$P(\rho, u) = (\Gamma - 1)\rho u, \quad (4.102)$$

where Γ is the adiabatic index of the gas which we set to be $\Gamma = 5/3$.

This test is used to check the physical accuracy of the code and thus the tree search itself. One example of our Sod shock simulations is presented in Fig. 4.4.

4.3.2 Sedov Blast Wave

A blast wave is the pressure and flow resulting from the deposition of a large amount of energy in a small very volume. There are different versions of this blast wave test and we compare it with the analytic solution for a point explosion as given by Sedov [179]. We make the assumption that the atmospheric pressure relative to the pressure inside the explosion is negligible. Here, we test the 2D blast wave. In this simulation, we use the ideal

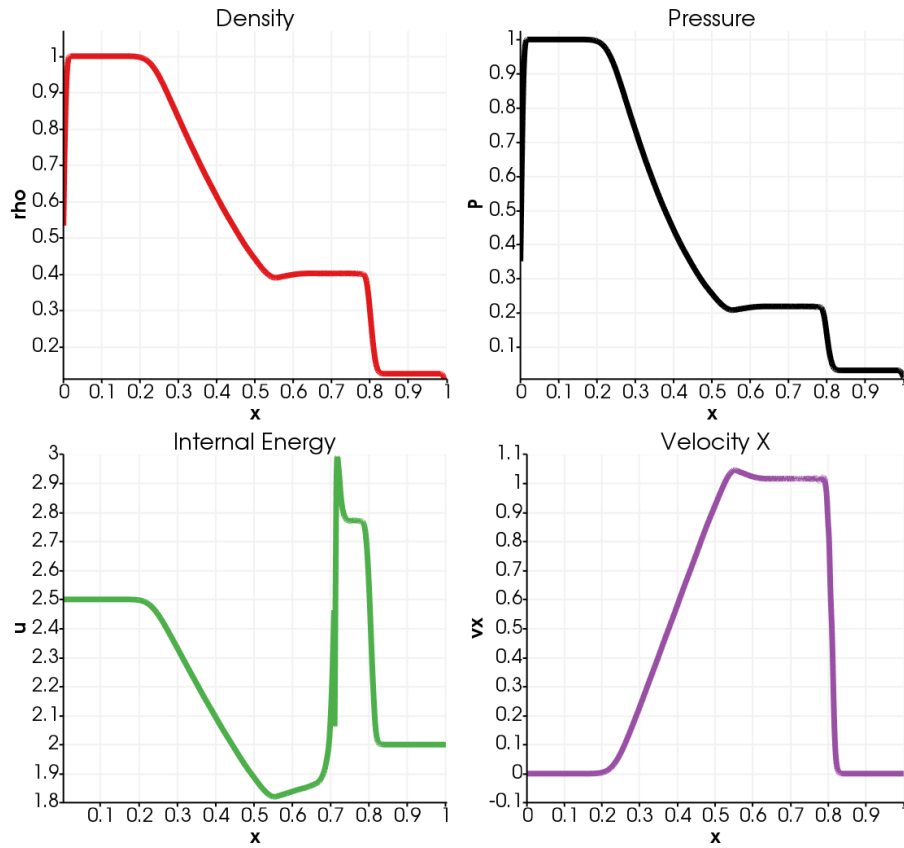


Figure 4.4 Sod shock tube with FleCSPH

gas EOS with $\Gamma = 5/3$ and we assume that the undisturbed area is at rest with a pressure $P_0 = 1.0^{-5}$. The density, ρ_0 , is constant in the pressurized region.

An example of our Sedov Blast wave simulations is presented on Fig. 4.5.

4.3.3 Fluid Flow Test

After performing the tests regarding the physics reliability, we worked on the fluid flow problem in 2D and 3D to reach a high number of particles. The details can be found in [180]. This test is based on an ideal EOS given by:

$$P = B \left[\left(\frac{\rho}{\rho_0} \right)^\gamma - 1 \right], \quad (4.103)$$

with $\gamma = 7$, $B = c_0 \rho_0 / \gamma$, and reference density $\rho_0 = 1000 \text{ kg.m}^{-3}$.

For this experiment, realistic boundary conditions were needed. Several methods are

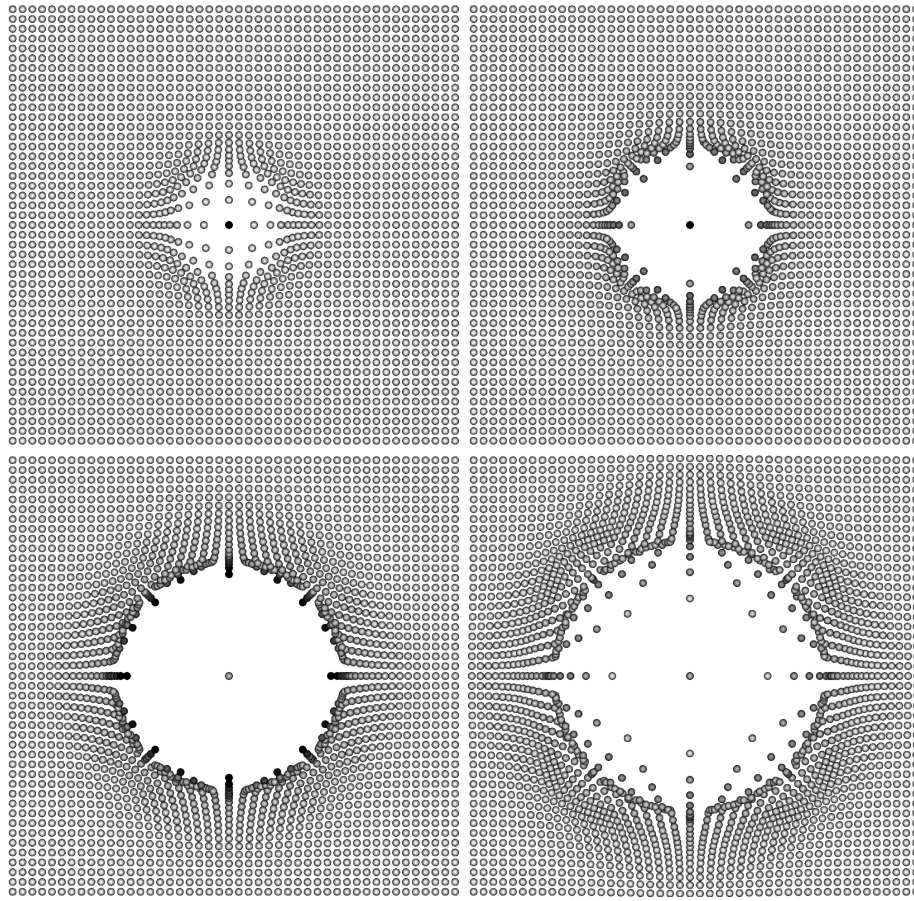


Figure 4.5 Sedov Blast Wave with FleCSPH at respectively $t = 0.01$, $t = 0.03$, $t = 0.06$ and $t = 0.1$

possible with SPH including the so-called mirror particles implementation [181] and the dummy particles implementation [182].

For the current work, we chose to use the dummy particles method. Fig 4.6 shows the representation of dummy particle to create the boundary i.e. the wall.

In this case, the wall particles are just considered as normal particles and their quantities are evolved during the run. However, the main difference between wall and fluid particles is that wall particles do not have their position evolved at the end of each time step. Furthermore, the pressure of a wall particle has to be obtained from the fluid particle to accurately approximate the pressure gradient in the fluid near the wall. This creates a stable boundary when fluid particles interact with the wall particles. A force balance at the

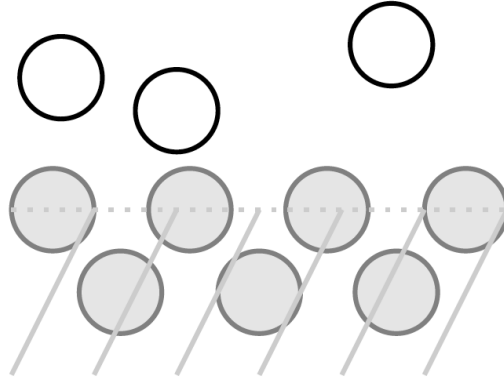


Figure 4.6 The representation of dummy particle. The gray filled circle indicates dummy particles that consist a boundary and white circle represent fluid particles

boundary interface gives

$$\frac{d\mathbf{v}_f}{dt} = -\frac{\nabla P_f}{\rho_f} + \mathbf{g} = \mathbf{a}_w, \quad (4.104)$$

where f and w label fluid and wall respectively, and \mathbf{g} is the gravitational acceleration. They are identified in the code with a specific type which is provided during data generation. Fig 4.7 shows a particle representation of a dam break simulation with the dummy particle method.

We used this code on up to 8 nodes on the ROMEO supercomputer cluster targeting 500,000 particles for a fluid simulation, the dam break test.

In Fig. 4.8, a test of strong scaling is presented on up to 128 cores. The first part, in blue, was done using 1 MPI process while increasing the OpenMP threads. Then, keeping 8 OpenMP threads, we increased the number of MPI processes up to 16, having a maximum of 128 cores.

The current limitation to strong scaling is related to synchronized communication. Indeed, after the local search, MPI collective operations like All_to_all or All_gather operations stop the execution.

From our several different fluid problem tests, we confirm that our implementation can solve/generate a variety of different systems correctly. This implies that using our

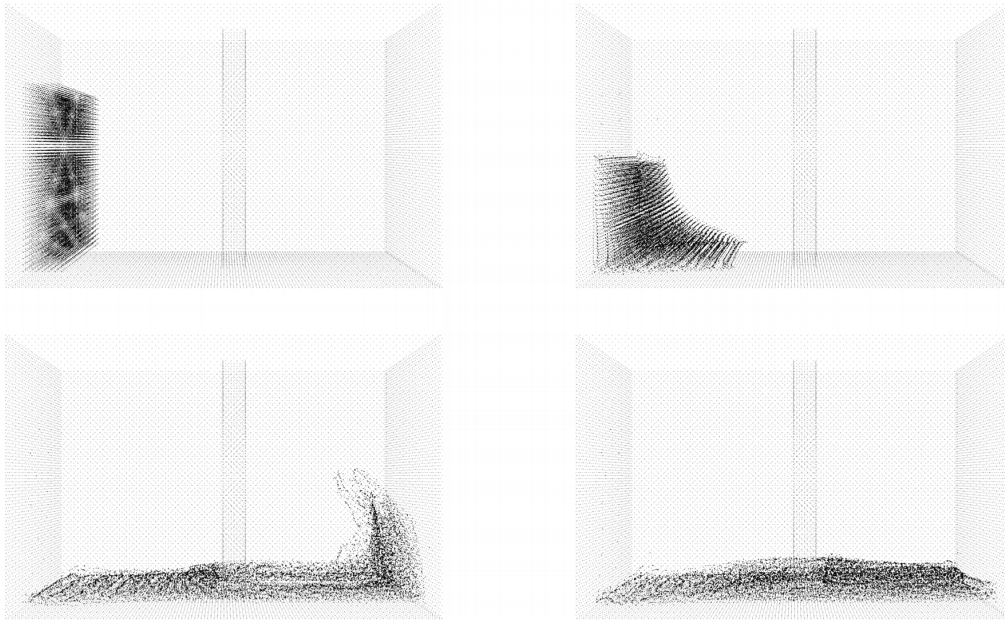


Figure 4.7 Fluid flow simulation, the dam break, for different time, $t = 0$, $t = 0.4$, $t = 0.8$, and $t = 1$ seconds. The boundary conditions are based on dummy particles method

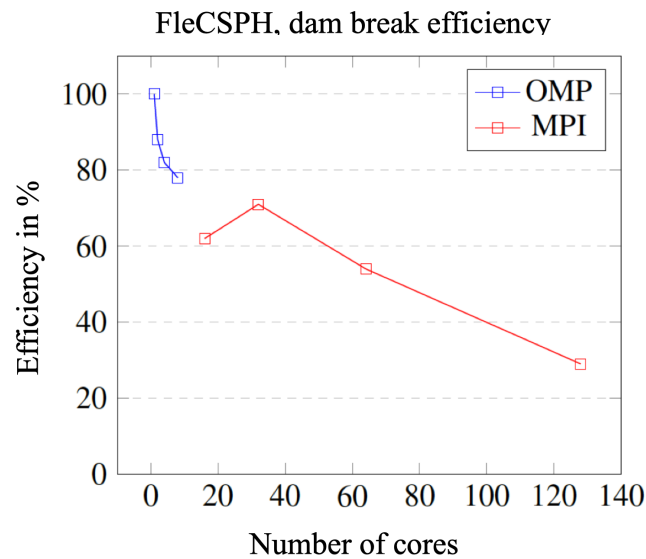


Figure 4.8 Efficiency on strong scaling

code is easily adaptable to different problems within astrophysics, fluid mechanics and mechanical engineering and provides some interesting testbeds for computer science in terms of efficiency within an HPC environment.

4.3.4 Binary Neutron Stars

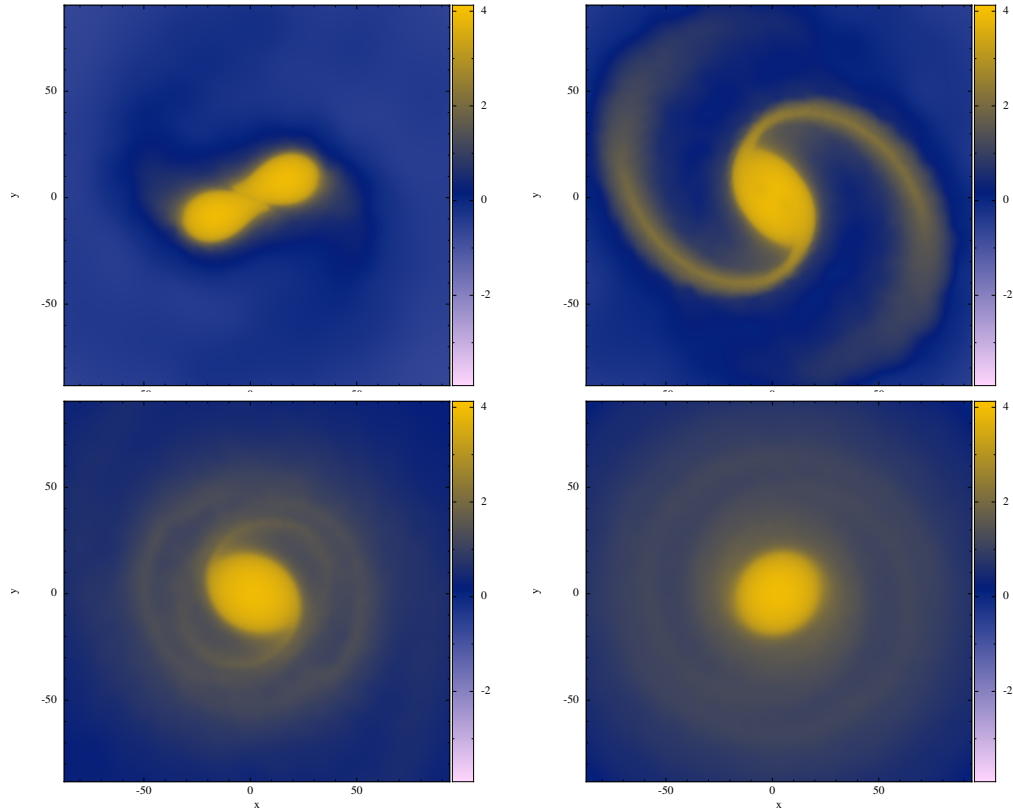


Figure 4.9 An equal-mass binary merger with 100,000 particles. The initial stars are generated by solving TOV equations (Sec. 4.2.3.1). The stars are sufficiently close that they merge via dynamical friction. The frames are at times (going from left to right) 9.8 ms, 12.0 ms, 16.0 ms, and 20.0 ms. Distances are in kilometers. The colormap shows density, integrated along the \hat{z} -axis in units of 10^{17} g/cm². The color bar is in a log scale. Images made with Splash [183].

In this section, we discuss the binary mergers we simulate. Note that some of these results are generated using the 2HOT code which was refactored during my 2016 co-design summer school fellowship at LANL. If the stars are sufficiently close to each other, dynamical friction will cause them to merge. As a proof of concept, we performed such a merger with 100,000 particles and the Lattimer-Swesty equation of state. Four frames of the merger are shown in Fig. 4.9. This binary is of equal mass, where each star is roughly $2M_{\odot}$. Fig. 4.10 shows particle representations of the binary.

As the stars merge, they lose material through the Lagrange points of the system, and

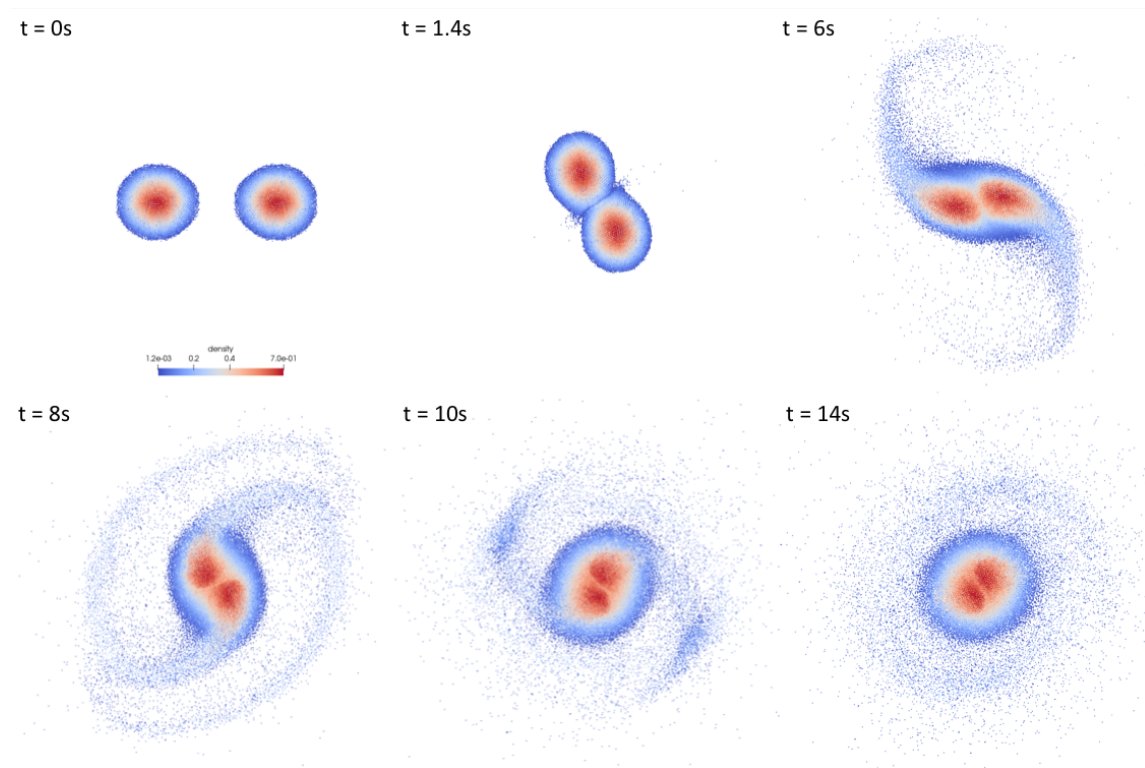


Figure 4.10 Particle representation for equal mass binary merger with 40,000 particles. This simulation is done by F1eCSPH. The initial data for the neutron stars is generated by solving the Lane-Emden equation (Sec. 4.2.3.2).

these form the spiral arms shown in the second frame of Fig. 4.9. Most of this material forms the accretion disk shown in the third and fourth frames and eventually accretes back onto the remnant. However, some of the material becomes gravitationally unbound.

This gravitationally unbound material, which we call the “ejecta” expands away from the remnant in a roughly toroidal topology, as shown in Fig. 4.11. Note that at this moment in time, the ejecta, in terms of spatial size, is roughly one order of magnitude larger than the remnant and the accretion disk system.

The ejecta are hot and neutron rich, making them an excellent location for r -process nucleosynthesis. To extract the thermodynamic properties of the ejecta, we track particles contained in the ejecta, and their thermodynamic quantities such as internal energy and temperature, back in time for the entire evolution of the system. Fig. 4.12 shows the trajectories of some of the particles in the ejecta.

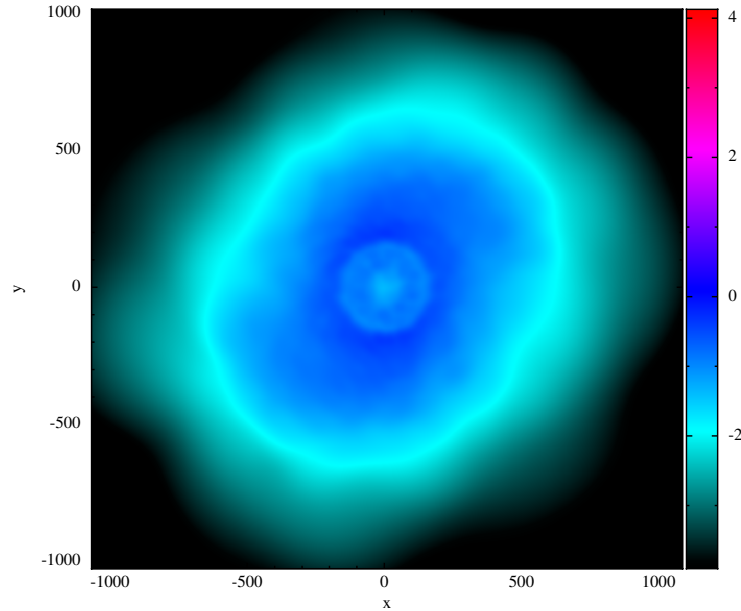


Figure 4.11 The morphology of the ejecta from the merger shown in figure 4.9. The time shown is 20 ms. Distances are in kilometers. The colormap shows density, integrated along the \hat{z} -axis in units of 10^{17} g/cm². The color bar is in a log scale. Images made with Splash [183].

After obtaining particle trajectories, we pass them into an r -process nucleosynthesis code which produces r -process residuals, as shown in Fig. 4.13. The solar system abundances are shown as green dots while the r -process abundances predicted by our ejecta are shown in purple. The highest peak in our r -process abundances plot is mass number around 196, which represents gold. Thus, we confirm that BNS mergers are a main source of gold which cannot be produced in the same abundances via core collapse supernovae. In addition, BNS mergers also play an important role in generating elements that have mass number around 130, which represents xenon (and possibly cesium).

As we push to larger particle numbers, we will be able to obtain significantly more accurate ejecta morphologies and commensurately more accurate r -process residuals.

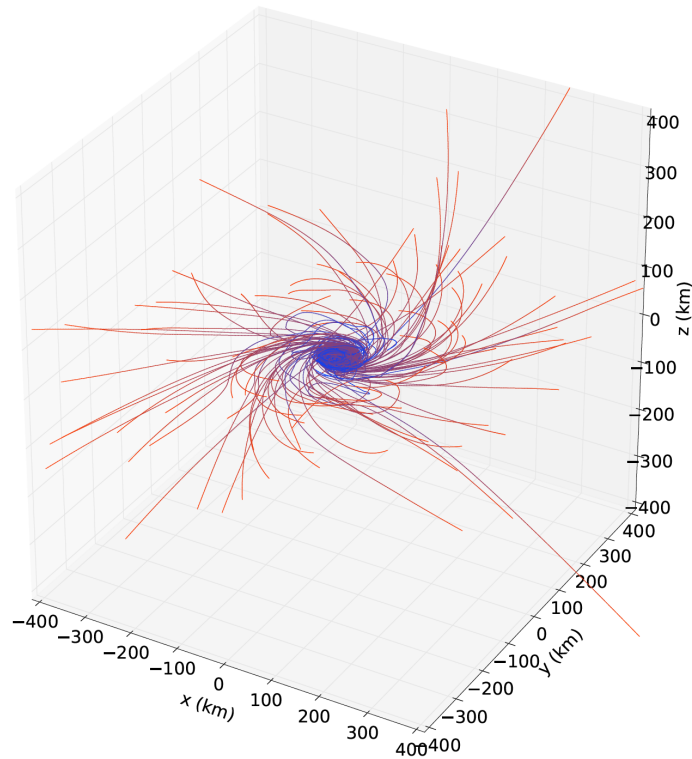


Figure 4.12 Traces of the trajectories of particles in the ejecta. The color indicates time—bluer is earlier and red is later.

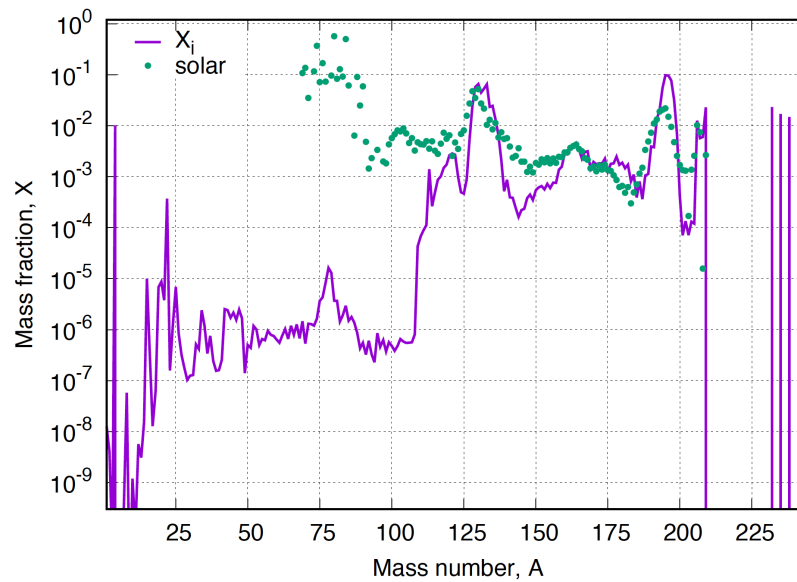


Figure 4.13 r-process residuals. We plot atomic abundance as a function of atomic mass. The green dots are observed solar abundances and the purple curve is as predicted by our ejecta.

4.4 Conclusion

In this chapter, we propose a new toolkit to solve merging BNS as well as other possible hydrodynamics problems including the investigation of fluid instabilities and turbulence using SPH. We have tested several basic fluid problems to validate our toolkit and have shown that our toolkit can solve various different problem in different spatial dimensions very efficiently. Further, all current development stage promises that we can incorporate with additional physics component such as neutrinos and

In addition, we investigate the ejecta morphology from binary neutron star coalescence. We develop realistic neutron star initial data for SPH and gravitational wave radiation. Furthermore, we test different EOS to see how this affects NSs. We run several stable BNS merger simulations with analysis of ejecta using SPH. The r -process nucleosynthesis results show that BNS coalescence is a primary source for the production of elements with mass numbers in the vicinity of 130 and 196. This result will be useful in providing detailed analyses of future BNS mergers and provide new insights into nuclear and neutrino physics in these extreme environments. Additionally, these simulations will provide guidance for the development of future observational instruments for kilonovae and their electromagnetic counterparts. Finally, current development stage promise that we can incorporate with more complicated physics component such as neutrinos and general relativistic effects.

Furthermore, the development of the computational methods will help guide software requirements for future exascale machines. Currently as main developer of FleCSPH, advanced parallel programming models such as task-based parallelism and potential GPU accelerator is already started to investigate and implement. These efforts for developing FleCSPH will provide efficient and state-of-art mesh free code for the larger community. All code developed for this project is intended to be open source in order to provide the broadest impact on not only the astrophysics community but also other disciplines.

Chapter 5

Exotic Primordial Black Hole and Neutron Star “Binaries”

5.1 Introduction

Many astrophysical observations suggest that on galactic as well as larger scales there is missing mass-energy. The observed and known luminous matter together with non-luminous matter for which we can account are not enough to explain galactic and cosmological scales dynamics. Some of this missing energy is referred to as dark matter (DM). Evidence for the existence of DM comes from the rotational velocities of galaxies, the equation of state of the primordial universe, and the gravitational lensing of colliding galactic clusters. However, both our theoretical understanding and direct detections of DM remain limited.

Many theoretical candidates for the DM have been considered in previous work. In particle physics, DM is associated with new stable particles beyond the Standard Model. A famous example of this are weakly interacting massive particles (WIMPs). However, there are alternative DM candidates which do not require new stable particles and which remain viable candidates. One interesting candidate of this type is primordial black holes (PBHs). PBHs can account for all or some parts of the DM [184, 185].

Previous studies have shown that NSs can be used to probe the DM [186]. In one

scenario [187], NS are used to set constraints on the characteristics of WIMPs. Some special WIMP classes would be trapped in NSs, concentrated towards the star's center, and become self-gravitating. This process may form a very small BH that eventually consumes the NS. This could be an interesting phenomenon because the NS transforms into a BH on a time scale shorter than the observed ages of NSs in many astrophysical systems. Another interesting possibility is that a PBH is captured by a NS and then settles into the center, growing rapidly until the supply of nuclear matter is exhausted by accretion and ejection. Indeed, there are a number of ongoing astrophysical searches to try to unveil DM that forms interior to a BH and which causes the implosion of the old NS [188, 189]. This, then, may provide yet another viable site for r -processes and nucleosynthesis [190]. Such have also been suggested as models for fast radio bursts [191].

With regard to nucleosynthesis, several recent studies suggest that interesting aspects of a combined NS and PBH may be used to explore the possible signature of DM in astrophysical observations. In [192], it is shown that NS disruptions by PBHs in a DM-rich environment, such as galactic centers and dwarf spheroidal galaxies, provide a viable site for r -process nucleosynthesis. The article [193] demonstrates how NS mergers observed in galaxies can be used to unmask DM with nucleon scattering cross-sections. Additionally, the article describes distinct signatures of DM-induced NS implosions, which may be discoverable by upcoming gravitational, optical, and radio surveys. In reference [194], the authors study the constraints on the fraction of PBHs in the total amount of DM that arises from the requirement that PBHs be captured by NSs. Finally, [195] proposes that DM may scatter into the dense core of pulsars and exceed the Schwarzschild limit within the lifetime of galactic center pulsars, especially millisecond pulsars.

In our work here, we construct a general relativistic model of such systems wherein a small BH has formed at the center of a NS by gravitational collapse. We follow the dynamics as the small BH consumes the NS from within. We numerically solve the Einstein equations coupled to hydrodynamics under spherical symmetry. Using numerical simulations, we explore the time scale on which the NS collapses into the small mass BH,

and consider the possible unbounded material and ejecta from the NS.

This project has been done in collaboration with William East during my fellowship at the Perimeter Institute for Theoretical Physics. My contribution to this work consisted of deriving all the systems of equations, developing good coordinate conditions, and developed the initial data. I developed all the necessary code using PAMR/AMRD driver in order to exploit this parallel AMR infrastructure. (See Appendix E.1.4 for more details regarding the implementation of this project.) I also performed all the tests and simulations.

5.2 Horizon Penetrating Coordinates

As already mentioned, our aim is to study the dynamics of a NS being consumed by a small mass BH from within. To do this, we need to capture all the dynamics of NS material that can pass within the event horizon of a small BH. Therefore, our choice of coordinate system is particularly crucial to obtaining a good numerical simulation.

As an example of the issue we are considering, recall the usual Schwarzschild line element in Boyer-Lindquist type coordinates

$$ds^2 = - \left(1 - \frac{2M}{r}\right) dt^2 + \left(1 - \frac{2M}{r}\right)^{-1} dr^2 + r^2 d\Omega^2, \quad (5.1)$$

where M is the mass of BH. Compare this with the usual 3+1 line element in the form $ds^2 = -\alpha^2 dt^2 + \gamma_{ij}(dx^i + \beta^i dt)(dx^j + \beta^j dt)$. We identify the lapse as $\alpha^2 = \left(1 - \frac{2M}{r}\right)$. As we know, the line element (Eqn. 5.1) is singular and the lapse goes to zero on the event horizon ($r = 2M$).

On the application of regular boundary conditions, we have a well-posed problem for all fields in the exterior of the BH; in our case within and exterior to the NS. In principle, we should be able to ask about the stability of such fields and compute their dynamics. Because the interior of the BH is not physically observable, the event horizon is a natural boundary. However, in this version of our model, the event horizon is not only a natural boundary, the coordinates in which we express this are, in fact, also singular on this boundary. This is

problematic from a numerical point of view because the equations of motion involving the metric can become unstable in the vicinity of such a coordinate singularity without some regularization technique.

One way to resolve this problem is to change to coordinate that are not singular at the event horizon and which “penetrate” the horizon in some sense. Such horizon penetrating coordinate systems would allow evolutions that do not suffer from the pathology of bad dynamical behavior on this, otherwise, natural boundary. So-called Kerr-Schild coordinates are one such coordinate system.

For example, the Schwarzschild solution in spherical Kerr-Schild coordinates is given by

$$\alpha = \sqrt{\frac{r}{r+2M}}, \quad (5.2)$$

$$\beta^r = \frac{2M}{r+2M}, \quad (5.3)$$

$$\beta_r = \frac{2M}{r}, \quad (5.4)$$

$$\beta^\theta = \beta^\phi = 0, \quad (5.5)$$

$$K_{ij} = \text{diag} \left[-\frac{2M(r+M)}{\sqrt{r^5(r+2M)}}, 2M\sqrt{\frac{r}{r+2M}}, K_{\theta\theta} \sin^2 \theta \right]. \quad (5.6)$$

The Schwarzschild solution in Cartesian type Kerr-Schild coordinates is given

$$\alpha = \sqrt{\frac{r}{r+2M}}, \quad (5.7)$$

$$\beta^i = \frac{2M}{r} \frac{x^i}{r+2M}, \quad (5.8)$$

$$\beta_i = \frac{2Mx_i}{r^2}, \quad (5.9)$$

$$K_{ij} = \frac{2M}{r^4} \sqrt{\frac{r}{r+2M}} \left[\left(\frac{M}{r} + 2 \right) x_i x_j - r^2 \delta_{ij} \right], \quad (5.10)$$

where $x^i = (x, y, z)$ are the usual spatial Cartesian coordinates. In both cases, we can see that the lapse is regular at the horizon ($r = 2M$).

A general spherically symmetric line element in polar-areal form becomes

$$ds^2 = -\alpha(r)^2 dt^2 + a(r)^2 dr^2 + r^2 d\Omega^2. \quad (5.11)$$

On comparing with the above Schwarzschild solution, clearly $\alpha = 1/a$.

Now consider a transformation of the Schwarzschild time coordinate, t , to a new generic coordinate \hat{t} according to

$$d\hat{t} = dt + a^2 \sqrt{1 - \frac{g}{a^2}} dr \quad (5.12)$$

where $g(r)$ is an arbitrary function. Substituting this into the line element gives

$$\begin{aligned} ds^2 &= -\alpha^2 \left(d\hat{t} - a^2 \sqrt{1 - \frac{g}{a^2}} dr \right)^2 + a^2 dr^2 + r^2 d\Omega^2 \\ &= -\alpha^2 d\hat{t}^2 + 2\sqrt{1 - \frac{g}{a^2}} d\hat{t} dr + g dr^2 + r^2 d\Omega^2. \end{aligned} \quad (5.13)$$

On comparing this with the usual 3+1 framework:

$$ds^2 = -\alpha^2 d\hat{t}^2 + \gamma_{ij} (dx^i + \beta^i d\hat{t}) (dx^j + \beta^j d\hat{t}), \quad (5.14)$$

and we recognize the lapse $\alpha = 1/\sqrt{g}$, the shift $\beta_i = (\sqrt{1 - g/a^2}, 0, 0)$ (or $\beta^i = \gamma^{ij} \beta_j$) and the spatial metric of the constant \hat{t} hypersurface $\gamma_{ij} = \text{diag}(g, r^2, r^2 \sin^2 \theta)$.

If we choose $\alpha = \sqrt{1 - 2M/r} = 1/a$ and $g = 1 + 2M/r$, we get

$$ds^2 = - \left(1 - \frac{2M}{r} \right) d\hat{t}^2 + \frac{4M}{r} d\hat{t} dr + \left(1 + \frac{2M}{r} \right) dr^2 + r^2 d\Omega^2, \quad (5.15)$$

which is the Schwarzschild metric in Kerr-Schild coordinates (also referred to as Eddington-Finkelstein coordinates). Correspondingly, $\alpha = \sqrt{r/(r+2M)}$, $\beta_i = (2M/r, 0, 0)$, and $\gamma_{ij} = \text{diag}(1 + 2M/r, r^2, r^2 \sin^2 \theta)$ which are the same as above.

As can be seen here, the Kerr-Schild form of the metric represents an analytic continuation of the Schwarzschild solution from the region $2M < r < \infty$ to $0 < r < \infty$. Thus, we apply this coordinate transformation to our equations.

It is convenient to rewrite the metric into a slightly more general form. For this we will use the 3 + 1 ADM form so as to consider time dependent case.

$$ds^2 = (-\alpha^2 + a^2 \beta^2) dt^2 + 2a^2 \beta dt dr + a^2 dr^2 + r^2 b^2 d\Omega^2. \quad (5.16)$$

The quantities α , a , b , and β are functions of r and t , and $d\Omega^2$ is the metric on the unit sphere. From this, we can calculate all the non-vanishing components of the connection coefficients and the Ricci tensor. These are

$$\begin{aligned}\Gamma^r_{rr} &= \frac{\partial_r a}{a}, & \Gamma^r_{\theta\theta} &= -\frac{rb\partial_r(rb)}{a^2}, & \Gamma^\theta_{r\theta} &= \frac{\partial_r(rb)}{rb}, \\ \Gamma^r_{\varphi\varphi} &= -\sin^2\theta \frac{rb\partial_r(rb)}{a^2}, & \Gamma^\varphi_{r\varphi} &= \Gamma^\theta_{r\theta}, \\ \Gamma^\theta_{\varphi\varphi} &= -\sin\theta \cos\theta, & \Gamma^\varphi_{\varphi\theta} &= -\cot\theta,\end{aligned}$$

$$R^r_r = -\frac{2}{arb} \partial_r \left(\frac{\partial_r(rb)}{a} \right), \quad (5.17)$$

$$R^\theta_\theta = \frac{1}{ar^2b^2} \left[a - \partial_r \left(\frac{rb\partial_r(rb)}{a} \right) \right]. \quad (5.18)$$

$$(5.19)$$

5.3 Theoretical Model

To begin, we use the perfect fluid approximation for the matter model. The stress-energy tensor takes the form

$$T_{ab} = (\rho + P)u_a u_b + P g_{ab}, \quad (5.20)$$

where $u^a(r, t)$ is the 4-velocity of a given perfect fluid element, $P(r, t)$ is the isotropic pressure, $\rho(r, t) = \rho_0(r, t)(1 + \varepsilon(r, t))$ is the energy density, $\rho_0(r, t)$ is the rest-mass energy density, and $\varepsilon(r, t)$ is the specific internal energy.

The equations of motion for this problem are derived from the local conservative equations for energy and baryon number such that

$$\nabla_a T^a_b = 0, \quad (5.21)$$

$$\nabla_a (\rho_0 u^a) = 0. \quad (5.22)$$

We use a high-resolution shock capturing method so we would like to write this in terms

of a flux-conservative form such that

$$\partial_t \mathbf{U} + \partial_i \mathbf{F}^i = \mathbf{S}, \quad (5.23)$$

where \mathbf{U} ,

$$\mathbf{U} = \begin{pmatrix} \sqrt{\gamma} W \rho_0 \\ \sqrt{\gamma} \alpha T_j^t \\ \alpha^2 \sqrt{\gamma} T^{tt} - \sqrt{\gamma} W \rho_0 \end{pmatrix}, \quad (5.24)$$

and \mathbf{F}^i

$$\mathbf{F}^i = \begin{pmatrix} \sqrt{\gamma} W \rho_0 v^i \\ \sqrt{\gamma} \alpha T_j^i \\ \alpha^2 \sqrt{\gamma} T^{ti} - \sqrt{\gamma} W \rho_0 v^i \end{pmatrix}, \quad (5.25)$$

and \mathbf{S}

$$\mathbf{S} = \begin{pmatrix} 0 \\ \frac{1}{2} \sqrt{\gamma} \alpha T^{ab} g_{ab,j} \\ \alpha^2 \sqrt{\gamma} (T^{at} \partial_a \alpha - \Gamma_{ab}^0 T^{ab} \alpha) \end{pmatrix}, \quad (5.26)$$

where W is Lorentz factor such that $W = \alpha u^t$ and $v^i = u^i / u^t$. Under our choice of spherical symmetry, $u^a = (u^t, u^r, 0, 0)$ so we can reduce our equations to

$$\partial_t (r^2 ab W \rho_0) + \partial_r (r^2 ab W \rho_0 v^r) = 0, \quad (5.27)$$

$$\partial_t (r^2 ab \alpha T_r^t) + \partial_r (r^2 ab \alpha T_r^r) = \frac{1}{2} r^2 ab T^{ab} g_{ab,r}, \quad (5.28)$$

$$\partial_t (\alpha^2 r^2 ab T^{tt} - r^2 ab W \rho_0) + \partial_r (\alpha^2 r^2 ab T^{tr} - r^2 ab W \rho_0 v^r) = \alpha r^2 ab (T^{at} \partial_a \alpha - \Gamma_{ab}^0 T^{ab} \alpha). \quad (5.29)$$

It is useful to define following variables

$$D = \rho_0 ab W, \quad (5.30)$$

$$E = \rho_0 h W^2 - P, \quad (5.31)$$

$$S = \rho_0 h W^2 v, \quad (5.32)$$

$$\tau = E - D. \quad (5.33)$$

The nonzero components of T^a_b which we are using become

$$T^t_t = -E, \quad (5.34)$$

$$T^t_r = \frac{ab}{\alpha} S, \quad (5.35)$$

$$T^r_r = Sv + P, \quad (5.36)$$

$$T^\theta_\theta = T^\phi_\phi = P, \quad (5.37)$$

where we have defined the fluid velocity as measured by an Eulerian observer to be

$$v = \frac{ab}{\alpha} v^r = \frac{abu^r}{\alpha u^t}. \quad (5.38)$$

We also define $W = 1/\sqrt{1-v^2}$ and $h = 1 + \varepsilon + P/\rho_0$ is the specific enthalpy. We can rewrite our equations as

$$\partial_t(r^2 D) + \partial_r \left(\frac{r^2 \alpha}{ab} Dv \right) = 0, \quad (5.39)$$

$$\partial_t(r^2 S) + \partial_r \left(\frac{r^2 \alpha}{ab} (Ev + P) \right) = \frac{1}{2} r^2 ab T^{ab} g_{ab,r}, \quad (5.40)$$

$$\partial_t(r^2 \tau) + \partial_r \left(\frac{r^2 \alpha}{ab} (S - Dv) \right) = \alpha r^2 ab (T^{at} \partial_a \alpha - \Gamma^0_{ab} T^{ab} \alpha), \quad (5.41)$$

this can take the form

$$\partial_t \mathbf{u} + \frac{1}{r^2} \partial_r (X r^2 \mathbf{f}) = \boldsymbol{\psi}, \quad (5.42)$$

where

$$\mathbf{u} = \begin{pmatrix} D \\ S \\ \tau \end{pmatrix}, \quad \mathbf{f} = \begin{pmatrix} Dv \\ Ev + P \\ S - Dv \end{pmatrix}, \quad \boldsymbol{\psi} = \begin{pmatrix} 0 \\ \frac{ab}{2} ab T^{ab} g_{ab,r} \\ \alpha ab (T^{at} \partial_a \alpha - \Gamma^0_{ab} T^{ab} \alpha) \end{pmatrix}, \quad (5.43)$$

and $X = \alpha/(ab)$ which is purely geometric factor.

Now consider the Einstein equations. We define the quantities appearing in the 3+1

equations as

$$\rho_{hydro} = n_a n_b T^{ab} = \rho_0 h W^2 - P, \quad (5.44)$$

$$J_i = -\gamma_{ia} n_b T^{ab} = \rho_0 h W u_i, \quad (5.45)$$

$$\perp T_{ij} = \gamma_{ia} \gamma_{jb} T^{ab} = P \gamma_{ij} + \rho_0 h u_i u_j, \quad (5.46)$$

$$\perp T = \gamma^{ij} \perp T_{ij} = 3P + \rho_0 h (W^2 - 1). \quad (5.47)$$

With all of this in hand, we can write the Einstein equations in ADM form as

$$\partial_t \gamma_{ij} = -2\alpha K_{ij} + D_i \beta_j + D_j \beta_i, \quad (5.48)$$

$$\begin{aligned} \partial_t K^i_j &= \alpha (R^i_j + K K^i_j) - D^i D_j \alpha - 8\pi \alpha \left(\perp T^i_j - \frac{1}{2} \delta^i_j (\perp T - \rho_{hydro}) \right) \\ &\quad + \beta^k \partial_k K^i_j + K^i_k \partial_j \beta^k - K^k_j \partial_k \beta^i, \end{aligned} \quad (5.49)$$

where D_i is the covariant derivative on spatial hypersurfaces. The momentum and Hamiltonian constraints are given by

$$R + K^2 - K_{ij} K^{ij} = 16\pi \rho_{hydro}, \quad (5.50)$$

$$D_i K^i_j - D_j K = 8\pi J_j. \quad (5.51)$$

On substituting the hydrodynamic source terms (ρ_{hydro} , J_i etc) from above we have

$$\begin{aligned} \partial_t K^i_j &= \alpha (R^i_j + K K^i_j) - \gamma^{ik} (\partial_i \partial_k \alpha - \Gamma^l_{ik} \partial_l \alpha) - 8\pi \alpha \left(\frac{1}{2} \delta^i_j (\rho_0 h - 2P) + \rho_0 h u^i u_j \right) \\ &\quad + \beta^k \partial_k K^i_j + K^i_k \partial_j \beta^k - K^k_j \partial_k \beta^i, \end{aligned} \quad (5.52)$$

$$\partial_t \gamma_{ij} = -2\alpha K_{ij} + D_i \beta_j + D_j \beta_i, \quad (5.53)$$

$$R + K^2 - K_{ij} K^{ij} = 16\pi (\rho_0 h W^2 - P), \quad (5.54)$$

$$D_i K^i_j - D_j K = 8\pi \rho_0 h W u_j. \quad (5.55)$$

From our choice of coordinates, we calculate the non-trivial connection coefficients and Ricci tensor. The metric form suggests that $\beta^i = (\beta^r, 0, 0)$, $K^i_j = \text{diag}(K^r_r, K^\theta_\theta, K^\theta_\theta)$.

Using these facts, the evolution equations for the geometric quantities are

$$\partial_t a = -\alpha a K_r^r + \partial_r(a\beta^r), \quad (5.56)$$

$$\partial_t b = -\alpha b K_\theta^\theta + \frac{\beta^r}{r} \partial_r(r\beta^r), \quad (5.57)$$

$$\begin{aligned} \partial_t K_r^r &= \beta^r \partial_r K_r^r + \alpha K_r^r K - \frac{1}{a} \partial_r \left(\frac{\partial_r \alpha}{a} \right) - \frac{2\alpha}{arb} \partial_r \left(\frac{\partial_r(rb)}{a} \right) \\ &\quad - 4\pi\alpha [(1 + 2u^r u_r)\rho_0 h - 2P], \end{aligned} \quad (5.58)$$

$$\begin{aligned} \partial_t K_\theta^\theta &= \beta^r \partial_r K_\theta^\theta + \alpha K_\theta^\theta K - \frac{\alpha}{r^2 b^2} - \frac{1}{ar^2 b^2} \partial_r \left(\frac{\alpha r b \partial_r(rb)}{a} \right) \\ &\quad - 4\pi\alpha(\rho_0 h - 2P). \end{aligned} \quad (5.59)$$

while the constraint equations are

$$\begin{aligned} \frac{1}{ar^2 b^2} \left[a - \partial_r \left(\frac{rb \partial_r(rb)}{a} \right) \right] - \frac{2}{arb} \partial_r \left(\frac{\partial_r(rb)}{a} \right) + 2K_\theta^\theta (K_\theta^\theta + 2K_r^r) &= 16\pi(\rho_0 h W^2 - P), \\ \frac{\partial_t(rb)}{rb} (K_\theta^\theta - K_r^r) - \partial_r K_\theta^\theta &= 4\pi\rho_0 h W u_r. \end{aligned} \quad (5.60)$$

Note that we have residual coordinate freedom including the choice of slicing condition. We can apply different choice of slicing (i.e. a gauge choice) to simplify and reduce the above system. Two possible choices include maximal ($\partial_t K = 0$) or polar slicing ($K_\theta^\theta + K_\varphi^\varphi = 0$).

5.3.1 Choice of Gauge

In the previous section, we derived our system of equations in horizon penetrating coordinates. This set of equations is not quite ready for numerical integration. We must still impose coordinate conditions by specifying the lapse α and the shift β^r . In general, finding kinematical conditions for the coordinates that allow for a well-behaved and a stable evolution is not trivial. Even if all the equations are written in terms of horizon penetrating coordinates, a simulation may fail without a good coordinate choice as matter passes through the event horizon. In this work, we present two possible choices of gauge. The first is maximal slicing ($\partial_t K = 0$). The other is 1+log slicing with Γ driver condition.

5.3.1.1 Maximal Slicing

We first consider maximal slicing. For this case, $K = \partial_t K = 0$ and the evolution equations become

$$\partial_t a = -\alpha a K^r_r + \partial_r(a\beta^r), \quad (5.61)$$

$$\partial_t b = \frac{\alpha b}{2} K^r_r + \frac{\beta^r}{r} \partial_r(r\beta^r), \quad (5.62)$$

$$\begin{aligned} \partial_t K^r_r = & \beta^r \partial_r K^r_r - \frac{1}{a} \partial_r \left(\frac{\partial_r \alpha}{a} \right) - \frac{2\alpha}{arb} \partial_r \left(\frac{\partial_r(rb)}{a} \right) - \frac{2\alpha}{r^2 b^2} - \frac{2}{ar^2 b^2} \partial_r \left(\frac{\alpha rb \partial_r(rb)}{a} \right) \\ & - 8\pi\alpha [(2 + 2u^r u_r)\rho_0 h - 4P]. \end{aligned} \quad (5.63)$$

with the constraints taking the form

$$\begin{aligned} \frac{1}{ar^2 b^2} \left[a - \partial_r \left(\frac{rb \partial_r(rb)}{a} \right) \right] - \frac{2}{arb} \partial_r \left(\frac{\partial_r(rb)}{a} \right) - \frac{3}{2} (K^r_r)^2 = 16\pi(\rho_0 h W^2 - P), \\ \partial_r K^r_r - \frac{3\partial_t(rb)}{rb} K^r_r = 8\pi\rho_0 h W u_r. \end{aligned} \quad (5.64)$$

The equation of motion for the fluid remains the same as previously

Because of the slicing condition, the lapse now obeys

$$\partial_t K = -D^2 \alpha + \alpha(K^{ij} K_{ij} + 4\pi(\rho + S)) + \beta^i D_i K. \quad (5.65)$$

In our choice of gauge, this can be re-written as

$$\begin{aligned} D^2 \alpha = & \alpha(K^{ij} K_{ij} + 4\pi(\rho + S)) = \alpha \left(K^{ij} K_{ij} + 8\pi \left[P + \rho_0 h \left(W^2 - \frac{1}{2} \right) \right] \right) \\ = & \alpha \left(2(K^r_r)^2 + 8\pi \left[P + \rho_0 h \left(W^2 - \frac{1}{2} \right) \right] \right). \end{aligned} \quad (5.66)$$

For the shift, we use

$$\partial_t \ln \sqrt{\gamma} = -\alpha K + D_i \beta^i, \quad (5.67)$$

which, in maximal slicing, reduces to

$$D_i \beta^i = -\partial_t \ln \sqrt{\gamma}. \quad (5.68)$$

Alternatively, we can write this equation as

$$\partial_i \beta^i = -\partial_t \sqrt{\gamma}. \quad (5.69)$$

This shows that the proper volume element $\sqrt{\gamma}$ satisfies a continuity equation in maximal slicing.

In terms of our metric choice and variables, this equation becomes

$$\partial_r \beta^r = \frac{b}{2b + \beta^r} \left[\frac{\alpha K_r^r}{2} - \frac{(\beta^r)^2 b}{r} - \frac{\partial_r a}{a} \beta^r \right]. \quad (5.70)$$

5.3.1.2 Spherical-BSSN formulation with 1+log and Γ driver

Another possible way to describe this system is by writing the equations in the BSSN formulation. This allows us to take advantage of the strong hyperbolicity of the BSSN formulation. We follow the usual treatment in defining BSSN variables. (See Appendix A or Chapter 2 for a detailed discussion of this formulation.) We will continue to use horizon penetrating coordinates described in the previous section.

On considering the usual BSSN formulation, we make the conformal transformation ($\bar{\gamma}_{ij} = e^{4\phi} \gamma_{ij}$) and define the conformal factor

$$\phi = \frac{1}{12} \ln(\gamma/\bar{\gamma}). \quad (5.71)$$

It is common to define a new variable $\chi = e^{-2\phi}$ instead of using the conformal factor ϕ .

The evolution equation for χ becomes

$$\partial_t \chi = \beta^r \partial_r \chi - \frac{1}{3} \chi \left[\alpha K - \partial_r \beta^r - \beta^r \left(\frac{\partial_r(ab^2)}{2ab^2} + \frac{2}{r} \right) \right]. \quad (5.72)$$

We also use the traceless part of the extrinsic curvature

$$\bar{A}_{ij} = e^{-4\phi} \left(K_{ij} - \frac{1}{3} \gamma_{ij} K \right). \quad (5.73)$$

In spherical symmetry, we have \bar{A}_r^r , and \bar{A}_θ^θ . Note that as \bar{A}_{ij} is traceless so $\bar{A}_r^r + 2\bar{A}_\theta^\theta =$

0. The evolution equations for the spatial metric eventually give

$$\partial_t a = \beta^r \partial_r a + 2a \partial_r \beta^r - \frac{2a}{3} \left[\partial_r \beta^r + \beta^r \left(\frac{\partial_r(ab^2)}{2ab^2} + \frac{2}{r} \right) \right] - 2\alpha a \bar{A}_r^r, \quad (5.74)$$

$$\partial_t b = \beta^r \partial_r b + 2b \frac{\beta^r}{r} - \frac{2b}{3} \left[\partial_r \beta^r + \beta^r \left(\frac{\partial_r(ab^2)}{2ab^2} + \frac{2}{r} \right) \right] + \alpha b \bar{A}_r^r. \quad (5.75)$$

The evolution equation for the trace of the extrinsic curvature K is given by

$$\begin{aligned} \partial_t K &= \beta^r \partial_r K - \frac{\chi^2}{\alpha} \left[\partial_r^2 \alpha - \partial_r \left(\frac{\partial_r a}{2a} - \frac{\partial_r b}{b} + \frac{\partial_r \chi}{\chi} - \frac{2}{r} \right) \right] \\ &+ \alpha \left(\frac{3}{2} (\bar{A}_r^r)^2 + \frac{1}{3} K^2 \right) + 4\pi\alpha (\rho_{hydro} + \perp T_r^r + \perp T_\theta^\theta). \end{aligned} \quad (5.76)$$

We have only one independent component of the traceless part of the extrinsic curvature.

The evolution equation for it is

$$\begin{aligned} \partial_t \bar{A}_r^r &= \beta^r \partial_r \bar{A}_r^r - \frac{\chi^2}{\alpha} \left(\left[\partial_r^2 \alpha - \partial_r \left(\frac{\partial_r a}{2a} - \frac{\partial_r \chi}{\chi} \right) \right] - \frac{1}{3} \left[\partial_r^2 \alpha - \partial_r \left(\frac{\partial_r a}{2a} - \frac{\partial_r b}{b} + \frac{\partial_r \chi}{\chi} - \frac{2}{r} \right) \right] \right) \\ &+ \frac{\alpha}{3} (2R_r^r - R^\theta_\theta) + \alpha K \bar{A}_r^r - 16\pi\alpha (\perp T_r^r - \perp T_\theta^\theta). \end{aligned} \quad (5.77)$$

Finally, we obtain the evolution equation for $\bar{\Delta}^r$ which is the radial component of the conformal connection functions which are defined by $\bar{\Delta}^i = \tilde{\gamma}^{mn} (\bar{\Gamma}_{mn}^i - \dot{\Gamma}_{mn}^i)$ where $\dot{\Gamma}_{mn}^i$ is defined by $\dot{\gamma}_{ij}$ which is the flat metric in spherical coordinate i.e. $\dot{\gamma} = r^2 \sin^2 \theta$. This equation is

$$\begin{aligned} \partial_t \bar{\Delta}^r &= \beta^r \partial_r \bar{\Delta}^r - \bar{\Delta}^r \partial_r \beta^r + \frac{1}{a} \partial^2 \beta^r + \frac{2}{b} \partial_r \left(\frac{\beta^r}{r} \right) \\ &+ \frac{1}{3} \left[\frac{1}{a} \left\{ \partial_r^2 \beta^r + \partial_r \beta^r \left(\frac{\partial_r (ab^2)}{2ab^2} + \frac{2}{r} \right) + \beta^r \partial_r \left(\frac{\partial_r (ab^2)}{2ab^2} + \frac{2}{r} \right) \right\} \right. \\ &\left. + 2\bar{\Delta}^r \left\{ \partial_r \beta^r + \beta^r \left(\frac{\partial_r (ab^2)}{2ab^2} + \frac{2}{r} \right) \right\} \right] \\ &- \frac{2}{a} (\bar{A}_r^r \partial_r \alpha + \alpha \partial_r \bar{A}_r^r) + 2\alpha \left(\bar{A}_r^r \bar{\Delta}^r + \frac{3}{rb} \bar{A}_r^r \right) \\ &+ \frac{2\alpha}{a} \left[\partial_r \bar{A}_r^r - \frac{2}{3} \partial_r K - \frac{3\bar{A}_r^r}{\chi} \partial_r \chi + \frac{3}{2} \bar{A}_r^r \left(\frac{2}{r} + \frac{\partial_r b}{b} \right) - 8\pi J_r \right]. \end{aligned} \quad (5.78)$$

We also evaluate the Hamiltonian and momentum constraints in order to monitor the fidelity of our numerical evolutions. These are

$$\mathcal{H} = -\frac{2}{arb} \partial_r \left(\frac{\partial_r (rb)}{a} \right) + \frac{1}{ar^2 b^2} \left[a - \partial_r \left(\frac{rb \partial_r (rb)}{a} \right) \right] - \frac{3}{2} (\bar{A}_r^r)^2 + \frac{2}{3} K^2 - 16\pi \rho_{hydro}, \quad (5.79)$$

$$\mathcal{M} = \partial_r \bar{A}_r^r - \frac{2}{3} \partial_r K - \frac{3\bar{A}_r^r}{2\chi} \partial_r \chi + \frac{2}{3} \bar{A}_r^r \left(\frac{2}{r} + \frac{\partial_r b}{b} \right) - 8\pi J_r. \quad (5.80)$$

Both of these are analytically zero. In numerical simulations, these will not be zero. How close to zero provides a measure of the accuracy of our numerical simulations.

In addition to BSSN variables in spherical symmetry, we still have two more variables that need to be chosen. This amounts to the gauge condition which we have not yet chosen. Here, we use the 1+log condition for the lapse and the gamma driver condition for the shift. The form of this slicing condition is

$$\partial_t \alpha = -2\alpha K. \quad (5.81)$$

As we have only the radial component for the shift vector, the gamma driver condition is simply

$$\partial_t B^r = \frac{3}{4} \partial_t \bar{\Delta}^r, \quad (5.82)$$

$$\partial_t \beta^r = B^r, \quad (5.83)$$

into which we have introduced the auxiliary variable B^r to reduce the derivative order in time from second to first.

5.3.2 Initial Data

Our initial NS model is approximated by a TOV star. After this initial data calculation, an in-going velocity profile is added to induce the star's collapse. We follow the method as described in [196]. We first specify the coordinate velocity

$$U \equiv \frac{dr}{dt} = \frac{u^r}{u^t}, \quad (5.84)$$

of the star. In general, the profile takes the algebraic form $U_g(x) = A_0(x^3 - B_0x)$ where $x \equiv r/R_{star}$ and R_{star} is the radius of the TOV solution.

In this work, we set two profiles

$$U(x) = \begin{cases} U_1(x) = U_{crit}(x^3 - 3x) & x < x_{lv} \\ U_2(x) = 0 & \text{otherwise} \end{cases}, \quad (5.85)$$

where U_{crit} is the amplitude that occurs for critical collapse, and x_{tlv} defines the region that forms a black hole.

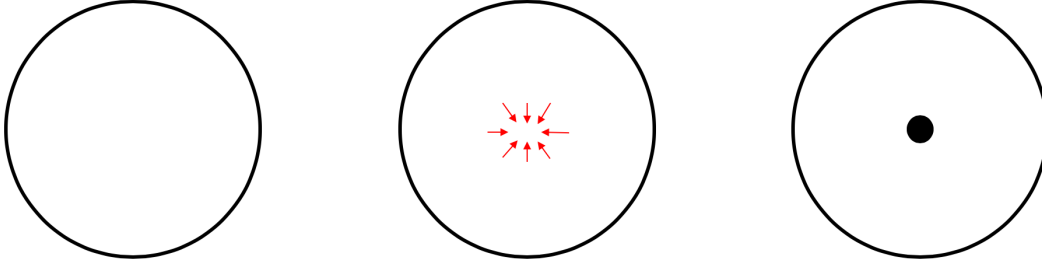


Figure 5.1 Description for generation of ID. Small mass BH is created in the center of TOV star by imposing large ingoing velocity that occurs critical collapse

Fig 5.1 shows a schematic view of our initial data problem. Our interest, of course, is in the interaction between the BH interior to the NS and the enveloping NS. We set x_{tlv} to be a small value such as 1% the size of the star i.e. $x_{tlv} = 0.01$ because x is the normalized star radius (x_{tlv} must be smaller than 1).

5.4 Results

In this section, we present some results of our research. One important question this model addresses is the collapse time of the neutron star, or the time when all the NS matter has been consumed by the BH. This quantity, should, in principle, vary with respect to the initial size of the small, interior BH. After the BH is formed due to gravitational collapse at the center of the NS, all the star material will eventually be consumed. Since the small BH can have different sizes by setting x_{tlv} , we expect that this collapse time will be different. In some sense, this scenario is a toy model for the similarly interesting astrophysical problem of a BH-NS binary merger. If a BH-NS binary has a total mass on the order of $1M_{\odot}$ (i.e. less than the Chandrasekhar limit), it is still expected that much of the NS material exterior to the BH will be consumed by the BH. It becomes important to check the time

scale in which BH-NS binaries may merge and have the NS material fall down the BH. Comparing it to our timescale might provide a similar measure. Of course, our simplified model concentrates on the spherically symmetric case, i.e. no rotation but this provides some basic insight for future studies.

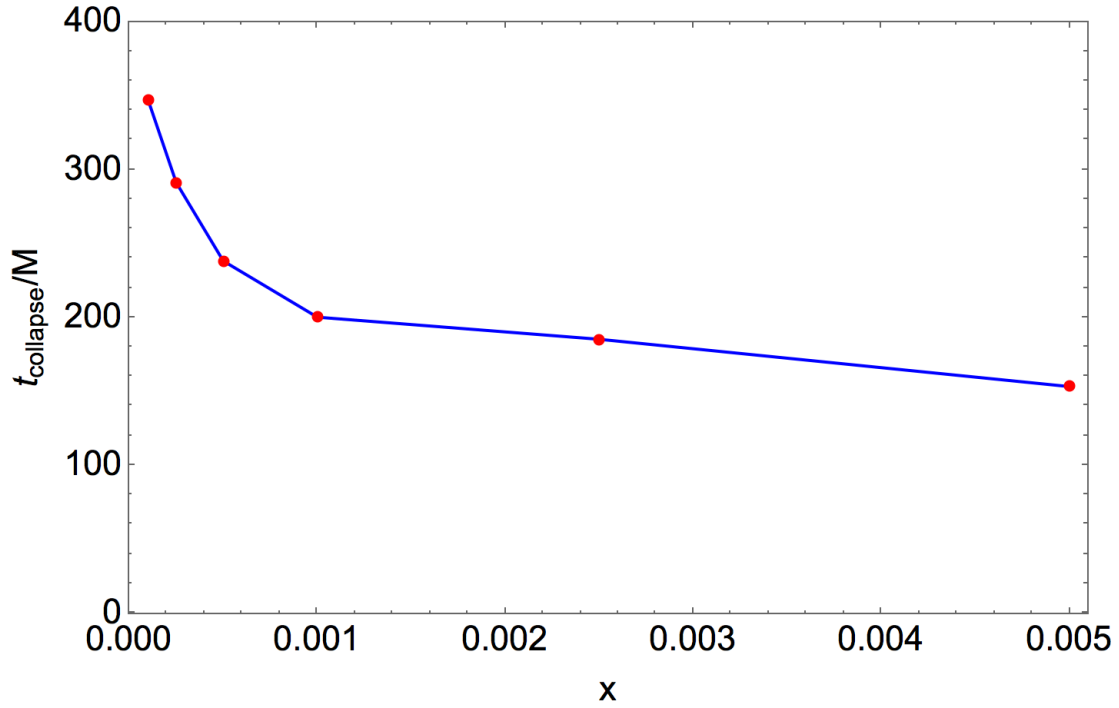


Figure 5.2 Different collapse times with respect to different initial sizes of the BH. The mass of the TOV star is $1M_{\odot}$ and x is the normalized radius for the BH. We set the star's radius R_{star} to be one.

Fig. 5.2 shows different collapse times with respect to different initial sizes of the interior BH. We measure $t_{collapse}$ when all the TOV material has been consumed by the BH. For this test, the mass of the TOV star is $1M_{\odot}$, the star radius $R_{star} = 1$, and x is the normalized radius that can be varied by setting x_{tlv} . We test various values, namely $x = 0.0001, 0.00025, 0.0005, 0.001, 0.0025$, and 0.005 . Note that there are distinct time differences until $x = 0.001$ but after this, the difference in collapse time becomes insignificant. Typical interesting relevant mass regimes for the PBH include $10^{-14} - 10^{-7}M_{\odot}$ which can be $10^{-3} - 0.5\%$ of $1R_{star}$ for $1M_{\odot}$ NS. We observe that there are distinct time differences until $x = 0.001$ (or 0.1% of $1R_{star}$) which is an interesting mass regime for the PBH. This

suggests that we can constrain the time scale for various sizes of BH formation and the NS materials that fall down the BH. For the $x \geq 0.001$ case, the time scale will not be distinguishable and might be shorter than the observed ages of NS in various astrophysical channels. Another interesting aspect for this system is searching for possible r -process nucleosynthesis. Because the mass of the BH is small, we expect that the resulting ejecta will have relatively cold neutron rich material. However, in this study, we do not observe any remnant or ejected materials. This does not necessarily imply that we will not have any ejecta or accretion around the PBH but finding the correct regime to have ejecta remains open. Our results do show different time scales and we expect that these time scales may differ when compared between the two major sources of r -process nucleosynthesis: BNS mergers and core-collapse supernovae. Therefore, combining different collapse times with possible ejecta or remnants will provide precise information about r -process nucleosynthesis channel which may explain the primary source for different heavy elements that might not be produced via BNS mergers or core collapse supernovae in the universe.

Because this work is limited to the spherically symmetric case, we would like to extend these results to axisymmetry and even the full 3D case thereby enabling rotations. One interesting question will be to explore the final phase of these dynamics. In the final stage of BH-NS binaries, the NS is either tidally disrupted or fully swallowed by the companion BH so we could compare these scenarios with our problem. Eventually, we will compute GW waveforms of this system and this will provide complementary information for future GW detectors as well as DM searches [14, 188, 189, 190, 192].

5.5 Conclusions

In this work, we have explored the interaction of a small mass BH interior to a NS by solving the Einstein equations coupled to hydrodynamics in spherical symmetry. As initial data, we started with a TOV star within which we formed a small BH. Once formed, we performed the dynamical evolution in order to examine the collapse times of the NS matter

into the BH as a function of the initial BH size. From our experiments, we found the forming a BH at the center of the NS is indeed possible and that the subsequent evolution results in the complete consumption of the NS. We do not observe any remnant or ejected material from this system. This indicates that the spherically symmetric system alone does not capture all the necessary physics to permit r -process nucleosynthesis from this system. Note that this is in contrast to other studies which have suggested possible remnant material and ejecta being present in this system. Although the theoretical understanding for the progenitors of this system is debatable, our dynamical, toy model version of this system at least suggests that interaction between a small mass BH and NS can provide new insights for future observations and detections.

Chapter 6

Conclusion

In this thesis, we have explored various aspects of dynamical compact objects. First, we explored BH dynamics in a modified theory of gravity. Using numerical simulations, we investigated dynamical black holes in EMDA theory. We consider a variety of initial data types in order to examine both the stability of single black holes in this theory as well as possible alternate scalar and electromagnetic field channels for emission. We also investigate BBH mergers in order to probe deviations from the standard gravitational wave signatures of GR. In addition, we explore quadratic gravity which is known to be perturbatively renormalizable as an effective quantum field theory. Corresponding unitarity problems only arise at trans-Planckian scales, where a full quantum theory of gravity is assumed to take over. Firstly, we review a proof of a well-posed initial value problem in quadratic gravity in generalized harmonic coordinates. We argue that this method can be applied to other possible alternative theories of gravity. Secondly, we explore possible physically stable dynamical black hole configurations in this theory using numerical simulations. Lastly, we discuss the possible generation of GW templates from BBH mergers in quadratic gravity.

Next we present some early results from a study of a more exotic merger scenario, namely intermediate mass ratio inspirals (IMRIs) of BBHs with mass ratios on the order of 100:1. To achieve this, we present a highly-scalable framework that targets problems of interest to the numerical relativity and broader astrophysics communities. This framework

combines a parallel octree-refined adaptive mesh with a wavelet adaptive multiresolution approach together with a physics module to solve the Einstein equations of GR in the BSSN formulation. These studies will be used to generate waveforms as used in LIGO data analysis and to calibrate semi-analytical approximate methods. Our framework consists of a distributed memory octree-based adaptive meshing framework in conjunction with a node-local code generator. The code generator makes our code portable across different architectures. The equations corresponding to the target application are written in symbolic notation and generators for different architectures can be added independently of the application. Additionally, this symbolic interface also makes our code extensible, and as such has been designed to easily accommodate many existing algorithms in astrophysics for plasma dynamics and radiation hydrodynamics. Our adaptive meshing algorithms and data-structures have been optimized for modern architectures with deep memory hierarchies. This enables our framework to achieve excellent performance and scalability on modern leadership architectures. We demonstrate excellent weak scalability up to 131K cores on ORNL's Titan for binary mergers for mass ratios up to 100.

Third, we study binary neutron star coalescence and their resulting ejecta. Observational signatures include GWs and faint supernova-like transients powered by radioactive decay of freshly synthesized heavy elements. Due to the complexity of the problem, the only way to understand these observations is to confront them with the predictions obtained via simulation. We use smoothed particle hydrodynamics, which is well suited for such problems, and introduce the highly scalable FLeCSPH code to simulate these mergers. Furthermore, we augment FLeCSPH by incorporating tabulated equations of state to improve the physics accuracy.

Finally, we consider a toy model of interactions between BHs and NSs. Several recent studies show that constraints can be put on primordial black holes (PBH) as a candidate for dark matter by considering their capture by NSs. If a PBH is captured by a NS, the NS is accreted onto the PBH and eventually collapses. A related scenario arises from asymmetric dark matter particles collecting in the center of NS, and eventually forming a small BH. In

this work, we study such systems where a small BH has formed in the center of a NS, and follow the dynamics as the small BH consumes the NS from inside. We solve numerically the coupled system of equations formed by the Einstein equations and hydrodynamics under spherical symmetry. Using numerical simulations, we examine the time scale for the NS to collapse into the PBH, and the possible unbounded material from the NS.

For future work, we will continue to explore additional aspects of compact object dynamics using various method and toolkits that have been described in this dissertation. These future works include:

For EMDA, we will perform more extreme cases such as high spin, large EM and scalar counterparts to explore possible deviations from GR. Also, improving the initial data for the BBHs may provide an interesting aspect to the simulations. Extending Maxwell fields to massive fields, or so-called Proca fields, will also be studied. For quadratic gravity, we will implement a full 3D code to extract GW information from the BBH merger. Comparing the linear stability analysis with our fully nonlinear evolutions will help determine stability considerations in higher order derivative gravity theories such as massive gravity, dRGT gravity. Also, searching for possible star formation scenario in this theory can provide interesting playgrounds for future detections. With IMRIs, we plan to perform more comprehensive tests for different mass ratio and create a GW catalogue for IMRIs across a range of mass ratios. This data might be used for future LISA detections. Also, we expect to include additional ingredients such as hydrodynamics, different EOS, an apparent horizon finder, and mesh generations for problems in different spacetime dimensions. Clearly, this will enhance the overall capabilities of our code. For BNS, we will include a nuclear network in order to perform more realistic nucleosynthesis analyses. Furthermore, coupling the BNS code with GRMHD will be considered to broaden the range of astrophysical phenomena that we can study. Relativistic formulations of SPH will also be studied and implemented. For PBH-NS, we will extend our work to situations more general than spherical symmetry, such as have rotating NS. Such dynamical studies may provide additional clues for constraining dark matter via PBH.

In summary, our detailed studies of compact objects can provide complementary information for not only current ground based GW detectors but also future space based GW detectors such as LISA. This will open new windows to study gravitational physics from quantum particles to cosmos.

Appendix A

At Introduction to the Mathematical Formulation of Numerical Relativity

A.1 The 3+1 Decomposition

There are numerous introductions to the Einstein equations of general relativity and how to determine their dynamical evolution. In this appendix, we will not consider all the mathematical and physical details associated with the dynamics of the Einstein equations. Our focus will be on a few formalisms and techniques needed to decompose the Einstein equations into forms amenable for numerical simulation. More detailed discussions can be found in the references [197, 198]

A.1.1 Foliation of Spacetime

Consider first a curved four dimensional spacetime manifold M and the metric in this spacetime g_{ab} . We assume that the spacetime (M, g_{ab}) can be foliated into a family of nonintersecting spacelike 3-surfaces Σ (hypersurface). These can be thought to arise as the level surfaces of a scalar function t that can be interpreted as a global time function. From

t , we can define the 1-form

$$\Omega_a = \nabla_a t, \quad (\text{A.1})$$

which, by construction, satisfies

$$\nabla_{[a} \Omega_{b]} = \nabla_{[a} \nabla_{b]} t = 0. \quad (\text{A.2})$$

The 4-metric g_{ab} defines the norm of Ω_a such that

$$\|\Omega\|^2 \equiv \Omega_a \Omega^a = \nabla_a t \nabla^a t = g^{ab} \nabla_a t \nabla_b t \equiv -\frac{1}{\alpha^2}, \quad (\text{A.3})$$

where α measures how much proper time elapses between neighboring time slices along the normal vector Ω^a to the slice, and is therefore called the *lapse* function. We assume $\alpha > 0$ so that Ω^a is future directed timelike and the hypersurface Σ is spacelike everywhere.

Further, we define the normalized 1-form

$$\omega_a \equiv \alpha \Omega_a, \quad (\text{A.4})$$

note that ω_a is rotation free i.e. $\omega_{[a} \nabla_b \omega_{c]} = 0$. With this, we can define the unit normal to the slices as

$$n^a \equiv -g^{ab} \omega_b, \quad (\text{A.5})$$

so that $n_a = -\omega_a$. Note that the negative sign is chosen so that n^a points in the direction of increasing t . As such, we, of course, have

$$n^a \omega_a = -g^{ab} \omega_a \omega_b = -\alpha^2 g^{ab} \Omega_a \Omega_b = 1, \quad (\text{A.6})$$

so that n^a is normalized and timelike by construction

$$n^a n_a = g^{ab} \omega_a \omega_b = -1. \quad (\text{A.7})$$

With a normal vector in hand, we can construct the spatial metric γ_{ab} that is induced on the 3-dimensional hypersurfaces by the foliation in terms of it and g_{ab} :

$$\gamma_{ab} = g_{ab} + n_a n_b. \quad (\text{A.8})$$

Thus γ_{ab} can be considered a projection operator or tensor that takes geometric objects out of the full spacetime into the hypersurfaces alone. It is also a metric as it allows us to compute distances within a slice, Σ . We can further see that γ_{ab} is purely spatial i.e. it resides entirely in Σ with no part lying along the orthogonal direction given by n^a :

$$n^a \gamma_{ab} = n^a (g_{ab} + n_a n_b) = n^a g_{ab} + n^a n_a n_b = n^a g_{ab} + \underbrace{n^a n_a}_{=-1} n_b = n_b - n_b = 0. \quad (\text{A.9})$$

The inverse spatial metric can be shown to be

$$\gamma^{ab} = g^{ac} g^{bd} \gamma_{cd} = g^{ac} g^{bd} (g_{cd} + n_c n_d) = g^{ab} + n^a n^b. \quad (\text{A.10})$$

Our intent here is to break up 4-dimensional tensors by decomposing them into purely spatial parts which lie wholly in the hypersurfaces Σ and timelike parts which remain normal to Σ . To do that, we need two projection operators. The first one, which projects a 4-dimensional tensor into a spatial slice, can be found by raising only one index of the spatial metric γ_{ab}

$$\gamma^a_b = g^a_b + n^a n_b = \delta^a_b + n^a n_b. \quad (\text{A.11})$$

To project higher rank tensors into the spatial surface, each free index has to be contracted with a projection operator. We sometimes denote this projection with a symbol \perp . For example, a completely projected second rank tensor will read

$$\perp T_{ab} = \gamma^c_a \gamma^d_b T_{cd}. \quad (\text{A.12})$$

Similarly, we define the normal projection operator

$$N^a_b \equiv -n^a n_b = \delta^a_b - \gamma^a_b. \quad (\text{A.13})$$

Using these two projection operators, we can decompose any tensor into its spatial and timelike parts. For example, an arbitrary vector v^a can be decomposed in the following way

$$v^a = \delta^a_b v^b = (\gamma^a_b + N^a_b) v^b = \perp v^a - n^a n_b v^b. \quad (\text{A.14})$$

Similarly, we find for a second rank tensor:

$$\begin{aligned}
T_{ab} &= \delta^c_a \delta^d_b T_{cd} = (\gamma^c_a + N^c_a)(\gamma^d_b + N^d_b)T_{cd} \\
&= (\gamma^c_a \gamma^d_b + \gamma^c_a N^d_b + \gamma^d_b N^c_a + N^c_a N^d_b)T_{cd} \\
&= \gamma^c_a \gamma^d_b T_{cd} - \gamma^c_a n^d_b T_{cd} - \gamma^d_b n^c_a T_{cd} + n^c_a n^d_b T_{cd} \\
&= \perp T_{ab} - n_b \perp (T_{ad} n^d) - n_a \perp (n^c T_{cb}) + n_a n_b n^c n^d T_{cd}. \tag{A.15}
\end{aligned}$$

In this example, note that the use of the \perp symbol must be used with some care, as it applies only to the free indices of the tensor on which it operates. To avoid confusion, we will often write out the projection operators explicitly.

We also need a 3-dimensional covariant derivative that maps spatial tensors into spatial tensors. It is uniquely defined by requiring that it be compatible with the 3-dimensional metric γ_{ab} . We can construct this derivative by projecting all indices present in a 4-dimensional covariant derivative into Σ . For example, for a scalar, we define

$$D_a f \equiv \gamma^b_a \nabla_b f. \tag{A.16}$$

For a higher rank tensor, the pattern simply repeats

$$D_a T_c^b \equiv \gamma^d_a \gamma^b_e \gamma^f_c \nabla_d T_f^e. \tag{A.17}$$

Other types tensors follow similar rules to those above. Further, the 3-dimensional covariant derivative is compatible with the spatial metric i.e. $D_a \gamma_{bc} = 0$. Note too, that for purely spatial vectors, the 3-dimensional covariant derivative satisfies the usual Leibniz rule.

As in the 4-dimensional case, we can define the connection coefficients, the Riemann tensor, and the Ricci tensor associated with the 3D hypersurfaces using D_a and γ_{ab}

$$\Gamma^a_{bc} = \frac{1}{2}\gamma^{ad}(\partial_c\gamma_{db} + \partial_b\gamma_{dc} - \partial_d\gamma_{bc}), \quad (\text{A.18})$$

$$2D_{[a}D_{b]}w_c = R^d{}_{cba}w_d, \quad R^d{}_{cba}n_d = 0, \quad (\text{A.19})$$

$$R_{abc}{}^d = \partial_b\Gamma^d{}_{ac} - \partial_a\Gamma^d{}_{bc} + \Gamma^e{}_{ac}\Gamma^d{}_{eb} - \Gamma^e{}_{bc}\Gamma^d{}_{ea}, \quad (\text{A.20})$$

$$R_{ab} = R^c{}_{acb}, \quad (\text{A.21})$$

$$R = R^a{}_a. \quad (\text{A.22})$$

A.1.2 The Extrinsic Curvature

The Einstein equations relate contractions of the 4-dimensional Riemann tensor ${}^{(4)}R^a{}_{bcd}$ to the stress-energy tensor T_{ab} . The goal of the 3+1 decomposition is to break ${}^{(4)}R^a{}_{bcd}$ into spatial tensors that live wholly in the 3-slices and other pieces that have “legs” in the orthogonal, timelike direction. This decomposition will naturally involve the 3-dimensional Riemann tensor $R^a{}_{bcd}$ which contains information about the intrinsic curvature of these spacelike slices. However, this cannot contain all the information residing in the full 4D spacetime. Indeed, $R^a{}_{bcd}$ is a purely spatial object and can be computed from spatial derivatives of the spatial metric alone, while ${}^{(4)}R^a{}_{bcd}$ describes the curvature of the full 4D spacetime. Of course, this will also contain time derivatives of the 4-dimensional metric. Another way to say this is that the 3-dimensional curvature $R^a{}_{bcd}$ only contains information about the curvature intrinsic to a slice Σ , but it gives no information about how these slices “fit” into the larger spacetime M in which it is embedded. This extra information is contained in a tensor called the *extrinsic* curvature.

The extrinsic curvature, K_{ab} , can be found by projecting gradients of the normal vector into the slice Σ . The projection of the gradient of the normal vector $\gamma^c{}_a\gamma^d{}_b\nabla_c n_d$ can be split into a symmetric part, also known as the expansion tensor

$$\theta_{ab} = \gamma^c{}_a\gamma^d{}_b\nabla_{(c}n_{d)}, \quad (\text{A.23})$$

and an antisymmetric part, also known as the rotation 2-form or twist

$$\omega_{ab} = \gamma^c{}_a \gamma^d{}_b \nabla_{[c} n_{d]}. \quad (\text{A.24})$$

We now define the extrinsic curvature, K_{ab} , as the negative expansion

$$K_{ab} = -\gamma^c{}_a \gamma^d{}_b \nabla_{(c} n_{d)} = -\gamma^c{}_a \gamma^d{}_b \nabla_c n_d. \quad (\text{A.25})$$

By definition, the extrinsic curvature is symmetric and purely spatial. It is a measure of the change in the normal vector n^a . Since the latter is normalized, it does so only in the direction in which it is pointing, and thus the extrinsic curvature is the indication of how much this direction changes from point to point across a spatial hypersurface. Said another way, the extrinsic curvature measures the rate at which the hypersurface deforms as it is carried forward along a normal.

We can express the extrinsic curvature in terms of the acceleration of the unit normal vector field

$$a_a \equiv n^b \nabla_b n_a. \quad (\text{A.26})$$

By definition, the acceleration is purely spatial i.e. $n^a a_a = 0$. Significantly, there is a key relationship between the lapse and the acceleration which we now derive. From the definition of a_a together with $n_a = -\omega_a = -\alpha \Omega_a = -\alpha \nabla_a t$ we can write

$$\begin{aligned} a_a &= n^b \nabla_b n_a = -n^b \nabla_b (\alpha \nabla_a t) \\ &= -n^b (\nabla_b \alpha) (\nabla_a t) - n^b \alpha \nabla_b \nabla_a t \\ &= \frac{1}{\alpha} n^b n_a \nabla_b \alpha - n^b \alpha \nabla_a \nabla_b t = \frac{1}{\alpha} n^b n_a \nabla_b \alpha - n^b \alpha \nabla_a \left(-\frac{1}{\alpha} n_b \right) \\ &= \frac{1}{\alpha} n^b n_a \nabla_b \alpha + n^b \alpha \frac{1}{\alpha} \nabla_a n_b + n^b n_b \alpha \nabla_a \frac{1}{\alpha}, \end{aligned} \quad (\text{A.27})$$

where we have used the fact that $\nabla_{[a} \nabla_{b]} t = 0$ to swap the order of the derivatives. We now

simplify this expression on using the relations $n^b \nabla_a n_b = 0$ and $n_b n^b = -1$ and write

$$\begin{aligned}
a_a &= \frac{1}{\alpha} n^b n_a \nabla_b \alpha + n^b \alpha \frac{1}{\alpha} \nabla_a n_b + n^b n_b \alpha \nabla_a \frac{1}{\alpha} = \frac{1}{\alpha} n^b n_a \nabla_b \alpha - \left(-\alpha \frac{1}{\alpha^2} \nabla_a \alpha \right) \\
&= \frac{1}{\alpha} (\nabla_a \alpha + n^b n_a \nabla_b \alpha) = \frac{1}{\alpha} (\delta_a^b \nabla_b \alpha + n^b n_a \nabla_b \alpha) = \frac{1}{\alpha} (\delta_a^b + n^b n_a) \nabla_b \alpha \\
&= \frac{1}{\alpha} \gamma_a^b \nabla_b \alpha = \frac{1}{\alpha} D_a \alpha \\
&= D_a \ln \alpha.
\end{aligned} \tag{A.28}$$

We see, therefore, that the acceleration is proportional to the gradient of the lapse.

We can now expand the RHS of Eqn. A.25

$$\begin{aligned}
K_{ab} &= -\gamma_a^c \gamma_b^d \nabla_c n_d = -(\delta_a^c + n_a n^c)(\delta_b^d + n_b n^d) \nabla_c n_d \\
&= -(\delta_a^c \delta_b^d \nabla_c n_d + \delta_a^c n_b n^d \nabla_c n_d + \delta_b^d n_a n^c \nabla_c n_d + n_a n^c n_b n^d \nabla_c n_d) \\
&= -(\nabla_a n_b + \underbrace{\delta_a^c n_b n^d \nabla_c n_d}_{=0} + n_a n^c \nabla_c n_b + n_a n^c n_b \underbrace{n^d \nabla_c n_d}_{=0}) \\
&= -(\nabla_a n_b + \underbrace{n_a n^c \nabla_c n_b}_{=a_b}) \\
&= -\nabla_a n_b - n_a a_b.
\end{aligned} \tag{A.29}$$

We can also express the extrinsic curvature as the Lie derivative of the spatial metric along n^a :

$$\begin{aligned}
\mathcal{L}_n \gamma_{ab} &= n^c \nabla_c \gamma_{ab} + \gamma_{ab} \nabla_b n^c + \gamma_{cb} \nabla_a n^c \\
&= n^c \nabla_c (g_{ab} + n_a n_b) + (g_{ac} + n_a n_c) \nabla_b n^c + (g_{cb} + n_b n_c) \nabla_a n^c \\
&= n^c \underbrace{[\nabla_c g_{ab} \nabla_c (n_a n_b)]}_{=0} + g_{ac} \nabla_b n^c + n_a \underbrace{n_c \nabla_b n^c}_{=0} + g_{cb} \nabla_a n^c + n_b \underbrace{n_c \nabla_a n^c}_{=0} \\
&= n^c n_b \nabla_c n_a + n^c n_a \nabla_c n_b + \nabla_b (g_a c n^c) + \nabla_a (g_c b n^c) \\
&= n_b \underbrace{n^c \nabla_c n_a}_{=a_a} + n_a \underbrace{n^c \nabla_c n_b}_{=a_b} + \nabla_b n_a + \nabla_a n_b \\
&= n_b a_a + n_a a_b + \nabla_b n_a + \nabla_a n_b = 2(n_{(a} a_{b)} + \nabla_{(a} n_{b)}).
\end{aligned} \tag{A.30}$$

Using Eqn. A.29 and the fact that the extrinsic curvature is symmetric then we can conclude

$$\mathcal{L}_n \gamma_{ab} = -2K_{ab}, \tag{A.31}$$

or, equivalently

$$K_{ab} = -\frac{1}{2}\mathcal{L}_n\gamma_{ab}. \quad (\text{A.32})$$

To sum up, we have three equivalent expressions for the extrinsic curvature

$$\begin{aligned} K_{ab} &= -\gamma_a^c \gamma_b^d \nabla_c n_d \\ &= -\nabla_a n_b - n_a a_b \\ &= -\frac{1}{2}\mathcal{L}_n\gamma_{ab}. \end{aligned}$$

The mean curvature is a scalar defined by the trace of the extrinsic curvature

$$K = g^{ab} K_{ab} = \gamma^{ab} K_{ab}. \quad (\text{A.33})$$

Taking the trace of Eqn. A.32 thus yields

$$K = \gamma^{ab} K_{ab} = -\frac{1}{2}\gamma^{ab}\mathcal{L}_n\gamma_{ab} = -\frac{1}{2\gamma}\mathcal{L}_n\gamma = -\frac{1}{2\sqrt{\gamma}}\mathcal{L}_n\sqrt{\gamma} = -\mathcal{L}_n \ln \sqrt{\gamma}. \quad (\text{A.34})$$

This expression can be given a geometrical interpretation. Because $\sqrt{\gamma}d^3x$ is the proper volume element in the spatial slice Σ , the negative of the mean curvature can be understood to measure the fractional change in the proper 3-volume along n^a .

A.1.3 The Equations of Gauss, Codazzi, and Ricci

In previous sections, we defined the spatial metric γ_{ab} and the extrinsic curvature K_{ab} . Of course, neither can be chosen arbitrarily. Instead, they must satisfy certain constraints which dictate how the 3D spatial slices fit into the larger, 4D spacetime M . To find these constraints, we need to understand the relation between the 3-dimensional Riemann tensor $R^a{}_{bcd}$ and the 4-dimensional Riemann tensor ${}^{(4)}R^a{}_{bcd}$.

First, we need to rewrite ${}^{(4)}R^a{}_{bcd}$ in terms of combinations of projections

$$\begin{aligned}
{}^{(4)}R_{abcd} &= \delta_a^p \delta_b^q \delta_c^r \delta_d^s {}^{(4)}R_{pqrs} \\
&= (\gamma_a^p - n^p n_a)(\gamma_b^q - n^q n_b)(\gamma_c^r - n^r n_c)(\gamma_d^s - n^s n_d) {}^{(4)}R_{pqrs} \\
&= (\gamma_a^p \gamma_b^q - \gamma_b^q n^p n_a - \gamma_a^p n^q n_b + n_a n_b n^p n^q)(\gamma_c^r - n^r n_c)(\gamma_d^s - n^s n_d) {}^{(4)}R_{pqrs} \\
&= \left(\gamma_a^p \gamma_b^q \gamma_c^r - \gamma_a^p \gamma_b^q n^r n_c - \gamma_a^p \gamma_c^r n^q n_b + \gamma_a^p n^q n_b n^r n_c - \gamma_b^q \gamma_c^r n^p n_a + \gamma_b^q n^p n_a n^r n_c \right. \\
&\quad \left. + \gamma_c^r n_a n_b n^p n^q - n^r n_c n_a n_b n^p n^q \right) (\gamma_d^s - n^s n_d) {}^{(4)}R_{pqrs} \\
&= \left[\gamma_a^p \gamma_b^q \gamma_c^r \gamma_d^s - \gamma_a^p \gamma_b^q \gamma_c^r n^s n_d - \gamma_a^p \gamma_b^q \gamma_d^s n^r n_c + \gamma_a^p \gamma_b^q n^r n_c n^s n_d - \gamma_a^p \gamma_c^r \gamma_d^s n^q n_b \right. \\
&\quad + \gamma_a^p \gamma_c^r n^q n_b n^s n_d + \gamma_a^p \gamma_d^s n^q n_b n^r n_c - \gamma_a^p n^q n_b n^r n_c n^s n_d - \gamma_b^q \gamma_c^r \gamma_d^s n^p n_a - \gamma_b^q \gamma_c^r n^p n_a n^s n_d \\
&\quad + \gamma_b^q \gamma_d^s n^p n_a n^r n_c - \gamma_b^q n^p n_a n^r n_c n^s n_d + \gamma_c^r \gamma_d^s n_a n_b n^p n^q - \gamma_c^r n_a n_b n^p n^q n^s n_d \\
&\quad \left. - \gamma_d^s n^r n_c n_a n_b n^p n^q + n_a n_b n_c n_d n^p n^q n^r n^s \right] {}^{(4)}R_{pqrs}. \tag{A.35}
\end{aligned}$$

There are a lot of terms in the above expression but we can simplify it. Using the symmetry properties of the Riemann tensor, we can eliminate terms like $n^p n^q {}^{(4)}R_{pqrs} = 0$ because $n^p n^q$ is symmetric under swapping p, q and ${}^{(4)}R_{pqrs}$ is anti-symmetric under this swapping.

Thus

$$\begin{aligned}
{}^{(4)}R_{abcd} &= \gamma_a^p \gamma_b^q \gamma_c^r \gamma_d^s {}^{(4)}R_{pqrs} - \gamma_a^p \gamma_b^q \gamma_c^r n^s n_d {}^{(4)}R_{pqrs} - \gamma_a^p \gamma_b^q \gamma_d^s n^r n_c {}^{(4)}R_{pqrs} + \underbrace{\gamma_a^p \gamma_b^q n^r n_c n^s n_d}_{=0} {}^{(4)}R_{pqrs} \\
&\quad - \gamma_a^p \gamma_c^r \gamma_d^s n^q n_b {}^{(4)}R_{pqrs} + \gamma_a^p \gamma_c^r n^q n_b n^s n_d {}^{(4)}R_{pqrs} + \gamma_a^p \gamma_d^s n^q n_b n^r n_c {}^{(4)}R_{pqrs} \\
&\quad - \underbrace{\gamma_a^p n^q n_b n^r n_c n^s n_d}_{=0} {}^{(4)}R_{pqrs} - \gamma_b^q \gamma_c^r \gamma_d^s n^p n_a {}^{(4)}R_{pqrs} - \gamma_b^q \gamma_c^r n^p n_a n^s n_d {}^{(4)}R_{pqrs} \\
&\quad + \gamma_b^q \gamma_d^s n^p n_a n^r n_c {}^{(4)}R_{pqrs} - \underbrace{\gamma_b^q n^p n_a n^r n_c n^s n_d}_{=0} {}^{(4)}R_{pqrs} + \underbrace{\gamma_c^r \gamma_d^s n_a n_b n^p n^q}_{=0} {}^{(4)}R_{pqrs} \\
&\quad - \underbrace{\gamma_c^r n_a n_b n^p n^q n^s n_d}_{(4)R_{pqrs}} - \underbrace{\gamma_d^s n^r n_c n_a n_b n^p n^q}_{=0} {}^{(4)}R_{pqrs} + \underbrace{n_a n_b n_c n_d n^p n^q n^r n^s}_{=0} {}^{(4)}R_{pqrs}. \tag{A.36}
\end{aligned}$$

A careful examination of this last expression shows that we have three types of projections: completely spatial projections, projections with one index projected in the normal direction,

and projections with two indices projected in the normal direction. All other projections vanish. On collecting the terms by these types of projection and rearranging dummy indices, we get

$$\begin{aligned}
{}^{(4)}R_{abcd} &= \gamma_a^p \gamma_b^q \gamma_c^r \gamma_d^s {}^{(4)}R_{pqrs} \\
&\quad - \gamma_a^p \gamma_b^q \gamma_c^r n_d^s {}^{(4)}R_{pqrs} - \underbrace{\gamma_a^p \gamma_b^q \gamma_d^r n_c^s}_{=\gamma_a^p \gamma_b^q \gamma_d^r n_c^s} {}^{(4)}R_{pqrs} - \underbrace{\gamma_a^p \gamma_c^r \gamma_d^s n_b^q}_{=\gamma_a^p \gamma_c^r \gamma_d^s n_b^q} {}^{(4)}R_{pqrs} \\
&\quad - \underbrace{\gamma_b^q \gamma_c^r \gamma_d^s n_a^p}_{=\gamma_b^q \gamma_c^r \gamma_d^s n_a^p} {}^{(4)}R_{pqrs} + \gamma_a^p \gamma_c^q n_b^r n_d^s {}^{(4)}R_{pqrs} + \underbrace{\gamma_a^p \gamma_d^q n_b^r n_c^s}_{=\gamma_a^p \gamma_d^q n_b^r n_c^s} {}^{(4)}R_{pqrs} \\
&\quad - \underbrace{\gamma_b^q \gamma_c^r n_a^p n_d^s}_{=\gamma_b^q \gamma_c^r n_a^p n_d^s} {}^{(4)}R_{pqrs} + \underbrace{\gamma_b^q \gamma_d^s n_a^p n_r^c}_{=\gamma_b^q \gamma_d^s n_a^p n_r^c} {}^{(4)}R_{pqrs} \\
&= \gamma_a^p \gamma_b^q \gamma_c^r \gamma_d^s {}^{(4)}R_{pqrs} \\
&\quad - \gamma_a^p \gamma_b^q \gamma_c^r n_d n^s {}^{(4)}R_{pqrs} - \gamma_a^p \gamma_b^q \gamma_d^r n_c n^s \underbrace{{}^{(4)}R_{pqsr}}_{=-{}^{(4)}R_{pqrs}} - \gamma_c^r \gamma_d^q \gamma_a^r n_b n^s \underbrace{{}^{(4)}R_{rspq}}_{=-{}^{(4)}R_{pqrs}} \\
&\quad - \gamma_c^r \gamma_d^q \gamma_b^r n_a n^s \underbrace{{}^{(4)}R_{srpq}}_{=-{}^{(4)}R_{pqrs}} + \gamma_a^p \gamma_c^q n_b n_d n^q n^s {}^{(4)}R_{pqrs} \\
&\quad + \gamma_a^p \gamma_d^r n_b n_c n^q n^s \underbrace{{}^{(4)}R_{pqsr}}_{=-{}^{(4)}R_{pqrs}} - \gamma_b^p \gamma_c^q n_a n_d n^q n^s \underbrace{{}^{(4)}R_{qprs}}_{=-{}^{(4)}R_{pqrs}} + \gamma_b^p \gamma_d^r n_a n_c n^q n^s \underbrace{{}^{(4)}R_{qpsr}}_{=-{}^{(4)}R_{pqrs}} \\
&= \gamma_a^p \gamma_b^q \gamma_c^r \gamma_d^s {}^{(4)}R_{pqrs} \\
&\quad - \gamma_a^p \gamma_b^q (\gamma_c n_d - \gamma_d n_c) n^s {}^{(4)}R_{pqrs} - \gamma_c^r \gamma_d^q (\gamma_a n_b - \gamma_b n_a) n^s {}^{(4)}R_{pqrs} \\
&\quad + \gamma_a^p (\gamma_c n_d - \gamma_d n_c) n_b n^q n^s {}^{(4)}R_{pqrs} - \gamma_b^p (\gamma_c n_d - \gamma_d n_c) n_a n^q n^s {}^{(4)}R_{pqrs}. \quad (\text{A.37})
\end{aligned}$$

Thus, the 4-dimensional Riemann tensor ${}^{(4)}R_{abcd}$ can be written as

$$\begin{aligned}
{}^{(4)}R_{abcd} &= \gamma_a^p \gamma_b^q \gamma_c^r \gamma_d^s {}^{(4)}R_{pqrs} - 2\gamma_a^p \gamma_b^q 2\gamma_{[c}^r n_{d]} n^s {}^{(4)}R_{pqrs} - 2\gamma_c^p \gamma_d^q \gamma_{[a}^r n_{b]} n^s {}^{(4)}R_{pqrs} \\
&\quad + 2\gamma_a^p \gamma_{[c}^r n_{d]} n_b n^q n^s {}^{(4)}R_{pqrs} - 2\gamma_b^p \gamma_{[c}^r n_{d]} n_a n^q n^s {}^{(4)}R_{pqrs}. \quad (\text{A.38})
\end{aligned}$$

Using the above projections, we can establish the relation between ${}^{(4)}R^a{}_{bcd}$ and $R^a{}_{bcd}$ which gives Gauss, Codazzi, and Ricci equations. Recall that the Riemann tensor can be

defined in terms of the noncommutivity of second covariant derivatives of a vector. So, we need to be able to relate 4-dimensional and 3-dimensional covariant derivatives.

Consider the first derivative of a spatial vector V^b . Using definitions which we made before, we can write

$$\begin{aligned} D_a V^b &= \gamma_a^p \gamma_q^b \nabla_p V^q = \gamma_a^p (\delta_q^b + n^b n_q) \nabla_p V^q \\ &= \gamma_a^p \delta_q^b \nabla_p V^q + \gamma_a^p n^b n_q \nabla_p V^q = \gamma_a^p \nabla_p V^b + \gamma_a^p n^b (\nabla_p (V^q n_q) - V^q \nabla_p n_q). \end{aligned} \quad (\text{A.39})$$

As V^q is assumed to be purely spatial then $n_q V^q = 0$, and we can write

$$\begin{aligned} D_a V^b &= \gamma_a^p \nabla_p V^b - \gamma_a^p n^b V^q \nabla_p n_q = \gamma_a^p \nabla_p V^b - n^b V^e \underbrace{\gamma_e^a \gamma_e^p \nabla_p n_q}_{=-K_{ae}} \\ &= \gamma_a^p \nabla_p V^b - n_b V^e K_{ae}. \end{aligned} \quad (\text{A.40})$$

Writing out the second derivative, we get

$$\begin{aligned} D_a D_b V^c &= \gamma_a^p \gamma_b^q \gamma_r^c \nabla_p \nabla_q V^r = \gamma_a^p \gamma_b^q \gamma_r^c \nabla_p (\gamma_e^d \gamma_e^r \nabla_d V^e) \\ &= \gamma_a^p \gamma_b^q \gamma_r^c \gamma_e^d \gamma_e^r \nabla_p \nabla_d V^e + \gamma_a^p \gamma_b^q \gamma_r^c \gamma_e^d \nabla_p \gamma_e^r \nabla_d V^e + \gamma_a^p \gamma_b^q \gamma_r^c \gamma_e^d \nabla_p \gamma_e^r \nabla_d V^e \\ &= \gamma_a^p \gamma_b^q \gamma_e^c \nabla_p \nabla_d V^e + \gamma_a^p \gamma_b^q \gamma_e^c \nabla_p (\delta_q^d + n^d n_q) \nabla_d V^e + \gamma_a^p \gamma_b^q \gamma_e^c \nabla_p (\delta_e^r + n^r n_e) \nabla_d V^e, \end{aligned} \quad (\text{A.41})$$

as δ_b^a is just a numerical tensor so the derivative of δ_b^a is zero. Recall that $\nabla_a (n_b n^c)$ can be expressed as

$$\begin{aligned} \nabla_a (n_b n^c) &= n_b \nabla_a n^c + n^c \nabla_a n_b \\ &= -n_b (K_a^c + n_a a^c) - n^c (K_{ab} + n_a a_b) \end{aligned} \quad (\text{A.42})$$

where we have used the definition of the extrinsic curvature in Eqn. A.29. The second derivative can now be expressed as

$$\begin{aligned} D_a D_b V^c &= \gamma_a^p \gamma_b^q \gamma_e^c \nabla_p \nabla_d V^e - \gamma_a^p \gamma_b^q \gamma_e^c [n^d (K_{pq} + n_p a_q) + n_q (K_p^d + n_p a^d)] \nabla_d V^e \\ &\quad - \gamma_a^p \gamma_b^q \gamma_e^c [n^r (K_{pe} + n_p a_e) + n_e (K_p^r + n_p a^r)] \nabla_d V^e. \end{aligned} \quad (\text{A.43})$$

Using the fact that $n_a \gamma^{ab} = 0$ we can simplify further

$$\begin{aligned}
D_a D_b V^c &= \gamma_a^p \gamma_b^d \gamma_e^c \nabla_p \nabla_d V^e - [\gamma_a^p \gamma_b^d \gamma_e^c n^d K_{pq} + n^d \underbrace{\gamma_b^d \gamma_e^c \gamma_a^p n_p}_{=0} a_q + \gamma_a^p \gamma_e^c \underbrace{\gamma_b^d n_q}_{=0} (K_p^d + n_p a^d)] \nabla_d V^e \\
&\quad - [\gamma_a^p \gamma_b^d \underbrace{\gamma_r n^r}_{=0} (K_{pe} + n_p a_e) + \gamma_a^p \gamma_b^d \gamma_r n_e K_p^r + \gamma_b^d \gamma_r \underbrace{\gamma_a^p n_p a^r}_{=0}] \nabla_d V^e \\
&= \gamma_a^p \gamma_b^d \gamma_e^c \nabla_p \nabla_d V^e - \gamma_e^c n^d K_{ab} \nabla_d V^e - \gamma_b^d K_a^c \underbrace{n_e \nabla_d V^e}_{=-V^e \nabla_d n_e} \\
&= \gamma_a^p \gamma_b^d \gamma_e^c \nabla_p \nabla_d V^e - K_{ab} \gamma_e^c n^d \nabla_d V^e + \gamma_b^d K_a^c V^e \nabla_d (\gamma_e^s n_s) \\
&= \gamma_a^p \gamma_b^d \gamma_e^c \nabla_p \nabla_d V^e - K_{ab} \gamma_e^c n^d \nabla_d V^e + V^e K_a^c \underbrace{\gamma_b^d \gamma_e^s \nabla_d n_s}_{=-K_{be}}. \tag{A.44}
\end{aligned}$$

On rearranging the indices (for convenience) the second derivative can now be written as

$$D_a D_b V^c = \gamma_a^p \gamma_b^d \gamma_r^c \nabla_p \nabla_q V^r - K_{ab} \gamma_r^c n^p \nabla_p V^r - K_a^c K_{bp} V^p. \tag{A.45}$$

The definition of the 3-dimensional Riemann tensor is

$$R^{dc}{}_{ba} V_d = 2D_{[a} D_{b]} V^c. \tag{A.46}$$

Inserting Eqn. A.45 into the RHS of with the anti-symmetrization we get

$$R^{dc}{}_{ba} V_d = 2\gamma_a^p \gamma_b^d \gamma_r^c \nabla_{[p} \nabla_{q]} V^r - 2K_{[ab]} \cdot \gamma_r^c n^p \nabla_p V^r - 2K_{[a}^c K_{b]p} V^p \tag{A.47}$$

Because K_{ab} is symmetric we have $K_{[ab]} = 0$ and because $\nabla_{[p} \nabla_{q]} V^r$ gives the 4-dimensional Riemann tensor via

$$2\nabla_{[p} \nabla_{q]} V^r = {}^{(4)}R^{dr}{}_{qp} V_d. \tag{A.48}$$

we can combine these results to get

$$R^{dc}{}_{ba} V_d = \gamma_a^p \gamma_b^d \gamma_r^c {}^{(4)}R^{dr}{}_{qp} V_d - 2K_{[a}^c K_{b]p} V^p, \tag{A.49}$$

$$\begin{aligned}
R_{dcba} V^d &= \gamma_a^p \gamma_b^d \gamma_c^r {}^{(4)}R_{drqp} V^d - 2K_{c[a} K_{b]d} V^d \\
&= \gamma_a^p \gamma_b^d \gamma_c^r \gamma_d^s {}^{(4)}R_{srqp} V^d - 2K_{c[a} K_{b]d} V^d. \tag{A.50}
\end{aligned}$$

This final relation must hold for any arbitrary spatial vector so we can conclude

$$R_{abcd} + K_{ab}K_{bd} - K_{ad}K_{cb} = \gamma_a^p \gamma_b^q \gamma_c^r \gamma_d^s {}^{(4)}R_{pqrs}. \quad (\text{A.51})$$

This final expression, Eqn. A.51, is called *Gauss' equation*. It relates the full spatial projection of ${}^{(4)}R^a{}_{bcd}$ to the 3-dimensional Riemann tensor, $R^a{}_{bcd}$, and terms quadratic in the extrinsic curvature.

Next, consider a spatial derivative of the extrinsic curvature

$$\begin{aligned} D_a K_{bc} &= \gamma_a^p \gamma_b^q \gamma_c^r \nabla_p K_{qr} = -\gamma_a^p \gamma_b^q \gamma_c^r \nabla_p (\nabla_q n_r + n_q a_r) \\ &= -(\gamma_a^p \gamma_b^q \gamma_c^r \nabla_p \nabla_q n_r + \underbrace{\gamma_a^p \gamma_c^r \gamma_b^q n_q}_{=0} \nabla_p a_r + \gamma_c^r a_r \underbrace{\gamma_a^p \gamma_b^q \nabla_p n_q}_{=-K_{ab}}) \\ &= -\gamma_a^p \gamma_b^q \gamma_c^r \nabla_p \nabla_q n_r + a_c K_{ab}. \end{aligned} \quad (\text{A.52})$$

On antisymmetrization:

$$\begin{aligned} D_{[a} K_{b]c} &= -\gamma_a^p \gamma_b^q \gamma_c^r \underbrace{\nabla_{[p} \nabla_{q]} n_r}_{=\frac{1}{2} {}^{(4)}R_{srqp} n^s} + a_c \underbrace{K_{[ab]}}_{=0} \\ &= -\frac{1}{2} \gamma_a^p \gamma_b^q \gamma_c^r n^s {}^{(4)}R_{pqrs}. \end{aligned} \quad (\text{A.53})$$

this can be rewritten as

$$D_b K_{ac} - D_a K_{bc} = \gamma_a^p \gamma_b^q \gamma_c^r n^s {}^{(4)}R_{pqrs}. \quad (\text{A.54})$$

The Eqn. A.54 is called the *Codazzi equation*. This relates the projection of Riemann with one index projected in the normal direction to the derivatives of the extrinsic curvature. Considering now the remaining projection namely with two indices projected in the normal direction, we look at a quantity involving a time derivative of K_{ab} . We compute the Lie derivative of the extrinsic curvature along n^a :

$$\begin{aligned} \mathcal{L}_n K_{ab} &= n^c \nabla_c K_{ab} + K_{ac} \nabla_b n^c + K_{cb} \nabla_a n^c \\ &= -n^c \nabla_c (\nabla_a n_b + n_a a_b) - K_{ac} (K_b^c + n_b a^c) - K_{cb} (K_a^c + n_a a^c) \\ &= -n^c \nabla_c \nabla_a n_b - n^c a_b \nabla_c n_a - n^c n_a \nabla_c a_b - (K_{ac} K_b^c + K_{cb} K_a^c) - (K_{ab} n_b + K_{cb} n_a) a^c. \end{aligned} \quad (\text{A.55})$$

Using the definition of the Riemann tensor we have

$${}^{(4)}R_{dbac}n^d = 2\nabla_{[c}\nabla_{a]}n_b = \nabla_c\nabla_a n_b - \nabla_a\nabla_c n_b, \quad (\text{A.56})$$

$$\rightarrow \nabla_c\nabla_a n_b = {}^{(4)}R_{dbac}n^d + \nabla_a\nabla_c n_b. \quad (\text{A.57})$$

Substituting Eqn. A.57 into Eqn. A.55 then gives

$$\begin{aligned} \mathcal{L}_n K_{ab} &= -{}^{(4)}R_{dbac}n^d n^c - n^c \nabla_a \nabla_c n_b - n^c a_b \nabla_c n_a - n^c n_a \nabla_c a_b - (K_{ac}K_b^c + K_{cb}K_a^c) \\ &\quad - (K_{ab}n_b + K_{cb}n_a)a^c. \end{aligned} \quad (\text{A.58})$$

Using the definition of $a_b = n^c \nabla_c n_b$ then we can find the relation

$$\begin{aligned} \nabla_a a_b &= \nabla_a (n^c \nabla_c n_b) = n^c \nabla_a \nabla_c n_b + \nabla_a n^c \nabla_c n_b, \quad (\text{A.59}) \\ \rightarrow n^c \nabla_a \nabla_c n_b &= \nabla_a a_b - \nabla_a n^c \nabla_c n_b \\ &= \nabla_a a_b - (K_a^c + n_a a^c)(K_{cb} + n_c a_b) \\ &= \nabla_a a_b - K_a^c K_{cb} - \underbrace{K_a^c n_c a_b}_{=0} - n_a a^c K_{cb} - n_a \underbrace{n_c a^c}_{=n_c n^d \nabla_d n^c = 0} a_b \\ &= \nabla_a a_b - K_a^c K_{cb} - n_a a^c K_{cb}. \end{aligned} \quad (\text{A.60})$$

On inserting Eqn. A.60 into Eqn. A.58 and simplifying, we get

$$\begin{aligned} \mathcal{L}_n K_{ab} &= -{}^{(4)}R_{dbac}n^d n^c - (\nabla_a a_b - K_a^c K_{cb} - n_a a^c K_{cb}) - n^c a_b \nabla_c n_a - n^c n_a \nabla_c a_b \\ &\quad - (K_{ac}K_b^c + K_{cb}K_a^c) - (K_{ab}n_b + K_{cb}n_a)a^c \\ &= -{}^{(4)}R_{dbac}n^d n^c - \nabla_a a_b - a_b \underbrace{n^c \nabla_c n_a}_{=a_a} - n^c n_a \nabla_c a_b - K_b^c K_{ac} - K_{ca}n_b a^c \\ &= -n^c n^d {}^{(4)}R_{dbac} - \nabla_a a_b - n^c n_a \nabla_c a_b - a_a a_b - K_b^c K_{ac} - K_{ca}n_b a^c. \end{aligned} \quad (\text{A.61})$$

Note that Eqn. A.61 is purely spatial. To see that, we project the equation into the normal direction:

$$\begin{aligned} n^a \mathcal{L}_n K_{ab} &= -\underbrace{n^a n^c n^d}_{=0} {}^{(4)}R_{dbac} - n^a \nabla_a a_b - n^c \underbrace{n^a n_a}_{=-1} \nabla_c a_b - \underbrace{n^a a_a}_{=0} a_b - K_b^c \underbrace{n^a K_{ac}}_{=0} - \underbrace{n^a K_{ca} n_b a^c}_{=0} \\ &= -n^a \nabla_a a_b + n^c \nabla_c a_b = 0. \end{aligned} \quad (\text{A.62})$$

where the last line came from swapping dummy indices (c to a). Since $\mathcal{L}_n K_{ab}$ is purely spatial, projecting the two free indices in Eqn. A.61 leaves the LHS unchanged

$$\gamma_a^i \gamma_b^j \mathcal{L}_n K_{qr} = \mathcal{L}_n (\gamma_a^i \gamma_b^j K_{qr}) = \mathcal{L}_n K_{ab}. \quad (\text{A.63})$$

The RHS, then, becomes

$$\begin{aligned} \mathcal{L}_n K_{ab} &= -\gamma_a^i \gamma_b^j n^c n^d {}^{(4)}R_{drqc} - \gamma_a^i \gamma_b^j \nabla_q a_r - \underbrace{\gamma_b^j n^c \gamma_a^i n_q}_{=0} \nabla_c a_r - \gamma_a^i \gamma_b^j a_q a_r - \gamma_a^i \gamma_b^j K_r^c K_{qc} - K_{cq} \underbrace{\gamma_a^i \gamma_b^j n_r a^c}_{=0} \\ &= -n^d n^c \gamma_a^i \gamma_b^j {}^{(4)}R_{drqc} - D_a a_b - a_a a_b - K_b^c K_{ac}. \end{aligned} \quad (\text{A.64})$$

We know that $a_a = D_a \ln \alpha$ so

$$\begin{aligned} D_a a_b &= D_a D_b \ln \alpha = D_a \left(\frac{1}{\alpha} D_b \alpha \right) \\ &= \frac{1}{\alpha} D_a D_b \alpha - \frac{1}{\alpha^2} D_a \alpha D_b \alpha = \frac{1}{\alpha} D_a D_b \alpha - \underbrace{\left(\frac{1}{\alpha} D_a \alpha \right)}_{=a_a} \underbrace{\left(\frac{1}{\alpha} D_b \alpha \right)}_{=a_b} \\ &= -a_a a_b + \frac{1}{\alpha} D_a D_b \alpha. \end{aligned} \quad (\text{A.65})$$

Therefore, we conclude that

$$\mathcal{L}_n K_{ab} = n^d n^c \gamma_a^i \gamma_b^j {}^{(4)}R_{drqc} - \frac{1}{\alpha} D_a D_b \alpha - K_b^c K_{ac}. \quad (\text{A.66})$$

Note that this final expression in Eqn. A.66 is referred to as *Ricci's equation*. It gives the twice projected Riemann tensor with two indices projected in the normal direction.

A.1.4 The Constraints and Evolution Equations

We are now ready to rewrite the Einstein equations in a 3+1 form. We just need to use the equations of Gauss, Codazzi, and Ricci to eliminate the 4-dimensional Riemann tensor using the Einstein equations

$$G_{ab} = {}^{(4)}R_{ab} - \frac{1}{2} {}^{(4)}R g_{ab} = 8\pi T_{ab}. \quad (\text{A.67})$$

In this section, we will complete the derivation of the constraint equations from the Gauss (Eqn. A.51) and the Codazzi's equations (Eqn. A.54). We will also then derive the evolution equations from Eqn. A.32 and Ricci's equation (Eqn. A.66).

Starting with Gauss' equation (Eqn. A.51)

$$R^a{}_{bcd} + K_c^a K_{bd} - K_d^a K_{cb} = \gamma^{pa} \gamma_b^q \gamma_c^r \gamma_d^s {}^{(4)}R_{pqrs}. \quad (\text{A.68})$$

we contract with γ_a^c

$$R^a{}_{bad} + K_a^a K_{bd} - K_d^c K_{cb} = \gamma^{pc} \gamma_b^q \gamma_c^r \gamma_d^s {}^{(4)}R_{pqrs}. \quad (\text{A.69})$$

Denoting $K_a^a = K$, for the trace of extrinsic curvature, and using the 3-dimensional Ricci tensor we have

$$\gamma^{pr} \gamma_b^q \gamma_d^s {}^{(4)}R_{pqrs} = R_{bd} + K K_{bd} - K_d^c K_{cb}. \quad (\text{A.70})$$

On rearranging indices

$$\gamma^{pr} \gamma_b^q \gamma^{sd} {}^{(4)}R_{pqrs} = R^b{}_d + K K_d^b - K^{cd} K_{cb}, \quad (\text{A.71})$$

and on contracting again with γ_d^b , we get

$$\begin{aligned} \gamma^{pr} \gamma_d^b \gamma_b^q \gamma^{sd} {}^{(4)}R_{pqrs} &= \gamma_d^b R^b{}_d + \gamma_d^b K K_d^b - \gamma_d^b K^{cd} K_{cb}, \\ \gamma^{pr} \gamma^{qs} {}^{(4)}R_{pqrs} &= R + K^2 - K^{cb} K_{cb}. \end{aligned} \quad (\text{A.72})$$

The LHS of the above equation can be written as

$$\begin{aligned} (g^{pr} + n^p n^r)(g^{qs} + n^q n^s) {}^{(4)}R_{pqrs} &= g^{pr} g^{qs} {}^{(4)}R_{pqrs} + g^{pr} n^q n^s {}^{(4)}R_{pqrs} + g^{qs} n^p n^r {}^{(4)}R_{pqrs} \\ &\quad + \underbrace{n^p n^q n^r n^s {}^{(4)}R_{pqrs}}_{=0} \\ &= {}^{(4)}R + n^q n^s {}^{(4)}R_{pq} + n^p n^r {}^{(4)}R_{pr} \\ &= {}^{(4)}R + 2n^p n^r {}^{(4)}R_{pr}, \end{aligned} \quad (\text{A.73})$$

where the last line comes from swapping dummy indices (q, s to p, r). So, we have

$${}^{(4)}R + 2n^p n^r {}^{(4)}R_{pr} = R + K^2 - K^{ab} K_{ab}. \quad (\text{A.74})$$

where again, we swap the dummy index (c to a). Contracting the Einstein tensor with $n^p n^r$ gives

$$\begin{aligned}
2n^p n^r G_{pr} &= 2n^p n^r \left({}^{(4)}R_{pr} - \frac{1}{2} g_{pr} {}^{(4)}R \right) \\
&= 2n^p n^r {}^{(4)}R_{pr} - n^p n^r (\gamma_{pr} + n_p n_r) {}^{(4)}R \\
&= 2n^p n^r {}^{(4)}R_{pr} - \underbrace{n^p n^r \gamma_{pr}}_{=0} {}^{(4)}R - \underbrace{n_p n^p}_{=-1} \underbrace{n^r n_r}_{=-1} {}^{(4)}R \\
&= {}^{(4)}R + 2n^p n^r {}^{(4)}R_{pr}. \tag{A.75}
\end{aligned}$$

Combining this with the above, we get

$$2n^p n^r G_{pr} = R + K^2 - K^{ab} K_{ab}. \tag{A.76}$$

From the Einstein equations, we have

$$2n^p n^r G_{pr} = 16\pi n^p n^r T_{pr}. \tag{A.77}$$

We now define the energy density ρ to be the total energy density as measured by a normal observer n^a

$$\rho \equiv n_a n_b T^{ab} = n^a n^b T_{ab}. \tag{A.78}$$

Finally, we get

$$R + K^2 - K^{ab} K_{ab} = 16\pi\rho. \tag{A.79}$$

The Eqn. A.79 is called the *Hamiltonian constraint*

Proceeding analogously with the Codazzi equation (Eqn. A.54), we have

$$D_b K_a^c - D_a K_b^c = \gamma_a^p \gamma_b^q \gamma^{rc} n^s {}^{(4)}R_{pqrs}. \tag{A.80}$$

Contracting this with γ_c^b gives

$$\begin{aligned}
D_b K_a^b - D_a K_b^b &= \gamma_a^p \gamma_b^q \underbrace{\gamma_c^b \gamma^{rc}}_{=\gamma^{rb}} n^s {}^{(4)}R_{pqrs}, \\
D_b K_a^b - D_a K &= \gamma_a^p \underbrace{\gamma_b^q \gamma^{rb}}_{\gamma^{qr}} n^s {}^{(4)}R_{pqrs} \\
&= \gamma_a^p (g^{qr} + n^q n^r) n^s {}^{(4)}R_{pqrs} \\
&= -\gamma_a^p n^s g^{qr} {}^{(4)}R_{qprs} + \underbrace{\gamma_a^p n^q n^r n^s {}^{(4)}R_{pqrs}}_{=0} \\
&= -\gamma_a^p n^s {}^{(4)}R_{ps}.
\end{aligned} \tag{A.81}$$

Likewise, contracting the Einstein tensor with n^s and γ_a^p yields

$$\begin{aligned}
\gamma_a^p n^s G_{ps} &= \gamma_a^p n^s \left({}^{(4)}R_{ps} - \frac{1}{2} g_{ps} {}^{(4)}R \right) \\
&= \gamma_a^p n^s {}^{(4)}R_{ps} - \frac{1}{2} \underbrace{\gamma_a^p g_{ps} n^s}_{=\gamma_a n^s=0} {}^{(4)}R \\
&= \gamma_a^p n^s {}^{(4)}R_{ps}.
\end{aligned} \tag{A.82}$$

Putting these together, we have

$$D_b K_a^b - D_a K = -\gamma_a^p n^s G_{ps}. \tag{A.83}$$

From the Einstein equations again we have

$$\gamma_a^p n^s G_{ps} = 8\pi \gamma_a^p n^s T_{ps}. \tag{A.84}$$

We now define S_a to be the momentum density as measured by a normal observer n^a

$$S_a \equiv -\gamma_a^b n^c T_{bc}, \tag{A.85}$$

and with that, we can finally write

$$D_b K_a^b - D_a K = 8\pi S_a. \tag{A.86}$$

The Eqn. A.86 is called the *momentum constraint*.

We also need to derive the evolution equations that evolve the data (γ_{ab}, K_{ab}) forward in time. These can be found from Eqn. A.32, which can be considered as the definition of the extrinsic curvature, and Ricci's equation (Eqn. A.66). These equations involve the Lie derivative along n^a . However, the \mathcal{L}_n is not a natural time derivative. This is because the normal, n^a , is not dual to the surface 1-form Ω_a :

$$n^a \Omega_a = -\alpha g^{ab} \nabla_a t \nabla_b t = \frac{1}{\alpha}. \quad (\text{A.87})$$

As an alternative, consider instead the new vector

$$t^a = \alpha n^a + \beta^a \quad (\text{A.88})$$

which is, in fact, dual to Ω_a for any spatial vector β^a

$$t^a \Omega_a = \underbrace{\alpha n^a \Omega_a}_{=1/\alpha} + \underbrace{\beta^a \Omega_a}_{=0} = 1. \quad (\text{A.89})$$

We will refer to the vector β^a as the *shift vector*. Geometrically, we will think of the vector t^a as connecting points with the same spatial coordinate on neighboring time slices. The shift vector β^a will then measure the amount by which the spatial coordinates are shifted within a slice with respect to the normal vector. As we discussed before, the lapse function α measures how much proper time elapses between neighboring time slices along the normal vector. Thus, the lapse and shift determine how the coordinates evolve in time. Note that the choice of α and β^a is quite arbitrary and there are many discussions of choices for lapse and shift in literatures. Recall that β^a is spatial and so $n^a \beta_a = 0$, hence only three of its components may be freely specified. Therefore, we have a total of four components or functions that can be freely chosen. These gauge functions, α and β^a , are completely arbitrarily and embody the four-fold coordinate degrees of freedom.

Consider now the Lie derivative of K_{ab} along t^a

$$\mathcal{L}_t K_{ab} = \mathcal{L}_{\alpha n + \beta} K_{ab} = \alpha \mathcal{L}_n K_{ab} + \mathcal{L}_\beta K_{ab}, \quad (\text{A.90})$$

which follows from the definition of the Lie derivative. Inserting this into Ricci's equation to eliminate $\mathcal{L}_n K_{ab}$ gives

$$\mathcal{L}_n K_{ab} = n^d n^c \gamma_a^d \gamma_b^r ({}^{(4)}R_{drcq}) - \frac{1}{\alpha} D_a D_b \alpha - K_b^c K_{ac}. \quad (\text{A.91})$$

Rewriting the first term of the RHS provides

$$\begin{aligned} n^d n^c \gamma_a^d \gamma_b^r ({}^{(4)}R_{drcq}) &= (\gamma^{dc} - g^{dc}) \gamma_a^d \gamma_b^r ({}^{(4)}R_{drcq}) \\ &= \gamma^{dc} \gamma_a^d \gamma_b^r ({}^{(4)}R_{drcq}) - \gamma_a^d \gamma_b^r ({}^{(4)}R_{rq}). \end{aligned} \quad (\text{A.92})$$

Using the Hamiltonian constraint (Eqn. 70), we have the relation

$$\gamma^{cd} \gamma_a^d \gamma_b^r ({}^{(4)}R_{drcq}) = R_{ab} + K K_{ab} - K_b^c K_{ac}. \quad (\text{A.93})$$

Contracting the Einstein equations with g^{rq} gives

$$\begin{aligned} g^{rq} \left(({}^{(4)}R_{rq}) - \frac{1}{2} g_{rq} ({}^{(4)}R) \right) &= g^{rq} ({}^{(4)}R_{rq}) - \frac{1}{2} \underbrace{g^{rq} g_{rq}}_{=4} ({}^{(4)}R) = - ({}^{(4)}R) \\ &= 8\pi g^{rq} T_{rq} = 8\pi T, \\ &\rightarrow ({}^{(4)}R) = -8\pi T. \end{aligned} \quad (\text{A.94})$$

Again, invoking the Einstein equations, we get

$$\begin{aligned} ({}^{(4)}R_{rq}) - \frac{1}{2} g_{rq} ({}^{(4)}R) &= 8\pi T_{rq}, \\ \rightarrow ({}^{(4)}R_{rq}) &= \frac{1}{2} g_{rq} ({}^{(4)}R) + 8\pi T_{rq} = 8\pi \left(T_{rq} - \frac{1}{2} g_{rq} T \right). \end{aligned} \quad (\text{A.95})$$

Using these results, Eqn. A.92 can be expressed as

$$\begin{aligned} n^d n^c \gamma_a^d \gamma_b^r ({}^{(4)}R_{drcq}) &= R_{ab} + K K_{ab} - K_b^c K_{ac} - 8\pi \gamma_a^d \gamma_b^r \left(T_{rq} - \frac{1}{2} g_{rq} T \right) \\ &= R_{ab} + K K_{ab} - K_b^c K_{ac} - 8\pi \left(\gamma_a^d \gamma_b^r T_{rq} - \frac{1}{2} \gamma_a^d \gamma_b^r g_{rq} g^{ef} T_{ef} \right) \\ &= R_{ab} + K K_{ab} - K_b^c K_{ac} - 8\pi \left(\gamma_a^d \gamma_b^r T_{rq} - \frac{1}{2} \gamma_{ab} (\gamma^{ef} - n^e n^f) T_{ef} \right). \end{aligned} \quad (\text{A.96})$$

We now define the spatial stress and its trace by

$$S_{ab} \equiv \gamma_a^c \gamma_b^d T_{cd}, \quad S \equiv S_a^a = \gamma^{ab} T_{ab}. \quad (\text{A.97})$$

Thus, we obtain

$$n^d n^c \gamma_a^d \gamma_b^c R_{drcq} = R_{ab} + KK_{ab} - K_b^c K_{ac} - 8\pi \left(S_{ab} - \frac{1}{2} \gamma_{ab} (S - \rho) \right), \quad (\text{A.98})$$

so that Ricci's equation can be rewritten as

$$\begin{aligned} \mathcal{L}_n K_{ab} &= R_{ab} + KK_{ab} - K_b^c K_{ac} - 8\pi \left(S_{ab} - \frac{1}{2} \gamma_{ab} (S - \rho) \right) - \frac{1}{\alpha} D_a D_b \alpha - K_b^c K_{ac} \\ &= R_{ab} + KK_{ab} - 2K_b^c K_{ac} - 8\pi \left(S_{ab} - \frac{1}{2} \gamma_{ab} (S - \rho) \right) - \frac{1}{\alpha} D_a D_b \alpha. \end{aligned} \quad (\text{A.99})$$

In terms of the Lie derivative along t^a , we rewrite

$$\frac{1}{\alpha} (\mathcal{L}_t K_{ab} - \mathcal{L}_\beta K_{ab}) = R_{ab} + KK_{ab} - 2K_b^c K_{ac} - 8\pi \left(S_{ab} - \frac{1}{2} \gamma_{ab} (S - \rho) \right) - \frac{1}{\alpha} D_a D_b \alpha. \quad (\text{A.100})$$

Finally, we have the first evolution equation

$$\mathcal{L}_t K_{ab} = \mathcal{L}_\beta K_{ab} - D_a D_b \alpha + \alpha (R_{ab} + KK_{ab} - 2K_b^c K_{ac}) - 8\pi \alpha \left(S_{ab} - \frac{1}{2} \gamma_{ab} (S - \rho) \right). \quad (\text{A.101})$$

Eqn. A.101 is the evolution equation for the extrinsic curvature. Note that all the differential operators together with the Ricci tensor R_{ab} are built out of the spatial metric γ_{ab} and not the full 4-dimensional metric g_{ab} .

Now we consider Eqn. A.32

$$\begin{aligned} K_{ab} &= -\frac{1}{2} \mathcal{L}_n \gamma_{ab} = -\frac{1}{2\alpha} (\mathcal{L}_t \gamma_{ab} - \mathcal{L}_\beta \gamma_{ab}), \\ \rightarrow \mathcal{L}_t \gamma_{ab} &= \mathcal{L}_\beta \gamma_{ab} - 2\alpha K_{ab}. \end{aligned} \quad (\text{A.102})$$

and Eqn. A.102 has become the evolution equation for the spatial metric. In summary, we have four constraint equations (Eqns. A.79 and A.86) and twelve evolution equations (Eqns. A.101 and A.102). These equations are completely equivalent to the Einstein equations.

A.1.5 The ADM Equations

So far, we have derived our equations in a covariant, coordinate-independent manner i.e. the basis vectors have been completely arbitrary and have no particular relationship to the 1-form Ω_a or to the congruence defined by t^a . Here we consider a specific choice of basis vector. (There are various motivations with regard to this choice but here we are concentrating only on the formulation itself.) We take it to be

$$t^a = (1, 0, 0, 0). \quad (\text{A.103})$$

This implies that the Lie derivative along t^a can be written as a partial derivative with respect to t i.e. $\mathcal{L}_t = \partial_t$. Since spatial tensors vanish when contracted with the normal vector, this also means that all components of spatial tensors with a contravariant index equal to zero must vanish. For the shift vector, this implies $n_a \beta^a = n_0 \beta^0 = 0$. Note, too, that spatial basis vectors, $e^a_{(i)}$, where the (i) distinguishes the vectors and not vector components, should satisfy $\Omega_a e^a_{(i)} = 0$. From this choice and condition, we can conclude $\Omega_a e^a_{(i)} = -\frac{1}{\alpha} n_a e^a_{(i)} = 0$ which implies that the covariant spatial components of the normal vector have to vanish

$$n_i = 0. \quad (\text{A.104})$$

Consequently, the shift vector should have the form

$$\beta^a = (0, \beta^i). \quad (\text{A.105})$$

Defining the vector $t^a = \alpha n^a + \beta^a$ gives the contravariant components

$$n^a = \left(\frac{1}{\alpha}, -\frac{\beta^i}{\alpha} \right). \quad (\text{A.106})$$

and from the normalization condition $n_a n^a = -1$ we find

$$n_a = (-\alpha, 0, 0, 0). \quad (\text{A.107})$$

Then from the definition of the spatial metric $\gamma_{ab} = g_{ab} + n_a n_b$ we have

$$\gamma_{ij} = g_{ij}. \quad (\text{A.108})$$

this implies that γ_{ij} , the metric on Σ , is just the spatial part of the covariant form of the 4-metric g_{ab} . Because the zeroth component of all spatial contravariant tensors must vanish, we also have $\gamma^{a0} = 0$. The inverse metric can therefore be expressed as

$$g^{ab} = \gamma^{ab} - n^a n^b = \begin{pmatrix} -\frac{1}{\alpha^2} & \frac{\beta^i}{\alpha^2} \\ \frac{\beta^j}{\alpha^2} & \gamma^{ij} - \frac{\beta^i \beta^j}{\alpha^2} \end{pmatrix}. \quad (\text{A.109})$$

From this, we can check

$$\gamma^{ik} \gamma_{kj} = (g^{ik} + n^i n^k) g_{kj} = g^{ik} g_{kj} + \underbrace{n^i n^k g_{kj}}_{=0} = \delta^i_j. \quad (\text{A.110})$$

This implies that γ^{ij} and γ_{ij} are 3-dimensional inverses, and can hence be used to raise and lower spatial indices of spatial tensors i.e. $\beta_i = \gamma_{ij} \beta^j$. Inverting g^{ab} gives g_{ab}

$$g_{ab} = \begin{pmatrix} -\alpha^2 + \beta_l \beta^l & \beta_i \\ \beta_j & \gamma_{ij} \end{pmatrix}. \quad (\text{A.111})$$

So the line element may be decomposed as

$$ds^2 = g_{ab} dx^a dx^b = -\alpha^2 dt^2 + \gamma_{ij} (dx^i + \beta^i dt) (dx^j + \beta^j dt), \quad (\text{A.112})$$

which is often referred to as the metric in 3+1 form. The entire content of any spatial tensor is available from their spatial components. This is true for contravariant components, since their zeroth component vanishes, but it is also true for covariant components. Therefore, the entire content of the decomposed Einstein equations is contained in their spatial components alone. Thus, we can rewrite the constraint and evolution equations in the same form as previously but only with reference to spatial indices.

For the Hamiltonian constraint (Eqn. A.79), we rewrite

$$\begin{aligned} R + K^2 - K^{ab} K_{ab} &= 16\pi\rho, \\ \Rightarrow R + K^2 - K^{ij} K_{ij} &= 16\pi\rho. \end{aligned} \quad (\text{A.113})$$

For the momentum constraint (Eqn. A.86), we again rewrite

$$\begin{aligned} D_b K_a^b - D_a K &= 8\pi S_a, \\ \Rightarrow D_j K_i^j - D_i K &= 8\pi S_i, \end{aligned} \quad (\text{A.114})$$

or equivalently

$$\begin{aligned} D_j K^{ji} - D^i K &= 8\pi S^i, \\ \Rightarrow D_j K^{ij} - \gamma^{ij} D_j K &= D_j (K^{ij} - \gamma^{ij} K) = 8\pi S^i. \end{aligned} \quad (\text{A.115})$$

For the evolution equation of the extrinsic curvature (Eqn. A.101), we get

$$\begin{aligned} \mathcal{L}_t K_{ab} &= \mathcal{L}_\beta K_{ab} - D_a D_b \alpha + \alpha (R_{ab} + K K_{ab} - 2K_b^c K_{ac}) - 8\pi \alpha \left(S_{ab} - \frac{1}{2} \gamma_{ab} (S - \rho) \right), \\ \Rightarrow \partial_t K_{ij} &= \mathcal{L}_\beta K_{ij} - D_i D_j \alpha + \alpha (R_{ij} + K K_{ij} - 2K_j^k K_{ik}) - 8\pi \alpha \left(S_{ij} - \frac{1}{2} \gamma_{ij} (S - \rho) \right) \\ &= \beta^k D_k K_{ij} + K_{ik} D_j \beta^k + K_{kj} D_i \beta^k - D_i D_j \alpha + \alpha (R_{ij} + K K_{ij} - 2K_j^k K_{ik}) \\ &\quad - 8\pi \alpha \left(S_{ij} - \frac{1}{2} \gamma_{ij} (S - \rho) \right). \end{aligned} \quad (\text{A.116})$$

For the evolution equation of the spatial metric (Eqn. A.102), we have

$$\begin{aligned} \mathcal{L}_t \gamma_{ab} &= \mathcal{L}_\beta \gamma_{ab} - 2\alpha K_{ab}, \\ \Rightarrow \partial_t \gamma_{ij} &= \mathcal{L}_\beta \gamma_{ij} - 2\alpha K_{ij} \\ &= -2\alpha K_{ij} + \underbrace{\beta^k D_k \gamma_{ij} + \gamma_{ik} D_j \beta^k + \gamma_{kj} D_i \beta^k}_{=0} \\ &= -2\alpha K_{ij} + D_j (\gamma_{ik} \beta^k) + D_i (\gamma_{kj} \beta^k) \\ &= -2\alpha K_{ij} + D_i \beta_j + D_j \beta_i. \end{aligned} \quad (\text{A.117})$$

with the matter sources appearing in the above equations defined by

$$\rho = n_a n_b T^{ab}, \quad S^i = -\gamma^{ij} n^a T_{aj}, \quad S_{ij} = \gamma_{ia} \gamma_{jb} T^{ab}, \quad S = \gamma^{ij} S_{ij} \quad (\text{A.118})$$

Eqns A.113, A.115, A.116, and A.117 are equivalent to the Einstein equations and comprise the standard 3+1 equations. These equations are also referred to as the ADM (Arnowitt-Deser-Misner) equations.

Using these ADM equations, we can derive other equations for the determinant of the spatial metric $\gamma = \det(\gamma_{ij})$ and the trace of the extrinsic curvature $K = K_i^i$.

Consider a version of Jacobi's formula for γ

$$\partial_t \ln \sqrt{\gamma} = \frac{1}{2\gamma} \partial_t \gamma = \frac{1}{2} \gamma^{ij} \partial_t \gamma_{ij}. \quad (\text{A.119})$$

Then, using Eqn. A.117, we can write

$$\begin{aligned}
\partial_t \ln \sqrt{\gamma} &= \frac{1}{2} \gamma^{jj} (-2\alpha K_{ij} + D_i \beta_j + D_j \beta_i) \\
&= -\alpha K + \frac{1}{2} \underbrace{(D^j \beta_j + D^i \beta_i)}_{=D^i \beta_i} \\
&= -\alpha K + D^i \beta_i = -\alpha K + D_i \beta^i.
\end{aligned} \tag{A.120}$$

Contracting Eqn. A.116 with γ^{jj} gives

$$\begin{aligned}
\gamma^{jj} \partial_t K_{ij} &= \gamma^{jj} \left[\beta^k D_k K_{ij} + K_{ik} D_j \beta^k + K_{kj} D_i \beta^k - D_i D_j \alpha + \alpha (R_{ij} + K K_{ij} - 2K_j^k K_{ik}) \right. \\
&\quad \left. - 8\pi\alpha \left(S_{ij} - \frac{1}{2} \gamma_{ij} (S - \rho) \right) \right] \\
&= \beta^k D_k \underbrace{(\gamma^{jj} K_{ij})}_{=K} + \underbrace{\gamma^{jj} K_{ik} D_j \beta^k}_{=K_k^j D_j \beta^k = K_k^i D_i \beta^k} + \underbrace{\gamma^{jj} K_{kj} D_i \beta^k}_{=K_k^i D_i \beta^k} - \underbrace{\gamma^{jj} D_i D_j \alpha}_{\equiv D^2} \\
&\quad + \alpha \left(\underbrace{\gamma^{jj} R_{ij}}_{=R} + \underbrace{K \gamma^{jj} K_{ij}}_{K^2} - 2 \underbrace{\gamma^{jj} K_j^k K_{ik}}_{K^{ik}} \right) - 8\pi\alpha \left(\underbrace{\gamma^{jj} S_{ij}}_{=S} - \frac{1}{2} \underbrace{\gamma^{jj} \gamma_{ij} (S - \rho)}_{=3} \right) \\
&= \beta^k D_k K + 2K_k^i D_i \beta^k - D^2 \alpha + \alpha \underbrace{(R + K^2 - K^{ik} K_{ik} - K^{ik} K_{ik})}_{=16\pi\rho} - 4\pi\alpha (3\rho - S) \\
&= \beta^k D_k K + 2K_k^i D_i \beta^k - D^2 \alpha - \alpha K^{ik} K_{ik} + 4\pi\alpha (S + \rho).
\end{aligned} \tag{A.121}$$

On evaluating the LHS expression $\gamma^{jj} \partial_t K_{ij}$, we get

$$\gamma^{jj} \partial_t K_{ij} = \partial_t (\gamma^{jj} K_{ij}) - K_{ij} \partial_t \gamma^{jj} = \partial_t K - K_{ij} \partial_t \gamma^{jj}. \tag{A.122}$$

We need to find the evolution equations for the inverse of the spatial metric. To do that, we use the fact $\gamma^{ik} \gamma_{kj} = \delta_j^i$

$$\begin{aligned}
\partial_t (\gamma^{ik} \gamma_{kj}) &= \partial_t \delta_j^i = 0 = \gamma^{ik} \partial_t \gamma_{kj} + \gamma_{kj} \partial_t \gamma^{ik}, \\
\Rightarrow \gamma_{kj} \partial_t \gamma^{ik} &= -\gamma^{ik} \partial_t \gamma_{kj}.
\end{aligned} \tag{A.123}$$

Now, contracting both sides with γ^{jl} and using Eqn. A.117, we get

$$\begin{aligned}\gamma^{jl}\gamma_{kj}\partial_t\gamma^{ik} &= -\gamma^{jl}\gamma^{ik}\partial_t\gamma_{kj}, \\ \Rightarrow \delta_k^l\partial_t\gamma^{ik} &= \partial_t\gamma^{il} = -\gamma^{jl}\gamma^{ik}(-2\alpha K_{kj} + D_k\beta_j + D_j\beta_k) = 2\alpha K^{il} - \gamma^{ik}D_k\beta^j - \gamma^{kj}D_k\beta^i.\end{aligned}\tag{A.124}$$

Thus, we have

$$\begin{aligned}\partial_t K - K_{ij}\partial_t\gamma^{ij} &= \partial_t K - K_{ij}(2\alpha K^{ij} - \gamma^{ik}D_k\beta^j - \gamma^{kj}D_k\beta^i) \\ &= \partial_t K - 2\alpha K^{ij}K_{ij} + K_j^k D_k\beta^j + K_i^k D_k\beta^i \\ &= \partial_t K - 2\alpha K^{ij}K_{ij} + 2K_i^k D_k\beta^i.\end{aligned}\tag{A.125}$$

On rearranging dummy indices in Eqns. A.121 and A.125 then we are able to find

$$\begin{aligned}\partial_t K - 2\alpha K_{ij}K^{ij} + 2K_k^i D_i\beta^k &= \beta^i D_i K + 2K_k^i D_i\beta^k - D^2\alpha - \alpha K^{ij}K_{ij} + 4\pi\alpha(S + \rho), \\ \Rightarrow \partial_t K &= -D^2\alpha + \alpha [K_{ij}K^{ij} + 4\pi(S + \rho)] + \beta^i D_i K.\end{aligned}\tag{A.126}$$

Here we have obtained additional evolution equations for the determinant of the spatial metric γ and the trace of the extrinsic curvature K . Eqns A.120 and A.126 will be used for some calculations in later sections.

A.2 Conformal Transformations

The previous section derived the decomposition of the Einstein equations into 3+1 form (or the ADM form) which gives the basic evolution and constraint equations for gravitational interactions. The constraint equations provide consistent initial data that we can use for black hole simulations. The evolution equations provide means by which the dynamics of spacetime metrics can be investigated. Oftentimes, solving those equations is very hard so techniques have been developed in order to try to solve them differently and attempt to isolate the pertinent degrees of freedom.

In this section, we consider aspects of conformal transformations. In particular, we consider conformal transformations of the spatial metric. In other contexts it may also

be useful to study conformal transformations of the spacetime metric. For our purposes, it has been found that a rewriting of the Einstein equations through a certain conformal transformation renders the relevant equations more stable than the ADM form we have previously derived.

A.2.1 Conformal Transformation of the Spatial Metric

Let's begin by writing the spatial metric γ_{ij} as a product of some power of a positive scaling factor ψ and a background metric $\bar{\gamma}_{ij}$

$$\gamma_{ij} = \psi^p \bar{\gamma}_{ij}. \quad (\text{A.127})$$

This identification is a conformal transformation of the spatial metric. We call ψ as the conformal factor and $\bar{\gamma}_{ij}$ as the conformally related metric. In general p could be any integer, but here we will choose $p = 4$ as the conformal power. From the property of the metric $\gamma_{ik} \gamma^{kj} = \bar{\gamma}_{ik} \bar{\gamma}^{kj} = \delta_j^i$, the conformal transformation for the inverse metric should be

$$\gamma^{jj} = \psi^{-4} \bar{\gamma}^{jj}. \quad (\text{A.128})$$

Superficially, the conformal transformation just shows rewriting one unknown as a product of two unknowns in order to make solving some equations easier but the conformal transformation serves to define an equivalence class of manifolds and metric. (See Wald's GR book for a more detailed discussion.)

Using this transformation, we can find a transformation rule for the connection coefficients and etc. In three dimensions, the connection coefficients must transform according

to

$$\begin{aligned}
\Gamma^i_{jk} &= \frac{1}{2} \gamma^{il} (\partial_k \gamma_{lj} + \partial_j \gamma_{lk} - \partial_l \gamma_{jk}) \\
&= \frac{1}{2} \psi^{-4} \bar{\gamma}^{il} (\partial_k (\psi^4 \bar{\gamma}_{lj}) + \partial_j (\psi^4 \bar{\gamma}_{lk}) - \partial_l (\psi^4 \bar{\gamma}_{jk})) \\
&= \frac{1}{2} \psi^{-4} \bar{\gamma}^{il} (\psi^4 \partial_k \bar{\gamma}_{lj} + \bar{\gamma}_{lj} \partial_k \psi^4 + \psi^4 \partial_j \bar{\gamma}_{lk} + \bar{\gamma}_{lk} \partial_j \psi^4 - \psi^4 \partial_l \bar{\gamma}_{jk} - \bar{\gamma}_{jk} \partial_l \psi^4) \\
&= \frac{1}{2} \psi^{-4} \bar{\gamma}^{il} \psi^4 (\partial_k \bar{\gamma}_{lj} + \partial_j \bar{\gamma}_{lk} - \partial_l \bar{\gamma}_{jk}) + \frac{1}{2} \psi^{-4} (\bar{\gamma}^{il} \bar{\gamma}_{lj} \partial_k \psi^4 + \bar{\gamma}^{il} \bar{\gamma}_{lk} \partial_j \psi^4 - \bar{\gamma}^{il} \bar{\gamma}_{jk} \partial_l \psi^4) \\
&= \bar{\Gamma}^i_{jk} + \frac{1}{2} \psi^{-4} (\delta_j^i \partial_k \psi^4 + \delta_k^j \partial_j \psi^4 - \bar{\gamma}^{il} \bar{\gamma}_{jk} \partial_l \psi^4), \tag{A.129}
\end{aligned}$$

where we define

$$\bar{\Gamma}^i_{jk} \equiv \frac{1}{2} \gamma^{il} (\partial_k \bar{\gamma}_{lj} + \partial_j \bar{\gamma}_{lk} - \partial_l \bar{\gamma}_{jk}), \tag{A.130}$$

and we can further simplify

$$\psi^{-4} \partial_i \psi^4 = \psi^{-4} (4\psi^3 \partial_i \psi) = \frac{4}{\psi} \partial_i \psi = 4\partial_i \ln \psi. \tag{A.131}$$

As a result, we get

$$\begin{aligned}
\Gamma^i_{jk} &= \bar{\Gamma}^i_{jk} + 2(\delta_j^i \partial_k \ln \psi + \delta_k^j \partial_j \ln \psi - \bar{\gamma}_{jk} \bar{\gamma}^{il} \partial_l \ln \psi) \\
&= \bar{\Gamma}^i_{jk} + 2(\delta_j^i \bar{D}_k \ln \psi + \delta_k^j \bar{D}_j \ln \psi - \bar{\gamma}_{jk} \bar{\gamma}^{il} \bar{D}_l \ln \psi). \tag{A.132}
\end{aligned}$$

In the last line, because ψ is a scalar function, we can replace partial derivatives with spatial covariant derivatives built from the conformally related metric. In addition, \bar{D}_i will be compatible with the conformally related metric i.e. $\bar{D}_i \bar{\gamma}_{jk} = 0$ in the usual way.

Without detailed proof, we can easily make relationship between D_i and \bar{D}_i . For an arbitrary vector v^i , $D_j v^i$ and $\bar{D}_j v^i$ are related by the formula

$$D_j v^i = \bar{D}_j v^i + C^i_{jk} v^k, \tag{A.133}$$

where C^i_{jk} is

$$C^i_{jk} \equiv \Gamma^i_{jk} - \bar{\Gamma}^i_{jk} = 2(\delta_j^i \bar{D}_k \ln \psi + \delta_k^j \bar{D}_j \ln \psi - \bar{\gamma}_{jk} \bar{\gamma}^{il} \bar{D}_l \ln \psi). \tag{A.134}$$

Thus C^i_{jk} has the same property as the connection coefficients. For example, $C^i_{jk} = C^i_{kj}$ for a torsion free manifold. Without loss of generality, this relation still works for higher rank tensors. For example,

$$D_i F_k^j = \bar{D}_i F_k^j + C^j_{il} F_k^l - C^l_{ki} F_l^j. \quad (\text{A.135})$$

This relation is useful for many calculations.

We next consider the Ricci tensor. From the Riemann tensor relation, $R_{dbac} n^i = 2D_{[l} D_{k]} n_j$, the Ricci tensor can be expressed as

$$R_{ij} v^j = D_j D_i v^j - D_i D_j v^j. \quad (\text{A.136})$$

Express D_i in terms of \bar{D}_i

$$\begin{aligned} R_{ij} v^j &= D_j D_i v^j - D_i D_j v^j \\ &= \bar{D}_j (D_i v^j) + C^j_{jk} D_i v^k - C^k_{ij} D_k v^j - (\bar{D}_i (D_j v^j) + \underbrace{C^j_{ik} D_j v_k - C^k_{ji} D_k v^j}_{=C^k_{ji} D_k v^j}) \\ &= \bar{D}_j (\bar{D}_i v^j + C^j_{ik} v^k) + C^j_{jk} (\bar{D}_i v^k + C^k_{il} v^l) - \bar{D}_i (\bar{D}_j v^j + C^j_{jk} v^k) - C^k_{ji} (\bar{D}_k v^j + C^j_{kl} v^l) \\ &= \bar{D}_j \bar{D}_i v^j + \underbrace{(\bar{D}_j C^j_{ik}) v^k}_{=(\bar{D}_k C^k_{ij}) v^j} + C^j_{ik} \bar{D}_j v^k + C^j_{jk} \bar{D}_i v^k + \underbrace{C^j_{jk} C^k_{il} v^l}_{=C^l_{lk} C^k_{ij} v^j} \\ &\quad - \bar{D}_i \bar{D}_j v^j - \underbrace{(\bar{D}_i C^j_{jk}) v^k}_{=(\bar{D}_i C^k_{kj}) v^j} - C^j_{jk} \bar{D}_i v^k - \underbrace{C^k_{jk} \bar{D}_k v^j}_{=C^j_{ki} \bar{D}_j v^k} - \underbrace{C^k_{ji} C^j_{kl} v^l}_{=C^k_{li} C^l_{kj} v^j} \\ &= \bar{D}_j \bar{D}_i v^j - \bar{D}_i \bar{D}_j v^j + (\bar{D}_k C^k_{ij}) v^j - (\bar{D}_i C^k_{kj}) v^j + C^l_{lk} C^k_{ij} v^j - C^k_{li} C^l_{kj} v^j, \end{aligned} \quad (\text{A.137})$$

where we have relabeled some dummy indices ($j \leftrightarrow k$ or $j \leftrightarrow l$). Defining a conformally related Ricci tensor

$$\bar{R}_{ij} v^j = \bar{D}_j \bar{D}_i v^j - \bar{D}_i \bar{D}_j v^j, \quad (\text{A.138})$$

where, because this is true for all arbitrary v^j , we can conclude

$$R_{ij} = \bar{R}_{ij} + \bar{D}_k C^k_{ij} - \bar{D}_i C^k_{kj} + C^l_{lk} C^k_{ij} - C^k_{li} C^l_{kj}. \quad (\text{A.139})$$

We can express this relation in terms of the conformally related metric and the conformal scalar function. Evaluating term by term in Eqn. A.139, we get

$$\begin{aligned}
\bar{D}_k C^k_{ij} &= 2\bar{D}_k(\delta_i^k \bar{D}_j \ln \psi + \delta_i^k \bar{D}_i \ln \psi - \bar{\gamma}_{ij} \bar{\gamma}^{kl} \bar{D}_l \ln \psi) \\
&= 2(\underbrace{\delta_i^k \bar{D}_k \bar{D}_j \ln \psi}_{=\bar{D}_i} + \underbrace{\delta_i^k \bar{D}_k \bar{D}_i \ln \psi}_{=\bar{D}_j} - \bar{\gamma}_{ij} \bar{\gamma}^{kl} \bar{D}_k \bar{D}_l \ln \psi) \\
&= 4\bar{D}_i \bar{D}_j \ln \psi - 2\bar{\gamma}_{ij} \bar{\gamma}^{kl} \bar{D}_k \bar{D}_l \ln \psi.
\end{aligned} \tag{A.140}$$

Here we have used the fact that $\bar{D}_i \bar{D}_j f = \bar{D}_j \bar{D}_i f$ i.e. covariant derivatives commute when applied to scalar functions. Now, we perform some calculations for later

$$\begin{aligned}
C^k_{kj} &= 2(\underbrace{\delta_k^k \bar{D}_j \ln \psi}_{=3} + \underbrace{\delta_j^k \bar{D}_k \ln \psi}_{=\bar{D}_j} - \underbrace{\bar{\gamma}_{kj} \bar{\gamma}^{kl} \bar{D}_l \ln \psi}_{=\delta_j^l \bar{D}_l = \bar{D}_j}) \\
&= 6\bar{D}_j \ln \psi,
\end{aligned} \tag{A.141}$$

$$\Rightarrow \bar{D}_i C^k_{kj} = 6\bar{D}_i \bar{D}_j \ln \psi. \tag{A.142}$$

$$\begin{aligned}
C^l_{lk} C^k_{ij} &= 12\bar{D}_k \ln \psi (\delta_i^k \bar{D}_j \ln \psi + \delta_j^k \bar{D}_i \ln \psi - \bar{\gamma}_{ij} \bar{\gamma}^{kl} \bar{D}_l \ln \psi) \\
&= 12(\underbrace{\delta_i^k \bar{D}_k \ln \psi \bar{D}_j \ln \psi}_{\bar{D}_i} + \underbrace{\delta_j^k \bar{D}_k \ln \psi \bar{D}_i \ln \psi}_{\bar{D}_j} - \bar{\gamma}_{ij} \bar{\gamma}^{kl} \bar{D}_k \ln \psi \bar{D}_l \ln \psi) \\
&= 24\bar{D}_i \ln \psi \bar{D}_j \ln \psi - 12\bar{\gamma}_{ij} \bar{\gamma}^{kl} \bar{D}_k \ln \psi \bar{D}_l \ln \psi.
\end{aligned} \tag{A.143}$$

$$\begin{aligned}
C^k_{li} C^l_{kj} &= 4(\delta_i^k \bar{D}_i \ln \psi + \delta_i^k \bar{D}_l \ln \psi - \bar{\gamma}_{li} \bar{\gamma}^{km} \bar{D}_m \ln \psi)(\delta_k^l \bar{D}_j \ln \psi + \delta_j^l \bar{D}_k \ln \psi - \bar{\gamma}_{kj} \bar{\gamma}^{lm} \bar{D}_m \ln \psi) \\
&= 4(\underbrace{\delta_i^k \delta_k^l \bar{D}_i \ln \psi \bar{D}_j \ln \psi}_{=3} + \underbrace{\delta_i^k \delta_j^l \bar{D}_i \ln \psi \bar{D}_k \ln \psi}_{=\delta_j^k} - \underbrace{\delta_i^k \bar{\gamma}_{kj} \bar{\gamma}^{lm} \bar{D}_i \ln \psi \bar{D}_m \ln \psi}_{\bar{\gamma}_{lj} \bar{\gamma}^{lm} = \delta_j^m} \\
&\quad + \underbrace{\delta_i^k \delta_k^l \bar{D}_l \ln \psi \bar{D}_j \ln \psi}_{=\delta_i^l} \\
&\quad + \delta_i^k \delta_j^l \bar{D}_l \ln \psi \bar{D}_k \ln \psi - \underbrace{\delta_i^k \bar{\gamma}_{kj} \bar{\gamma}^{lm} \bar{D}_l \ln \psi \bar{D}_m \ln \psi}_{=\bar{\gamma}_{ij}} - \underbrace{\delta_k^l \bar{\gamma}_{ki} \bar{\gamma}^{km} \bar{D}_m \ln \psi \bar{D}_j \ln \psi}_{=\bar{\gamma}_{li} \bar{\gamma}^{lm} = \delta_i^m} \\
&\quad - \underbrace{\delta_j^l \bar{\gamma}_{lj} \bar{\gamma}^{km} \bar{D}_k \ln \psi \bar{D}_m \ln \psi}_{\bar{\gamma}_{ij}} + \underbrace{\bar{\gamma}_{li} \bar{\gamma}^{lm} \bar{\gamma}_{kj} \bar{\gamma}^{km} \bar{D}_m \ln \psi \bar{D}_m \ln \psi}_{=\delta_i^m \delta_j^m}) \\
&= 4(3\bar{D}_i \ln \psi \bar{D}_j \ln \psi + \bar{D}_i \ln \psi \bar{D}_j \ln \psi - \bar{D}_i \ln \psi \bar{D}_j \ln \psi + \bar{D}_i \ln \psi \bar{D}_j \ln \psi \\
&\quad + \bar{D}_j \ln \psi \bar{D}_i \ln \psi - \bar{\gamma}_{ij} \bar{\gamma}^{lm} \bar{D}_l \ln \psi \bar{D}_m \ln \psi - \bar{D}_i \ln \psi \bar{D}_j \ln \psi \\
&\quad - \underbrace{\bar{\gamma}_{ij} \bar{\gamma}^{km} \bar{D}_k \ln \psi \bar{D}_m \ln \psi + \bar{D}_i \ln \psi \bar{D}_j \ln \psi}_{=\bar{\gamma}_{ij} \bar{\gamma}^{lm} \bar{D}_l \ln \psi \bar{D}_m \ln \psi}) \\
&= 20\bar{D}_i \ln \psi \bar{D}_j \ln \psi - 8\bar{\gamma}_{ij} \bar{\gamma}^{lm} \bar{D}_l \ln \psi \bar{D}_m \ln \psi. \tag{A.144}
\end{aligned}$$

Combining these results gives

$$\begin{aligned}
R_{ij} &= \bar{R}_{ij} + \bar{D}_k C^k_{ij} - \bar{D}_i C^k_{kj} + C^l_{lk} C^k_{ij} - C^k_{li} C^l_{kj} \\
&= 4\bar{D}_i \bar{D}_j \ln \psi - 2\bar{\gamma}_{ij} \underbrace{\bar{\gamma}^{kl} \bar{D}_k \bar{D}_l \ln \psi}_{=\bar{\gamma}^{ml} \bar{D}_m} - 6\bar{D}_i \bar{D}_j \ln \psi + 24\bar{D}_i \ln \psi \bar{D}_j \ln \psi - 12\bar{\gamma}_{ij} \underbrace{\bar{\gamma}^{kl} \bar{D}_k \ln \psi \bar{D}_l \ln \psi}_{=\bar{\gamma}^{ml} \bar{D}_m} \\
&\quad - (20\bar{D}_i \ln \psi \bar{D}_j \ln \psi - 8\bar{\gamma}_{ij} \bar{\gamma}^{lm} \bar{D}_l \ln \psi \bar{D}_m \ln \psi) \\
&= \bar{R}_{ij} - 2(\bar{D}_i \bar{D}_j \ln \psi + \bar{\gamma}_{ij} \bar{\gamma}^{lm} \bar{D}_l \bar{D}_m \ln \psi) + 4(\bar{D}_i \ln \psi \bar{D}_j \ln \psi - \bar{\gamma}_{ij} \bar{\gamma}^{lm} \bar{D}_l \ln \psi \bar{D}_m \ln \psi). \tag{A.145}
\end{aligned}$$

For the Ricci scalar, we obtain

$$\begin{aligned}
R &= \gamma^{jj} R_{ij} \\
&= \psi^{-4} \bar{\gamma}^{jj} [\bar{R}_{ij} - 2(\bar{D}_i \bar{D}_j \ln \psi + \bar{\gamma}_{ij} \bar{\gamma}^{lm} \bar{D}_l \bar{D}_m \ln \psi) + 4(\bar{D}_i \ln \psi \bar{D}_j \ln \psi - \bar{\gamma}_{ij} \bar{\gamma}^{lm} \bar{D}_l \ln \psi \bar{D}_m \ln \psi)] \\
&= \psi^{-4} [\bar{\gamma}^{jj} \bar{R}_{ij} - 2(\bar{\gamma}^{jj} \bar{D}_i \bar{D}_j \ln \psi + \underbrace{\bar{\gamma}^{jj} \bar{\gamma}_{ij} \bar{\gamma}^{lm}}_{=3} \bar{D}_l \bar{D}_m \ln \psi) \\
&\quad + 4(\bar{\gamma}^{jj} \bar{D}_i \ln \psi \bar{D}_j \ln \psi - \underbrace{\bar{\gamma}^{jj} \bar{\gamma}_{ij} \bar{\gamma}^{lm}}_{=3} \bar{D}_l \ln \psi \bar{D}_m \ln \psi)]. \tag{A.146}
\end{aligned}$$

Defining the conformally related Ricci scalar as $\bar{R} = \bar{\gamma}^{jj} \bar{R}_{ij}$ and defining the covariant Laplacian associated with $\bar{\gamma}_{ij}$ as $\bar{D}^2 = \bar{D}^i \bar{D}_i = \bar{\gamma}^{jj} \bar{D}_i \bar{D}_j$, we get

$$R = \psi^{-4} [\bar{R} - 8(\bar{D}^2 \ln \psi - \bar{\gamma}^{jj} \bar{D}_i \ln \psi \bar{D}_j \ln \psi)]. \tag{A.147}$$

Here again we do some relabeling of dummy indices. We can rewrite this using

$$\begin{aligned}
\bar{D}^2 \ln \psi &= \bar{\gamma}^{jj} \bar{D}_i \bar{D}_j \ln \psi = \bar{\gamma}^{jj} \bar{D}_i \left(\frac{1}{\psi} \bar{D}_j \psi \right) \\
&= \bar{\gamma}^{jj} \left(\frac{1}{\psi} \bar{D}_i \bar{D}_j \psi - \frac{1}{\psi^2} \bar{D}_i \psi \bar{D}_j \psi \right) \\
&= \frac{1}{\psi} \bar{\gamma}^{jj} \bar{D}_i \bar{D}_j \psi - \bar{\gamma}^{jj} \underbrace{\left(\frac{1}{\psi} \bar{D}_i \psi \right)}_{=\bar{D}_i \ln \psi} \underbrace{\left(\frac{1}{\psi} \bar{D}_j \psi \right)}_{=\bar{D}_j \ln \psi} \\
&= \frac{1}{\psi} \bar{D}^2 \psi - \bar{\gamma}^{jj} \bar{D}_i \ln \psi \bar{D}_j \ln \psi. \tag{A.148}
\end{aligned}$$

Therefore, we can get

$$R = \psi^{-4} \bar{R} - 8\psi^{-5} \bar{D}^2 \psi. \tag{A.149}$$

A.2.2 Conformal Transformation of the Extrinsic Curvature

It is useful to rescale the extrinsic curvature K_{ij} conformally. Practically, it is convenient to split K_{ij} into its trace part K and a traceless, symmetric part A_{ij} such that

$$K_{ij} = A_{ij} + \frac{1}{3} \gamma_{ij} K. \tag{A.150}$$

It is easy to check that the above definition makes A_{ij} traceless

$$\gamma^{ij}A_{ij} = \underbrace{\gamma^{ij}K_{ij}}_{=K} - \frac{1}{3} \underbrace{\gamma^{ij}\gamma_{ij}}_{=3} K = 0. \quad (\text{A.151})$$

Consider the transformations

$$a^{ij} = \psi^\alpha \bar{A}^{ij}, \quad (\text{A.152})$$

$$K = \psi^\beta \bar{K}. \quad (\text{A.153})$$

where α and β are arbitrary exponents. The choices of α and β are completely free. The purpose of these transformations is that the transformation should bring the systems of equation into a simple and solvable form. We can choose different exponents subject to the system we want to solve. For example, $\alpha = -10$ is widely used to solve the constraint equations while $\alpha = -4$ is chosen to build the standard BSSN formalism.

A.3 Reformulating: The Evolution Equations

At this point, we derive the 3+1 decomposition of the Einstein equations (Eqns. A.113, A.114, A.116, and A.117, i.e. the ADM equations) that provides the initial value problem (or Cauchy problem). Unfortunately, despite many efforts over many years (including careful numerical implementation and methodological preparation), most simulations would crash a rather short time.

Ultimately, it came to be realized that The ADM equations are not in a form that is suitable for stable numerical integration. The failure of these equations can be understood in terms of their mathematical properties, including notions of hyperbolicity and well-posedness. Detailed discussions of these problems can be found in a number of places [37, 55, 56, 57]. Here, we comment only that the ADM equations are indeed only weakly hyperbolic. Consequently, the ADM equations are not well-posed, and we have no guarantee with which to expect the solutions of numerical implementations will be stable well-behaved for long simulation times.

In the following, we will discuss a different approach that leads to different reformulations of the evolution equations that are both strongly hyperbolic and that have been used successfully in numerical simulations. There are many different approaches but we will concentrate on the BSSN (Baumgarte, Shapiro, Shibata, Nakamura) formulation.

A.3.1 The BSSN Equations

To derive this formulation, we choose the conformal factor ψ as $\psi = e^{-\phi}$ so that we have the following conformal transformation for the spatial metric:

$$\gamma_{ij} = e^{4\phi} \bar{\gamma}_{ij}. \quad (\text{A.154})$$

We then require that the determinant of the conformally related metric $\bar{\gamma}_{ij}$ be equal to that of the flat metric η_{ij} in whatever coordinate system we are using such that

$$\phi = \frac{1}{12} \ln \frac{\gamma}{\eta}. \quad (\text{A.155})$$

In the following, we adopt a Cartesian coordinate system so that $\bar{\gamma} = \eta = 1$.

Also, we choose a conformal rescaling for \tilde{A}_{ij} such that

$$A_{ij} = e^{4\phi} \tilde{A}_{ij}. \quad (\text{A.156})$$

For the inverse A^{ij} , we get

$$\begin{aligned} A^{ij} &= \gamma^{ik} \gamma^{jl} A_{kl} = e^{-4\phi} \bar{\gamma}^{ik} e^{-4\phi} \bar{\gamma}_{jl} e^{4\phi} \tilde{A}_{kl} \\ &= e^{-4\phi} \tilde{A}^{ij}. \end{aligned} \quad (\text{A.157})$$

Here we use \tilde{A} instead of \bar{A} to avoid confusion between different conformal transformations for the evolution scheme and the constraint equations. So, the extrinsic curvature can be expressed

$$K_{ij} = A_{ij} + \frac{1}{3} \gamma_{ij} K = e^{4\phi} \left(\tilde{A}_{ij} + \frac{1}{3} \bar{\gamma}_{ij} K \right). \quad (\text{A.158})$$

Now we need to find evolution equations for the new variables $\bar{\gamma}_{ij}$, \tilde{A}_{ij} , and ϕ . From the evolution equation for γ_{ij} (Eqn. A.117)

$$\partial_t \gamma_{ij} = \mathcal{L}_\beta \gamma_{ij} - 2\alpha K_{ij}. \quad (\text{A.159})$$

Applying the conformal transformation gives

$$\begin{aligned} \partial_t (e^{4\phi} \bar{\gamma}_{ij}) &= \mathcal{L}_\beta (e^{4\phi} \bar{\gamma}_{ij}) - 2\alpha e^{4\phi} \left(\tilde{A}_{ij} + \frac{1}{3} \bar{\gamma}_{ij} K \right), \\ 4e^{4\phi} \bar{\gamma}_{ij} \partial_t \phi + e^{4\phi} \partial_t \bar{\gamma}_{ij} &= 4e^{4\phi} \bar{\gamma}_{ij} \mathcal{L}_\beta \phi + e^{4\phi} \mathcal{L}_\beta \bar{\gamma}_{ij} - 2\alpha e^{4\phi} \left(\tilde{A}_{ij} + \frac{1}{3} \bar{\gamma}_{ij} K \right). \end{aligned} \quad (\text{A.160})$$

Multiplying both sides by $e^{-4\phi}$ we get

$$4\bar{\gamma}_{ij} \partial_t \phi + \partial_t \bar{\gamma}_{ij} = 4\bar{\gamma}_{ij} \mathcal{L}_\beta \phi + \mathcal{L}_\beta \bar{\gamma}_{ij} - 2\alpha \left(\tilde{A}_{ij} + \frac{1}{3} \bar{\gamma}_{ij} K \right). \quad (\text{A.161})$$

Now we need to find the evolution equation for ϕ . To do that, contract both sides with $\bar{\gamma}^{ij}$

$$4 \underbrace{\bar{\gamma}^{ij} \bar{\gamma}_{ij}}_{=3} \partial_t \phi + \bar{\gamma}^{ij} \partial_t \bar{\gamma}_{ij} = 4 \underbrace{\bar{\gamma}^{ij} \bar{\gamma}_{ij}}_{=3} \mathcal{L}_\beta \phi + \bar{\gamma}^{ij} \mathcal{L}_\beta \bar{\gamma}_{ij} - 2\alpha \left(\underbrace{\bar{\gamma}^{ij} \tilde{A}_{ij}}_{=0} + \frac{1}{3} \underbrace{\bar{\gamma}^{ij} \bar{\gamma}_{ij}}_{=3} K \right). \quad (\text{A.162})$$

and use a version of Jacobi's formula:

$$\bar{\gamma}^{ij} \partial_t \bar{\gamma}_{ij} = \frac{1}{2\bar{\gamma}} \partial_t \bar{\gamma} = 0. \quad (\text{A.163})$$

Note that the final equality is because $\bar{\gamma} = 1$. We now get

$$\begin{aligned} 12\partial_t \phi &= 12\mathcal{L}_\beta \phi + \bar{\gamma}^{ij} (\beta^k \underbrace{\bar{D}_k \bar{\gamma}_{ij}}_{=0} + \bar{\gamma}_{ik} \bar{D}_j \beta^k + \bar{\gamma}_{kj} \bar{D}_i \beta^k) - 2\alpha K \\ &= 12\beta^i \bar{D}_i \phi + \underbrace{\bar{\gamma}^{ij} \bar{\gamma}_{ik}}_{\delta_k^j} \bar{D}_j \beta^k + \underbrace{\bar{\gamma}^{ij} \bar{\gamma}_{kj}}_{\delta_k^i} \bar{D}_i \beta^k - 2\alpha K. \end{aligned} \quad (\text{A.164})$$

Here we require that the determinant of the conformally related metric be equal to the flat metric so we can replace the conformally related covariant derivative \bar{D}_i with the usual partial derivative ∂_i . Rearranging the indices and simplifying yields

$$\partial_t \phi = -\frac{1}{6} \alpha K + \beta^k \partial_k \phi + \frac{1}{6} \partial_k \beta^k. \quad (\text{A.165})$$

Substituting Eqn. A.165 into Eqn. A.161

$$\begin{aligned}
& 4\bar{\gamma}_{ij} \left(-\frac{1}{6}\alpha K + \beta^k \partial_k \phi + \frac{1}{6}\partial_k \beta^k \right) + \partial_t \bar{\gamma}_{ij} = 4\bar{\gamma}_{ij} \mathcal{L}_\beta \phi + \mathcal{L}_\beta \bar{\gamma}_{ij} - 2\alpha \left(\tilde{A}_{ij} + \frac{1}{3}\bar{\gamma}_{ij} K \right), \\
\Rightarrow \partial_t \bar{\gamma}_{ij} &= \frac{2}{3}\bar{\gamma}_{ij} \alpha K - 4\bar{\gamma}_{ij} \beta^k \partial_k \phi - \frac{2}{3}\bar{\gamma}_{ij} \partial_k \beta^k + 4\bar{\gamma}_{ij} \underbrace{\mathcal{L}_\beta \phi}_{=\beta^k \partial_k \phi} + \mathcal{L}_\beta \bar{\gamma}_{ij} - 2\alpha \tilde{A}_{ij} - \frac{2}{3}\bar{\gamma}_{ij} \alpha K \\
&= -2\alpha \tilde{A}_{ij} + \beta^k \partial_k \bar{\gamma}_{ij} + \bar{\gamma}_{ik} \partial_j \beta^k + \bar{\gamma}_{kj} \partial_i \beta^k - \frac{2}{3}\bar{\gamma}_{ij} \partial_k \beta^k. \tag{A.166}
\end{aligned}$$

Again partial derivatives come from the Lie derivative of $\bar{\gamma}_{ij}$. Next consider splitting the evolution equation for the extrinsic curvature. From Eqn. A.126

$$\partial_t K = -D^2 \alpha + \alpha [K_{ij} K^{ij} + 4\pi(S + \rho)] + \beta^i \partial_i K. \tag{A.167}$$

Note that we change $D_i K$ to $\partial_i K$ because K is a scalar function. The only term that affected under the transformation is $K^{ij} K_{ij}$, so

$$K^{ij} K_{ij} = \left(e^{4\phi} \tilde{A}_{ij} + \frac{1}{3} e^{4\phi} \bar{\gamma}_{ij} K \right) \left(e^{-4\phi} \tilde{A}^{ij} + \frac{1}{3} e^{-4\phi} \bar{\gamma}^{ij} K \right) = \tilde{A}^{ij} \tilde{A}_{ij} + \frac{1}{3} K^2. \tag{A.168}$$

Thus, we get

$$\begin{aligned}
\partial_t K &= -D^2 \alpha + \alpha \left[\tilde{A}^{ij} \tilde{A}_{ij} + \frac{1}{3} K^2 + 4\pi(S + \rho) \right] + \beta^i \partial_i K \\
&= -\gamma^{ij} D_j D_i \alpha + \alpha \left(\tilde{A}^{ij} \tilde{A}_{ij} + \frac{1}{3} K^2 \right) + 4\pi \alpha (\rho + S) + \beta^i \partial_i K. \tag{A.169}
\end{aligned}$$

From Eqn. A.116, we obtain

$$\begin{aligned}
\partial_t K_{ij} &= \beta^k D_k K_{ij} + K_{ik} D_j \beta^k + K_{kj} D_i \beta^k - D_i D_j \alpha + \alpha (R_{ij} + K K_{ij} - 2K_j^k K_{ik}) \\
&\quad - 8\pi \alpha \left(S_{ij} - \frac{1}{2} \gamma_{ij} (S - \rho) \right) \\
&= \mathcal{L}_\beta K_{ij} - D_i D_j \alpha + \alpha (R_{ij} + K K_{ij} - 2K_j^k K_{ik}) - 8\pi \alpha \left(S_{ij} - \frac{1}{2} \gamma_{ij} (S - \rho) \right).
\end{aligned}$$

For convenience, we define the operator $d_t \equiv \partial_t - \mathcal{L}_\beta$. We can then rewrite

$$d_t K_{ij} = -D_i D_j \alpha + \alpha (R_{ij} + K K_{ij} - 2K_j^k K_{ik}) - 8\pi \alpha \left(S_{ij} - \frac{1}{2} \gamma_{ij} (S - \rho) \right). \tag{A.170}$$

Using d_t we can also write

$$\begin{aligned}\partial_t \gamma_j &= \mathcal{L}_\beta \gamma_j - 2\alpha K_{ij}, \\ \Rightarrow d_t \gamma_j &= -2\alpha K, \end{aligned} \quad (\text{A.171})$$

$$\begin{aligned}\partial_t K &= -D^2 \alpha + \alpha [K_{ij} K^{ij} + 4\pi(S + \rho)] + \underbrace{\beta^i \partial_i K}_{=\mathcal{L}_\beta K}, \\ \Rightarrow d_t K &= -D^2 \alpha + \alpha [K_{ij} K^{ij} + 4\pi(S + \rho)]. \end{aligned} \quad (\text{A.172})$$

The LHS of Eqn. A.170 can now be written as

$$\begin{aligned}d_t K_{ij} &= d_t \left(A_{ij} + \frac{1}{3} \gamma_j K \right) = d_t A_{ij} + \frac{1}{3} \gamma_j d_t K + \frac{1}{3} K d_t \gamma_j \\ &= d_t A_{ij} + \frac{1}{3} \gamma_j (-D^2 \alpha + \alpha [\underbrace{K_{ij} K^{ij}}_{R+K^2-16\pi\rho} + 4\pi(S + \rho)]) + \frac{1}{3} K (-2\alpha K_{ij}) \\ &= d_t A_{ij} - \frac{1}{3} \gamma_j (D^2 \alpha - \alpha [R + K^2 + 4\pi(S - 3\rho)]) - \frac{2}{3} \alpha K K_{ij}. \end{aligned} \quad (\text{A.173})$$

So Eqn. A.170 can be rewritten as

$$\begin{aligned}d_t A_{ij} &= \frac{1}{3} \gamma_j D^2 \alpha - \frac{1}{3} \gamma_j \alpha R - \frac{1}{3} \gamma_j K^2 + \frac{4}{3} \pi \gamma_j \alpha (S - 3\rho) - \frac{2}{3} \alpha K K_{ij} - D_i D_j \alpha \\ &\quad + \alpha (R_{ij} + K K_{ij} - 2K_j^k K_{ik}) - 8\pi \alpha S_{ij} - 4\pi \alpha \gamma_j (S - \rho) \\ &= - \left(D_i D_j \alpha - \frac{1}{3} \gamma_j D^2 \alpha \right) + \alpha \left(R_{ij} - \frac{1}{3} \gamma_j R \right) - 8\pi \alpha \left(S_{ij} - \frac{1}{3} \gamma_j S \right) \\ &\quad + \alpha \left(\frac{5}{3} K K_{ij} - 2K_j^k K_{ik} - \frac{1}{3} \gamma_j K^2 \right), \end{aligned} \quad (\text{A.174})$$

where we can further split

$$\begin{aligned}\frac{5}{3} K K_{ij} - 2K_j^k K_{ik} - \frac{1}{3} \gamma_j K^2 &= \frac{5}{3} K \left(A_{ij} + \frac{1}{3} K \gamma_j \right) - 2 \left(A_{ik} + \frac{1}{3} \gamma_{ik} K \right) \left(A_j^k + \frac{1}{3} K \delta_j^k \right) - \frac{1}{3} \gamma_j K^2 \\ &= \frac{5}{3} K A_{ij} + \frac{5}{9} K^2 \gamma_j - 2 \left(A_{ik} A_j^k + \frac{1}{3} K \underbrace{A_{ik} \delta_j^k}_{A_{ij}} + \frac{1}{3} K \underbrace{\gamma_{ik} A_j^k}_{A_{ij}} + \frac{1}{9} \underbrace{\gamma_{ik} \delta_j^k}_{\gamma_j} K^2 \right) \\ &\quad - \frac{1}{3} \gamma_j K^2 \\ &= \frac{5}{3} K A_{ij} + \frac{5}{9} \gamma_j K^2 - 2A_{ik} A_j^k - \frac{4}{3} K A_{ij} - \frac{2}{9} \gamma_j K^2 - \frac{1}{3} \gamma_j K^2 \\ &= \frac{1}{3} K A_{ij} - 2A_{ik} A_j^k, \end{aligned} \quad (\text{A.175})$$

and we define the trace-free part of a tensor such that

$$D_i D_j \alpha - \frac{1}{3} \gamma_{ij} D^2 \alpha = (D_i D_j \alpha)^{TF}, \quad (\text{A.176})$$

$$R_{ij} - \frac{1}{3} \gamma_{ij} R = R_{ij}^{TF}, \quad (\text{A.177})$$

$$S_{ij} - \frac{1}{3} \gamma_{ij} S = S_{ij}^{TF}. \quad (\text{A.178})$$

Therefore, we get

$$d_t A_{ij} = -(D_i D_j \alpha)^{TF} + \alpha (R_{ij}^{TF} - 8\pi S_{ij}^{TF}) + \alpha \left(\frac{1}{3} K A_{ij} - 2A_{ik} A_j^k \right) \quad (\text{A.179})$$

We now perform the conformal transformation for A_{ij} . The LHS of the above equation gives

$$d_t (e^{4\phi} \tilde{A}_{ij}) = 4e^{4\phi} \tilde{A}_{ij} d_t \phi + e^{4\phi} d_t \tilde{A}_{ij}. \quad (\text{A.180})$$

From Eqn. A.165, we can get $d_t \phi$

$$\begin{aligned} d_t (e^{4\phi} \tilde{A}_{ij}) &= 4e^{4\phi} \tilde{A}_{ij} \left(\frac{1}{6} \partial_k \beta^k - \frac{1}{6} \alpha K \right) + e^{4\phi} d_t \tilde{A}_{ij} \\ &= e^{4\phi} d_t \tilde{A}_{ij} + \frac{2}{3} e^{4\phi} \tilde{A}_{ij} \partial_k \beta^k - \frac{2}{3} e^{4\phi} \alpha K \tilde{A}_{ij}. \end{aligned} \quad (\text{A.181})$$

For $A_{ik} A_j^k$, we have

$$\begin{aligned} A_{ik} A_j^k &= \gamma^{kl} A_{ik} A_{jl} = e^{-4\phi} \tilde{\gamma}^{kl} e^{4\phi} \tilde{A}_{ik} e^{4\phi} \tilde{A}_{jl} \\ &= e^{4\phi} \tilde{\gamma}^{kl} \tilde{A}_{ik} \tilde{A}_{jl} = e^{4\phi} \tilde{A}_{ik} \tilde{A}_j^k. \end{aligned} \quad (\text{A.182})$$

Combining these results, we get

$$\begin{aligned} e^{4\phi} d_t \tilde{A}_{ij} + \frac{2}{3} e^{4\phi} \tilde{A}_{ij} \partial_k \beta^k - \frac{2}{3} e^{4\phi} \alpha K \tilde{A}_{ij} &= -(D_i D_j \alpha)^{TF} + \alpha (R_{ij}^{TF} - 8\pi S_{ij}^{TF}), \\ &\quad + \alpha \left(\frac{1}{3} K e^{4\phi} \tilde{A}_{ij} - 2e^{4\phi} \tilde{A}_{ik} \tilde{A}_j^k \right) \\ \Rightarrow e^{4\phi} d_t \tilde{A}_{ij} &= -\frac{2}{3} e^{4\phi} \tilde{A}_{ij} \partial_k \beta^k - (D_i D_j \alpha)^{TF} + \alpha (R_{ij}^{TF} - 8\pi S_{ij}^{TF}) \\ &\quad + \alpha \left(K e^{4\phi} \tilde{A}_{ij} - 2e^{4\phi} \tilde{A}_{ik} \tilde{A}_j^k \right). \end{aligned} \quad (\text{A.183})$$

Multiplying both sides by $e^{-4\phi}$ and returning to $d_t = \partial_t - \mathcal{L}_\beta$, we have

$$\begin{aligned} \partial_t \tilde{A}_{ij} &= e^{-4\phi} [-(D_i D_j \alpha)^{TF} + \alpha(R_{ij}^{TF} - 8\pi S_{ij}^{TF})] + \alpha(K\tilde{A}_{ij} - 2\tilde{A}_{ik}\tilde{A}_j^k) + \mathcal{L}_\beta \tilde{A}_{ij} - \frac{2}{3}\tilde{A}_{ij}\partial_k \beta^k \\ &= e^{-4\phi} [-(D_i D_j \alpha)^{TF} + \alpha(R_{ij}^{TF} - 8\pi S_{ij}^{TF})] + \alpha(K\tilde{A}_{ij} - 2\tilde{A}_{ik}\tilde{A}_j^k) \\ &\quad + \beta^k \partial_k \tilde{A}_{ij} + \tilde{A}_{ik}\partial_j \beta^k + \tilde{A}_{kj}\partial_i \beta^k - \frac{2}{3}\tilde{A}_{ij}\partial_k \beta^k. \end{aligned} \quad (\text{A.184})$$

Note that the divergence of the shift $\partial_i \beta^i$ appears because the choice $\bar{\gamma} = 1$ makes ϕ a tensor density of weight 1/6, and $\tilde{\gamma}_{ij}$ and \tilde{A}_{ij} are tensor densities of weight $-2/3$.

Using Eqn. A.145, we can split the Ricci tensor into two terms

$$R_{ij} = \bar{R}_{ij} + R_{ij}^\phi, \quad (\text{A.185})$$

where R_{ij}^ϕ only depends on the conformal factor such that

$$R_{ij}^\phi = -2(\bar{D}_i \bar{D}_j \phi + \tilde{\gamma}_{ij} \tilde{\gamma}^{lm} \bar{D}_l \bar{D}_m \phi) + 4(\bar{D}_i \phi \bar{D}_j \phi - \tilde{\gamma}_{ij} \tilde{\gamma}^{lm} \bar{D}_l \phi \bar{D}_m \phi), \quad (\text{A.186})$$

and \bar{R}_{ij} is defined in terms of $\tilde{\gamma}_{ij}$ such that

$$\bar{R}_{ij} = \frac{1}{2} \tilde{\gamma}^{kl} (\partial_i \partial_l \tilde{\gamma}_{kj} + \partial_k \partial_j \tilde{\gamma}_{il} - \partial_i \partial_j \tilde{\gamma}_{kl} - \partial_k \partial_l \tilde{\gamma}_{ij}) + \tilde{\gamma}^{kl} (\bar{\Gamma}^m_{il} \bar{\Gamma}_{mkj} - \bar{\Gamma}^m_{ij} \bar{\Gamma}_{mkl}). \quad (\text{A.187})$$

In \bar{R}_{ij} , we still have the mixed second derivative terms that we are trying to avoid. To eliminate these, we define the conformal connection functions

$$\begin{aligned} \bar{\Gamma}^i &\equiv \tilde{\gamma}^{jk} \bar{\Gamma}^i_{jk} = \tilde{\gamma}^{jk} \frac{1}{2} \tilde{\gamma}^{il} (\partial_j \tilde{\gamma}_{lk} + \partial_k \tilde{\gamma}_{jl} - \partial_l \tilde{\gamma}_{jk}) \\ &= \frac{1}{2} (\tilde{\gamma}^{il} \tilde{\gamma}^{jk} \partial_j \tilde{\gamma}_{lk} + \tilde{\gamma}^{il} \tilde{\gamma}^{jk} \partial_k \tilde{\gamma}_{jl} - \underbrace{\tilde{\gamma}^{il} \tilde{\gamma}^{jk} \partial_l \tilde{\gamma}_{jk}}_{=0}) \\ &= \frac{1}{2} (\tilde{\gamma}^{il} \partial^k \tilde{\gamma}_{lk} + \underbrace{\tilde{\gamma}^{il} \partial^j \tilde{\gamma}_{jl}}_{\partial^k \tilde{\gamma}_{lk}}) = \tilde{\gamma}^{il} \partial^k \tilde{\gamma}_{lk} \\ &= \underbrace{\partial^k (\tilde{\gamma}^{il} \tilde{\gamma}_{lk})}_{=\partial^k \delta_k^i = 0} - \tilde{\gamma}_{lk} \partial^k \tilde{\gamma}^{il} = -\partial_l \tilde{\gamma}^{il}. \end{aligned} \quad (\text{A.188})$$

Using these, we can express the Ricci tensor \bar{R}_{ij} in terms of these conformal connection functions. First, consider the following term

$$\begin{aligned} \partial_i \partial_l (\tilde{\gamma}^{kl} \tilde{\gamma}_{kj}) &= \partial_i (\tilde{\gamma}^{kl} \partial_l \tilde{\gamma}_{kj} + \tilde{\gamma}_{kj} \partial_l \tilde{\gamma}^{kl}) \\ &= \partial_i \tilde{\gamma}_{kj} \partial_l \tilde{\gamma}^{kl} + \tilde{\gamma}^{kl} \partial_i \partial_l \tilde{\gamma}_{kj} + \partial_i \tilde{\gamma}^{kl} \partial_l \tilde{\gamma}_{kj} + \tilde{\gamma}_{kj} \partial_i \partial_l \tilde{\gamma}^{kl}, \end{aligned} \quad (\text{A.189})$$

Because the LHS of Eqn. A.189 is zero so we have

$$\begin{aligned}\bar{\gamma}^{kl} \partial_i \partial_l \bar{\gamma}_{kj} &= -\partial_i \bar{\gamma}_{kj} \underbrace{\partial_l \bar{\gamma}^{kl}}_{-\bar{\Gamma}^k} - \partial_i \bar{\gamma}^{kl} \partial_l \bar{\gamma}_{kj} - \bar{\gamma}_{kj} \partial_i \underbrace{\partial_l \bar{\gamma}^{kl}}_{\bar{\Gamma}^k} \\ &= \partial_i \bar{\gamma}_{kj} \bar{\Gamma}^k + \bar{\gamma}_{kj} \partial_i \bar{\Gamma}^k - \partial_i \bar{\gamma}^{kl} \partial_l \bar{\gamma}_{kj}.\end{aligned}\quad (\text{A.190})$$

Using this, we can rewrite Eqn. A.187

$$\begin{aligned}\bar{R}_{ij} &= -\frac{1}{2} \bar{\gamma}^{kl} \partial_k \partial_l \bar{\gamma}_{ij} + \frac{1}{2} (\partial_i \bar{\gamma}_{kj} \bar{\Gamma}^k + \bar{\gamma}_{kj} \partial_i \bar{\Gamma}^k - \partial_i \bar{\gamma}^{kl} \partial_l \bar{\gamma}_{kj}) + \frac{1}{2} (\partial_j \bar{\gamma}_{il} \bar{\Gamma}^l + \bar{\gamma}_{il} \partial_j \bar{\Gamma}^l - \partial_k \bar{\gamma}^{il} \partial_j \bar{\gamma}_{kl}) \\ &\quad - \frac{1}{2} \bar{\gamma}^{kl} \partial_i \partial_j \bar{\gamma}_{kl} + \bar{\gamma}^{kl} \bar{\Gamma}^m_{il} \bar{\Gamma}^m_{mkj} - \bar{\gamma}^{kl} \bar{\Gamma}^m_{ij} \bar{\Gamma}^m_{mkl}.\end{aligned}\quad (\text{A.191})$$

The last term in the above equation can be written

$$\bar{\gamma}^{kl} \bar{\Gamma}^m_{ij} \bar{\Gamma}^m_{mkl} = \bar{\gamma}^{kl} \bar{\Gamma}^m_{kl} \bar{\Gamma}^m_{mij} = -\bar{\Gamma}^m \bar{\Gamma}^m_{mij}.\quad (\text{A.192})$$

Rearranging dummy indices and collecting terms yields

$$\begin{aligned}\bar{R}_{ij} &= -\frac{1}{2} \bar{\gamma}^{kl} \partial_k \partial_l \bar{\gamma}_{ij} + \frac{1}{2} (\bar{\gamma}_{kj} \partial_i \bar{\Gamma}^k + \bar{\gamma}_{kj} \partial_j \bar{\Gamma}^k) + \left(\frac{1}{2} \partial_i \bar{\gamma}_{kj} + \frac{1}{2} \partial_j \bar{\gamma}_{ik} + \bar{\Gamma}^k_{kij} \right) \bar{\Gamma}^k \\ &\quad + \bar{\gamma}^{kl} \bar{\Gamma}^m_{il} \bar{\Gamma}^m_{mkj} - \frac{1}{2} (\partial_i \bar{\gamma}^{kl} \partial_l \bar{\gamma}_{kj} + \partial_k \bar{\gamma}^{il} \partial_j \bar{\gamma}_{kl} + \bar{\gamma}^{kl} \partial_i \partial_j \bar{\gamma}_{kl}).\end{aligned}\quad (\text{A.193})$$

After some additional algebra we can conclude

$$\bar{R}_{ij} = -\frac{1}{2} \bar{\gamma}^{lm} \partial_m \partial_l \bar{\gamma}_{ij} + \bar{\gamma}_{k(i} \partial_j) \bar{\Gamma}^k + \bar{\Gamma}^k \bar{\Gamma}^m_{(ij)k} + \bar{\gamma}^{lm} (2\bar{\Gamma}^k_{l(i} \bar{\Gamma}^m_{j)km} + \bar{\Gamma}^k_{im} \bar{\Gamma}^m_{klj}).\quad (\text{A.194})$$

Because we are treating $\bar{\Gamma}^i$ as a new independent function, we have to find the evolution equation for it. Taking its time derivative gives

$$\partial_t \bar{\Gamma}^i = \partial_t (-\partial_j \bar{\gamma}^{ij}) = -\partial_j (\partial_t \bar{\gamma}^{ij}).\quad (\text{A.195})$$

Here we interchange a partial time and spatial derivative because they commute. Now we need to evaluate $\partial_t \bar{\gamma}^{ij}$. From Eqn. A.166 we have

$$\begin{aligned}\partial_t \bar{\gamma}^{ij} &= -\bar{\gamma}^{ik} \bar{\gamma}^{jl} \partial_t \bar{\gamma}_{kl} \\ &= -\bar{\gamma}^{ik} \bar{\gamma}^{jl} \left(-2\alpha \tilde{A}_{kl} + \beta^m \partial_m \bar{\gamma}_{kl} + \bar{\gamma}_{ml} \partial_k \beta^m + \bar{\gamma}_{mk} \partial_l \beta^m - \frac{2}{3} \bar{\gamma}_{kl} \partial_m \beta^m \right) \\ &= 2\alpha \underbrace{\bar{\gamma}^{ik} \bar{\gamma}^{jl} \tilde{A}_{kl}}_{=\tilde{A}^{ij}} - \beta^m \underbrace{\bar{\gamma}^{ik} \bar{\gamma}^{jl} \partial_m \bar{\gamma}_{kl}}_{=-\partial_m \bar{\gamma}^{ij}} - \bar{\gamma}^{ik} \bar{\gamma}^{jl} \bar{\gamma}_{ml} \partial_k \beta^m - \bar{\gamma}^{ik} \bar{\gamma}^{jl} \bar{\gamma}_{mk} \partial_l \beta^m + \frac{2}{3} \underbrace{\bar{\gamma}^{ik} \bar{\gamma}^{jl} \bar{\gamma}_{kl}}_{=\bar{\gamma}^{ij}} \partial_m \beta^m.\end{aligned}\quad (\text{A.196})$$

We perform some re-arrangement of metrics and indices to simplify further

$$\begin{aligned}\bar{\gamma}^{ik}\bar{\gamma}^{jl}\bar{\gamma}_{ml}\partial_k\beta^m &= \delta_m^j\partial^i\beta^m = \delta_m^j\partial^i(\delta_i^m\beta^i) = \delta_m^j\delta_i^m\partial^i\beta^i = \delta_m^j\delta_i^m\bar{\gamma}^{im}\partial_m\beta^i \\ &= \bar{\gamma}^{jm}\partial_m\beta^i.\end{aligned}\quad (\text{A.197})$$

Similarly, we obtain

$$\begin{aligned}\bar{\gamma}^{ik}\bar{\gamma}^{jl}\bar{\gamma}_{mk}\partial_l\beta^m &= \delta_m^i\partial^j\beta^m = \delta_m^i\partial^j(\delta_j^m\beta^j) = \delta_m^i\delta_j^m\partial^j\beta^j = \delta_m^i\delta_j^m\bar{\gamma}^{jm}\partial_m\beta^j \\ &= \bar{\gamma}^{im}\partial_m\beta^j.\end{aligned}\quad (\text{A.198})$$

Thus $\partial_t\bar{\gamma}^{ij}$ is

$$\partial_t\bar{\gamma}^{ij} = 2\alpha\tilde{A}^{ij} - \bar{\gamma}^{mj}\partial_m\beta^i - \bar{\gamma}^{mi}\partial_m\beta^j + \frac{2}{3}\bar{\gamma}^{jj}\partial_l\beta^l + \beta^l\partial_l\bar{\gamma}^{ij}, \quad (\text{A.199})$$

where we relabel some dummy indices. So $\partial_t\bar{\Gamma}^i$ is

$$\begin{aligned}\partial_t\bar{\Gamma}^i &= -\partial_j\left(2\alpha\tilde{A}^{ij} - \bar{\gamma}^{mj}\partial_m\beta^i - \bar{\gamma}^{mi}\partial_m\beta^j + \frac{2}{3}\bar{\gamma}^{jj}\partial_l\beta^l + \beta^l\partial_l\bar{\gamma}^{ij}\right) \\ &= -2\tilde{A}^{ij}\partial_j\alpha - 2\alpha\partial_j\tilde{A}^{ij} + \underbrace{\partial_j\bar{\gamma}^{mj}}_{=-\bar{\Gamma}^m}\partial_m\beta^i + \bar{\gamma}^{mj}\partial_j\partial_m\beta^i + \partial_j\bar{\gamma}^{mi}\partial_m\beta^j + \underbrace{\bar{\gamma}^{mi}\partial_j\partial_m\beta^j}_{=\bar{\gamma}^{ji}\partial_j\partial_l\beta^l} \\ &\quad - \frac{2}{3}\underbrace{\partial_j\bar{\gamma}^{jj}}_{=-\bar{\Gamma}^i}\partial_l\beta^l - \frac{2}{3}\bar{\gamma}^{jj}\partial_j\partial_l\beta^l - \partial_j\beta^l\partial_l\bar{\gamma}^{ij} - \underbrace{\beta^l\partial_j\partial_l\bar{\gamma}^{ij}}_{=\partial_m\beta^j\partial_j\bar{\gamma}^{im}},\end{aligned}\quad (\text{A.200})$$

where we apply the definition of $\bar{\Gamma}^i$. Again, on relabeling indices and simplifying

$$\partial_t\bar{\Gamma}^i = -2(\alpha\partial_j\tilde{A}^{ij} + \tilde{A}^{ij}\partial_j\alpha) - \bar{\Gamma}^j\partial_j\beta^i + \bar{\gamma}^{jj}\partial_l\partial_j\beta^j + \frac{1}{3}\bar{\gamma}^{ii}\partial_l\partial_j\beta^j + \frac{2}{3}\bar{\Gamma}^i\partial_j\beta^j + \beta^j\partial_j\bar{\Gamma}^i. \quad (\text{A.201})$$

The divergence of \tilde{A}^{ij} can be simplified by using the momentum constraint. From Eqn. A.114, we have

$$D_j(K^{ij} - \gamma^{ij}K) = 8\pi S^i. \quad (\text{A.202})$$

Splitting the extrinsic curvature via $K^{ij} + \gamma^{ij}K/3$ gives

$$D_j\left(A^{ij} - \frac{2}{3}\gamma^{ij}K\right) = D_jA^{ij} - \frac{2}{3}\gamma^{ij}D_jK = 8\pi S^i. \quad (\text{A.203})$$

We need to find a conformal relation between $D_j A^{ij}$ and $\bar{D}_j \tilde{A}^{ij}$. From the conformal transformation section, we know the relation between $D_j A^{ij}$ and $\bar{D}_j A^{ij}$

$$D_j A^{ij} = \bar{D}_j A^{ij} + C^i_{jk} A^{kj} + C^j_{jk} A^{ik}, \quad (\text{A.204})$$

where again $C^i_{jk} \equiv \Gamma^i_{jk} - \bar{\Gamma}^i_{jk}$ and $\psi = e^\phi$

$$C^i_{jk} = 2(\delta_j^i \bar{D}_k \phi + \delta_k^i \bar{D}_j \phi - \bar{\gamma}_{jk} \bar{\gamma}^{il} \bar{D}_l \phi), \quad (\text{A.205})$$

so $C^j_{jk} = 6\bar{D}_k \phi$. Substituting these into Eqn. A.204

$$\begin{aligned} D_j A^{ij} &= \bar{D}_j A^{ij} + 2(\delta_j^i \bar{D}_k \phi + \delta_k^i \bar{D}_j \phi - \bar{\gamma}_{jk} \bar{\gamma}^{il} \bar{D}_l \phi) A^{kj} + 6A^{ik} \bar{D}_k \phi \\ &= \bar{D}_j A^{ij} + 2(\delta_j^i A^{kj} \bar{D}_k \phi + \delta_k^i A^{kj} \bar{D}_j \phi - \underbrace{A^{kj} \bar{\gamma}_{jk} \bar{\gamma}^{il} \bar{D}_l \phi}_{=0}) + 6A^{ik} \bar{D}_k \phi \\ &= \bar{D}_j A^{ij} + 2(A^{ik} \bar{D}_k \phi + A^{ij} \bar{D}_j \phi) + 6A^{ik} \bar{D}_k \phi = \bar{D}_j A^{ij} + 10A^{ij} \bar{D}_j \phi. \end{aligned} \quad (\text{A.206})$$

Note that we can further simplify $\bar{D}_j A^{ij} + 10A^{ij} \bar{D}_j \phi = e^{-10\phi} \bar{D}_j (e^{10\phi} A^{ij})$. From this relation, we can rescale A^{ij} as $A^{ij} = e^{-10\phi} \tilde{A}^{ij}$ and this is widely used in solving the momentum constraints. Here we keep our choice $A^{ij} = e^{-4\phi} \tilde{A}^{ij}$

$$\begin{aligned} D_j A^{ij} &= \bar{D}_j (e^{-4\phi} \tilde{A}^{ij}) + 10e^{-4\phi} \tilde{A}^{ij} \bar{D}_j \phi \\ &= e^{-4\phi} \bar{D}_j \tilde{A}^{ij} - 4e^{-4\phi} \tilde{A}^{ij} \bar{D}_j \phi + 10e^{-4\phi} \tilde{A}^{ij} \bar{D}_j \phi \\ &= e^{-4\phi} (\partial_j \tilde{A}^{ij} + \bar{\Gamma}^i_{jk} \tilde{A}^{kj} + \bar{\Gamma}^j_{jk} \tilde{A}^{ij} + 6e^{-4\phi} \tilde{A}^{ij} \bar{D}_j \phi). \end{aligned} \quad (\text{A.207})$$

Using the following formula

$$\bar{\Gamma}^j_{jk} = \frac{1}{2} \bar{\gamma}^{lm} \partial_k \bar{\gamma}_{lm} = \frac{1}{2\bar{\gamma}} \partial_k \bar{\gamma} = 0, \quad (\text{A.208})$$

we obtain

$$D_j A^{ij} = e^{-4\phi} (\partial_j \tilde{A}^{ij} + \bar{\Gamma}^i_{jk} \tilde{A}^{kj} + 6\tilde{A}^{ij} \partial_j \phi). \quad (\text{A.209})$$

Substituting this into the momentum constraint gives

$$\begin{aligned} D_j A^{ij} - \frac{2}{3} \gamma^{ij} D_j K &= e^{-4\phi} (\partial_j \tilde{A}^{ij} + \bar{\Gamma}^i_{jk} \tilde{A}^{kj} + 6\tilde{A}^{ij} \partial_j \phi) - \frac{2}{3} e^{-4\phi} \bar{\gamma}^{ij} D_j K = 8\pi S^i \\ \Rightarrow e^{-4\phi} (\partial_j \tilde{A}^{ij} + \bar{\Gamma}^i_{jk} \tilde{A}^{kj} + 6\tilde{A}^{ij} \partial_j \phi) - \frac{2}{3} e^{-4\phi} \bar{\gamma}^{ij} D_j K &= 8\pi S^i = 8\pi \gamma^{ij} S_j = 8\pi e^{-4\phi} \bar{\gamma}^{ij} S_j. \end{aligned} \quad (\text{A.210})$$

Multiplying both sides by $e^{4\phi}$ and replacing covariant derivatives of ϕ and K with partial derivatives then $\partial_j \tilde{A}^{ij}$ can be expressed

$$\partial_j \tilde{A}^{ij} = -\bar{\Gamma}^i{}_{jk} \tilde{A}^{kj} - 6\tilde{A}^{ij} \partial_j \phi + \frac{2}{3} \bar{\gamma}^{jj} \partial_j K + 8\pi \bar{\gamma}^{ij} S_j. \quad (\text{A.211})$$

Substitute this into Eqn. A.201

$$\begin{aligned} \partial_t \bar{\Gamma}^i = & -2\tilde{A}^{ij} \partial_j \alpha + 2\alpha \left(\bar{\Gamma}^i{}_{jk} \tilde{A}^{kj} + 6\tilde{A}^{ij} \partial_j \phi - \frac{2}{3} \bar{\gamma}^{jj} \partial_j K - 8\pi \bar{\gamma}^{ij} S_j \right) \\ & + \beta^j \partial_j \bar{\Gamma}^i - \bar{\Gamma}^j \partial_j \beta^i + \frac{2}{3} \bar{\Gamma}^i \partial_j \beta^j + \bar{\gamma}^{lj} \partial_l \partial_j \beta^i + \frac{1}{3} \bar{\gamma}^{li} \partial_l \partial_j \beta^j. \end{aligned} \quad (\text{A.212})$$

Eqns. A.165, A.166, A.169, A.184, and A.212 are called the BSSN equations. In summary, we have

$$\partial_t \phi = -\frac{1}{6} \alpha K + \beta^k \partial_k \phi + \frac{1}{6} \partial_k \beta^k, \quad (\text{A.213})$$

$$\partial_t \bar{\gamma}_{ij} = -2\alpha \tilde{A}_{ij} + \beta^k \partial_k \bar{\gamma}_{ij} + \bar{\gamma}_{ik} \partial_j \beta^k + \bar{\gamma}_{kj} \partial_i \beta^k - \frac{2}{3} \bar{\gamma}_{ij} \partial_k \beta^k, \quad (\text{A.214})$$

$$\partial_t K = -\gamma^{jj} D_j D_j \alpha + \alpha \left(\tilde{A}^{ij} \tilde{A}_{ij} + \frac{1}{3} K^2 \right) + 4\pi \alpha (\rho + S) + \beta^i \partial_i K, \quad (\text{A.215})$$

$$\begin{aligned} \partial_t \tilde{A}_{ij} = & e^{-4\phi} \left[-(D_i D_j \alpha)^{TF} + \alpha (R_{ij}^{TF} - 8\pi S_{ij}^{TF}) \right] + \alpha \left(K \tilde{A}_{ij} - 2\tilde{A}_{ik} \tilde{A}_j^k \right) \\ & + \beta^k \partial_k \tilde{A}_{ij} + \tilde{A}_{ik} \partial_j \beta^k + \tilde{A}_{kj} \partial_i \beta^k - \frac{2}{3} \tilde{A}_{ij} \partial_k \beta^k, \end{aligned} \quad (\text{A.216})$$

$$\begin{aligned} \partial_t \bar{\Gamma}^i = & -2\tilde{A}^{ij} \partial_j \alpha + 2\alpha \left(\bar{\Gamma}^i{}_{jk} \tilde{A}^{kj} + 6\tilde{A}^{ij} \partial_j \phi - \frac{2}{3} \bar{\gamma}^{jj} \partial_j K - 8\pi \bar{\gamma}^{ij} S_j \right) \\ & + \beta^j \partial_j \bar{\Gamma}^i - \bar{\Gamma}^j \partial_j \beta^i + \frac{2}{3} \bar{\Gamma}^i \partial_j \beta^j + \bar{\gamma}^{lj} \partial_l \partial_j \beta^i + \frac{1}{3} \bar{\gamma}^{li} \partial_l \partial_j \beta^j. \end{aligned} \quad (\text{A.217})$$

The ADM equations have (γ_{ij}, K_{ij}) as evolved variables and the BSSN equations split these variables into $(\phi, \bar{\gamma}_{ij}, K, \tilde{A}_{ij}, \bar{\Gamma}^i)$. Note that there are variants of the BSSN formulation. For example, people set $\chi = e^{-4\phi}$ so that (this is used in order to regularize the system near puncture)

$$\partial_a \chi = -4e^{-4\phi} \partial_a \phi = -4\chi \partial_a \phi. \Rightarrow \partial_a \phi = -\frac{1}{4\chi} \partial_a \chi \quad (\text{A.218})$$

So, Eqn. A.165 is changed

$$-\frac{1}{4\chi}\partial_t\chi = -\frac{1}{6}\alpha K + \beta^k\left(-\frac{1}{4\chi}\partial_k\chi\right) + \frac{1}{6}\partial_k\beta^k, \quad (\text{A.219})$$

$$\Rightarrow \partial_t\chi = \frac{2}{3}\chi(\alpha K - \partial_k\beta^k) + \beta^k\partial_k\chi. \quad (\text{A.220})$$

There are also minor changes to the equations for \tilde{A}_{ij} and $\bar{\Gamma}^i$. The χ method is used for many binary black hole simulations. (Our code is also based on this.) The BSSN equations for the black hole case, i.e. the vacuum Einstein equations, with this method are

$$\partial_t\chi = \frac{2}{3}\chi(\alpha K - \partial_k\beta^k) + \beta^k\partial_k\chi, \quad (\text{A.221})$$

$$\partial_t\tilde{\gamma}_{ij} = -2\alpha\tilde{A}_{ij} + \beta^k\partial_k\tilde{\gamma}_{ij} + \tilde{\gamma}_{ik}\partial_j\beta^k + \tilde{\gamma}_{kj}\partial_i\beta^k - \frac{2}{3}\tilde{\gamma}_{ij}\partial_k\beta^k, \quad (\text{A.222})$$

$$\partial_t K = -\gamma^{ij}D_jD_i\alpha + \alpha\left(\tilde{A}^{ij}\tilde{A}_{ij} + \frac{1}{3}K^2\right) + \beta^i\partial_i K, \quad (\text{A.223})$$

$$\begin{aligned} \partial_t\tilde{A}_{ij} = & \chi\left[-(D_iD_j\alpha)^{TF} + \alpha R_{ij}^{TF}\right] + \alpha\left(K\tilde{A}_{ij} - 2\tilde{A}_{ik}\tilde{A}_j^k\right) \\ & + \beta^k\partial_k\tilde{A}_{ij} + \tilde{A}_{ik}\partial_j\beta^k + \tilde{A}_{kj}\partial_i\beta^k - \frac{2}{3}\tilde{A}_{ij}\partial_k\beta^k, \end{aligned} \quad (\text{A.224})$$

$$\begin{aligned} \partial_t\bar{\Gamma}^i = & -2\tilde{A}^{ij}\partial_j\alpha + 2\alpha\left(\bar{\Gamma}^i{}_{jk}\tilde{A}^{kj} - \frac{3}{2\chi}\tilde{A}^{ij}\partial_j\chi - \frac{2}{3}\tilde{\gamma}^{ij}\partial_j K\right) \\ & + \beta^j\partial_j\bar{\Gamma}^i - \bar{\Gamma}^j\partial_j\beta^i + \frac{2}{3}\bar{\Gamma}^i\partial_j\beta^j + \tilde{\gamma}^{ij}\partial_l\partial_j\beta^j + \frac{1}{3}\tilde{\gamma}^{ij}\partial_l\partial_j\beta^j. \end{aligned} \quad (\text{A.225})$$

These are a relatively standard form of the BSSN equations for the binary black hole problem.

A.4 The Generalized Harmonic Formulation

There is an alternative hyperbolic formalism for the Einstein equations, the so called generalized harmonic formalism. The idea is to generalize of the harmonic coordinate condition in the form

$$\square x^a = H^a, \quad (\text{A.226})$$

where \square is the usual wave operator, x^a is coordinate, and H^a is an arbitrary source function.

Starting from Eqn. A.226, we can have

$$H^a = \square x^a = \frac{1}{\sqrt{-g}} \partial_c (\sqrt{-g} g^{cb} \underbrace{\partial_b x^a}_{=\delta_b^a}) = \frac{1}{\sqrt{-g}} \partial_c (\sqrt{-g} g^{ca}), \quad (\text{A.227})$$

therefore

$$\begin{aligned} H_a &= g_{ab} H^b = g_{ab} \frac{1}{\sqrt{-g}} \partial_c (\sqrt{-g} g^{cb}) \\ &= g_{ab} g^{cb} \underbrace{\frac{1}{\sqrt{-g}} \partial_c \sqrt{-g}}_{=\partial_c \ln \sqrt{-g}} + g_{ab} \frac{1}{\sqrt{-g}} \sqrt{-g} \partial_c g^{cb} \\ &= \partial_a \ln \sqrt{-g} + g_{ab} \partial_c g^{cb}. \end{aligned} \quad (\text{A.228})$$

Using the fact that $g_{ab} \partial_c g^{cb} = -g^{cb} \partial_c g_{ab}$ we can also write

$$H_a = \partial_a \ln \sqrt{-g} - g^{cb} \partial_c g_{ab}. \quad (\text{A.229})$$

On taking the gradient

$$\partial_b H_a = \partial_b \partial_a \ln \sqrt{-g} - \partial_b g^{cd} \partial_c g_{ad} - g^{cd} \partial_b \partial_c g_{ad}, \quad (\text{A.230})$$

and symmetrizing

$$H_{(a,b)} = (\ln \sqrt{-g})_{,ab} - g^{cd} g_{c(a,b)d} - g^{cd} {}_{(,a} g_{b)c,d}, \quad (\text{A.231})$$

where $f_{,a} = \partial_a f$ which is usual partial derivative

On considering the Einstein equation

$$R_{ab} - \frac{1}{2} g_{ab} R = 8\pi T_{ab}. \quad (\text{A.232})$$

we rewrite this into its trace reversed form

$$R_{ab} = 4\pi(2T_{ab} - g_{ab} T). \quad (\text{A.233})$$

In vacuum, we need only consider the Ricci tensor decomposition. This gets expressed as

$$R_{ab} = \Gamma^c{}_{ab,c} - \Gamma^c{}_{cb,a} + \Gamma^d{}_{ab} \Gamma^c{}_{dc} - \Gamma^d{}_{cb} \Gamma^c{}_{da}. \quad (\text{A.234})$$

Using the definition of the connection coefficients, we can express the Ricci tensor in terms of the metric. Evaluating term by term

$$\begin{aligned}\Gamma^c_{ab,c} &= \partial_c \left[\frac{1}{2} g^{cd} (\partial_a g_{db} + \partial_b g_{da} - \partial_d g_{ab}) \right] \\ &= \frac{1}{2} \partial_c g^{cd} (\partial_a g_{db} + \partial_b g_{da} - \partial_d g_{ab}) + \frac{1}{2} g^{cd} (\partial_c \partial_a g_{db} + \partial_c \partial_b g_{da} - \partial_c \partial_d g_{ab}).\end{aligned}\quad (\text{A.235})$$

Using the identity $\Gamma^a_{ab} = \partial_b \ln \sqrt{-g}$, the second and third terms of Eqn. A.234 can be expressed as

$$\Gamma^c_{cb,a} = \partial_a \partial_b \ln \sqrt{-g}, \quad (\text{A.236})$$

$$\Gamma^d_{ab} \Gamma^c_{dc} = \partial_d \ln \sqrt{-g} \Gamma^d_{ab}. \quad (\text{A.237})$$

Substituting all of these into Eqn. A.234 gives

$$\begin{aligned}R_{ab} &= \frac{1}{2} \partial_c g^{cd} (\partial_a g_{db} + \partial_b g_{da} - \partial_d g_{ab}) + \frac{1}{2} g^{cd} (\partial_c \partial_a g_{db} + \partial_c \partial_b g_{da} - \partial_c \partial_d g_{ab}) \\ &\quad - \partial_a \partial_b \ln \sqrt{-g} + \partial_d \ln \sqrt{-g} \Gamma^d_{ab} + \Gamma^d_{cb} \Gamma^c_{da} \\ &= -\frac{1}{2} g^{cd} \partial_c \partial_d g_{ab} + \underbrace{\frac{1}{2} g^{cd} (\partial_c \partial_a g_{db} + \partial_c \partial_b g_{da})}_{=g^{cd} \partial_c \partial_{(a} g_{b)d} = g^{cd} g_{c(a,b)d}} + \underbrace{\partial_c g^{cd} \frac{1}{2} (\partial_a g_{db} + \partial_b g_{da} - \partial_d g_{ab})}_{=\Gamma_{dab}} \\ &\quad - \partial_a \partial_b \ln \sqrt{-g} + \partial_d \ln \sqrt{-g} \Gamma^d_{ab} + \Gamma^d_{cb} \Gamma^c_{da} \\ &= -\frac{1}{2} g^{cd} g_{ab,cd} + g^{cd}{}_{,c} \Gamma_{dab} - (\ln \sqrt{-g})_{,ab} + g^{cd} g_{c(a,b)d} + (\ln \sqrt{-g})_{,d} \Gamma^d_{ab} + \Gamma^d_{cb} \Gamma^c_{da}.\end{aligned}\quad (\text{A.238})$$

Using Eqn. A.231, we have

$$-(\ln \sqrt{-g})_{,ab} + g^{cd} g_{c(a,b)d} = -H_{(a,b)} - g^{cd}{}_{(a} g_{b)c,d}. \quad (\text{A.239})$$

On using Eqn. A.228, we can rewrite

$$(\ln \sqrt{-g})_{,d} \Gamma^d_{ab} = (H_d - g_{de} g^{ce}{}_{,c}) \Gamma^d_{ab} = H_d \Gamma^d_{ab} - g^{ce}{}_{,c} \Gamma_{eab}. \quad (\text{A.240})$$

Substituting these into Eqn A.238 we get

$$\begin{aligned}FR_{ab} &= -\frac{1}{2} g^{cd} g_{ab,cd} + g^{cd}{}_{,c} \Gamma_{dab} - H_{(a,b)} - g^{cd}{}_{(a} g_{b)c,d} + H_d \Gamma^d_{ab} - \underbrace{g^{ce}{}_{,c} \Gamma_{eab}}_{=g^{cd}{}_{,c} \Gamma_{dab}} + \Gamma^d_{cb} \Gamma^c_{da} \\ &= -\frac{1}{2} g^{cd} g_{ab,cd} - H_{(a,b)} - g^{cd}{}_{(a} g_{b)c,d} + H_d \Gamma^d_{ab} + \Gamma^d_{cb} \Gamma^c_{da}.\end{aligned}\quad (\text{A.241})$$

Thus, the Einstein equations can be written as

$$-\frac{1}{2}g^{cd}g_{ab,cd} - H_{(a,b)} - g^{cd}{}_{(,a}g_{b)c,d} + H_d\Gamma^d{}_{ab} + \Gamma^d{}_{cb}\Gamma^c{}_{da} = 4\pi(2T_{ab} - g_{ab}T), \quad (\text{A.242})$$

or more explicitly

$$g^{cd}\partial_c\partial_d g_{ab} + \partial_a g^{cd}\partial_d g_{bc} + \partial_b g^{cd}\partial_d g_{ac} + 2\partial_{(b}H_{a)} - 2H_d\Gamma^d{}_{ab} + \Gamma^d{}_{cb}\Gamma^c{}_{da} = -8\pi(2T_{ab} - g_{ab}T). \quad (\text{A.243})$$

A.5 The 3+1 Decomposition for Other Fields

The techniques that we have described so far can be applied to other quantities in addition to the metric, including scalar fields, $U(1)$ gauge fields, and fluids. In this section, we present an example of the 3+1 decomposition for some of these other fields which are part of EMDA and which are discussed in Sec. 2.2

A.5.1 Decomposition of the dilaton field, ϕ

We will start with the equation of motion for the dilaton, ϕ .

$$\nabla_a\nabla^a\phi = -\frac{1}{2}\alpha_0 e^{-2\alpha_0\phi}F^2 + \frac{1}{2}\alpha_0 e^{4\alpha_1\phi}(\nabla\kappa)^2. \quad (\text{A.244})$$

First, we consider the LHS of the equation

$$\begin{aligned} \nabla_a\nabla^a\phi &= g^b{}_a\nabla_b(g^{ac}\nabla_c\phi) \\ &= (\gamma^b{}_a - n_a n^b)\nabla_b((\gamma^{ac} - n^a n^c)\nabla_c\phi) \\ &= \gamma^b{}_a\nabla_b(\gamma^{ac}\nabla_c\phi) - n_a n^b\nabla_b(\gamma^{ac}\nabla_c\phi) - \gamma^b{}_a\nabla_b(n^a n^c\nabla_c\phi) + n_a n^b\nabla_b(n^a n^c\nabla_c\phi). \end{aligned} \quad (\text{A.245})$$

Note that the derivatives of the physical field are being projected into and normal to the spacelike hypersurfaces, Σ . Using this, we will define a new variable $\Pi = -n^a\nabla_a\phi$. This, of course, is the time derivative (the velocity, essentially) of the field as measured by an observer following the world line of n^a . Knowing that γ_{ab} lives in Σ , the quantity $\gamma^{ab}\nabla_b$

turns out to simply be the 3D covariant derivative, or gradient, which we will call D^a and which lives on the slices Σ . Using these definitions, we rewrite the equation for ϕ as

$$\nabla_a \nabla^a \phi = D_a(D^a \phi) - n_a n^b \nabla_b(D^a \phi) - \gamma^b{}_a \nabla_b(-n^a \Pi) + n_a n^b \nabla_b(-n^a \Pi). \quad (\text{A.246})$$

The first term on the right hand side is simply the spatial Laplacian while the remaining pieces all include some kind of time component. The second term on the right hand side of Eqn. A.246, can be rewritten as $n_a n^b \nabla_b(D^a \phi) = n^b \nabla(n^a D_a \phi) - (n^b \nabla_b n_a) D^a \phi$. As D_a is defined as a spatial gradient living wholly on Σ and n_a is orthogonal to Σ , clearly $n^a D_a \phi = 0$. Using the definition of n_a , $n^b \nabla_b n_a = D_a \ln \alpha$. So we have $n_a n^b \nabla_b(D^a \phi) = -D_a \ln \alpha D^a \phi$. Turning to the third term on the RHS of Eqn. A.246, we see that

$$\gamma^b{}_a \nabla_b(n^a \Pi) = \gamma^b{}_a n^a (\nabla_b \Pi) + \gamma^b{}_a (\nabla_b n^a) \Pi = \gamma^b{}_a (-K^a{}_b) \Pi = -K \Pi,$$

where we have used the fact that $\gamma^b{}_a n^a = 0$ and the definition of the extrinsic curvature $K_{ab} = -\gamma_a^c \gamma_b^d \nabla_c n_d$ and that its trace is $K = \gamma^{ab} K_{ab}$.

The final term on the RHS in Eqn. A.246 is

$$n_a n^b \nabla_b(n^a \Pi) = n_a n^b (\nabla_b n_a) \Pi + n_a n^a n^b \nabla_b \Pi = n^b \nabla_b \Pi, \quad (\text{A.247})$$

where we have used $n_a n^a = -1$ and the orthogonality between n_a and the extrinsic curvature living on Σ . Using all of these, the LHS of equation for ϕ can be written as

$$\nabla_a \nabla^a \phi = n^a \nabla_a \Pi + \frac{1}{\alpha} D_a(\alpha D^a \phi) - K \Pi. \quad (\text{A.248})$$

In order to consider the RHS of the dilaton equation, we must also consider the 3 + 1 split of the Maxwell tensor, F_{ab} . Of course, we follow an analogous approach in that we project along both free indices:

$$\begin{aligned} F_{ab} &= g_a^c g_b^d F_{cd} = (\gamma_a^c - n_a n^c)(\gamma_b^d - n_b n^d) F_{cd} \\ &= \gamma_a^c \gamma_b^d F_{cd} - n_a n^c F_{cd} \gamma_b^d - \gamma_a^c F_{cd} n_b n^d + n_a n^c n_b n^d F_{cd}. \end{aligned} \quad (\text{A.249})$$

The fourth term on the RHS vanishes by the antisymmetry of F_{cd} . We define the quantity $\perp E_a = \gamma_a^b F_{bc} n^c$ and recognize it as the electric 3-vector measured by an observer moving

normal to the spatial hypersurfaces. We also make the definition $\varepsilon_{abc}B^c \equiv \gamma_a^c \gamma_b^d F_{cd}$ which identifies the purely spatial components of the Maxwell tensor as the magnetic field components on Σ . Combining these, we can write

$$F_{ab} = \varepsilon_{abc} \perp B^c + n_a \perp E_b - n_b \perp E_a. \quad (\text{A.250})$$

Straightforward computation then shows

$$F^2 = 2[(\perp B)^2 - (\perp E)^2]. \quad (\text{A.251})$$

Kinetic terms such as $(\nabla \kappa)^2$ can be decomposed in a now obvious way

$$\begin{aligned} (\nabla \kappa)^2 &\equiv \nabla_a \kappa \nabla^a \kappa = g_a^b \nabla_b \kappa g^{ca} \nabla_c \kappa \\ &= (\gamma_a^b - n^b n_a) \nabla_b \kappa (\gamma^{ca} - n^c n^a) \nabla_c \kappa \\ &= D^a \kappa D_a \kappa - (\Xi)^2, \end{aligned} \quad (\text{A.252})$$

where we have used $n_a \gamma^{ab} = 0$ and the previously made definition of Ξ .

Using all of the above, the equation of motion for the dilaton ϕ can now be written in terms of two first order in time equations for ϕ and Π given by

$$n^a \nabla_a \phi = \frac{1}{\alpha} (\partial_t \phi - \beta^i \partial_i \phi) = -\Pi, \quad (\text{A.253})$$

$$\begin{aligned} n^b \nabla_b \Pi &= \frac{1}{\alpha} (\partial_t \Pi - \beta^i \partial_i \Pi) \\ &= -\frac{1}{\alpha} D_a (\alpha D^a \phi) + K \Pi \\ &\quad - \alpha_0 e^{-2\alpha_0 \phi} [(\perp B)^2 - (\perp E)^2] + \frac{1}{2} \alpha_0 e^{4\alpha_1 \phi} (D^a \kappa D_a \kappa - (\Xi)^2). \end{aligned} \quad (\text{A.254})$$

Note we have used the notation $n^a \nabla_a f = \frac{1}{\alpha} (\partial_t - \beta^i \partial_i) f$ for f a scalar. (This will not be true for a higher rank tensor field.) Note as well that the definition of Ξ provides the evolution equation for κ :

$$\partial_t \kappa = \beta^i \partial_i \kappa - \alpha \Xi. \quad (\text{A.255})$$

We will return to this momentarily. Finally, we note that D_a and ∂_a both indicate partial derivatives but D_a is a spatial derivative living on Σ . In general, however, we cannot say that D_a and ∂_a are equivalent.

A.5.2 Decomposition of the axion field, κ

We now consider the axion equations

$$\nabla_a \left(e^{4\alpha_1 \phi} \nabla^a \kappa \right) = F_{ab} (*F)^{ab}, \quad (\text{A.256})$$

or we can rewrite the equations into

$$\nabla_a \nabla^a \kappa = -4\alpha_1 \nabla_a \phi \nabla^a \kappa + e^{-4\alpha_1 \phi} F_{ab} (*F)^{ab}. \quad (\text{A.257})$$

For LHS of the equations, we use a completely analogous manner in A.5.1, Note that in a completely analogous manner, $\nabla_a \nabla^a \kappa$ can be decomposed as

$$\nabla_a \nabla^a \kappa = n^b \nabla_b \Xi + \frac{1}{\alpha} D_a (\alpha D^a \kappa) - K \Xi, \quad (\text{A.258})$$

where we define the auxiliary variable, $\Xi = -n^a \nabla_a \kappa$.

Using our previous results, it is straightforward to show that the decomposition of the term on the RHS involving κ becomes

$$\nabla_a \phi \nabla^a \kappa = D^a \kappa D_a \phi - \Pi \Xi,$$

Using earlier definitions, it is tedious but straightforward to verify that the contraction of the Maxwell tensor with its dual is the usual $\vec{E} \cdot \vec{B}$ invariant, or in our notation

$$F_{ab} (*F)^{ab} = 4 \perp E^c \perp B_c. \quad (\text{A.259})$$

Using all of the above, the equation of motion for the axion, κ , can be written as

$$n^b \nabla_b \Xi = -\frac{1}{\alpha} D_a (\alpha D^a \kappa) + K \Xi + 4\alpha_1 D^a \kappa D_a \phi + 4\alpha_1 \Pi \Xi + 4e^{-4\alpha_1 \phi} \perp B^c \perp E_c, \quad (\text{A.260})$$

or, in terms of derivatives with respect to the time coordinate

$$\partial_t \Xi = \beta^a \partial_a \Xi - D_a (\alpha D^a \kappa) + \alpha K \Xi + 4\alpha \alpha_1 D^a \kappa D_a \phi + 4\alpha \alpha_1 \Pi \Xi + 4\alpha e^{-4\alpha_1 \phi} \perp B^c \perp E_c. \quad (\text{A.261})$$

A.5.3 Decomposition of the electromagnetic field equation

We will next consider the equations for the electromagnetic field

$$\nabla_a \left(e^{-2\alpha_0\phi} F^{ab} \right) = -(*F)^{ab} \nabla_a \kappa, \quad (\text{A.262})$$

or we rewrite the equations into more relevant form for the decomposition.

$$\nabla_a F^{ab} = 2\alpha_0 (\nabla_a \phi) F^{ab} - e^{2\alpha_0\phi} (\nabla_a \kappa) (*F)^{ab}. \quad (\text{A.263})$$

Again, the procedure has already been outlined and we will not follow each step. As this is a vector equation, we must decompose it along and perpendicular to the spatial slices, Σ . To this end, we contract Eqn. A.263 along n_a and γ_{ab} separately. The first will give us an evolution equation and the second a constraint equation. With work, we can show that the EM equation contracted with the normal vector, n^a , becomes

$$D_i \perp E^i = 2\alpha_0 (D_i \phi) \perp E^i + e^{2\alpha_0\phi} (D_i \kappa) \perp B^i. \quad (\text{A.264})$$

Likewise, the EM equation projected into the spatial slice with γ_{bc} is

$$\begin{aligned} \gamma_{bc} D_a \perp F^{ab} + K_{ac} \perp E^a + \perp F_c^a D_a \ln \alpha + K \perp E_c \\ + n^d \nabla_d \perp E_c - 2\alpha_0 \Pi \perp E^a \gamma_{bc} - e^{2\alpha_0\phi} \Xi \perp B^a \gamma_{bc} = 0. \end{aligned} \quad (\text{A.265})$$

Rearranging, we can express both equations as

$$D_a \perp F^{ab} = 2\alpha_0 \Pi \perp E^a + e^{2\alpha_0\phi} \Xi \perp B^a \quad (\text{A.266})$$

$$n^d \nabla_d \perp E_c = -K_{ac} \perp E^a - \perp F_c^a D_a \ln \alpha - K \perp E_c. \quad (\text{A.267})$$

These equations are only half the Maxwell equations; namely the sourced EM equation. We have the homogeneous Maxwell equations which we need to decompose this in an analogous way as for the F_{ab} equation.

In particular, we write the equation as

$$\nabla_a (*F)^{ab} = 0, \quad (\text{A.268})$$

and project it again against n_b and γ_{bc} . We follow the same procedure as for the F_{ab} equation. On contracting with n_b , we can get this immediately with the definition of $\perp B_a = -\gamma_a^b (*F)_{bc} n^c$

$$n_b \nabla_a (*F)^{ab} = -D_a \perp B^a = 0. \quad (\text{A.269})$$

This is a divergence free condition for the magnetic field (in more familiar notation we might say $\nabla \cdot \mathbf{B} = 0$). On contraction with γ_{bc} , we can write

$$\gamma_{bc} \nabla_a (*F)^{ab} = \gamma_{bc} D_a \perp (*F)^{ab} - K_{ac} \perp B^a + \perp (*F)_c^a D_a \ln \alpha - K \perp B_c - n^d \nabla_d \perp B_c = 0. \quad (\text{A.270})$$

Thus, it gives

$$D_a \perp (*F)^{ab} = 0, \quad (\text{A.271})$$

$$n^d \nabla_d \perp B_c = -K_{ac} \perp B^a + \perp (*F)_c^a D_a \ln \alpha - K \perp B_c. \quad (\text{A.272})$$

In addition to this, we incorporate with constraint damping terms. The necessity of additional constraint damping terms is discussed in Sec. 2.2.3. In this appendix, we focus on the decomposition technique. We have a new set of EM equations with constraint dampings

$$\nabla_a (F^{ab} + g^{ab} \Psi) - 2\alpha_0 (\nabla_a \phi) F^{ab} + e^{2\alpha_0 \phi} (\nabla_a \kappa) (*F)^{ab} = \eta_1 n^b \Psi, \quad (\text{A.273})$$

$$\nabla_a ((*F)^{ab} + g^{ab} \Phi) = \eta_2 n^b \Phi, \quad (\text{A.274})$$

where η_1 and η_2 are positive real constants.

We follow our earlier techniques for decomposing this equation. Indeed, most of the terms have already been decomposed. We need only consider the decomposition of the constraint damping fields. After a little bit of work, we have

$$n^a \nabla_a \Psi + D_a \perp E^a - 2\alpha_0 (D_c \phi) \perp E^c - e^{2\alpha_0 \phi} (D_c \kappa) \perp B^c = -\eta_1 \Psi, \quad (\text{A.275})$$

$$n^a \nabla_a \Phi - D_a \perp B^a = -\eta_2 \Phi. \quad (\text{A.276})$$

In terms of time derivatives, these equations become

$$\partial_t \Psi = \beta^a \partial_a \Psi - \alpha D_a \perp E^a + 2\alpha \alpha_0 (D_c \phi) \perp E^c + \alpha e^{2\alpha_0 \phi} (D_c \kappa) \perp B^c - \alpha \eta_1 \Psi, \quad (\text{A.277})$$

$$\partial_t \Phi = \beta^a \partial_a \Phi + \alpha D_a \perp B^a - \alpha \eta_2 \Phi. \quad (\text{A.278})$$

Combining these with Eqns. A.265 and A.270, we have

$$\begin{aligned} n^d \nabla_d \perp E_c &= -\gamma_{bc} D_a \perp F^{ab} - K_{ac} \perp E^a - \perp F_c^a D_a \ln \alpha - K \perp E_c + \\ &+ 2\alpha_0 \Pi \perp E^a \gamma_{bc} + e^{2\alpha_0 \phi} \Xi \perp B^a \gamma_{bc} - D_c \Psi, \end{aligned} \quad (\text{A.279})$$

$$n^d \nabla_d \perp B_c = \gamma_{bc} D_a \perp (*F)^{ab} - K_{ac} \perp B^a + \perp (*F)_c^a D_a \ln \alpha - K \perp B_c + D_c \Phi_0. \quad (\text{A.280})$$

Here we need to evaluate $n^a \nabla_a \perp A^b$ in order to write the time derivative term in scalar form. Since $\perp A^b$ is a vector, we cannot write this as in the scalar field case. Let f^a be an arbitrary spatial vector. We want to show

$$\gamma_{bc} n^a \nabla_a f^b = \gamma_{bc} (n^a \partial_a f^b - K_a^b f^a + \frac{1}{\alpha} f^a \partial_a \beta^b). \quad (\text{A.281})$$

To show this, we use the definition of the Lie derivative:

$$\mathcal{L}_x y^a \equiv x^b \nabla_b y^a - y^b \nabla_b x^a = x^b \partial_b y^a - y^b \partial_b x^a. \quad (\text{A.282})$$

From the RHS of Eqn. A.281, we have

$$\begin{aligned} n^a \partial_a f^b - K_a^b f^a + \frac{1}{\alpha} f^a \partial_a \beta^b &= \frac{1}{\alpha} (\partial_t f^b - \beta^i \partial_i f^b) - K_a^b f^a + \frac{1}{\alpha} f^a \partial_a \beta^b \\ &= -\frac{1}{\alpha} \mathcal{L}_\beta f^b - f^a K_a^b + \frac{1}{\alpha} \partial_t f^b, \end{aligned}$$

while from the LHS of Eqn. A.281, we have

$$n^a \nabla_a f^b = n^a \nabla_a f^b - f^a \nabla_a n^b + f^a \nabla_a n^b = \mathcal{L}_n f^b + f^a \nabla_a n^b.$$

Combining these results gives

$$\gamma_{bc} (\mathcal{L}_n f^b + f^a \nabla_a n^b) = \gamma_{bc} (-\frac{1}{\alpha} \mathcal{L}_\beta f^b - f^a K_a^b + \frac{1}{\alpha} \partial_t f^b). \quad (\text{A.283})$$

On evaluating this, we get

$$\begin{aligned} \gamma_{bc} f^a \nabla_a n^b &= \gamma_{bc} f^d (\gamma^a_d - n^a n_d) \nabla_a n^b = \gamma_{bc} f^d \gamma^a_d \nabla_a n^b - \gamma_{bc} f^d n_d n^a \nabla_a n^b \\ &= f^d (\gamma_{bc} \gamma^a_d \nabla_a n^b) = -f^d K_{dc}. \end{aligned} \quad (\text{A.284})$$

Because f^d is an arbitrary spatial vector, $f^d n_d = 0$, and therefore we can rewrite Eqn. A.283 as

$$\begin{aligned} \gamma_{bc}(\mathcal{L}_n f^b + \frac{1}{\alpha} \mathcal{L}_\beta f^b) &= \gamma_{bc}(\frac{1}{\alpha} \partial_t f^b - f^a \nabla_a n^b - f^a K_a^b) \\ &= \gamma_{bc} \frac{1}{\alpha} \partial_t f^b + f^d K_{dc} - f^a \gamma_{bc} K_a^b \\ &= \gamma_{bc} \frac{1}{\alpha} \partial_t f^b. \end{aligned} \quad (\text{A.285})$$

Alternatively, we could also write this as

$$\gamma_{bc}(\alpha \mathcal{L}_n + \mathcal{L}_\beta) f^b = \gamma_{bc} \partial_t. \quad (\text{A.286})$$

By the definition of n^a and β^a , $\alpha n^a + \beta^a = \alpha \left(\frac{1}{\alpha}, -\frac{\beta^i}{\alpha} \right) + (0, \beta^i) = (1, 0, 0, 0) = t^a$. And using the property of the Lie derivative

$$(\alpha \mathcal{L}_n + \mathcal{L}_\beta) f^b = \mathcal{L}_{\alpha n + \beta} f^b = \mathcal{L}_t f^b = t^a \partial_a f^b - f^a \partial_a t^b = \partial_t f^b, \quad (\text{A.287})$$

which shows that Eqn. A.281 is indeed correct. Using this property, we can write

$$n^a \nabla_a \perp E^b = n^a \partial_a \perp E^b - K_a^b \perp E^a + \frac{1}{\alpha} \perp E^a \partial_a \beta^b. \quad (\text{A.288})$$

This can be applied to the electric and magnetic spatial vectors. To sum up, the decomposition of the EM equations becomes

$$\begin{aligned} n^a \nabla_a \perp E^b &= -\frac{1}{\alpha} D_a (\alpha \perp F^{ab}) - \perp E^a K_a^b + \perp E^b K - D^b \Psi - 2\alpha_0 [\Pi \perp E^b - \varepsilon^{abc} D_a \phi \perp B_c] \\ &\quad - e^{2\alpha_0 \phi} [\Xi \perp B^b + D_a K \perp (*F^{ab})], \end{aligned} \quad (\text{A.289})$$

$$n^a \nabla_a \perp B^b = \frac{1}{\alpha} D_a (\alpha \varepsilon^{abc} \perp E_c) - \perp B^a K_a^b + \perp B^b K + D^b \Phi. \quad (\text{A.290})$$

In terms of time derivatives we can write these as

$$\begin{aligned} \partial_t \perp E^b &= \beta^i \partial_i \perp E^b - \perp E^a \partial_a \beta^b - D_a (\alpha \varepsilon^{abc} \perp B_c) + \alpha \perp E^b K - \alpha D^b \Psi \\ &\quad - 2\alpha \alpha_0 [\Pi \perp E^b - \varepsilon^{abc} D_a \phi \perp B_c] - \alpha e^{2\alpha_0 \phi} [\Xi \perp B^b + D_a K \varepsilon^{abc} \perp E_c], \end{aligned} \quad (\text{A.291})$$

$$\partial_t \perp B^b = \beta^i \partial_i \perp B^b - \perp B^a \partial_a \beta^b + D_a (\alpha \varepsilon^{abc} \perp E_c) + \alpha \perp B^b K + \alpha D^b \Phi. \quad (\text{A.292})$$

Appendix B

Black Hole Initial data for EMDA

In this appendix, we describe several analytic BH solutions as possible initial data. We present several different types of BHs in both EMD and EMDA theories. First, we explore several different BH solutions for EMD then we review a BH solution for EMDA.

To use these BH solutions, they will need to be adapted to gauge or coordinate conditions that are useful for numerical simulations. In particular, we will need to consider BH initial data in so called quasi-isotropic coordinates. For the single BHs that we will consider, the (r, θ) part of the metric will, in general, be conformally related to

$$\frac{dr^2}{\Delta(r)} + d\theta^2, \quad (\text{B.1})$$

where $\Delta(r)$ is some radial factor related to the labeling of surfaces of constant r . For numerical purposes, this coordinate choice can be prone to singularity formation. So, a better coordinate system, again for numerical purposes, would be of the form

$$\frac{dr^2}{\Delta(r)} + d\theta^2 = \frac{1}{\bar{r}^2} (d\bar{r}^2 + \bar{r}^2 d\theta^2). \quad (\text{B.2})$$

We denote \bar{r} as a new radial coordinate which can be used to express this portion of the metric in a form that is conformally flat and with respect to standard circular type coordinates. This implies a coordinate transformation defined as

$$\int \frac{d\bar{r}}{\bar{r}} = \int \frac{dr}{\sqrt{\Delta}}, \quad (\text{B.3})$$

such that $\bar{r} \rightarrow r$ as $\bar{r} \rightarrow \infty$. We will generally rewrite our BH initial data in terms of this isotropic coordinate.

We also note that because our use of the BSSN variables in our code will be based on a Cartesian style coordinate system, this will be based the isotropic radial coordinate and standard angular coordinate $(\bar{r}, \theta, \varphi)$. The resulting Cartesian coordinate system will be denoted $\bar{x}, \bar{y}, \bar{z}$. We will also continue to reference the Boyer-Lindquist type r for convenience, but it should be understood that r is not a coordinate here but rather a complicated function of \bar{r} and other related coordinates.

B.1 The Garfinkle-Horowitz-Strominger Black Hole

Garfinkle, Horowitz, and Strominger (GHS) [40] constructed a charged EMD black hole ($\alpha_0 = 1$). This BH can be electrically or magnetically charged. We will consider the magnetically charged case first. The solution to the static, spherically symmetric, EMD equations with a regular event horizon and magnetic charge takes the following form in Schwarzschild like coordinates

$$ds^2 = - \left(1 - \frac{2M}{r}\right) dt^2 + \left(1 - \frac{2M}{r}\right)^{-1} dr^2 + r \left(r - \frac{Q_m^2 e^{-2\phi_0}}{M}\right) d\Omega^2, \quad (\text{B.4})$$

$$F_{\theta\varphi} = Q_m \sin \theta, \quad (\text{B.5})$$

$$e^{-2\phi} = e^{-2\phi_0} \left(1 - \frac{Q_m^2 e^{-2\phi_0}}{Mr}\right), \quad (\text{B.6})$$

where Q_m is the magnetic charge, M is the ADM mass, and ϕ_0 is the asymptotic value of the dilaton. There is a curvature singularity at $r = Q_m^2 e^{-2\phi_0}/M$ and a regular horizon at $r = 2M$ if $Q_m^2 < 2M^2 e^{2\phi_0}$ which becomes singular when the inequality is saturated, in other words, the extremal limit.

There is a discrete electromagnetic duality in this theory which leaves the equations of motion unchanged although the action does change as the F^2 term picks up a minus sign.

The transformation is (with $\alpha_0 = 1$)

$$F_{ab} \rightarrow \frac{1}{2} e^{-2\phi} \epsilon_{abcd} F^{cd}, \quad (\text{B.7})$$

$$\phi \rightarrow -\phi, \quad (\text{B.8})$$

$$g_{ab} \rightarrow g_{ab}. \quad (\text{B.9})$$

In this theory, this amounts to a means of generating new solutions. In particular, one can take the above magnetically charged solution and generate an electrically charged black hole solution that is also static and spherically symmetric. The solution is

$$ds^2 = - \left(1 - \frac{2M}{r}\right) dt^2 + \left(1 - \frac{2M}{r}\right)^{-1} dr^2 + r \left(r - \frac{Q_e^2 e^{2\phi_0}}{M}\right) d\Omega^2, \quad (\text{B.10})$$

$$F_{tr} = \frac{Q_m}{r^2} e^{-2\phi_0} = \frac{Q_e}{r}, \quad (\text{B.11})$$

$$e^{2\phi} = e^{2\phi_0} \left(1 - \frac{Q_e^2 e^{2\phi_0}}{Mr}\right), \quad (\text{B.12})$$

where Q_e is the electric charge (up to a constant), and M , ϕ_0 are the ADM mass and (*negative*) asymptotic value of the dilaton which are the same as in the magnetically charged case. There is a curvature singularity at $r = Q_e^2 e^{-2\phi_0} / M$ and a regular horizon at $r = 2M$ if $Q_e^2 < 2M^2 e^{2\phi_0}$ which becomes singular in the extremal limit.

Another generalization to consider is the case of these same black holes across theories i.e. with general α_0 . In this case, the solution to the static, spherically symmetric, EMD equations with a regular event horizon and magnetic charge takes the form (with $\beta = 2\alpha_0^2 / (1 + \alpha_0^2)$)

$$ds^2 = - \left(1 - \frac{r_+}{r}\right) \left(1 - \frac{r_-}{r}\right)^{1-\beta} dt^2 + \left(1 - \frac{r_+}{r}\right)^{-1} \left(1 - \frac{r_-}{r}\right)^{\beta-1} dr^2 + r^2 \left(1 - \frac{r_-}{r}\right)^\beta d\Omega^2, \quad (\text{B.13})$$

$$F_{\theta\phi} = Q_m \sin \theta, \quad (\text{B.14})$$

$$e^{-2\alpha_0\phi} = e^{-2\alpha_0\phi_0} \left(1 - \frac{r_-}{r}\right)^\beta, \quad (\text{B.15})$$

where r_+ and r_- give the outer and inner event horizons and their combination gives the

ADM mass, M , magnetic charge, Q_m , and asymptotic dilaton value, ϕ_0 , as follows

$$2M = r_+(1 - \beta)r_-, \quad (\text{B.16})$$

$$2Q_m^2 = e^{2\alpha_0\phi_0}r_+r_-(2 - \beta). \quad (\text{B.17})$$

Note that for $\alpha_0 = 0$ ($\beta = 0$) this reduces to Reissner-Nordstrom with a constant dilaton while $\alpha_0 = 1$ ($\beta = 1$) reproduces the above magnetic solution for low energy string theory.

Again, we can use a discrete electromagnetic duality to generate the electrically charged solution. The transformation is

$$F_{ab} \rightarrow \frac{1}{2}e^{-2\alpha_0\phi} \epsilon_{abcd}F^{cd}, \quad (\text{B.18})$$

$$\phi \rightarrow -\phi, \quad (\text{B.19})$$

$$g_{ab} \rightarrow g_{ab}. \quad (\text{B.20})$$

Taking the previous magnetically charged solution then we get

$$ds^2 = - \left(1 - \frac{r_+}{r}\right) \left(1 - \frac{r_-}{r}\right)^{1-\beta} dt^2 + \left(1 - \frac{r_+}{r}\right)^{-1} \left(1 - \frac{r_-}{r}\right)^{\beta-1} dr^2 + r^2 \left(1 - \frac{r_-}{r}\right)^\beta d\Omega^2, \quad (\text{B.21})$$

$$F_{tr} = \frac{Q_m}{r^2} e^{-2\alpha_0\phi_0} = \frac{Q_e}{r}, \quad (\text{B.22})$$

$$e^{-2\alpha_0\phi} = e^{2\alpha_0\phi_0} \left(1 - \frac{r_-}{r}\right)^\beta, \quad (\text{B.23})$$

where, again, r_+ and r_- give the outer and inner event horizons and their combinations gives the ADM mass, M , electric charge, Q_e , and asymptotic value of the dilaton, ϕ_0 , as follows

$$2M = r_+(1 - \beta)r_-, \quad (\text{B.24})$$

$$2Q_e^2 = e^{-2\alpha_0\phi_0}r_+r_-(2 - \beta). \quad (\text{B.25})$$

Note that we have not changed the value of the constant, ϕ_0 , in the duality transformation. It was originally introduced in the magnetic case as the asymptotic value of the dilaton. Notice that here, under the duality transformation, it is the *negative* of the asymptotic value of the

dilaton in the electric case. In a similar vein, while Q_e is a constant, it is not, in fact, strictly the electric charge on doing the surface integral at infinity of the relevant divergenceless quantity. That would actually be Q_m without the exponential factor. Nonetheless, we are going to call Q_e the electric charge and understand that it is effectively in slightly different units.

Now, express these solutions in isotropic coordinates. Define a new radial coordinate \bar{r} such that

$$\frac{d\bar{r}}{\bar{r}} = \frac{dr}{\sqrt{(r-r_+)(r-r_-)}}, \quad (\text{B.26})$$

On performing the integration and letting \bar{r} approach r at spatial infinity, we get

$$r = \frac{1}{\bar{r}} \left[\left(\bar{r} + \frac{r_+ + r_-}{4} \right)^2 - \frac{r_+ r_-}{4} \right], \quad (\text{B.27})$$

where we have

$$r_+ = M \left[1 + \left(1 - (1 - \alpha_0^2) \frac{Q^2}{M^2} \right)^{1/2} \right], \quad (\text{B.28})$$

$$r_- = \frac{Q^2}{M^2} (1 + \alpha_0^2) \left[1 + \left(1 - (1 - \alpha_0^2) \frac{Q^2}{M^2} \right)^{1/2} \right]^{-1}, \quad (\text{B.29})$$

where $Q^2 = Q_m^2 e^{-2\alpha_0\phi_0}$ for the magnetic case and $Q^2 = Q_e^2 e^{2\alpha_0\phi_0}$ for the electric case.

With this isotropic, radial coordinate, the metric for both magnetic and electric solutions takes the form

$$\begin{aligned} ds^2 &= -\alpha^2 dt^2 + \chi^{-1} (d\bar{r}^2 + \bar{r}^2 d\Omega^2) \\ &= -\frac{(\bar{r} - \bar{r}_H)^2 (\bar{r} + \bar{r}_H)^{2(1-\beta)}}{(\bar{r} + \bar{r}_H)^{2-\beta} (\bar{r} + \bar{r}_2)^{2-\beta}} dt^2 + \frac{1}{\bar{r}^4} (\bar{r} + \bar{r}_1)^{2-\beta} (\bar{r} + \bar{r}_2)^{2-\beta} (\bar{r} + \bar{r}_H)^{2\beta} [d\bar{r}^2 + \bar{r}^2 d\Omega^2], \end{aligned} \quad (\text{B.30})$$

where we have defined

$$\bar{r}_1 = \frac{1}{4} (\sqrt{r_+} - \sqrt{r_-})^2, \quad (\text{B.31})$$

$$\bar{r}_2 = \frac{1}{4} (\sqrt{r_+} + \sqrt{r_-})^2, \quad (\text{B.32})$$

$$\bar{r}_H = \frac{1}{2} (r_+ - r_-), \quad (\text{B.33})$$

with \bar{r}_H being the radial location of the horizon in these coordinates.

In the magnetically charged case, the EM and dilaton fields take the form

$$F_{\theta\phi} = Q_m \sin \theta, \quad (\text{B.34})$$

$$\perp B^{\bar{r}} = Q_m \frac{\bar{r}^4}{(\bar{r} + \bar{r}_1)^3 (\bar{r} + \bar{r}_2)^3} \left[\frac{(\bar{r} + \bar{r}_1)(\bar{r} + \bar{r}_2)}{(\bar{r} + \bar{r}_H)^2} \right]^{3\beta/2} = \frac{Q_m}{\bar{r}^2} \chi^{3/2}, \quad (\text{B.35})$$

$$e^{-2\alpha_0\phi} = e^{-2\alpha_0\phi_0} \frac{(\bar{r} + \bar{r}_H)^{2\beta}}{(\bar{r} + \bar{r}_1)^\beta (\bar{r} + \bar{r}_2)^\beta}. \quad (\text{B.36})$$

In the electrically charged case, the EM and dilaton fields take the form

$$F_{t\bar{r}} = Q_e \frac{(\bar{r}^2 - \bar{r}_H^2)}{(\bar{r} + \bar{r}_1)^2 (\bar{r} + \bar{r}_H)^2}, \quad (\text{B.37})$$

$$\perp E^{\bar{r}} = -Q_e \frac{\bar{r}^4}{(\bar{r} + \bar{r}_1)^3 (\bar{r} + \bar{r}_2)^3} \left[\frac{(\bar{r} + \bar{r}_1)(\bar{r} + \bar{r}_2)}{(\bar{r} + \bar{r}_H)^2} \right]^{\beta/2} = -Q_e \frac{\bar{r}^2}{(\bar{r} + \bar{r}_1)^2 (\bar{r} + \bar{r}_2)^2} \chi^{1/2}, \quad (\text{B.38})$$

$$e^{2\alpha_0\phi} = e^{2\alpha_0\phi_0} \frac{(\bar{r} + \bar{r}_H)^{2\beta}}{(\bar{r} + \bar{r}_1)^\beta (\bar{r} + \bar{r}_2)^\beta}. \quad (\text{B.39})$$

It turns out there is quantity associated with the dilaton that is also conserved. It is not an additional, independent conserved quantity as it is defined in terms of other conserved quantities. GHS define this dilaton charge according to

$$D = \frac{1}{4\pi} \int d\Sigma^a \nabla_a \phi, \quad (\text{B.40})$$

where the integral is over an S^2 at infinity. In the spherically symmetric magnetic case with $\alpha_0 = 1$ and a nonzero asymptotic value for the dilaton, this becomes

$$D = \frac{1}{4\pi} \lim_{r \rightarrow \infty} \int_{S^2} \partial_r \phi \left[r \left(r - \frac{Q_m^2 e^{-2\phi_0}}{M} \right) \right] \sin \theta d\theta d\phi = -\frac{Q_m^2 e^{-2\phi_0}}{2M}. \quad (\text{B.41})$$

For the electric case, this dilaton charge becomes

$$D = -\frac{1}{4\pi} \lim_{r \rightarrow \infty} \int_{S^2} \partial_r \phi \left[r \left(r - \frac{Q_e^2 e^{2\phi_0}}{M} \right) \right] \sin \theta d\theta d\phi = \frac{Q_e^2 e^{2\phi_0}}{2M}. \quad (\text{B.42})$$

We can extend this with general α_0 . For the magnetic case, the dilaton charge is

$$D = -\frac{\alpha_0 Q_m^2}{M} \frac{1}{1 + \sqrt{1 + (\alpha_0 - 1) Q_m^2 / M^2}}. \quad (\text{B.43})$$

For the electric case, the dilaton charge becomes

$$D = \frac{\alpha_0 Q_e^2}{M} \frac{1}{1 + \sqrt{1 + (\alpha_0 - 1) Q_e^2 / M^2}}. \quad (\text{B.44})$$

Note that as $\alpha_0 \rightarrow \infty$, the dilaton charge is just the EM charge

$$\lim_{\alpha_0 \rightarrow \infty} D = -|Q_m|, \quad (\text{B.45})$$

(or Q_e for the electrically charged case). As well, in the limit of infinite α_0 , the spherically symmetric metric becomes (with $\beta \rightarrow 2$)

$$ds^2 = - \left(1 - \frac{r_+}{r}\right) \left(1 - \frac{r_-}{r}\right)^{-1} dt^2 + \left(1 - \frac{r_+}{r}\right)^{-1} \left(1 - \frac{r_-}{r}\right) dr^2 + (r - r_-)^2 d\Omega^2. \quad (\text{B.46})$$

while the dilaton become $\phi = \phi_0$. The form for the EM field in either the electric or magnetic case is unchanged. However, the charge in each case is proportional to $2 - \beta$ which goes to zero in the limit. Thus, the EM field vanishes. Also note that if we define $r_s = r - r_-$ with $r_+ - r_- = 2M$, then we can identify this metric as the Schwarzschild metric with respect to the new Schwarzschild radial coordinate r_s .

B.2 The Horne-Horowitz-Kaluza-Klein Black Hole

The Horne-Horowitz-Kaluza-Klein (HHKK) black hole is a rotating member of the EMD family of theories with $\alpha_0 = \sqrt{3}$. It is constructed in the following way [41]. We begin by working in 5D GR. We take a product of 4D Kerr with \mathbb{R} . The resulting 5D manifold is, of course, a vacuum solution to the 5D Einstein equations. Now take this solution and make a simple Lorentz boost in the extra dimension (which we will denote as x^5):

$$t' = \gamma(t - vx^5), \quad (\text{B.47})$$

$$x' = \gamma(x^5 - vt). \quad (\text{B.48})$$

This boosted product manifold is a 5D BH and a solution which can be dimensionally reduced to 4D according to usual the Kaluza-Klein compactification prescription. On

performing the compactification, we can re-interpret the solution, now in 4D, in terms of a 4D metric with the additional Maxwell field and dilaton. The electrically charged solution in this case is

$$ds^2 = -\frac{B\rho^2}{\Sigma} dt^2 + \frac{\Sigma}{B\rho^2} \sin^2 \theta \left[d\phi - \frac{2amr}{\Sigma\sqrt{1-v^2}} dt \right] + B\rho^2 \left(\frac{dr^2}{\Delta} + d\theta^2 \right), \quad (\text{B.49})$$

$$A_a dx^a = \frac{v}{2(1-v^2)} \frac{2mr}{B^2\rho^2} dt - \frac{av}{2\sqrt{1-v^2}} \frac{2mr}{B^2\rho^2} \sin^2 \theta d\phi, \quad (\text{B.50})$$

$$\phi = -\frac{\sqrt{3}}{2} \ln B, \quad (\text{B.51})$$

where each quantity is defined with respect to the Boyer-Lindquist radial type coordinate, r :

$$\rho^2 = r^2 + a^2 \cos^2 \theta, \quad (\text{B.52})$$

$$\Delta = r^2 - 2mr + a^2, \quad (\text{B.53})$$

$$B^2 = 1 + \frac{v^2}{1-v^2} \frac{2mr}{\rho^2}, \quad (\text{B.54})$$

$$\Sigma = B^2 \rho^2 (r^2 + a^2) + 2mra^2 \sin^2 \theta. \quad (\text{B.55})$$

Here, the constants m , a , and v are parameters carried over from the 4D Kerr solution and the subsequent boost. The ADM mass and spin parameters of this BH solution are

$$M = m \left[1 + \frac{v^2}{2(1-v^2)} \right], \quad (\text{B.56})$$

$$Q_e = \frac{mv}{1-v^2}, \quad (\text{B.57})$$

$$J = \frac{ma}{\sqrt{1-v^2}}. \quad (\text{B.58})$$

The event horizon and ergosphere are in the same coordinate locations as in Kerr, i.e. $\Delta = 0$ and $\Delta = a^2 \sin^2 \theta$ respectively. The radial isotropic coordinate is again defined by

$$\int \frac{d\bar{r}}{\bar{r}} = \int \frac{dr}{\sqrt{\Delta}}, \quad (\text{B.59})$$

so that

$$r = \frac{1}{\bar{r}} \left[\left(\bar{r} + \frac{m}{2} \right)^2 - \frac{a^2}{4} \right]. \quad (\text{B.60})$$

The metric determinant remains $\sqrt{-g} = B\rho^2 \sin \theta$ while the components of the inverse metric are

$$g^{tt} = -\frac{\Sigma}{\Delta B\rho^2}, \quad (\text{B.61})$$

$$g^{t\varphi} = -\frac{2amr}{\Delta B\rho^2 \sqrt{1-v^2}}, \quad (\text{B.62})$$

$$g^{\varphi\varphi} = \frac{1}{B\Delta \sin^2 \theta} \left(1 - \frac{2mr}{\rho^2}\right), \quad (\text{B.63})$$

$$g^{rr} = \frac{\Delta}{B\rho^2}, \quad (\text{B.64})$$

$$g^{\theta\theta} = \frac{1}{B\rho^2}. \quad (\text{B.65})$$

The nonzero components of the extrinsic curvature are

$$K_{r\varphi} = \frac{am \sin^2 \theta}{\sqrt{1-v^2}} \frac{1}{B\rho^2} \frac{(\partial_r \Sigma)r - \Sigma}{\sqrt{\Delta B\rho^2 \Sigma}}, \quad (\text{B.66})$$

$$K_{\theta\varphi} = \frac{am \sin^2 \theta}{\sqrt{1-v^2}} \frac{1}{B\rho^2} \frac{\partial_\theta \Sigma}{\sqrt{\Delta B\rho^2 \Sigma}}. \quad (\text{B.67})$$

The nonzero components of the Maxwell tensor are

$$F_{tr} = \frac{mv}{1-v^2} \frac{2r^2 - \rho^2}{B^4 \rho^4}, \quad (\text{B.68})$$

$$F_{t\theta} = -\frac{mv}{1-v^2} \frac{2r}{B^4 \rho^4} a^2 \sin \theta \cos \theta, \quad (\text{B.69})$$

$$F_{r\varphi} = \frac{amv}{\sqrt{1-v^2}} \frac{2r^4 - \rho^2}{B^4 \rho^4} \sin^2 \theta, \quad (\text{B.70})$$

$$F_{\theta\varphi} = -\frac{2amrv}{\sqrt{1-v^2}} \frac{1}{B^4 \rho^4} (B^2 \rho^2 + a^2 \sin^2 \theta) \sin \theta \cos \theta. \quad (\text{B.71})$$

We could rewrite the components of the extrinsic curvature in Cartesian like coordinates

$(\bar{x}, \bar{y}, \bar{z})$ as

$$K_{\bar{x}\bar{x}} = -\frac{2\bar{x}\bar{y}}{\bar{r}\bar{\rho}^3} \left(\bar{\rho} \frac{\partial r}{\partial \bar{r}} K_{r\varphi} + \frac{\bar{z}}{\bar{r}} K_{\theta\varphi} \right), \quad (\text{B.72})$$

$$K_{\bar{x}\bar{y}} = -\frac{\bar{x}^2 - \bar{y}^2}{\bar{r}\bar{\rho}^3} \left(\bar{\rho} \frac{\partial r}{\partial \bar{r}} K_{r\varphi} + \frac{\bar{z}}{\bar{r}} K_{\theta\varphi} \right), \quad (\text{B.73})$$

$$K_{\bar{x}\bar{z}} = -\frac{\bar{y}}{\bar{r}\bar{\rho}^2} \left(\bar{z} \frac{\partial r}{\partial \bar{r}} K_{r\varphi} - \frac{\bar{\rho}}{\bar{r}} K_{\theta\varphi} \right), \quad (\text{B.74})$$

$$K_{\bar{y}\bar{y}} = -K_{\bar{x}\bar{z}}, \quad (\text{B.75})$$

$$K_{\bar{y}\bar{z}} = -\frac{\bar{x}}{\bar{y}} K_{\bar{x}\bar{z}}, \quad (\text{B.76})$$

$$K_{\bar{z}\bar{z}} = 0. \quad (\text{B.77})$$

Similar to the above, the components of Maxwell components in Cartesian like coordinates are

$$F_{t\bar{x}} = \frac{\bar{x}}{\bar{r}\bar{\rho}} \left(\bar{\rho} \frac{\partial r}{\partial \bar{r}} F_{tr} + \frac{\bar{z}}{\bar{r}} F_{t\theta} \right) \quad (\text{B.78})$$

$$F_{t\bar{y}} = \frac{\bar{y}}{\bar{x}} F_{t\bar{x}} \quad (\text{B.79})$$

$$F_{t\bar{z}} = \frac{1}{\bar{r}} \left(\bar{z} \frac{\partial r}{\partial \bar{r}} F_{tr} - \frac{\bar{\rho}}{\bar{r}} F_{t\theta} \right) \quad (\text{B.80})$$

Note that these components refer to the electric field. The remaining components are (implies B field)

$$F_{\bar{x}\bar{y}} = \frac{1}{\bar{r}\bar{\rho}} \left(\bar{\rho} \frac{\partial r}{\partial \bar{r}} F_{r\varphi} + \frac{\bar{z}}{\bar{r}} F_{\theta\varphi} \right), \quad (\text{B.81})$$

$$F_{\bar{x}\bar{z}} = \frac{\bar{y}}{\bar{r}\bar{\rho}} \left(\bar{z} \frac{\partial r}{\partial \bar{r}} F_{r\varphi} - \frac{\bar{\rho}}{\bar{r}} F_{\theta\varphi} \right), \quad (\text{B.82})$$

$$F_{\bar{y}\bar{z}} = -\frac{\bar{x}}{\bar{y}} F_{\bar{x}\bar{z}}. \quad (\text{B.83})$$

The components of the shift vector are

$$\beta^{\bar{x}} = -\bar{y}\beta^\varphi, \quad (\text{B.84})$$

$$\beta^{\bar{y}} = \bar{x}\beta^\varphi, \quad (\text{B.85})$$

$$\beta^{\bar{z}} = 0. \quad (\text{B.86})$$

Because of the discrete electromagnetic duality present in EMD, we can also write down a magnetically charged black hole solution as we could for the spherically symmetric case. The duality transformation is as before and the metric takes the same form as for the electric case. The Maxwell and dilaton fields in the magnetically charged case become

$$A_a dx^2 = -\frac{amv}{\sqrt{1-v^2}} \frac{\cos \theta}{\rho^2} dt + \frac{mv}{1-v^2} \frac{r^2 + a^2}{\rho^2} \cos \theta d\varphi, \quad (\text{B.87})$$

$$\phi = \frac{\sqrt{3}}{2} \ln B. \quad (\text{B.88})$$

And, the nonzero components of the Maxwell tensors are

$$F_{tr} = -\frac{2amrv}{\sqrt{1-v^2}} \frac{\cos \theta}{\rho^4}, \quad (\text{B.89})$$

$$F_{t\theta} = \frac{amv}{\sqrt{1-v^2}} \frac{2r^2 - \rho^4}{\rho^4} \sin \theta, \quad (\text{B.90})$$

$$F_{r\varphi} = -\frac{2mrv}{1-v^2} \frac{a^2 \sin^2 \theta \cos \theta}{\rho^4}, \quad (\text{B.91})$$

$$F_{\theta\varphi} = -\frac{mv}{1-v^2} \frac{2r^2 - \rho^2}{\rho^4} (r^2 + a^2) \sin \theta. \quad (\text{B.92})$$

This represents a magnetic monopole with a radial field and magnetic charge $Q_m = -mv/(1-v^2)$. Radial and poloidal electric field components as well as a poloidal magnetic field component are generated by the rotation of the black hole

B.3 The Rasheed-Larsen Black Hole

Rasheed [43] and Larsen [42] present a dyonic generalization of the above rotating, charged, EMD black hole for $\alpha_0 = \sqrt{3}$. We refer the reader to those works for the details of the derivations. We do note that we largely follow Laresn's notation. This solution has metric,

dilaton and gauge potential such that

$$ds^2 = -\frac{H_3}{\sqrt{H_1 H_2}}(dt + \tilde{B}d\phi)^2 + \sqrt{H_1 H_2} \left[\frac{1}{\Delta} dr^2 + d\theta^2 + \frac{\Delta}{H_3} \sin^2 \theta d\phi^2 \right], \quad (\text{B.93})$$

$$\phi = -\frac{\sqrt{3}}{2} \ln \sqrt{\frac{H_2}{H_1}}, \quad (\text{B.94})$$

$$\begin{aligned} A_a dx^a = & -\frac{1}{H_2} \left[Q \left(r + \frac{p-2m}{2} \right) + q \sqrt{\frac{q(p^2-4m^2)}{16m^2(p+q)}} a \cos \theta \right] dt \\ & - \frac{1}{H_2} \left[P(H_2 a^2 \sin^2 \theta) \cos \theta \right. \\ & \left. + \sqrt{\frac{p(q^2-4m^2)}{16m^2(p+q)}} \left(pr - m(p-2m) \frac{q(p^2-4m^2)}{p+q} \right) a \sin^2 \theta \right] d\phi, \end{aligned} \quad (\text{B.95})$$

where each quantity is defined (in Boyer-Lindquist radial type coordinate r)

$$H_1 = r^2 + a^2 \cos^2 \theta + r(p-2m) \frac{p(p-2m)(q-2m)}{2(p+q)} - \frac{p\sqrt{(q^2-4m)(p^2-4m)}}{2m(p+q)} a \cos \theta, \quad (\text{B.96})$$

$$H_2 = r^2 + a^2 \cos^2 \theta + r(q-2m) \frac{p(p-2m)(q-2m)}{2(p+q)} + \frac{q\sqrt{(q^2-4m)(p^2-4m)}}{2m(p+q)} a \cos \theta, \quad (\text{B.97})$$

$$H_3 = r^2 + a^2 \cos^2 \theta - 2mr, \quad (\text{B.98})$$

$$\tilde{B} = \sqrt{pg} \frac{(pq + 4m^2)r - m(p-2m)(q-2m)}{2m(p+q)H_3} a \sin^2 \theta, \quad (\text{B.99})$$

$$\Delta = r^2 + a^2 - 2mr, \quad (\text{B.100})$$

and the constants m and a are the mass and rotation parameters carried over from the 4D Kerr solution. The constants p and q define the magnetic and electric charges.

$$P^2 = \frac{p(p^2-4m)}{4(p+q)}, \quad (\text{B.101})$$

$$Q^2 = \frac{q(q^2-4m)}{4(p+q)}. \quad (\text{B.102})$$

Note that $p > 2m$ and $q > 2m$ in order to obtain real values of P and Q . The zero magnetic charge case or the zero electric charge case correspond to $p = 2m$ and $q = 2m$ respectively.

The mass and angular momentum of this spacetime are

$$M = \frac{p+q}{4}, \quad (\text{B.103})$$

$$J = \sqrt{pq} \frac{pq+4m^2}{4m(p+q)} a. \quad (\text{B.104})$$

Alternative forms of the metric can be written as

$$ds^2 = -\frac{\sqrt{H_1 H_2} \Delta}{\Sigma_{pq}} dt^2 + \frac{\Sigma_{pq} \sin^2 \theta}{\sqrt{H_1 H_2}} \left[d\varphi - \frac{a\sqrt{pq}(c_0(r-m)+r)}{\Sigma_{pq}} dt \right]^2 + \sqrt{H_1 H_2} \left(\frac{dr^2}{\Delta} + d\theta^2 \right) \quad (\text{B.105})$$

$$= -\frac{\rho^2 - 2mr}{\sqrt{H_1 H_2}} dt^2 - \frac{2a \sin^2 \theta \sqrt{pq}(c_0(r-m)+r)}{\sqrt{H_1 H_2}} dt d\varphi + \sqrt{H_1 H_2} \left(\frac{dr^2}{\Delta} + d\theta^2 \right) + \frac{\Sigma_{pq} \sin^2 \theta}{\sqrt{H_1 H_2}} d\varphi^2, \quad (\text{B.106})$$

where we define

$$\begin{aligned} \Sigma_{pq} &= (r^2 + a^2) \left[\rho^2 + r(q-2m) + r(p-2m) + \frac{1}{2}(p-2m)(q-2m) \right] + 2mra^2 \sin^2 \theta \\ &\quad + \Delta c_1 (q-p) a \cos \theta \\ &\quad + (p-2m)(q-2m) \left[r^2 + \frac{r}{2(p+q)} (q(p-2m) + p(q-2m)) \right] + pq(c_0^2 m^2 - c_1^2 a^2) \end{aligned} \quad (\text{B.107})$$

$$\rho^2 = r^2 + a^2 \cos^2 \theta, \quad (\text{B.108})$$

$$c_0 = \frac{(p-2m)(q-2m)}{2m(p+q)} = \frac{pq+4m^2}{2m(p+q)} - 1, \quad (\text{B.109})$$

$$c_1 = \frac{\sqrt{(p^2-4m^2)(q^2-4m^2)}}{2m(p+q)} = \sqrt{c_0(c_0+2)}. \quad (\text{B.110})$$

A convenient identity used in the calculations is the following:

$$(\rho^2 - 2mr)\Sigma_{pq} + pqa^2 \sin^2 \theta (c_0(r-m)+r)^2 = H_1 H_2 \Delta. \quad (\text{B.111})$$

Again, the event horizon and ergosphere are in the same coordinate locations as in Kerr (like the HHKK BH case). The radial isotropic coordinate is defined by

$$\int \frac{d\bar{r}}{\bar{r}} = \int \frac{dr}{\sqrt{\Delta}}, \quad (\text{B.112})$$

so

$$r = \frac{1}{\bar{r}} \left[\left(\bar{r} + \frac{m}{2} \right)^2 - \frac{a^2}{4} \right]. \quad (\text{B.113})$$

The components of the inverse metric are

$$g^{tt} = -\frac{\Sigma_{pq}}{\Delta\sqrt{H_1H_2}}, \quad (\text{B.114})$$

$$g^{t\varphi} = -\frac{a\sqrt{pq}(c_0(r-m)+r)}{\Delta\sqrt{H_1H_2}}, \quad (\text{B.115})$$

$$g^{\varphi\varphi} = \frac{\rho^2 - 2mr}{\Delta\sqrt{H_1H_2}\sin^2\theta}, \quad (\text{B.116})$$

$$g^{rr} = \frac{\Delta}{\sqrt{H_1H_2}}, \quad (\text{B.117})$$

$$g^{\theta\theta} = \frac{1}{\sqrt{H_1H_2}}. \quad (\text{B.118})$$

The nonzero components of the extrinsic curvature are

$$K_{r\varphi} = -\frac{a\sqrt{pq}\sin^2\theta}{2\sqrt{\Delta\Sigma_{pq}\sqrt{H_1H_2}}} [(c_0+1)\Sigma_{pq} - (c_0(r-m)+r)\partial_r\Sigma_{pq}], \quad (\text{B.119})$$

$$K_{\theta\varphi} = -\frac{a\sqrt{pq}\sin^2\theta}{2\sqrt{\Delta\Sigma_{pq}\sqrt{H_1H_2}}} \frac{\partial_\theta\Sigma_{pq}}{\sqrt{H_1H_2}} (c_0(r-m)+r). \quad (\text{B.120})$$

The nonzero components of the Maxwell tensors are

$$F_{tr} = \frac{Q}{H_2} - \frac{\partial_r H_2}{(H_2)^2} \left[Q \left(r + \frac{p-2m}{2} \right) + q \sqrt{\frac{q(p^2-4m)}{16m^2(p+q)}} a \cos\theta \right], \quad (\text{B.121})$$

$$F_{t\theta} = -\frac{\partial_r H_\theta}{(H_2)^2} \left[Q \left(r + \frac{p-2m}{2} \right) + q \sqrt{\frac{q(p^2-4m)}{16m^2(p+q)}} a \cos\theta \right] \\ - \frac{q}{H_2} \sqrt{\frac{q(p^2-4m)}{16m^2(p+q)}} a \sin\theta, \quad (\text{B.122})$$

$$F_{r\varphi} = \frac{\partial_r H_2}{(H_2)^2} \left[Pa^2 \sin^2 \theta \cos \theta + \sqrt{\frac{p(q^2 - 4m^2)}{16m^2(p+q)}} \left(pr - m(p-2m) + \frac{q(p^2 - 4m^2)}{p+q} \right) a \sin^2 \theta \right] - \frac{1}{H_2} p \sqrt{\frac{p(q^2 - 4m^2)}{16m^2(p+q)}} a \sin^2 \theta, \quad (\text{B.123})$$

$$F_{\theta\varphi} = P \sin \theta + \frac{1}{H_2} \left[Pa^2 \sin \theta (\sin^2 \theta - 2 \cos^2 \theta) - \sqrt{\frac{p(q^2 - 4m^2)}{16m^2(p+q)}} \left(pr - m(p-2m) + \frac{q(p^2 - 4m^2)}{p+q} \right) 2a \sin \theta \cos \theta \right] + \frac{\partial_\theta H_2}{(H_2)^2} \left[Pa^2 \sin^2 \theta \cos \theta + \sqrt{\frac{p(q^2 - 4m^2)}{16m^2(p+q)}} \left(pr - m(p-2m) + \frac{q(p^2 - 4m^2)}{p+q} \right) a \sin^2 \theta \right], \quad (\text{B.124})$$

where derivatives are

$$\partial_r H_1 = 2r + p - 2m, \quad (\text{B.125})$$

$$\partial_r H_2 = 2r + q - 2m, \quad (\text{B.126})$$

$$\partial_\theta H_1 = -2a^2 \sin \theta \cos \theta + pc_1 a \sin \theta, \quad (\text{B.127})$$

$$\partial_\theta H_2 = -2a^2 \sin \theta \cos \theta - 1c_1 a \sin \theta, \quad (\text{B.128})$$

$$\begin{aligned} \partial_r \Sigma_{pq} &= 2r \left[\rho^2 + r(q-2m) + r(p-2m) + \frac{1}{2}(p-2m)(q-2m) \right] \\ &\quad + (r^2 + a^2)(2r + q + p - 4m) + 2ma^2 \sin^2 \theta + 2(r-m)c_1(q-p) \cos \theta \\ &\quad + (p-2m)(q-2m) \left[2r + \frac{1}{2(p+q)}(q(p-2m) + p(q-2m)) \right], \end{aligned} \quad (\text{B.129})$$

$$\partial_\theta \Sigma_{pq} = -2(r^2 + a^2)a^2 \sin \theta \cos \theta + 4mra^2 \sin \theta \cos \theta - c_1(q-p)\Delta \sin \theta. \quad (\text{B.130})$$

The BSSN variables are

$$ds^2 = -\alpha^2 dt^2 + e^{4\xi} \tilde{\gamma}_{ij} (dx^i + \beta^i dt)(dx^j + \beta^j dt), \quad (\text{B.131})$$

where ξ is a conformal factor and $\det \tilde{\gamma} = 1$. From here, we have

$$\alpha^2 = \frac{\sqrt{H_1 H_2} \Delta}{\Sigma_{pq}}, \quad (\text{B.132})$$

$$\beta^\varphi = -\frac{a\sqrt{pq}(c_0(r-m) + r)}{\Sigma_{pq}}. \quad (\text{B.133})$$

We can write the metric $\bar{\gamma}_{ij}$ in Cartesian like coordinates (with $\bar{r}^2 = \bar{x}^2 + \bar{y}^2 + \bar{z}^2$, $\bar{\rho}^2 = \bar{x}^2 + \bar{y}^2$)

$$e^{12\xi} = \frac{\sqrt{H_1 H_2 \Sigma_{pq}}}{\bar{r}^6}, \quad (\text{B.134})$$

$$\bar{\gamma}_{\bar{x}\bar{x}} = \frac{C^{-2/3}}{\bar{\rho}^2} (\bar{x}^2 C + \bar{y}^2), \quad (\text{B.135})$$

$$\bar{\gamma}_{\bar{x}\bar{y}} = \frac{\bar{x}\bar{y}C^{-2/3}}{\bar{\rho}^2} (C - 1), \quad (\text{B.136})$$

$$\bar{\gamma}_{\bar{y}\bar{y}} = \frac{C^{-2/3}}{\bar{\rho}^2} (\bar{y}^2 C + \bar{x}^2), \quad (\text{B.137})$$

$$\bar{\gamma}_{\bar{z}\bar{z}} = C^{1/3}, \quad (\text{B.138})$$

where $C = H_1 H_2 / \Sigma_{pq}$. The components of the inverse metric are

$$\bar{\gamma}^{\bar{x}\bar{x}} = \frac{C^{-1/3}}{\bar{\rho}^2} (\bar{y}^2 C + \bar{x}^2), \quad (\text{B.139})$$

$$\bar{\gamma}^{\bar{x}\bar{y}} = \frac{\bar{x}\bar{y}C^{-1/3}}{\bar{\rho}^2} (1 - C), \quad (\text{B.140})$$

$$\bar{\gamma}^{\bar{y}\bar{y}} = \frac{C^{-1/3}}{\bar{\rho}^2} (\bar{x}^2 C + \bar{y}^2), \quad (\text{B.141})$$

$$\bar{\gamma}^{\bar{z}\bar{z}} = C^{-1/3}. \quad (\text{B.142})$$

The components of conformal connection coefficients are

$$\bar{\Gamma}^{\bar{x}} = -\bar{x} \left[\frac{C^{-1/3}}{\bar{\rho}^2} (3 - C) + (\bar{r} \sin^2 \theta + \partial_{\bar{r}} + \sin \theta \cos \theta \partial_{\theta}) \frac{C^{-1/3}}{\bar{\rho}^2} \right], \quad (\text{B.143})$$

$$\bar{\Gamma}^{\bar{y}} = \frac{\bar{y}}{\bar{x}} \bar{\Gamma}^{\bar{x}}, \quad (\text{B.144})$$

$$\bar{\Gamma}^{\bar{z}} = -\partial_{\bar{z}} C^{-1/3}. \quad (\text{B.145})$$

We can rewrite other quantities such as the extrinsic curvature components and the Maxwell tensor in Cartesian like coordinates in a way similar to the HHKK solution as described in the previous section.

B.4 The Kerr-Sen Black Hole

A rotating black hole solution also exists in EMDA for $\alpha_0 = \alpha_1 = 1$. Sen finds [38] this solution in the low-energy limit of heterotic string theory. The action in this case is given, in the string frame, as

$$\int d^4x \sqrt{-g_{str}} e^{-2\phi} \left(R + 4(\nabla\phi)^2 - \frac{1}{12} H^2 - F^2 \right). \quad (\text{B.146})$$

We prefer to consider things in the Einstein frame and thus we perform the conformal transformation to give the action in the Einstein frame which is

$$\int d^4x \sqrt{-g} \left(R - 2(\nabla\phi)^2 - \frac{1}{12} e^{-4\phi} H^2 - e^{-2\phi} F^2 \right). \quad (\text{B.147})$$

Note that $H^2 \equiv H_{abc}H^{abc}$ is the kinetic term for the 3-form field and is derivable from a 2-form gauge potential according to $H = dB - A \wedge F$ where F is our $U(1)$ gauge field in the action and given, in turn, by $F = dA$.

Without derivation, Sen's solution for the metric in Boyer-Lindquist coordinate is then

$$ds^2 = -\frac{\Delta\rho^2}{\Sigma} dt^2 + \rho^2 \left(\frac{dr^2}{\Delta} + d\theta^2 \right) + \frac{\Sigma \sin^2 \theta}{\rho^2} \left(d\varphi - \frac{2amr \cosh^2 \alpha}{\Sigma} dt \right)^2, \quad (\text{B.148})$$

where we use the definitions

$$\Delta = r^2 - 2mr + a^2, \quad (\text{B.149})$$

$$\rho^2 = \Delta - a^2 \sin^2 \theta + 2mr \cosh^2 \alpha, \quad (\text{B.150})$$

$$\Sigma = (\rho^2 + a^2 \sin^2 \theta)^2 - \Delta a^2 \sin^2 \theta, \quad (\text{B.151})$$

and the associated gauge, dilaton and two-form fields being

$$A_a dx^a = \frac{mr \sinh 2\alpha}{\sqrt{2}\rho^2} (dt - a \sin^2 \theta d\varphi), \quad (\text{B.152})$$

$$e^{-\Phi} = e^{-2\phi} = \frac{\rho^2}{r^2 + a^2 \cos^2 \theta}, \quad (\text{B.153})$$

$$B_{t\varphi} = \frac{2mar \sinh^2 \alpha}{\rho^2} \sin^2 \theta. \quad (\text{B.154})$$

The quantities m , α , and a are integration parameters. (Note that α here is not the lapse function.) Thus, the mass, electric charge, angular momentum, and magnetic dipole moment of the spacetime can be expressed in terms of these parameters

$$M = m \cosh^2 \alpha, \quad (\text{B.155})$$

$$Q = \frac{m}{\sqrt{2}} \sinh 2\alpha, \quad (\text{B.156})$$

$$J = aM, \quad (\text{B.157})$$

$$\mu = aQ. \quad (\text{B.158})$$

The event horizon is at $r_H = m + \sqrt{m^2 - a^2}$ such that a BH exists provided $m^2 > a^2$ or $M^2 \geq |J| + Q^2/2$. The inverse relations are

$$m = M - \frac{Q^2}{2M}, \quad (\text{B.159})$$

$$\sinh 2\alpha = \frac{2\sqrt{2}QM}{2M^2 - Q^2}, \quad (\text{B.160})$$

$$a = \frac{J}{M}. \quad (\text{B.161})$$

We choose to use the axion pseudo-scalar rather than the 3 form field H . To this end, we need to find the expression of the 3 form field in terms of the axion field κ . To do that, we must solve

$$H_{abc} = \frac{1}{2} e^{4\phi} \varepsilon_{abc}{}^d \partial_d \kappa, \quad (\text{B.162})$$

where we need to know what the 3 form H_{abc} , is in terms of its potential, the 2 form B_{ab} .

For the normalization in the current action, the 3 form H_{abc} is given as

$$H_{abc} = \partial_a B_{bc} + \partial_b B_{ca} + \partial_c B_{ab} - 2(A_a F_{bc} + A_b F_{ca} + A_c F_{ab}). \quad (\text{B.163})$$

From this, we obtain the following integrability conditions for the axion field κ

$$\partial_\theta \kappa = 2e^{-4\phi} \frac{1}{\sqrt{-g}} \frac{1}{g^{\theta\theta}} \partial_r B_{t\phi}, \quad (\text{B.164})$$

$$\partial_r \kappa = -2e^{-4\phi} \frac{1}{\sqrt{-g}} \frac{1}{g^{rr}} \left(\partial_\theta B_{t\phi} + 2A_t^2 \partial_\theta \frac{A_\phi}{A_t} \right). \quad (\text{B.165})$$

Integrating both above expressions gives

$$\kappa = 4ma \sinh^2 \alpha \frac{\cos \theta}{r^2 + a^2 \cos^2 \theta}. \quad (\text{B.166})$$

This shows self-consistency of field and this suggests that the definition above for the H field in terms of B and A is correct.

The radial isotropic coordinate is defined by

$$\int \frac{d\bar{r}}{\bar{r}} = \int \frac{dr}{\sqrt{\Delta}}, \quad (\text{B.167})$$

so

$$\bar{r} = \frac{1}{2}(r - m \pm \sqrt{(r - m)^2 - (m^2 - a^2)}). \quad (\text{B.168})$$

This is the quasi-isotropic coordinates for this case. The inverse should be obtained by

$$\begin{aligned} r &= \bar{r} + m + \frac{m^2 - a^2}{4\bar{r}} \\ &= \frac{1}{\bar{r}} \left[\left(\bar{r} + \frac{m}{2} \right)^2 - \frac{a^2}{4} \right] \\ &= \frac{1}{\bar{r}} (\bar{r} + \bar{r}_1)(\bar{r} + \bar{r}_2), \end{aligned} \quad (\text{B.169})$$

where $r_{1,2} = (m \pm a)/2$ such that $\bar{r}_1 < \bar{r}_2$. The metric with respect to the isotropic coordinate becomes

$$ds^2 = -\frac{\Delta \rho^2}{\Sigma} dt^2 + \frac{\rho^2}{\bar{r}^2} (d\bar{r}^2 + \bar{r}^2 d\theta^2) + \frac{\Sigma \sin^2 \theta}{\rho^2} \left(d\phi - \frac{2amr \cosh^2 \alpha}{\Sigma} dt \right)^2. \quad (\text{B.170})$$

Note that now r is a function of \bar{r} i.e. $r(\bar{r})$. All metric components depend on the isotropic

coordinate, \bar{r}

$$\begin{aligned}
\Delta(\bar{r}) &= r^2 - 2mr + a^2 = (r - m)^2 - (m^2 - a^2) \\
&= \left(\bar{r} + \frac{m^2 - a^2}{4\bar{r}} \right)^2 - (m^2 - a^2) = \left(\bar{r} + \frac{\bar{r}_1 \bar{r}_2}{\bar{r}} \right)^2 - 4\bar{r}_1 \bar{r}_2 \\
&= \bar{r}^2 - 2\bar{r}_1 \bar{r}_2 + \frac{\bar{r}_1^2 \bar{r}_2^2}{\bar{r}^2} \\
&= \left(\bar{r} - \frac{\bar{r}_1 \bar{r}_2}{\bar{r}} \right)^2 = \frac{1}{\bar{r}^2} (\bar{r}^2 - \bar{r}_1 \bar{r}_2)^2,
\end{aligned} \tag{B.171}$$

$$\begin{aligned}
\rho^2(\bar{r}) &= \Delta - a^2 \sin^2 \theta + 2mr \cosh^2 \alpha = r^2 + a^2 \cos^2 \theta + 2mr \sinh^2 \alpha \\
&= \frac{1}{\bar{r}^2} (\bar{r} + \bar{r}_1)^2 (\bar{r} + \bar{r}_2)^2 + a^2 \cos^2 \theta + \frac{2m}{\bar{r}} (\bar{r} + \bar{r}_1) (\bar{r} + \bar{r}_2) \sinh^2 \alpha,
\end{aligned} \tag{B.172}$$

$$\Sigma(\bar{r}) = (\rho^2(\bar{r}) + a^2 \sin^2 \theta)^2 - \Delta(\bar{r}) a^2 \sin^2 \theta. \tag{B.173}$$

The nonzero components of the inverse metric are

$$g^{tt} = -\frac{\Sigma}{\Delta \rho^2}, \tag{B.174}$$

$$g^{t\phi} = -\frac{2amr \cosh^2 \alpha}{\Delta \rho^2}, \tag{B.175}$$

$$g^{\phi\phi} = \frac{1}{\Delta \sin^2 \theta} \left(1 - \frac{2mr \cosh^2 \alpha}{\rho^2} \right), \tag{B.176}$$

$$g^{rr} = \frac{\Delta}{\rho^2}, \tag{B.177}$$

$$g^{\theta\theta} = \frac{1}{\rho^2}. \tag{B.178}$$

The nonzero components of the extrinsic curvature are

$$K_{r\phi} = am \sin^2 \theta \cosh^2 \alpha \frac{(\partial_r \Sigma)r - \Sigma}{\rho^2 \sqrt{\Delta \rho^2 \Sigma}}, \tag{B.179}$$

$$K_{r\theta} = am \sin^2 \theta \cosh^2 \alpha \frac{\partial_\theta \Sigma}{\rho^2 \sqrt{\Delta \rho^2 \Sigma}}. \tag{B.180}$$

The nonzero components of the Maxwell fields are

$$F_{tr} = \frac{m}{\sqrt{2}} \sinh 2\alpha \left(\frac{r^2 - a^2 \cos^2 \theta}{\rho^4} \right), \quad (\text{B.181})$$

$$F_{t\theta} = -\sqrt{2}m \sinh 2\alpha \left(\frac{r}{\rho^4} \right) a^2 \sin \theta \cos \theta, \quad (\text{B.182})$$

$$F_{r\phi} = \frac{am}{\sqrt{2}} \sinh 2\alpha \left(\frac{r^2 - a^2 \cos^2 \theta}{\rho^4} \right) \sin^2 \theta, \quad (\text{B.183})$$

$$F_{\theta\phi} = -\sqrt{2}ma \sinh 2\alpha \left(\frac{r}{\rho^4} \right) (r^2 + a^2 2mr \sinh^2 \alpha) \sin \theta \cos \theta. \quad (\text{B.184})$$

Again, the BSSN variables are written such that

$$ds^2 = -\alpha^2 dt^2 + e^{4\xi} \bar{\gamma}_{ij} (dx^i + \beta^i dt)(dx^j + \beta^j dt). \quad (\text{B.185})$$

So, we have

$$\alpha^2 = \frac{\Delta \rho^2}{\Sigma}, \quad (\text{B.186})$$

$$\beta^\phi = -\frac{2amr \cosh^2 \alpha}{\Sigma}, \quad (\text{B.187})$$

with respect to the radial isotropic coordinates. Calculate derivatives of metric quantities that we need in several places

$$\partial_r(\rho^2) = 2r + 2m \sinh^2 \alpha, \quad (\text{B.188})$$

$$\partial_\theta(\rho^2) = -2a^2 \sin \theta \cos \theta, \quad (\text{B.189})$$

$$\partial_r \Sigma = 4\rho^2(r + m \sinh^2 \alpha) + 2a^2 \sin^2 \theta (r + m \cosh 2\alpha), \quad (\text{B.190})$$

$$\partial_\theta \Sigma = -2\Delta a^2 \sin \theta \cos \theta. \quad (\text{B.191})$$

We can write the metric $\bar{\gamma}_{ij}$ in Cartesian like coordinates (with $\bar{r}^2 = \bar{x}^2 + \bar{y}^2 + \bar{z}^2$, $\bar{\rho}^2 =$

$\bar{x}^2 + \bar{y}^2$)

$$e^{12\bar{\phi}} = \frac{\rho^2 \Sigma}{\bar{r}^6}, \quad (\text{B.192})$$

$$\bar{\gamma}_{\bar{x}\bar{x}} = \frac{C^{-2/3}}{\bar{\rho}^2} (\bar{x}^2 C + \bar{y}^2), \quad (\text{B.193})$$

$$\bar{\gamma}_{\bar{x}\bar{y}} = \frac{\bar{x}\bar{y}C^{-2/3}}{\bar{\rho}^2} (C - 1), \quad (\text{B.194})$$

$$\bar{\gamma}_{\bar{y}\bar{y}} = \frac{C^{-2/3}}{\bar{\rho}^2} (\bar{y}^2 C + \bar{x}^2), \quad (\text{B.195})$$

$$\bar{\gamma}_{\bar{z}\bar{z}} = C^{1/3} \quad (\text{B.196})$$

where $C = \rho^4 / \Sigma$. The components of the inverse metric are

$$\bar{\gamma}^{\bar{x}\bar{x}} = \frac{C^{-1/3}}{\bar{\rho}^2} (\bar{y}^2 C + \bar{x}^2), \quad (\text{B.197})$$

$$\bar{\gamma}^{\bar{x}\bar{y}} = \frac{\bar{x}\bar{y}C^{-1/3}}{\bar{\rho}^2} (1 - C), \quad (\text{B.198})$$

$$\bar{\gamma}^{\bar{y}\bar{y}} = \frac{C^{-1/3}}{\bar{\rho}^2} (\bar{x}^2 C + \bar{y}^2), \quad (\text{B.199})$$

$$\bar{\gamma}^{\bar{z}\bar{z}} = C^{-1/3}. \quad (\text{B.200})$$

The components of the conformal connection coefficients are

$$\bar{\Gamma}^{\bar{x}} = -\bar{x} \left[\frac{C^{-1/3}}{\bar{\rho}^2} (3 - C) + (\bar{r} \sin^2 \theta + \partial_{\bar{r}} + \sin \theta \cos \theta \partial_{\theta}) \frac{C^{-1/3}}{\bar{\rho}^2} \right], \quad (\text{B.201})$$

$$\bar{\Gamma}^{\bar{y}} = \frac{\bar{y}}{\bar{x}} \bar{\Gamma}^{\bar{x}}, \quad (\text{B.202})$$

$$\bar{\Gamma}^{\bar{z}} = -\partial_{\bar{z}} C^{-1/3}. \quad (\text{B.203})$$

We can rewrite other quantities such as the extrinsic curvature components and the components of the Maxwell tensor in Cartesian like coordinates in a way similar to the HHKK solution as described in the previous section.

Of course, there are other possible BH solutions in different energy sector and even higher dimensions. Searching more black hole forms with respect to EMD and EMDA can give more parameterization of theories that we described so far.

Appendix C

Sparse Wavelet Representation via Iterated Interpolating Wavelets

In this Appendix, we review the wavelet method that we use in this work. We will describe the essential ingredients that are crucial in order to obtain wavelet adaptivity. Two of these essential ingredients are the construction of iterative interpolating wavelets and the notion of sparse wavelet representations for the numerical grid [199, 200, 201].. Both of these we describe below.

C.1 Interpolating Bases

The starting point for our wavelet method are iterated interpolating functions. Before we define these functions, we need to provide a mathematical definition of the notion of grids. To begin, let us work in one dimension and assume that we have a set of points (or grids) on the real line described by V_j . Each grid V_j is set of uniformly spaced grid points that satisfies $V_j = \{x_{j,k} \in \mathbb{R} : x_{j,k} = 2^{-j}k, k \in \mathbb{Z}\}$ where j is an integer, and $x_{j,k}$ are the coordinate locations of the points on level j . The integer k labels points on level j .

It is clear that grid V_{j+1} contains all grid points in V_j with additional points inserted halfway in between the points in V_j . The locations of points on these grids are illustrated in

Fig 3.2. The index k runs from 0 to $2^j N_0$ for some number of base grid points $N_0 + 1$. The number of grid points at level j is given by $N_j = 2^j N_0 + 1$. One of the crucial aspects of our grids is that every point in V_j is also in V_{j+1} at all higher level grids. If the index k is evenly divided by $2^{(j-m)}$, then the point in question also corresponds to a point on level V_m (for $m < j$). Another way of saying this is that, at a given level, the points on an odd index appear at that level for the first time. All points with even index exist on at least one of the previous levels.

Now consider function values on our set of grids, V_j . Call this $\{f_{j,k}\}_{k \in \mathbb{Z}^+}$ where $f_{j,k} = f(x_{j,k})$, is a function defined on the grid points in V_j . We would like to extend these function values to higher levels of the grid using interpolation. To do that, we define a very simple basis set. Take the function $\phi_{j,k}(x)$ to be defined so that at points at level j , e.g. $x_{j,l}$, the function is either zero or one:

$$\phi_{j,k}(x_{j,l}) = \delta_{k,l}. \quad (\text{C.1})$$

Of course, at other x values, $\phi_{j,k}(x)$ will take on values other than zero or one. Indeed, our task will be to construct $\phi_{j,k}(x)$. So, we want to begin with this sequence of numbers and turn it into a function by using interpolation to fill in values at higher levels, copying the values from lower levels onto corresponding higher levels. Note that the basis functions satisfy the following translation property:

$$\phi_{j,k}(x) = \phi_{j,m}(x + (x_{j,m} - x_{j,k})), \quad (\text{C.2})$$

which implies that they have the same shape but are translated in x . Further, these functions will satisfy a scaling relation given by:

$$\phi_{j,k}(2x) = \phi_{j+1,2k}(x). \quad (\text{C.3})$$

In general, the basis we are constructing is a dilated translated version of a single abstract function $\phi(x)$ such that:

$$\phi_{j,k}(x) = \phi(2^j x - x_{j,k}). \quad (\text{C.4})$$

Note another interesting property of these functions. If we consider a basis element with an odd position index, then the value of the function on all grid points at a lower level will be zero:

$$\phi_{j,k}(x_{g,m}) = 0 \text{ when } g < j. \quad (\text{C.5})$$

This is because the one nonzero value at level j will be for a location that is not in any of the other lower level grids.

The idea of interpolation from level $j + 1$ to level j can be expressed as

$$\phi_{j,k} = \sum_{l=0}^{N_{j+1}} h_{j,k}^{j+1,l} \phi_{j+1,l}(x), \quad (\text{C.6})$$

where the coefficients $h_{j,k}^{j+1,l}$ define the chosen interpolation scheme. These coefficients are easily found. Consider a basis function at level j evaluated on level $j + 1$

$$\begin{aligned} \phi_{j,k}(x_{j+1,m}) &= \sum_{l=0}^{N_{j+1}} h_{j,k}^{j+1,l} \phi_{j+1,l}(x_{j+1,m}) \\ &= \sum_{l=0}^{N_{j+1}} h_{j,k}^{j+1,l} \delta_{l,m} = h_{j,k}^{j+1,m}. \end{aligned}$$

The h coefficients are thus just values of the j level basis on the nearby $j + 1$ level points, which are either known from Eqn. C.1 or can be found using the interpolation scheme. Note that the vast majority of these values are zero, and it is only a few points near the index $2k$ (on level $j + 1$) that will contribute non-zero coefficients. For the particular case of four point interpolation we have that the most relevant coefficients are:

m	$h_{j,k}^{j+1,m}$
$2k - 4$	0
$2k - 3$	-1/16
$2k - 2$	0
$2k - 1$	9/16
$2k$	1
$2k + 1$	9/16
$2k + 2$	0
$2k + 3$	-1/16
$2k + 4$	0

Table C.1 The coefficients for four point interpolation

This procedure can be iterated to any higher level because the basis at any level can be written in terms of the basis at any higher level. Indeed, there was nothing special in the previous expressions about the difference in levels being 1. For any two levels j and g , where $g > j$:

$$\phi_{j,k}(x) = \sum_{l=0}^{N_g} h_{j,k}^{g,l} \phi_{g,l}(x), \quad (\text{C.7})$$

where

$$h_{j,k}^{g,l} = \phi_{j,k}(x_{g,l}). \quad (\text{C.8})$$

C.2 An Interpolating Wavelet Basis

So far, we have defined a basis for different levels on our set of grids. The basis on one level can be written in terms of the basis at a higher level. Using this, we can efficiently represent complicated functions on a very high level i.e. a very fine grid. Consider a basis made from the following functions:

$$\{\phi_{0,k}, \psi_{j,k}\}, \quad (\text{C.9})$$

where

$$\Psi_{j,k} = \phi_{j+1,2k+1}, \quad (\text{C.10})$$

and j takes the values from 0 to $J - 1$, where J is some specified maximum refinement level of the grid. The $\Psi_{j,k}$ functions are those from the basis that do not live on the base grid. However, this set of functions can act as a basis for all the finer grid levels available. Each point of the finest grid can be associated with a single function from the given set. Note that even though $\Psi_{j,k}$ is centered on a point at level $j + 1$, we nevertheless index the Ψ function at level j .

Thus, we can write any function on the finest grid as a weighted sum of these functions:

$$f(x) = \sum_{k=0}^{N_0} f_{0,k} \phi_{0,k}(x) + \sum_{j=0}^{J-1} \sum_{k=0}^{N_j-1} d_{j,k} \Psi_{j,k}(x). \quad (\text{C.11})$$

It is easy to verify the coefficients, $f_{0,k}$, in the first term are just those values of the function, f , on the base grid, $f(x_{0,l}) = f_{0,l}$.

To derive the coefficients $d_{j,k}$, we can proceed recursively. Start with a point on level 1:

$$\begin{aligned} f(x_{1,l}) &= \sum_{k=0}^{N_0} f_{0,k} \phi_{0,k}(x_{1,l}) + \sum_{j=0}^{J-1} \sum_{k=0}^{N_j-1} d_{j,k} \Psi_{j,k}(x_{1,l}), \\ f_{1,l} &= \sum_{k=0}^{N_0} f_{0,k} \phi_{0,k}(x_{1,l}) + \sum_{j=0}^{J-1} \sum_{k=0}^{N_j-1} d_{j,k} \phi_{j+1,2k+1}(x_{1,l}) \\ &= \sum_{k=0}^{N_0} f_{0,k} \phi_{0,k}(x_{1,l}) + \sum_{j=0}^{J-1} \sum_{k=0}^{N_0-1} d_{0,k} \phi_{j+1,2k+1}(x_{1,l}) \\ &= \sum_{k=0}^{N_0} f_{0,k} \phi_{0,k}(x_{1,l}) + \sum_{j=0}^{J-1} \sum_{k=0}^{N_0-1} d_{0,k} \delta_{2k+1,l} \\ &= \sum_{k=0}^{N_0} f_{0,k} \phi_{0,k}(x_{1,l}) + d_{0,(l-1)/2}, \end{aligned} \quad (\text{C.12})$$

where we have used the notational shorthand $f(x_{j,k}) = f_{j,k}$ and Eqn. C.10. Note that we also use Eqn. C.5 to eliminate all the terms in the second summation that vanish. Finally, we use Eqn. C.1, and the Kronecker delta to eliminate all the rest of the terms. We find:

$$d_{0,(l-1)/2} = f_{1,l} - \sum_{k=0}^{N_0} f_{0,k} \phi_{0,k}(x_{1,l}). \quad (\text{C.13})$$

This can be simplified slightly by using Eqn. C.6 in the second term, and applying Eqn. C.1:

$$\begin{aligned}
d_{0,(l-1)/2} &= f_{1,l} - \sum_{k=0}^{N_0} f_{0,k} \phi_{0,k}(x_{1,l}) \\
&= f_{1,l} - \sum_{k=0}^{N_0} f_{0,k} \sum_{m=0}^{N_1} h_{0,k}^{1,m} \phi_{1,m}(x_{1,l}) \\
&= f_{1,l} - \sum_{k=0}^{N_0} f_{0,k} \sum_{m=0}^{N_1} h_{0,k}^{1,m} \delta_{m,l} \\
&= f_{1,l} - \sum_{k=0}^{N_0} f_{0,k} \sum_{m=0}^{N_1} h_{0,k}^{1,l}.
\end{aligned}$$

For the case of four point interpolation the last sum expands out to give (see Table 3.1 for the specific h values),

$$\begin{aligned}
d_{0,(l-1)/2} &= f_{1,l} - f_{0,(l-3)/2} h_{0,(l-3)/2}^{1,l} - f_{0,(l-1)/2} h_{0,(l-1)/2}^{1,l} \\
&\quad - f_{0,(l+1)/2} h_{0,(l+1)/2}^{1,l} - f_{0,(l+3)/2} h_{0,(l+3)/2}^{1,l}.
\end{aligned} \tag{C.14}$$

and note that the summation is over locations in the base level:

$$d_{0,(l-1)/2} = f_{1,l} - \left(-\frac{1}{16} f_{0,(l-3)/2} + \frac{9}{16} f_{0,(l-1)/2} + \frac{9}{16} f_{0,(l+1)/2} - \frac{1}{16} f_{0,(l+3)/2} \right). \tag{C.15}$$

We recognize this last sum as being exactly the formula for interpolating from level 0 to level 1. As such we can now interpret the coefficients $d_{0,(l-1)/2}$ as just the difference between the actual function value at level 1 points and those interpolated values at level 1 based on the level 0 values. For brevity, we will on occasion call the term in parenthesis $P_{1,l}$, the interpolation at point l on level 1 from points on the previous level. We can extend this notation to arbitrary levels:

$$P_{j,k}(f) = -\frac{1}{16} f_{j-1,(k-3)/2} + \frac{9}{16} f_{j-1,(k-1)/2} + \frac{9}{16} f_{j-1,(k+1)/2} - \frac{1}{16} f_{j-1,(k+3)/2}. \tag{C.16}$$

Of course for other interpolation schemes the coefficients may be different. However, this leads to the general expression for the coefficients as given by:

$$d_{0,(l-1)/2} = f_{1,l} - P_{1,l}(f). \tag{C.17}$$

C.2.1 Computing Wavelet Coefficients

As just noted in Eqn. C.17, we recognize the $d_{0,k}$ coefficients as the difference between the function value at the level 1 points and the interpolations of the level 0 values at that point. We can always recombine the sum in Eqn. C.11 to start at level 1.

We will treat each term in Eqn. C.11 separately. We start with the sum over the base level functions:

$$\begin{aligned}
\sum_{k=0}^{N_0} f_{0,k} \phi_{0,k}(x) &= \sum_{k=0}^{N_0} f_{0,k} \sum_{l=0}^{N_1} h_{0,k}^{1,l} \phi_{1,l}(x) \\
&= \sum_{k=0}^{N_0} f_{1,2k} \sum_{l=0}^{N_1} h_{0,k}^{1,l} \phi_{1,l}(x) \\
&= \sum_{k=0}^{N_0} f_{1,2k} \sum_{l=0}^{N_1} h_{0,k}^{1,2l} \phi_{1,2l}(x) + \sum_{k=0}^{N_0} f_{1,2k} \sum_{l=0}^{N_0-1} h_{0,k}^{1,2l+1} \phi_{1,2l+1}(x) \\
&= \sum_{k=0}^{N_0} f_{1,2k} \phi_{1,2k}(x) + \sum_{k=0}^{N_0} f_{0,k} \sum_{l=0}^{N_0-1} h_{0,k}^{1,2l+1} \phi_{1,2l+1}(x), \tag{C.18}
\end{aligned}$$

where we have used the fact that $f_{0,k} = f_{1,2k}$.

For the second term in Eqn. C.11, the basic idea is to split the sum over levels into the zero level contribution and everything else. We plug in our result for $d_{0,l}$ and get

$$\begin{aligned}
\sum_{j=0}^{J-1} \sum_{k=0}^{N_j} d_{j,k} \psi_{j,k}(x) &= \sum_{k=0}^{N_0-1} d_{0,k} \psi_{0,k}(x) + \sum_{j=1}^{J-1} \sum_{k=0}^{N_j-1} d_{j,k} \psi_{j,k}(x) \\
&= \sum_{k=0}^{N_0-1} d_{0,k} \phi_{1,2k+1}(x) + \sum_{j=1}^{J-1} \sum_{k=0}^{N_j-1} d_{j,k} \psi_{j,k}(x) \\
&= \sum_{k=0}^{N_0-1} \left(f_{1,2k+1} - \sum_{w=0}^{N_0} f_{0,w} h_{0,w}^{1,2k+1} \right) \phi_{1,2k+1}(x) + \sum_{j=1}^{J-1} \sum_{k=0}^{N_j-1} d_{j,k} \psi_{j,k}(x) \\
&= \sum_{k=0}^{N_0-1} f_{1,2k+1} \phi_{1,2k+1}(x) - \sum_{w=0}^{N_0} f_{0,w} \sum_{k=0}^{N_0} h_{0,w}^{1,2k+1} \phi_{1,2k+1}(x) \\
&\quad + \sum_{j=1}^{J-1} \sum_{k=0}^{N_j-1} d_{j,k} \psi_{j,k}(x). \tag{C.19}
\end{aligned}$$

Here, the second term in Eqn. C.18 will now exactly cancel the second term from Eqn. C.19

resulting in the final expression

$$\begin{aligned}
 f(x) &= \sum_{k=0}^{N_0} f_{1,2k} \phi_{1,2k}(x) + \sum_{k=0}^{N_0-1} f_{1,2k+1} \phi_{1,2k+1}(x) + \sum_{j=1}^{J-1} \sum_{k=0}^{N_j-1} d_{j,k} \psi_{j,k}(x) \\
 &= \sum_{k=0}^{N_1} f_{1,k} \phi_{1,k}(x) + \sum_{j=1}^{J-1} \sum_{k=0}^{N_j-1} d_{j,k} \psi_{j,k}(x).
 \end{aligned} \tag{C.20}$$

Eqn. C.20 is essentially identical to the original expansion in the multilevel basis, Eqn. C.11. The upshot of this is that we can now easily find the $d_{1,k}$ coefficients. They will be, by analogy with Eqn. C.17, $d_{1,k} = f_{2,2k+1} - P_{2,2k+1}(f)$. This process can be repeated to add all levels and clearly will yield the following expression:

$$d_{j,k} = f_{j+1,2k+1} - P_{j+1,2k+1}(f). \tag{C.21}$$

The coefficients $d_{j,k}$ are called wavelet coefficients.

C.3 The Sparse Wavelet Representation

Having constructed our expansion for functions in terms of the wavelet coefficients with the wavelet basis, we will use these not only to express our function but also to define our adaptive grid. First realize that we have two possible expressions for our function. One is the collection of function values at all the points. The other are the function values at the base level with $d_{j,k}$ values at the points in between. These $d_{j,k}$ values are not the function values, but we can use them to recreate all function values that we might need. This involves rearranging C.21 for the values of the functions at the higher levels.

Notice that in the above, we only ever needed to use a few values of the basis functions. This is because of the iterated interpolation property of each of the basis functions and the manner in which they are constructed. Their very construction creates the details on the finer levels from the interpolation from the coarser levels. We emphasize that this is not the same as saying that the functions only have non-zero values at a few locations. Every basis function, at every level, has a large number of nonzero values. The special way that

we construct them allows us to use a few values to generate others, but this is not the same as saying these functions are only defined in a few places.

Now that we are able to compute the $d_{j,k}$ coefficients given some function $f(x)$, we can now represent that function with fewer points than exist on the finest level. As we have seen, the wavelet coefficients, $d_{j,k}$, give the difference between the function values and the interpolated values. If this coefficient at a certain point is close to zero, then if we are willing to accept a certain level of error in our representation, we can ignore that coefficient because the interpolated value will be close enough for our needs. This is the essence of the sparse representation.

If we choose some tolerance ε , we can group the terms in Eqn. C.11 according to whether $|d_{j,k}| < \varepsilon$ or not. For those terms that are less than ε , we can ignore them in our sum, and we will pick up an error there that is on the order of ε . Each of the terms in Eqn. C.11 are naturally associated with a grid point. The $f_{0,k}$ terms are associated with points on the base level, and the $d_{j,k}$ terms are associated with points on higher levels. When ignoring d terms that are below the tolerance, we may want to throw out points of the grid. This could potentially save on computational costs.

However, just because we do not have to store extra information at points where $|d_{j,k}| < \varepsilon$, does not mean there is no value at that location. Indeed, the basis elements of all the other terms that will overlap the missing point will contribute to the value of the function represented by the expression Eqn. C.11. So the lack of one of the locations in the grid is really a statement that all the relevant information at that location is covered by other functions. In addition, because all of the base level points will always be in the grid, there will be a number of the base level basis functions that will contribute at every location in the grid.

Appendix D

FleCSPH Implementation

D.1 FleCSI Framework

FleCSI [202] is a compile-time configurable framework designed to support multi-physics application development. It is developed at the Los Alamos National Laboratory as part of the Los Alamos Ristra project. As such, FleCSI provides a very general set of infrastructure design patterns that can be specialized and extended to suit the needs of a broad variety of solver and data requirements. FleCSI currently supports multi-dimensional mesh topology, geometry, and adjacency information, as well as n -dimensional hashed-tree data structures, graph partitioning interfaces, and dependency closures.

FleCSI introduces a functional programming model with control, execution, and data abstractions that are consistent both with MPI and with state-of-the-art, task-based runtimes such as Legion[203] and Charm++[204]. The abstraction layer insulates developers from the underlying runtime, while allowing support for multiple runtime systems including conventional models like asynchronous MPI.

The intent is to provide developers with a concrete set of user-friendly programming tools that can be used now, while allowing flexibility in choosing runtime implementations and optimizations that can be applied to future architectures and runtimes.

FleCSI's control and execution models provide formal nomenclature for describing

poorly understood concepts such as kernels and tasks. FleCSI's data model provides a low-buy-in approach that makes it an attractive option for many application projects, as developers are not locked into particular layouts or data structure representations.

FleCSI currently provides a parallel but not distributed implementation of Binary, Quad and Oct-tree topology. This implementation is based on space filling curve domain decomposition, the Morton order. The current version allows the user to specify the code main loop and the data distribution requested. The data distribution feature is not available for the tree data structure needed in our SPH code and we provide it in the FleCSPH implementation. The next step will be to incorporate it directly from FleCSPH to FleCSI as we reach a decent level of performance. As FleCSI is an in-development code the structure may change in the future and we will keep track of these updates in FleCSPH.

Based on FleCSI the intent is to provide a binary, quad and oct-tree data structure and the methods to create, search and share information for it. In FleCSPH this will be dedicated, applied and tested on the SPH method. In this part we first present the domain decomposition, based on space filling curves, and the tree data structure. We describe the HDF5 file structure used for the I/O. Then we describe the distributed algorithm for the data structure over the MPI processes.

D.2 Domain Decomposition

The number of particles can be high and represent a huge amount of data that does not fit in a single node of memory. This implies the distribution of the particles over several computational nodes. As the particles move during the simulation, static distribution is not possible and they have to be redistributed at some point in the execution. Furthermore, this distribution needs to keep local particles in the same computation node to optimize exchanges and the computation itself.

A common approach is to distribute the particles over computational nodes using *space filling curves* as in [205, 206, 207]. It intends to assign to each particle a key which is based

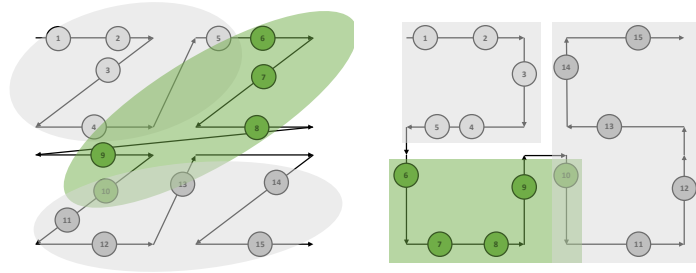


Figure D.1 Morton and Hilbert space filling curves

on its spatial coordinates, then sorting particles based on those keys keeps particles grouped locally. Many space filling curves exist: Morton, Hilbert, Peano, Moore, Gosper, etc.

This domain decomposition is used in several layers of our implementation. On one hand, the domain decomposition must spread the particles over all the MPI processes and provide a decent load balancing regarding the number of particles. On the other hand, the same domain decomposition must be used locally to store efficiently the particles and provide a $O(N \log(N))$ neighbor search, instead of $O(N^2)$, using a tree representation described in Sec. D.3.

Several space filling curves can suit our purposes. **Morton curves** [208], or Z-Order, is the most widespread method. This method can produce irregularly shaped domain decomposition as shown in green in Fig. D.1. The main advantage of these curves is their speed and their ease of computation. Key to this is their ability to directly interlace three coordinate directions without invoking rotations. **Hilbert curves** [209] are constructed by interlacing bits but also by adding rotation based on the Gray code [210]. This work is based on Peano curves and are also sometimes referred to as Hilbert-Peano curves. Their construction is more complicated than for Morton curves but they do allow for a better distribution of data.

We present the Morton (left) and Hilbert (right) space-filling curves in Fig. D.1. The particles are distributed over 3 processes. The set of particles in the second process appears in green. As we can see there are discontinuities in the Morton case due to the Z-order “jump” over the space. This can lead to non-local particles and over-sharing of particles

that will not be needed during the computation. In the Hilbert curve, the locality over the processes is conserved.

In this first implementation of FleCSPH we used Morton ordering due to the computational cost. In future work, we will compare computation time of different space filling curves.

Technically the keys are generated for each particle at each iteration because their position is expected to change over time. To be more efficient, the keys can stay the same during several steps and the final comparison can be made on the real particles' positions. This increases the search time but allows fewer tree reconstructions.

We use 64 bits to represent the keys to avoid conflicts. The FleCSI code allows us to use a combination of memory words to reach the desired size of keys (possibly more than 64 bits) but there is an extra cost in memory occupancy. The particle keys are generated by normalizing the space and then converting the floating-point representation to 64 bits for each dimension. Then the Morton interlacing is done and the keys are created. Unfortunately, in some arrangements, as with isolated particles, or in scenarios with particles very close to each other, the keys can be either badly distributed or duplicate keys can appear. Indeed, if the distance between two particles is less than $2^{-64} \approx 10^{-20}$, in a normalized space, the key generated through the algorithm will be the same. This problem is handled during the particle sort and the tree generation. In both cases, two particles can be differentiated based on their unique ID generated at runtime.

D.3 Hierarchical Trees

The method we use for the tree data structure creation and search comes from Barnes-Hut trees presented in [211, 212]. By reducing the search complexity from $O(N^2)$ for direct summation to $O(N \log(N))$ it allows us to do very large simulations with billions of particles. It also allows the use of the tree data structure to compute gravitation using multipole methods.

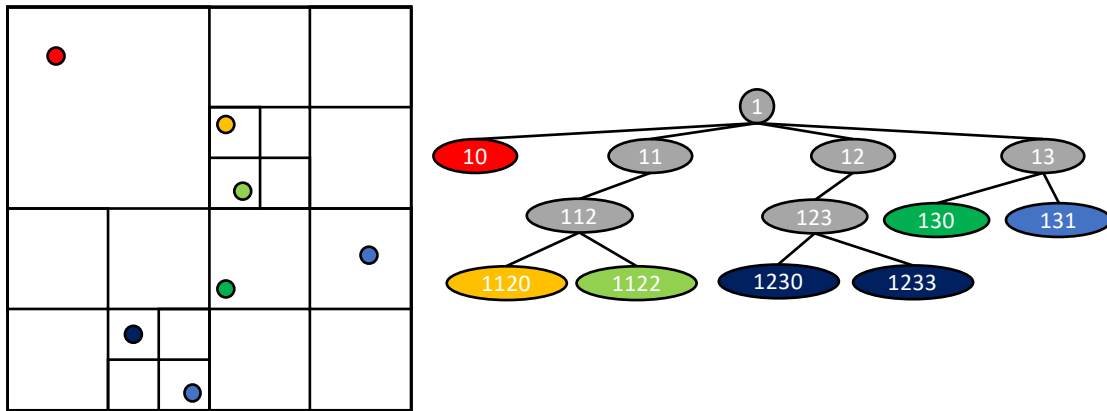


Figure D.2 Quadtree, space and data representation

We consider binary trees, for 1 dimension, quad-trees, for 2 dimensions, and oct-trees, for 3 dimensions. The construction of those trees is directly based on the domain decomposition using keys and space-filling curves presented in Sec. D.2.

As explained in the previous section, we use 64 bit keys. That gives us up to 63, 31 and 21 levels in the tree for respectively 1, 2 and 3 dimensions. As presented on in Fig. D.2 the first bit is used to represent the root of the tree, 1. This allows us to have up to 2^{63} different keys and unique particles.

D.4 Tree Generation

After each particle gets distributed on its final process using its space-filling curve key, we can recursively construct the tree. Each particle is added and the branches are created recursively if there is an intersection between keys. Starting from the root of key “1” the branches are added at each level until the particles are reached. An example of a final tree is shown in Fig. D.2.

D.5 Tree Search

When all the particles have been added, the data regarding the tree nodes are computed with a bottom up approach. Then we add all necessary information such as summing up the mass, position called COM, and the boundary box of all sub-particles of this tree node.

For the search algorithm the basic idea would be to do a tree traversal for all the particles and once we reach a particle or a node that interacts with the particle smoothing length, we add it for computation or in a neighbor list. Besides being easy to implement and to use in parallel, this algorithm requires a full tree traversal for every particle and will not take advantage of the particle's locality.

Our search algorithm, presented on Algorithm D.1, is a two step algorithm as with Barnes trees. We first create the interaction lists and then use them on the sub-tree particles. In the first step we look down for nodes with a target sub-mass of particles $tmass$. Then for those branches we compute an interaction list and continue the recursive tree search. When a particle is reached, we compute the physics using the interaction list as the neighbors. The interaction list is computed using an opening-angle criterion comparing the boundary box and a user defined angle. In this way we will not need a full tree traversal for each particle but a full tree traversal for every group of particles.

D.6 Distribution Strategies

The previous section presented the tree data structure that can be used locally on every node. The distribution layer is added on top of it, keeping each sub-tree on the computation nodes. The current version of FleCSPH is still based on synchronous communications using the Message Passing Interface (MPI).

The main distributed algorithm is presented in Algorithm D.2:

ALGORITHM D.1 Tree search algorithm

```
1: procedure FIND_NODES
2:   stack  $stk \leftarrow$  root
3:   while not_empty( $stk$ ) do
4:     branch  $b \leftarrow stk.pop()$ 
5:     if  $b$  is leaf then
6:       for each particles  $p$  of  $b$  do
7:          $apply\_sub\_tree(p, interaction\_list(p))$ 
8:       end for
9:     else
10:      for each child branch  $c$  of  $b$  do
11:         $stk.push(c)$ 
12:      end for
13:    end if
14:  end while
15: end procedure
```



```
16: procedure APPLY_SUB_TREE(node n, node-list nl)
17:   stack stk  $\leftarrow$  n
18:   while not_empty(stk) do
19:     branch b  $\leftarrow$  stk.pop()
20:     if b is leaf then
21:       for each particles p of b do
22:         apply_physics(p,nl)
23:       end for
24:     else
25:       for each child branch c of b do
26:         stk.push(c)
27:       end for
28:     end if
29:   end while
30: end procedure
```

```
31: function INTERACTION_LIST(node n)
32:   stack stk  $\leftarrow$  root
33:   node-list nl  $\leftarrow$   $\emptyset$ 
34:   while not_empty(stk) do
35:     branch b  $\leftarrow$  stk.pop()
36:     if b is leaf then
37:       for each particles p of b do
38:         if within() then
39:           nl  $\leftarrow$  nl + p
40:         end if
41:       end for
42:     else
43:       for each child branch c of b do
44:         if mac(c,angle) then
45:           nl  $\leftarrow$  nl + c
46:         else
47:           stk.push(c)
48:         end if
49:       end for
50:     end if
51:   end while
52: end function
```

ALGORITHM D.2 Main algorithm

- 1: **procedure** SPECIALIZATION_DRIVER(input data file f)
- 2: Read f in parallel
- 3: Set physics constant from f
- 4: **while** iterations **do**
- 5: Distribute the particles using distributed quick sort
- 6: Compute total range
- 7: Generate the local tree
- 8: Share branches
- 9: Compute the ghosts particles
- 10: Update ghosts data
- 11: *PHYSICS*
- 12: Update ghosts data
- 13: *PHYSICS*
- 14: Distributed output to file
- 15: **end while**
- 16: **end procedure**

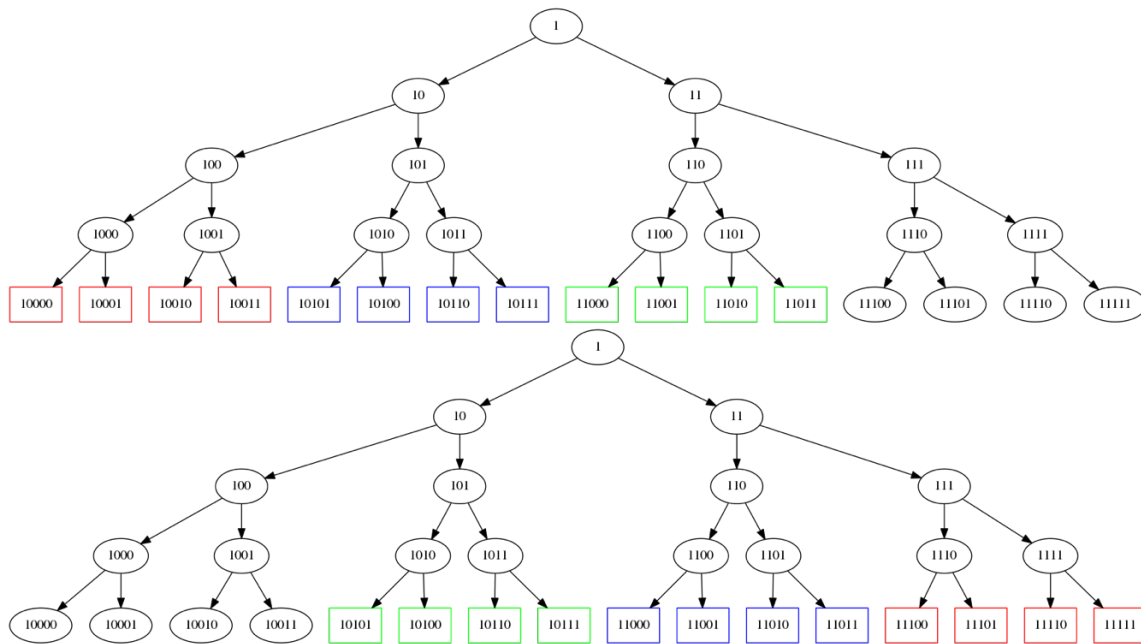


Figure D.3 Binaries tree for a 2 processes system. Exclusive, Shared and Ghosts particles resp. red, blue, green.

Particle distribution

The sort step, line 5, is based on a distributed quick sort algorithm. The keys are generated using the Morton order described in Sec. D.2. As we associate a unique number to each particle we are able to sort them using the keys and, in case of collision keys, using their unique ID. This gives us a global order for the particles. Each process sends to a master node (or submaster for larger cases) a sample of its keys. We determined this size to be 256 KB of key data per process for our test cases but it can be refined for larger simulations. Then the master determines the general ordering for all the processes and shares the pivots. Each process locally sorts its local keys and, in a global communication step, the particles are distributed to the process on which they belong. This algorithm gives us a good partition in terms of number of particles. Some downsides can be identified:

- The ordering may not be balanced in terms of the number of particles per process. But by optimizing the data exchanged to the master can lead to better affectation.
- The load balance also depends on the number of neighbors of each particle. If a

particle finds itself in a non-smooth region or large distance from other particles, this can lead to bad load balancing too.

This is why we also provide another load balancing based on the particles' neighbors. Depending on the problem, the choice can be to distribute the particles on each process regarding the number of neighbors, having the same amount of physical computation to perform on each process.

After this first step, the branches are shared between the different processes, line Algorithm 8. All of them communicate with their neighbors' several boundaries boxes. Then particles from the neighbors are computed, exchanged and added in the local tree. Those particles are labeled as `NON_LOCAL` particles. At this point a particle can be referenced as `EXCLUSIVE` in which they will never be exchanged and will only be used on this process. `SHARED` particles may be needed by another process during the computation. `GHOST` particles have information that the process needs to retrieve from another process. An example is given for 2 processes on Fig. D.3.

Exchange Shared and Ghost particles

The previous distribution shares the particles and the general information about neighbors' particles. Each process is able to do synchronous or asynchronous communications to gather distant particles. In the current version of FleCSPH an extra step is required to synchronously share data needed during the next tree traversal and physics components. After this step, ghost data can be exchanged as wanted several times during the same time step.

D.7 I/O

Regarding the high number of particles, an efficient, parallel and distributed I/O implementation is required. Several choices were available but we wanted a solution that can be specific for our usage. The first requirement is to allow the user to work directly with the

Paraview visualization tool and splash¹ [183].

We base this first implementation on the HDF5 [213] file structure with H5Part and H5Hut [214]. The HDF5 file structure supports MPI runtime with distributed read and write in a single or multiple files. We added the library H5hut to add normalization in the code to represent global data, steps, steps' data and the particles data for each step. The I/O code was developed internally at LANL and provides a simple way to write and read the data in H5Part format. The usage of H5Hut to generate H5part data files allows us to directly read the output in Paraview without using a XDMF descriptor as requested in HDF5 format.

¹<http://users.monash.edu.au/~dprice/splash/>

Appendix E

Artifact Evaluations

The purpose of artifact evaluation is mainly reproducing computational results which are presented here. In this appendix, we provide instructions for various codes that we use as part of this dissertation. More detailed information and codes themselves can be found Hyun Lim's github webpage (<https://github.com/hlim88>).

E.1 Toolkits/Softwares

E.1.1 HAD

HAD is a computational toolkit providing distributed adaptive mesh refinement for partial differential equations. It provides a flexible environment for diverse applications which can be combined or employed individually.

This code is used in Chap 2 to study the dynamics of BHs in EMDA theory and quadratic gravity. To install the code, please follow the instructions at <http://relativity.phys.lsu.edu/postdocs/matt/software/docs/had/docs/index.html>. We recommend that you create your own environment file that enables the correct export of compilers and project source codes. For example, we show below the local environment file to build HAD using intel complers for the EMDA project.

```
export NEED_LORENE_NAMESPACE="true"
```

```
export CCTK_C_PTR_TYPE="integer8"
export CCTK_F77_QUAD_TYPE="real16"
export MPIHOME="/apps/mpich/3.1.4_intel-15.0.3"
export LSV=""
export IMPI="-I$MPIHOME/include"
export LMPI="-L$MPIHOME/lib -lmpi"
export MPIF90="mpif90"
export CC="mpicc"
export CXX_COMMAND="icpc"
export LOCALIB="-L$HOME/intel-15.0.3/lib -L/opt/gm/lib"
export LOCALINCLUDE="-I$HOME/intel-15.0.3/include"
export F90FLAGS="-O3 -r8 -mmodel=medium -fpic"
export CCFLAGS="-O3 -fpic -mmodel=medium"
export F90_FREEFORM_FLAG="-FR"
export F90_FIXEDFORM_FLAG="-FI"
export MASS_SCALE_FACTOR="1.0"
export LFFTW=-lfftw3

export EQS="hyperEMDA"
export EXECNAME="hemda"
```

Note that the last two lines show which project you would like to include and the name of the binary executable. All projects can be found in the `src` directory. Using this, you can create `.ENV_local` which is sourced prior to invoking `make`. After that, one simply types `make` to build the code.

E.1.1.1 Running the Application

After successful installation of HAD, you will see the binary executable in the bin directory.

In this work, you should have hemda for simulation.

After collecting all the necessary files, you can run simulations. For example, EMDA projects can be simulated via

```
$ mpirun -np <number of mpi tasks> hemda <parameter file name>
```

All parameter files can be found in <https://github.com/hlim88/EMDA-material/tree/master/params>.

You may not need to change many parameters but the following do are some key parameters that are used particularly for EMDA project.

- `dil_mass`, `dil_alpha`, `dil_infty`

Parameters of the dilaton field.

- `axn_mass`, `axn_alpha`, `axn_infty`

Parameters of the axion field.

- `geometry_type`

Options for various BH initial data. Put numbers to choose initial data. Options for relevant EMDA projects are:

- 11: Garfinkle-Horowitz-Strominger BH
- 12: Horne-Horowitz-Kaluza-Klein BH
- 13: Kerr-Sen BH
- 14: Rasheed-Larsen BH
- 15: BBH for EMDA

- `sf_amp`

Initial amplitude for the dilaton field

- `sf_amp_axn`
Initial amplitude for the axion field
- `sen_alpha`
 α constant for Kerr-Sen BH. If you are not choosing `geometry_type = 13`, this value defaults to zero

E.1.2 Dendro-GR

Dendro-GR is a distributed memory partial differential equations (PDEs) solver using numerical methods such as Finite Differences, Finite Elements, the Wavelet Method etc. It uses adaptive octree meshes as the geometric discretization. This code is used in Chap 3 to study BH IMRIs.

The DENDRO simulation code is freely available at GitHub (<https://github.com/paralab/Dendro-GR>) under the MIT License. The latest version of the code can be obtained by cloning the repository

```
$ git clone git@github.com:paralab/Dendro-GR.git
```

The following dependencies are required to compile DENDRO

- C/C++ compilers with C++11 standards and OpenMP support
- MPI implementation (e.g. `openmpi`, `mvapich2`)
- ZLib compression library (used to write `.vtu` files in binary format with compression enabled)
- BLAS and LAPACK are optional and not needed for the current version of DENDRO
- CMake 2.8 or higher version

Note: We have tested the compilation and execution of DENDRO with `intel`, `gcc` 4.8 or higher, `openmpi`, `mpich2` and `intelmpi` and `craympi` (in **Titan**) using the linux operating systems.

To compile the code, execute the following commands

```
$ cd <path to DENDRO directory >
$ mkdir build
$ cd build
$ cmake ../
```

The following options for DENDRO can then be set in cmake:

- `DENDRO_COMPUTE_CONSTRAINTS`: Enables the computation of Hamiltonian and momentum constraints.
- `DENDRO_CONSEC_COMM_SELECT`: If `ON` sub-communicators are selected from consecutive global ranks, otherwise sub-communicators are selected from a complete binary tree of global ranks (note that in this case the global communicator size needs to be a power of two).
- `DENDRO_ENABLE_VTU_CONSTRAINT_OUT`: Enables constraint variable output while time-stepping.
- `DENDRO_ENABLE_VTU_OUTPUT`: Enables evolution variable output while time-stepping.
- `DENDRO_VTK_BINARY`: If `ON` vtu files are written in binary format, else ASCII format (binary format is recommended).
- `DENDRO_VTK_ZLIB_COMPRES`: If `ON` binary format is compressed (only effective if `DENDRO_VTK_BINARY` is `ON`).
- `HILBERT_ORDERING`: Hilbert space filling curve is used if `ON`, otherwise the Morton curve is used. (The Hilbert curve is recommended to reduce communication costs.)
- `NUM_NPES_THRESHOLD`: When running in large scale set this to \sqrt{p} where p is the number of mpi tasks for better performance.

- `RK_SOLVER_OVERLAP_COMM_AND_COM`: If ON non blocking communication is used and enables overlapping of communication and computation *unzip* (recommended option), otherwise blocking synchronized *unzip* is used.

After configuring DENDRO, generate the Makefile (use `c` to configure and `g` to generate). Then execute `make all` to build all the targets. On completion, `bssnSolver` will be the main executable as related to the current project.

E.1.2.1 Getting Started: Running `bssnSolver`

`bssnSolver` can be run as follows.

```
$ mpirun -np <number of mpi tasks >\
  ./bssnSolver \
  <parameter file name>.par
```

Example parameter files can be found in `BSSN_GR/pars/`. The following is an example parameter file for equal mass ratio binary inspirals.

```
{
  "DENDRO_VERSION": 5.0,
  "BSSN_RESTORE_SOLVER": 0,
  "BSSN_IO_OUTPUT_FREQ": 10,
  "BSSN_REMESH_TEST_FREQ": 5,
  "BSSN_CHECKPT_FREQ": 50,
  "BSSN_VTU_FILE_PREFIX": "bssn_gr",
  "BSSN_CHKPT_FILE_PREFIX": "bssn_cp",
  "BSSN_PROFILE_FILE_PREFIX": "bssn_rl",
  "BSSN_DENDRO_GRAIN_SZ": 100,
  "BSSN_ASYNC_COMM_K": 4,
  "BSSN_DENDRO_AMR_FAC": 1e0,
  "BSSN_WAVELET_TOL": 1e-4,
  "BSSN_LOAD_IMB_TOL": 1e-1,
  "BSSN_RK_TIME_BEGIN": 0,
  "BSSN_RK_TIME_END": 1000,
  "BSSN_RK_TIME_STEP_SIZE": 0.01,
  "BSSN_DIM": 3,
  "BSSN_MAXDEPTH": 12,
  "ETA_CONST": 2.0,
  "ETA_R0": 30.0,
  "ETA_DAMPING": 1.0,
  "ETA_DAMPING_EXP": 1.0,
  "BSSN_LAMBDA": {
  "BSSN_LAMBDA_1": 1,
  "BSSN_LAMBDA_2": 1,
```

```

"BSSN_LAMBDA_3": 1,
"BSSN_LAMBDA_4": 1
},
"BSSN_LAMBDA_F": {
"BSSN_LAMBDA_F0": 1.0,
"BSSN_LAMBDA_F1": 0.0
},
"CHI_FLOOR": 1e-4,
"BSSN_TRK0": 0.0,
"KO DISS SIGMA": 1e-1,
"BSSN_BH1": {
"MASS":0.48528137423856954,
"X": 4.00000000e+00,
"Y":0.0,
"Z": 1.41421356e-05,
"V_X": -0.00132697,
"V_Y": 0.1123844,
"V_Z": 0,
"SPIN": 0,
"SPIN_THETA":0,
"SPIN_PHI": 0
},
"BSSN_BH2": {
"MASS":0.48528137423856954,
"X": -4.00000000e+00,
"Y":0.0,
"Z":1.41421356e-05,
"V_X": 0.00132697,
"V_Y": -0.1123844,
"V_Z": 0,
"SPIN": 0,
"SPIN_THETA":0,
"SPIN_PHI": 0
}
}

```

Here we list a few key options for `bssnSolver` with a short description.

- `BSSN_RESTORE_SOLVER`: Set to 1 to restore *RK* solver from latest checkpoint.
- `BSSN_IO_OUTPUT_FREQ`: IO (i.e. `vtu` files) output frequency
- `BSSN_CHECKPT_FREQ`: Checkpoint file output frequency
- `BSSN_REMESH_TEST_FREQ`: Remesh test frequency (i.e. frequency in time steps that is being tested for re-meshing)
- `BSSN_DENDRO_GRAIN_SZ`: Number of octants per core

- BSSN_ASYNC_COMM_K: Number of variables that are being processed during an asynchronous *unzip* (< 24)
- BSSN_DENDRO_AMR_FAC: Safety factor for coarsening i.e. coarsen if and only if $W_c \leq AMR_FAC \times WAVELET_TOL$ where W_c is the computed wavelet coefficient.
- BSSN_WAVELET_TOL: Wavelet tolerance for WAMR.
- BSSN_MAXDEPTH: Maximum level of refinement allowed (≤ 30)
- KO_DISS_SIGMA: Kreiss-Oliger dissipation factor for BSSN formulation
- MASS: Mass of the black hole
- X: x coordinate of the black hole
- Y: y coordinate of the black hole
- Z: z coordinate of the black hole
- V_X: momentum of the black hole in x direction
- V_Y: momentum of the black hole in y direction
- V_Z: momentum of the black hole in z direction
- SPIN: magnitude of the spin of the black hole
- SPIN_THETA: magnitude of the spin of the black hole along θ
- SPIN_PHI: magnitude of the spin of the black hole along ϕ

E.1.2.2 Generating Your Own Parameters

The initial data parameters for a binary black hole [215] depend on the total mass ($M = m_1 + m_2$), the mass ratio q and the separation distance d . These parameters are calculated using the Python script `BSSN_GR/scripts/id.py`. The command to generate parameters for $q = 10$, total mass $M = 5$ and separation $d = 16$ is

```
$ python3 id.py -M 5 -r 10 16
```

```
PUNCTURE PARAMETERS (par file format)
```

```
"BSSN_BH1": {
"MASS":4.489529,
"X":1.454545,
"Y":0.000000,
"Z": 0.000014,
"V_X": -0.020297,
"V_Y": 0.423380,
"V_Z": 0.000000,
"SPIN": 0.000000,
"SPIN_THETA":0.000000,
"SPIN_PHI": 0.000000
},
"BSSN_BH2": {
"MASS":0.398620,
"X":-14.545455,
"Y":0.000000,
"Z":0.000014,
"V_X": 0.020297,
"V_Y": -0.423380,
"V_Z": 0.000000,
"SPIN": 0.000000,
"SPIN_THETA":0.000000,
"SPIN_PHI": 0.000000
}
}
```

The tangential momentum is just an estimate, and the value for a
for a circular orbit is likely between (0.5472794147860968, 0.29947988193805547)

E.1.2.3 Symbolic Interface and Code Generation

The BSSN formulation is a decomposition of the Einstein equations into 24 coupled hyperbolic PDEs. Writing the computation code for the BSSN formulation can be a tedious task. Hence we have written a symbolic Python interface to generate optimized C code to compute the BSSN equations. All the symbolic utilities necessary to write the BSSN formulation in symbolic Python can be found in `GR/rhs_scripts/bssn/dendro.py` and the symbolic BSSN code can be found in `GR/rhs_scripts/bssn/bssn.py`. This could be modified for more advanced uses of the code such as including new equations to describe additional physics or for introducing a different formulation of the Einstein equations.

E.1.2.4 Profiling the Code

DENDRO contains built-in profiler code which enables one to profile the code extensively. On configuration, a user can enable/disable the internal profiling flags using `ENABLE_DENDRO_PROFILE_COUNTERS` and the profile output can be changed between a human readable version and a tab separated format using the flag `BSSN_PROFILE_HUMAN_READABLE`.

Note that in order to profile communication, internal profile flags need to be enabled. The following is an example of profiling output for the first 10 time steps.

```

active npes: 16
global npes: 16
current step: 10
partition tol: 0.1
wavelet tol: 0.0001
maxdepth: 12
Elements: 4656
DOF(zip): 279521
DOF(unzip): 2078609
===== MESH =====
step          min( #)   mean( #)  max( #)
ghost Elements 634      824.062   1065
local Elements 263      291       319
ghost Nodes   43781    55671.7   71693
local Nodes   14292    17470.1   20705
send Nodes    18760    24872.9   36861
recv Nodes    18113    24872.9   33777
===== RUNTIME =====
step          min(s)    mean(s)   max(s)
++2:l balance 0         0         0
++mesh        1.9753    1.98299   1.98946
++rkstep      20.159    20.1856   20.1996
++ghostExchge. 1.81442   3.15703   4.49568
++unzip_sync  8.27839   9.67293   11.0991
++unzip_async 0         0         0
++isReMesh    0.04642   0.117357  0.207305
++gridTransfer 1.53709   1.54899   1.56531
++deriv       1.98942   2.34851   2.76695
++compute_rhs 4.00119   4.61547   5.11566
--compute_rhs_a 0.0137962 0.0245449 0.0351532
--compute_rhs_b 0.0296426 0.0503471 0.069537
--compute_rhs_gt 0.111898  0.12846   0.15463
--compute_rhs_chi 0.0170642 0.0315392 0.044856
--compute_rhs_At 2.40738   2.72922   3.05622
--compute_rhs_K 0.358215  0.39879   0.457139
--compute_rhs_Gt 0.774211  0.933581  1.05702
--compute_rhs_B 0.0575426 0.071209  0.0855094
++boundary con 0         0.0421986 0.134712
++zip         0.23529   0.260862  0.291513
++vtu        0.0872362 0.101362  0.128246
++checkpoint  3.27e-06  3.85e-06  5.7469e-06

```


derivative	grid points	$\ \cdot\ _2$	$\ \cdot\ _\infty$
∂_x	4913	0.0201773	0.00144632
∂_x	99221	0.000849063	2.74672e-05

Table E.1 The normed difference between the numerical and analytic derivatives evaluated at grid points for the function $f(x, y, z) = \sin(2\pi x) \sin(2\pi y) \sin(2\pi z)$. In both cases, the wavelet tolerance is 10^{-8} while the maximum refinement level (i.e. `maxDepth`) has been increased from 4 to 6. Note that when `maxDepth` increases, the number of grid points increases. Hence, the normed difference goes down significantly.

In this section, we present experimental evaluations that we performed to ensure the accuracy of the simulation code.

E.1.2.5 Accuracy of Stencil Operators

In order to test the accuracy and convergence of WAMR and the derivative stencils, we used a known function to generate an adaptive octree grid based on the wavelet expansion. Then we compute numerical derivatives using finite difference stencils which are compared against the analytical derivatives of $f(x, y, z)$. L_2 and L_∞ norms of the comparison are given in Table E.1.

E.1.2.6 Accuracy of Symbolic Interface and Code Generation

Given the complexity of the BSSN equations, writing code to evaluate these equations can be an error-prone and tedious task. For example, to evaluate the equations we need to calculate more than 300 finite derivatives. Hence DENDRO provides a symbolic framework written in SymPy for automatically generating C++ code for the equations. The user writes equations in a high-level representation that more closely resembles their symbolic form. We can then use the computational graph of the equations to generate optimized C++ code. The accuracy of the symbolic framework and code generation is certified by comparing results from the generated C++ code to those from the HAD code, an established and tested code for numerical relativity.

This table shows a comparison of BSSN equations (i.e. all 24 equations), evaluated over a grid of 128^3 points by arbitrary, non-zero functions by both the DENDRO and HAD codes. All spatial derivatives in the equations are evaluated using finite differences, for both the DENDRO and HAD codes. The table reports the L_2 norm of the difference in the equations as evaluated in both codes, as well as the L_2 norms of the functions used to evaluate the equations. Equations with residual norms of order 10^{-15} are clearly at machine zero, but this low level is reached for only the simplest equations. Residuals with norms of order 10^{-12} arise in complicated equations, where finite precision errors can accumulate in hundreds of floating point operations. The optimized equations require about 4500 floating point operations to evaluate at a single point.

L_2 Norms differences in the HAD and DENDRO equations on 128^3 points.

diff 0 = 1.18413e-15, rhs HAD = 1.28114, rhs DENDRO = 1.28114
diff 1 = 7.53412e-16, rhs HAD = 2.18877, rhs DENDRO = 2.18877
diff 2 = 4.98271e-16, rhs HAD = 1.66315, rhs DENDRO = 1.66315
diff 3 = 1.03346e-15, rhs HAD = 0.720477, rhs DENDRO = 0.720477
diff 4 = 5.82489e-16, rhs HAD = 1.40142, rhs DENDRO = 1.40142
diff 5 = 4.93128e-16, rhs HAD = 0.797567, rhs DENDRO = 0.797567
diff 6 = 1.94194e-11, rhs HAD = 19.0107, rhs DENDRO = 19.0107
diff 7 = 2.14958e-11, rhs HAD = 19.5221, rhs DENDRO = 19.5221
diff 8 = 4.0673e-12, rhs HAD = 8.96364, rhs DENDRO = 8.96364
diff 9 = 1.58532e-11, rhs HAD = 10.1459, rhs DENDRO = 10.1459
diff 10 = 4.31184e-12, rhs HAD = 8.70053, rhs DENDRO = 8.70053
diff 11 = 4.95696e-12, rhs HAD = 10.8644, rhs DENDRO = 10.8644
diff 12 = 4.27211e-15, rhs HAD = 19.3546, rhs DENDRO = 19.3546
diff 13 = 2.29542e-10, rhs HAD = 93.4829, rhs DENDRO = 93.4829
diff 14 = 1.7484e-11, rhs HAD = 40.0804, rhs DENDRO = 40.0804
diff 15 = 4.79216e-11, rhs HAD = 30.9279, rhs DENDRO = 30.9279
diff 16 = 2.03434e-11, rhs HAD = 26.0603, rhs DENDRO = 26.0603
diff 17 = 1.07479e-15, rhs HAD = 15.5765, rhs DENDRO = 15.5765
diff 18 = 3.00151e-15, rhs HAD = 32.9891, rhs DENDRO = 32.9891
diff 19 = 7.79103e-16, rhs HAD = 8.10107, rhs DENDRO = 8.10107
diff 20 = 7.77113e-16, rhs HAD = 9.01369, rhs DENDRO = 9.01369
diff 21 = 1.74839e-11, rhs HAD = 46.3141, rhs DENDRO = 46.3141
diff 22 = 4.79217e-11, rhs HAD = 32.7981, rhs DENDRO = 32.7981
diff 23 = 2.03434e-11, rhs HAD = 32.7256, rhs DENDRO = 32.7256

E.1.2.7 Constraint Equations

Similar to the Maxwell equations of electrodynamics, the Einstein equations contain both hyperbolic evolution equations and elliptic constraint equations, which must be satisfied at all times. Following the common practice in numerical relativity, we evolve the hyperbolic

Time (M)	$\ \mathcal{H}_{r>a}\ _2$	$\ M1_{r>a}\ _2$	$\ M2_{r>a}\ _2$	$\ M3_{r>a}\ _2$
0	0.000777861	1.01855e-05	1.23443e-05	7.17572e-06
0.976562	0.000808294	3.05681e-05	2.91217e-05	2.79631e-05
1.95312	0.000793783	5.03912e-05	3.93872e-05	4.06693e-05
2.92969	0.00079551	7.54643e-05	5.068e-05	5.42466e-05
3.90625	0.000956987	0.000102901	7.38208e-05	7.47156e-05
4.88281	0.00247348	0.000200055	0.000140583	0.000139008

Table E.2 Violation of constraint equations with time for an equal mass ratio binary merger simulation done using OT. Note that \mathcal{H} , $M1$, $M2$, $M3$ denotes the Hamiltonian and 3 momentum component constraints that is being monitored through the evolution.

equations and monitor the quality of the solution by checking that the constraint equations are satisfied. The choice of coordinates for the BBH evolution (the puncture gauge) does induce constraint violations in the vicinity of each black hole. The violations of the constraint equations in our runs are consistent with the discretization error expected for the numerical derivatives in the constraint equations and the constraint violations near the black holes (punctures). An example of monitored constraint violations are listed in the Table E.2.

E.1.2.8 Single Black Hole

Prior to simulating binary inspirals, we perform simple experiments with a single black hole to ensure the accuracy of the simulation code. While a single black hole is a stable, static solution of the Einstein equations, there is some transient time dependence due to our particular coordinate conditions. The black hole parameters for this test is given below. (The parameter file can be found in SC18_AE/par/single_bh1.par in the repository.) Note that to generate initial data for a single black hole, we place one of the black holes in the binary far from the computational domain and set its mass to zero.

```
"BSSN_BH1": { "MASS":1.0, "X":0.0, "Y":0.0,
```

```

"Z": 0.00123e-6, "V_X": 0.0, "V_Y": 0.0,
"V_Z": 0.0, "SPIN": 0,
"SPIN_THETA":0, "SPIN_PHI": 0 },

"BSSN_BH2": { "MASS":1e-15,"X":1e15, "Y":0.0,
"Z":0.00123e-6, "V_X": 0.0, "V_Y": 0.0,
"V_Z": 0.0, "SPIN": 0,
"SPIN_THETA":0, "SPIN_PHI": 0 }

```

E.1.2.8.1 Boosted Single Black Hole The next experiment is an extension of the single BH test. It “boosts” the BH with constant velocity in the x -direction. The constant velocity of the BH should be apparent in the evolution. The parameter file for this test can be found in the repository at `SC18_AE/par/single_bh1_boost.par`. The black hole parameters are given below (note that *BSSN_BH1* has a momentum of 0.114 in x -direction).

```

"BSSN_BH1": { "MASS":1.0, "X":0.0, "Y":0.0,
"Z": 0.00123e-6, "V_X": 0.114, "V_Y": 0.0,
"V_Z": 0.0, "SPIN": 0,
"SPIN_THETA":0, "SPIN_PHI": 0 },

"BSSN_BH2": { "MASS":1e-15,"X":1e15, "Y":0.0,
"Z":0.00123e-6, "V_X": 0.0, "V_Y": 0.0,
"V_Z": 0.0, "SPIN": 0,
"SPIN_THETA":0, "SPIN_PHI": 0 }

```

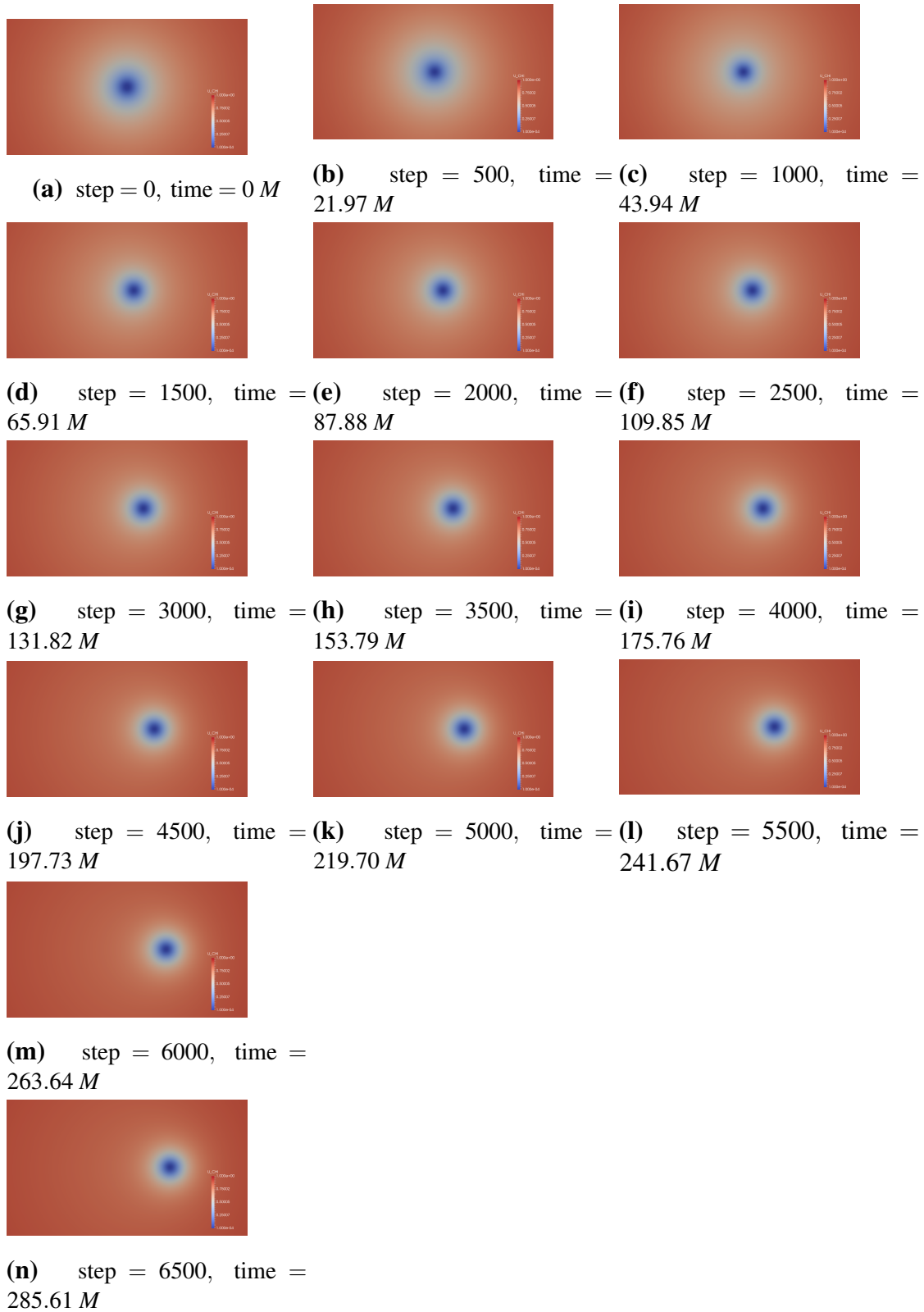


Figure E.1 A single black hole boosted in the x -direction, with MAXDEPTH=12 and wavelet tolerance of 10^{-3} . Time is given in terms of the black hole mass, M .

E.1.2.9 Binary Black Holes

E.1.2.9.1 Mass ratio $q = 1$ We performed a series of short-term binary BH evolutions with different mass ratios. This run was used to validate the code by comparing the trajectories of the BHs calculated using DENDRO to the trajectories calculated by HAD. Frames from the evolution are shown in the Figure E.3, and the BH parameters used for this run are listed below.

```
"BSSN_BH1": {
  "MASS":0.485 ,
  "X": 4.00e+00, "Y":0.0 , "Z": 1.41-05,
  "V_X": -0.00133, "V_Y": 0.112, "V_Z": 0,
  "SPIN": 0, "SPIN_THETA":0, "SPIN_PHI": 0 },
"BSSN_BH2": {
  "MASS":0.485 ,
  "X":-4.00+00, "Y":0.0 , "Z":1.41-05,
  "V_X": 0.00132, "V_Y": -0.112, "V_Z": 0,
  "SPIN": 0, "SPIN_THETA":0, "SPIN_PHI": 0 }
```

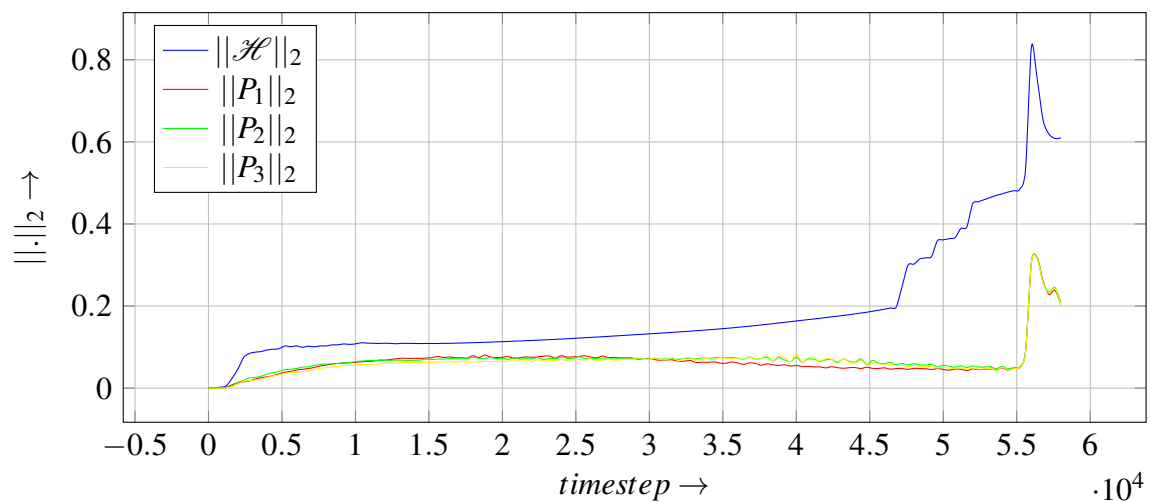


Figure E.2 L_2 norm of the Hamiltonian and momentum constraint violation as hyperbolic equations are evolved in time for equal mass ratio binary black hole configuration test run performed using DENDRO-GR. Note that the final spike in constraint violation happens during the merging event of the binary black holes.

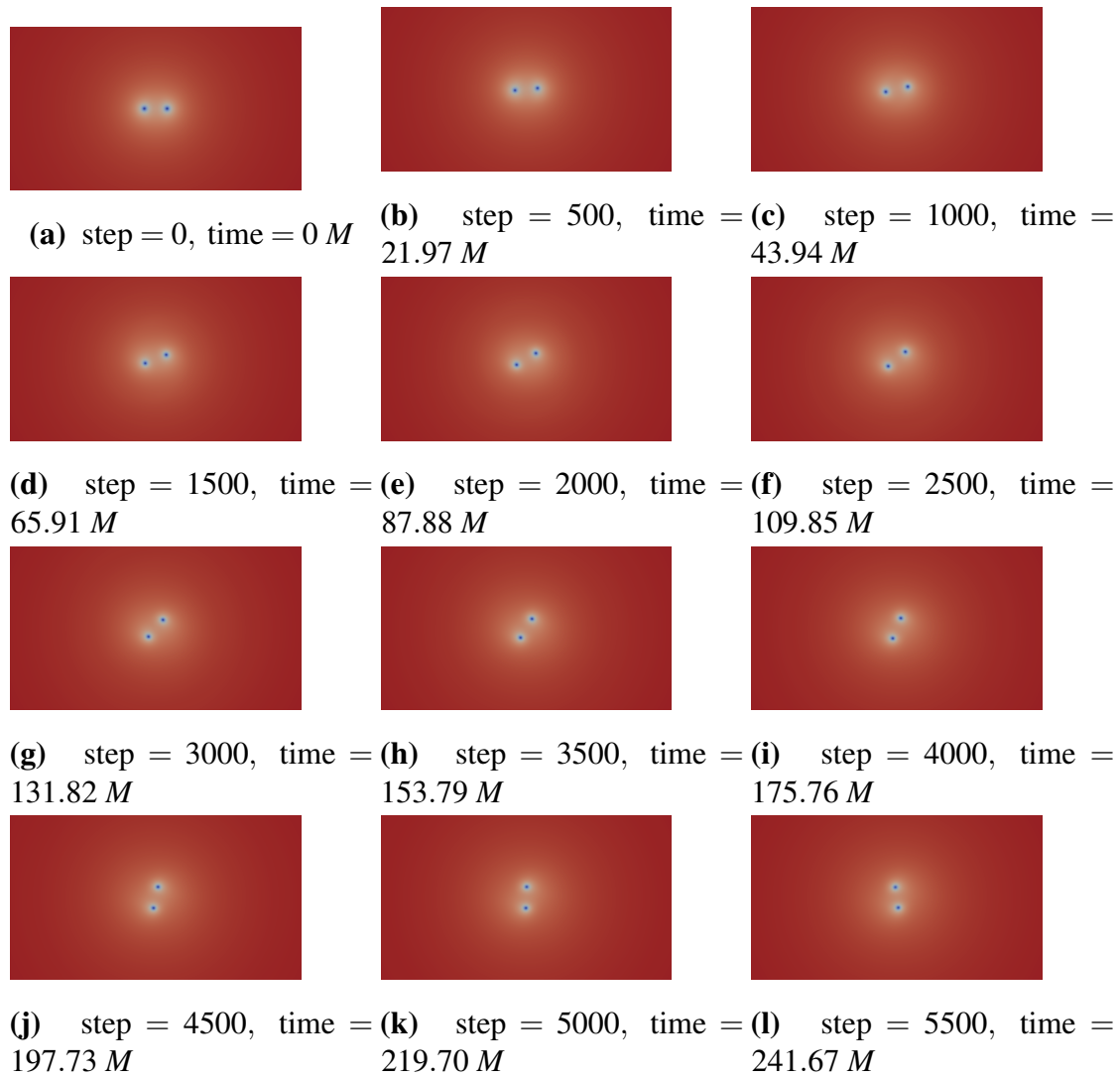


Figure E.3 This figure shows frames from the evolution of a black hole binary with an equal mass ratio, $q = 1$. Time is measured in terms of the total black hole mass M .

E.1.2.9.2 Mass ratio $q = 10$ We performed a short simulation with a mass ratio $q = 10$. This is a short demonstration run to show that DENDRO easily handles large mass ratios and gives consistent results for the binary evolution. Frames from the evolution are shown in Figure E.4, and the BH parameters used for this run are listed below.

```
"BSSN_BH1": {
  "MASS":0.903 ,
  "X":5.45-01 , "Y":0.0 , "Z": 1.41-05 ,
  "V_X": -3.90e-04 , "V_Y": 0.0470 , "V_Z": 0 ,
  "SPIN": 0 , "SPIN_THETA":0 , "SPIN_PHI": 0 } ,
"BSSN_BH2": {
  "MASS":0.0845 ,
  "X":-5.45+00 , "Y":0.0 , "Z":1.41-05 ,
  "V_X": 3.90e-04 , "V_Y": -0.0470 , "V_Z": 0 ,
  "SPIN": 0 , "SPIN_THETA":0 , "SPIN_PHI": 0 }
```

E.1.2.9.3 Mass ratio $q = 100$ We performed a short simulation with a mass ratio $q = 100$. This is a short demonstration run to show that DENDRO produces the proper grid structure for this system and reasonable results for a very challenging binary configuration. Frames from the evolution are shown in Figure E.5 and the BH parameters used for this run are listed below.

```
"BSSN_BH1": {
  "MASS":0.989 ,
  "X":5.94-02 , "Y":0.0 , "Z": 1.41-05 ,
  "V_X": -5.60-06 , "V_Y": 5.61-03 , "V_Z": 0 ,
  "SPIN": 0 , "SPIN_THETA":0 , "SPIN_PHI": 0 } ,
"BSSN_BH2": {
  "MASS":0.00914 ,
  "X":-5.94+00 , "Y":0.0 , "Z":1.41421356e-05 ,
  "V_X": 5.60-06 , "V_Y": -5.61-03 , "V_Z": 0 ,
```

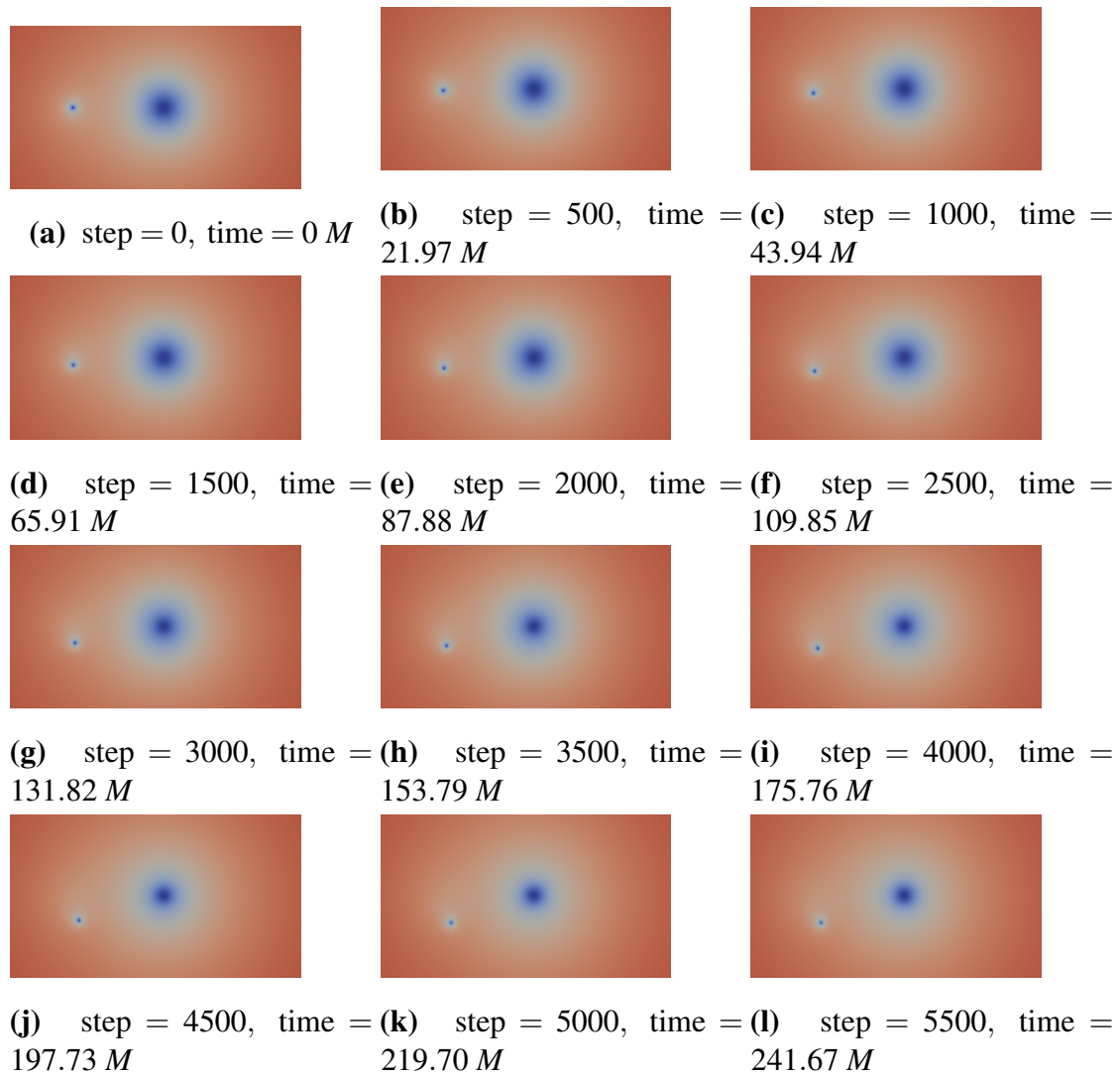



Figure E.4 This figure shows frames from the evolution of a black hole binary with mass ratio $q = 10$. Time is measured in terms of the total black hole mass M .

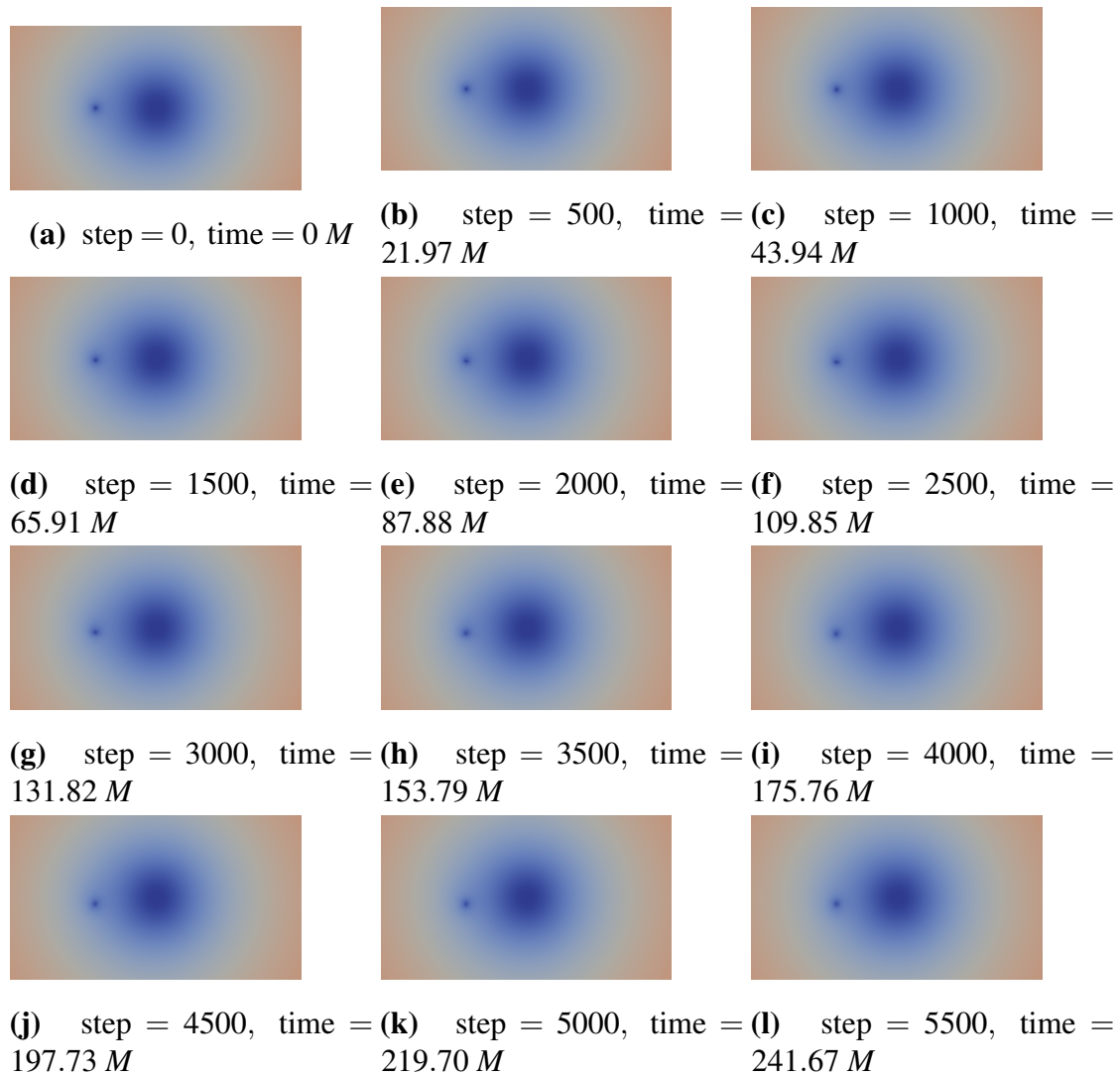


Figure E.5 This figure shows frames from the evolution of a black hole binary with mass ratio $q = 100$. Time is given in terms of the total mass M .

```
"SPIN": 0, "SPIN_THETA":0, "SPIN_PHI": 0 }
```

E.1.3 FleCSPH

FleCSPH implements the method of smoothed particle hydrodynamics (SPH) in order to simulate fluids and gases using the FleCSI framework. Currently, particle affinity and gravitation is handled using the parallel implementation of the octree data structure provided by FleCSI.

This code is used in Chap 4 to study BNS mergers. To build the code, please follow the instructions at <https://github.com/laristra/flecsph#building-the-flecsph>.

E.1.3.1 Running FleCSPH Applications

Current FleCSPH contains several initial data generators and two evolution drivers: hydro and Newtonian. Initial data generators are located in `app/id_generators/`:

- `sodtube`: 1D/2D/3D sodtube shock test;
- `sedov`: 2D and 3D Sedov blast wave;
- `noh`: 2D and 3D Noh implosion test.
- etc.

Evolution drivers are located in `app/drivers`:

- `hydro`: 1D/2D/3D hydro evolution without gravity;
- `newtonian`: 3D hydro evolution with self-gravity.

To run a test, one needs a valid input parameter file which specify the parameters of the problem. The link at <https://github.com/laristra/flecsph/tree/master/data> shows parameter files for various simulations. One can easily copy one of these parameter files and use it. In the next section, we provide a description of some of these parameters. Running an application consists of two steps:

- generating initial data;
- running evolution code.

For instance, to run a `sodtube` 1D shock test, do the following (assuming you are in your build directory after having successfully built FleCSPH):

```
cp ../data/sodtube_t1_n1000.par sodtube.par
# edit the file sodtube.par to adjust the number of particles etc.
app/id_generators/sodtube_1d_generator sodtube.par
app/driver/hydro_1d sodtube.par
```

E.1.3.2 Description of Parameters

In this section, we provide key parameters that are used in any type of application

- `sph_variable_h`
yes: Particles can have variable smoothing length over and in each iteration. no:
Particles will have the same smoothing length in each iteration.
- `initial_dt` (default 1)
Value of the initial timestep
- `adaptive_timestep` (default no)
yes: The timestep will be recomputed at every iteration based on either: acceleration,
viscosity and energy.
no: Constant timestep.
- `timestep_cfl_factor` (default 0.25)
Courant-Friedrichs-Lewy factor (between the timestep and particle separation)
- `initial_iteration` (default 0)
initial iteration
- `final_iteration` (default, 10)
final iteration (= total iterations + 1, if counting from 0)
- `initial_time` (default, 0)
initial time

- `final_time` (default, 1.0)
final time
- `initial_dt` (default, 0.001)
initial timestep

E.1.4 PAMR/AMRD lib

AMRD is a library consisting of a main function which implements a Berger and Olinger (B&O) style adaptive mesh refinement (AMR) driver, with modifications to allow one to solve elliptic equations within the B&O time stepping framework. In addition, PAMR is a library of routines designed to manage distributed grid hierarchies with AMRD. The project code can be found in <https://github.com/hlim88/PBH-NS>.

This code is used in Chap 5 to study interactions of small mass BH with NS

First, we need to build PAMR/AMRD lib. To build this lib, you will need the following dependencies:

- C++14 compatible compiler such as `gcc >=5.4`
- MPI lib, compiled with the compiler above
- Basic GSL lib
- HDF5 lib

Once you have all these, you need to configure via:

```
./configure -prefix=/usr/local
```

(If you are trying to build this code on a supercomputer, you may want to configure it with `./configure -prefix=/home/<your username>/local`)

If no problems arise during configuration, type `make install` to build the lib. If the lib has been correctly built you should find `libpamr.a` and `libamrd.a` in `/usr/local` (or `/home/<your username>/local`)

Next, we need to build the code. If you have built PAMR/AMRD correctly, you can simply type `make` to compile the code. You will see `ss_gr_hydroSolver`. The link at https://github.com/hlim88/PBH-NS/tree/master/ss_gr_hydro/Instruction shows different makefiles with respect to various architecture. You can also use these for future reference.

After you have successfully built the code, you can simply use `runScript` to run the code. This script can be found in https://github.com/hlim88/PBH-NS/tree/master/ss_gr_hydro/run.

E.1.4.1 Parameter Generation

There are two different types of parameter files: `.rtparam` and `.fparam`.

`.fparam` contains fixed parameters that define evolution variables and AMR related variables. Most users will not need to change this parameter file unless they introduce new variables.

`.rtparam` contains run-time parameters that are used for simulations. Also, you can choose which variables are output.

Both `.rtparam` and `.fparam` can be found at <https://github.com/hlim88/PBH-NS/tree/master/par>. Note as well that both parameter files are required to start simulations. Check `runScript` (https://github.com/hlim88/PBH-NS/tree/master/ss_gr_hydro/run) on how to use these parameter files.

Below we show some key parameters for our work

- `Pressure_c`

Central pressure value for initial data

- `U_amp`

Initial fluid velocity. We find that 0.48639 provides collapsing into BH

- `MG_*`

Parameters with `MG_` are related with multigrid methods for solving elliptic equations.

If you are using BSSN form, you do not need these.

E.2 Visualization of Results

Data visualization tools and technologies are key to analyze massive amounts of information. By using visual elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand physical consequences in data. There are number of ways to visualize large data from industrial packages such as Matlab and Mathematica to lower level coding. In this work, we are using many different techniques to visualize our data but we review specific toolkits that could be adapted different projects

E.2.1 DV/xvs

[DV/xvs](#) is a set of visualization toolkits to aid in the development of numerical codes that solve partial differential equations via grid-based methods, in particular those utilizing adaptive mesh refinement (AMR) and/or running in a parallel environment. Both HAD and PAMR/AMRD generate output file for DVxvs/

E.2.1.1 Getting DV/xvs and Installation

Both DV and xvs require several dependencies. In particular they require

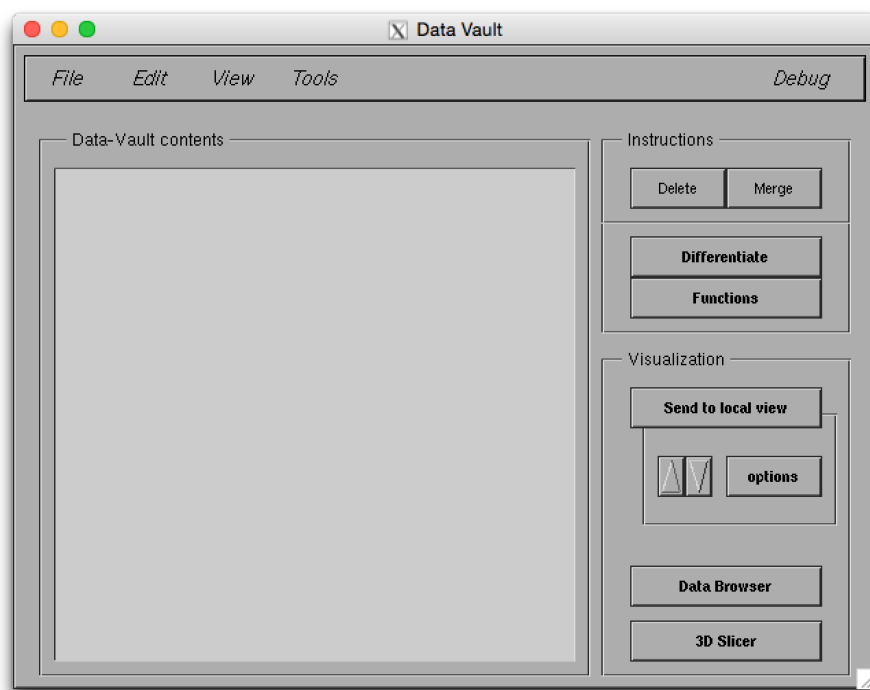
- C99 compatible compiler such as gcc >=4.8
- Fortran 77 compiler such as f77 or gfortran
- Perl
- OpenGL
- Jpeg libs for Mpeg encoding
- For linux user, Flex and GNU Bison
- XForms
- RNPL and related libs

Most of these dependencies can be downloaded from <http://laplace.physics.ubc.ca/Doc/rnpletal/>.

Note that there may be additional packages or libraries that are required due to different computer operating systems. For example, Ubuntu users may want to additionally install libform (<https://launchpad.net/ubuntu/precise/+source/libforms>) with xforms to satisfy some contingencies.

Once all the required packages have been downloaded, installation can be done by following the instructions at <http://laplace.physics.ubc.ca/Doc/rnpletal/>

After successful installation, starting DV/xvs can be done by simply typing DV (or xvs) in your terminal. Then you will see below start window.



E.2.1.2 Visualizing sdf Files

This toolkit is suitable to visualize sdf files. Once you have your simulation results, you can send the sdf files to this visualization server. Note that XVS aims to visualize 1D results and DV visualizes higher dimensional results but it also can visualize 1D result. In this section, we will concentrate on DV and review its basic usage. (xvs has a very similar structure so a user familiar with DV should fairly readily be able to pick up xvs.) More

detailed instructions for DV can be found at <http://laplace.physics.ubc.ca/Doc/DV/> and for *xvs* at <http://laplace.physics.ubc.ca/Doc/xvs/>

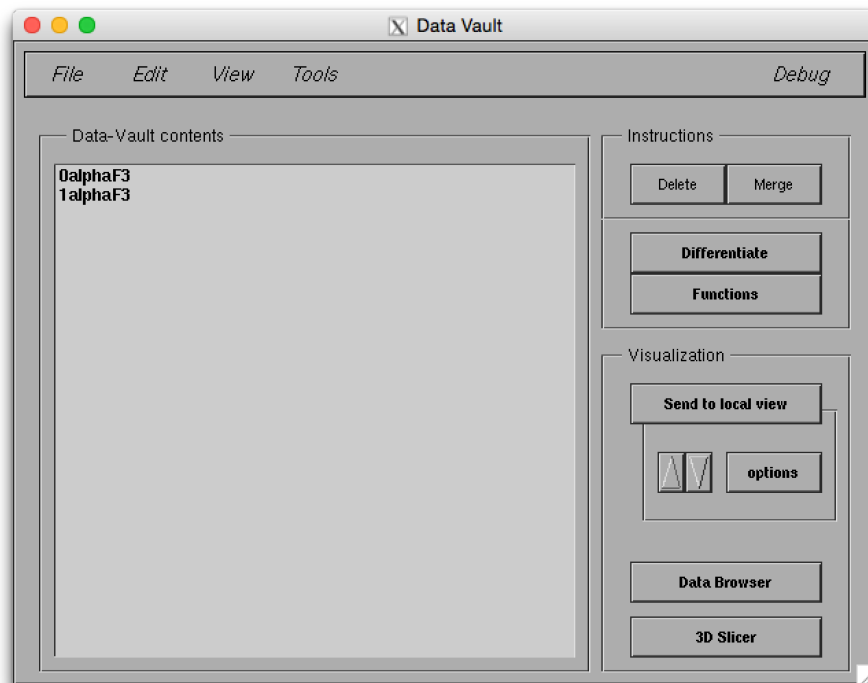
Starting DV can be done by typing DV at a command line on your local machine. Once DV is running, you can send data files that are in sdf format to the DV server using `sdftodv` (`sdftoxvs` for *xvs*) command. Before using `sdftodv`, you will need to specify the host address of the machine on which DV is running. You can do this by setting

```
export DVHOST=<your hostname> (for xvs, you need to specify XVSHOST).
```

In a different OS, you may need to set `env DVHOST <your hostname>`. After setting the hostname, you can send the sdf file through

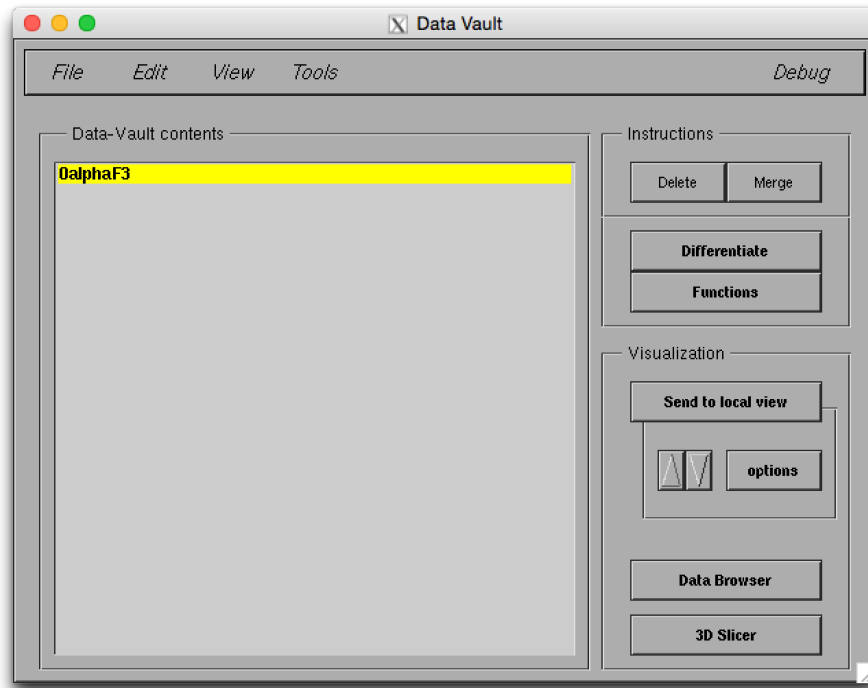
```
sdftodv -i <ivec> <your sdf file name>
```

where `ivec` is an option to control which data sets within the sdf files will be sent to DV. If you successfully send data sets to DV, you will see data register in the DV contents panel. An example is given below

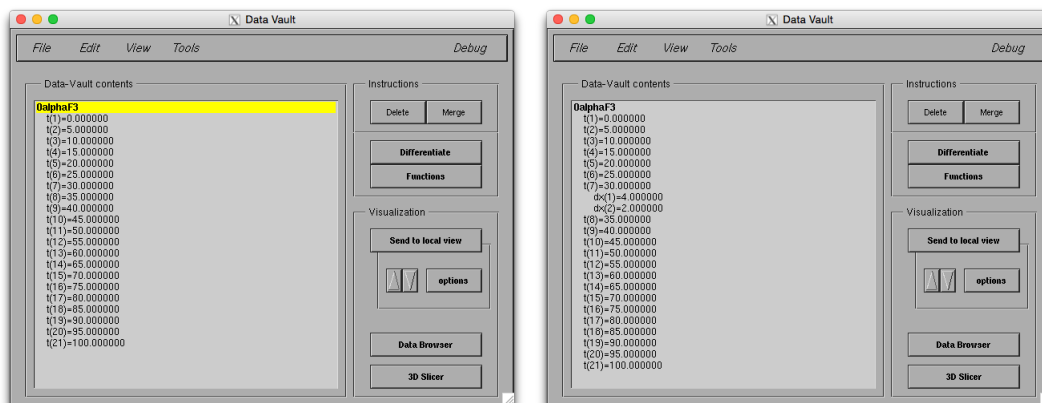


In the above example, we have two different registers that represent the lapse function (α). We can merge these two data into one single register that contains all the grid functions for a variable, and it will be identified by its time, resolution, and coordinate

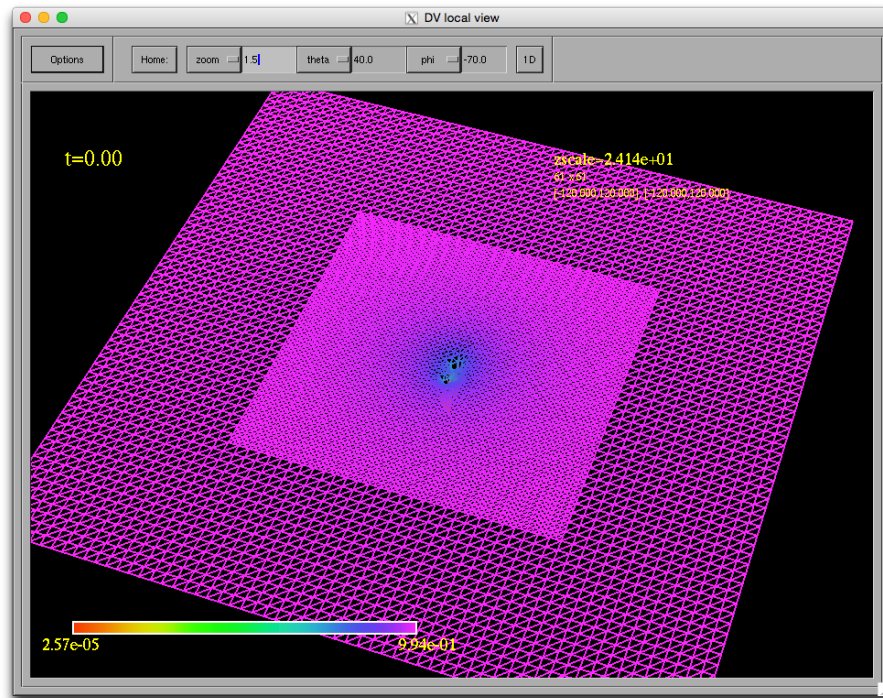
domain. The Merge button on the right side of the main window can be used to collate the different data sets. After this, you will see two data sets merge into one data set as in the picture below. Note that you can merge with a specific time rather than the whole register.



Also, you can expand the register to see different time and coordinate resolutions by double-clicking on the data contents as shown in the figure below.



To visualize your data, simply click the Send to local view button on the right side of the main window. After sending your data into local view, you will see a separate window that shows your data as in the figure below.



As you can see from above, there are different options for the visualization panel. See http://laplace.physics.ubc.ca/Doc/DV/sec_33.html#Other_functions for details of these functionalities.

E.2.2 Paraview

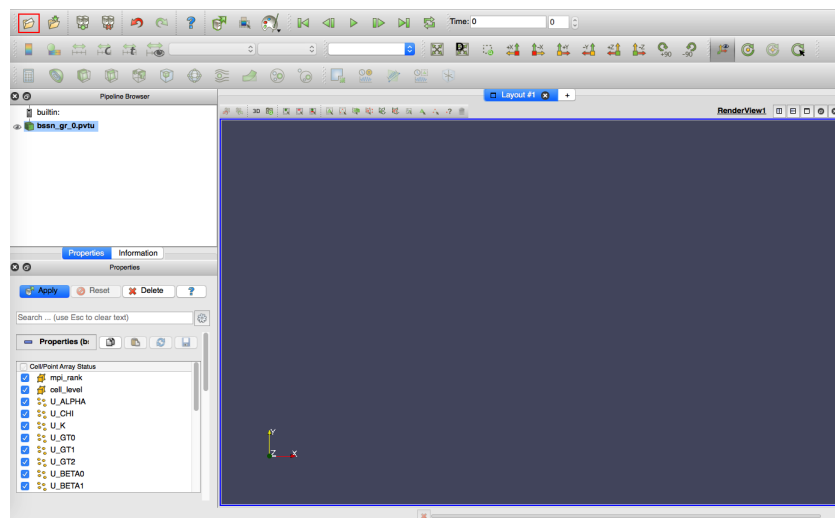
Paraview is an open source multiple-platform application for interactive, scientific visualization. It has a client-server architecture to facilitate remote visualization of datasets, and generates level of detail (LOD) models to maintain interactive frame rates for large datasets. It is an application built on top of the Visualization Toolkit (VTK) libraries. ParaView is an application designed for data parallelism on shared-memory or distributed-memory multicomputers and clusters. It can also be run as a single-computer application. It can be downloaded via <https://www.paraview.org/download/>. One can easily download and install Paraview on different operating systems. It also provides a user manual through download webpages.

After successfully installing Paraview, we are ready to plot our data. In this instruction,

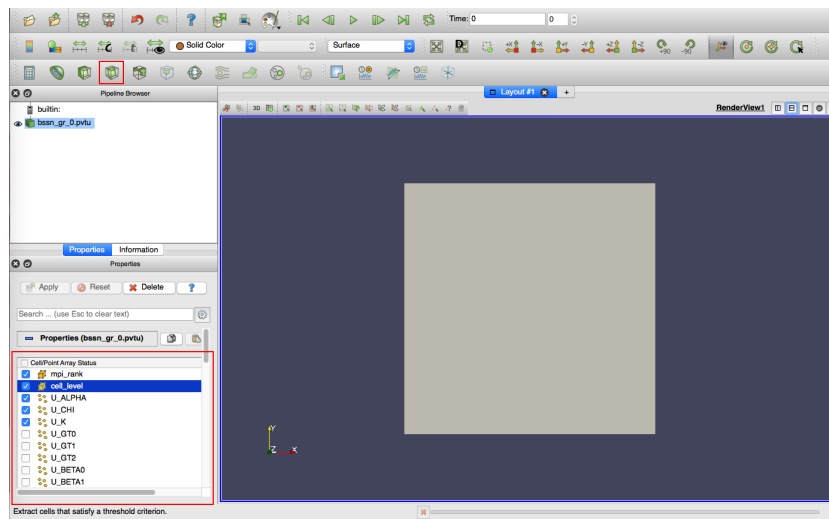
we only consider results from our BSSN_GR project. Other possible applications of and with Paraview can be found in the manual pages.

Here we use simple functionalities in Paraview. Once you finish your simulation, you will get a bunch of different output files. Here, we only need to consider `.pvtu` for visualization purposes. Our example below shows a step-by-step procedure to plot our binary black hole data.

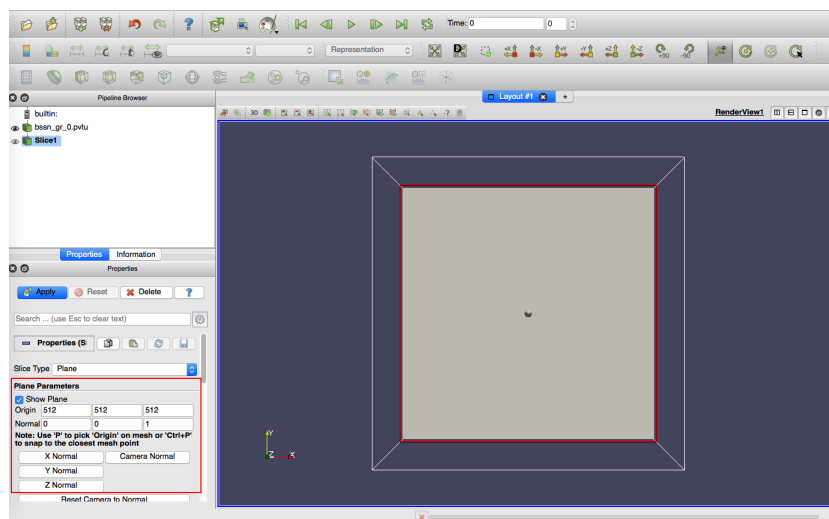
1. Click the Open icon (in the red box in the figure below) or `Ctrl-0` (Command-0 for mac) then load the `<filename>.pvtu` file into your directory. Once the data has been successfully loaded, you will see something akin to the figure below.



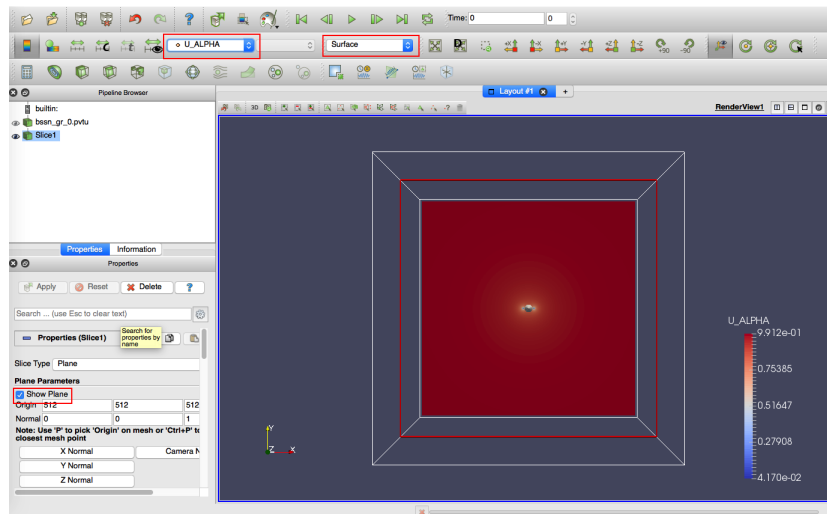
2. The lower left part of the panel shows your data fields (red-boxed area in figure). You can choose the data set that you want to plot. Here, the data fields show BSSN variables. After you choose your data, click Apply right above. Then you should see something to the picture below. This shows the whole 3D box of data that we have. To see the data more precisely, we slice the box. Click on the slice icon (the red-boxed icon in the third-from-the-top menu bar) or find the slice option in the Filters/Common tab



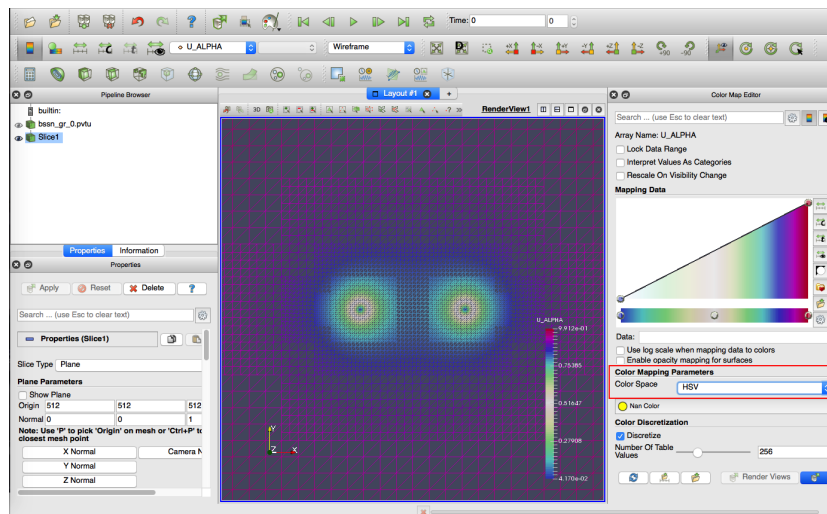
3. After applying Slice, you will see many options in the Properties section (the red-boxed area in the figure below). This shows the direction of the slice axes. You can choose your own coordinates or just choose X , Y , or Z axes. After choosing one of these options click Apply. You should see a picture similar to below.



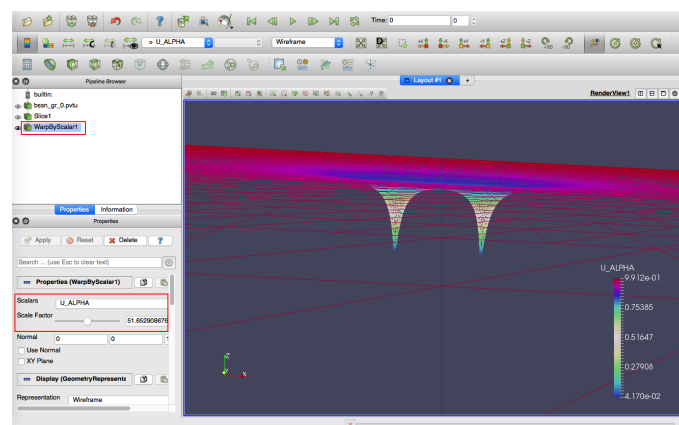
4. We can change the data field that we want to see. The default set up is Solid Color. You can change the fields via second top lines in the below figure (the first red-boxed area in the figure). In this example, U_Alpha is chosen. Also, Paraview has different representations such as Surface, Points, and Wireframe etc. The second red-boxed area in the figure shows different options for these representations. You can also change the representation in Properties section (left lower section). In this figure, we choose Surface. If you want to turn off the plane frame in the figure, disable Show Plane (the red-boxed area in the Properties section in below figure).



5. If you want to change the color scheme and/or the level, go to the View tab, then enable Color Map Editor. Once you enable, you can see mapping data in the color map editor panel. In many cases, the default color space is Diverging. In this example, I choose HSV as the color space and Wireframe as the representation.



6. With our previous steps, we have created a visualization of our data via a 2D surface. We would, of course, also like to elevate our data to see how each field scales. There are several different representations in Paraview. Here, we use Warp By Scalar. Go to the Filters/Alphabetical tab and then enable Warp By Scalar. After this, you should see a picture similar to below. The scaling can be changed via Scale Factor in the Properties window (red-boxed area in the left lower corner).



Individual image files can be saved and video can also be created of these results within

Paraview. Go to the File tab, and check Save Screenshot and Save Animation to make images and videos

These are just a few, basic features of Paraview used to plot our binary black hole simulation results. Additional functions include `clip` or `contour`. Check <https://www.paraview.org/download/> to learn about some of these additional features.

E.2.2.1 Visualize Data using a Script

Normally, production runs will generate huge output files. Using a personal laptop or sequential usage with Paraview can require a great deal of time for plotting the data. Fortunately, Paraview can be run in parallel.

Paraview has a function name `Trace` which records all your commands/usages during a plotting session. With `Trace`, it will generate a python script which will automatically perform all procedures saved. On accessing the Dendro-5.0 repository, one can find a python script called `bssnVis.py`. `bssnVis.py` was created from Paraview trace and includes additional parser features to generate images efficiently. This script can be found in the `Dendro-5.0/scripts` directory.

To use this, a user will need `PvBatch` and `PvPython`. `PvPython` is the Python interface to Paraview. You can think of `PvPython` as `ParaView` with a Python interface. `PvBatch` is like `PvPython`, with two exceptions. `PvBatch` only accepts commands from input scripts, and `PvBatch` will run in parallel if it was built using MPI. Input is exactly like `PvPython`.

In these instructions, we will use `bssnVis.py`. Additional details about these functions can be found at https://www.paraview.org/Wiki/ParaView_and_Batch and <https://www.paraview.org/Wiki/ParaView>

On successful installation of Paraview, `pvbatch` will be found in the bin directory of Paraview. For example, `pvbatch` should be in `/Users/<user name>/<path to Paraview build>/bin/` directory. The command to use `bssnVis.py` is:

```
pvbatch -b <beginning time steps> -e <end time> -f <frequency> -pvtu
<path>/bssn_gr
(or file name) -img <image saving path> (if not, save locally)/<image name>
```


As an example:

```
pvbatch bssnVis.py -b 0 -e 400 -f 50 -pvtu gr_sp1/bssn_gr -img img1/img_slice
```

On using this, we can run Paraview in parallel with:

```
mpirun -np <number of cores> pvbatch bssnVis.py -b 0 -e 400 -f 50 -pvtu  
gr_sp1/bssn_gr  
-img img1/img_slice
```

After generation, image files are labelled automatically. Using these image files, one can generate video files. For example, using ffmpeg:

```
ffmpeg -r 10 -i 0%03d.png <movie name>.mp4
```

For more details, please refer ffmpeg website (<https://www.ffmpeg.org/>).

We assume that most supercomputers will have Paraview as a module. So, you can follow either ways to plot the data. If you have more questions about it, please check the Paraview manual or contact Milinda Fernando (milindasf@gmail.com) for more specific inquiries.

E.2.2.2 pvbatch in Marylou

In this section, we explore how we can use pvbatch in marylou. First, a user will need specific modules. These include

```
module load gcc/6.4  
module load python/2.7_gcc-6  
paraview/5/4
```

Once the correct modules have been loaded, the script can be used in the normal way. If an error arises related to the X-server, a possible fix is to try with the no display option such that

```
DISPLAY=:0 pvbatch -b <beginning time steps> -e <end time> -f <frequency>  
-pvtu <path>/bssn_gr (or file name) -img <image saving path>/<image name>
```

This can be exported as the default with `export DISPLAY=:0`. pvbatch can be run in

parallel using the usual `mpirun` command.

E.2.2.3 Remote Visualization

Paraview supports remote access for visualization purposes. Instead of using a script, one accesses the supercomputer (or some other server) remotely and then visualization of data can happen without transferring the data. Here is a simple example of how can use this functionality. We use `marylou` as an example.

To begin using Paraview for remote data processing and visualization, we must first start the server application `pvserver` on the remote system. To do this, call the `paraview` module in `marylou` then type

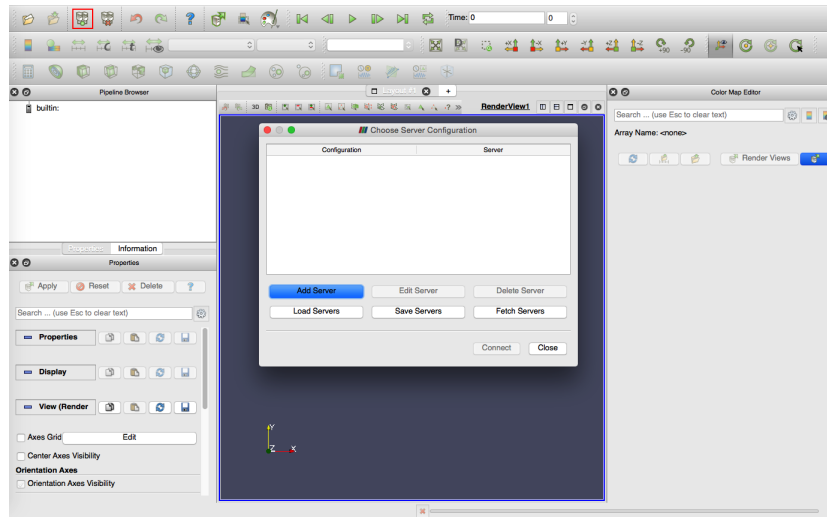
```
pvserver
```

You will see this startup message on the terminal

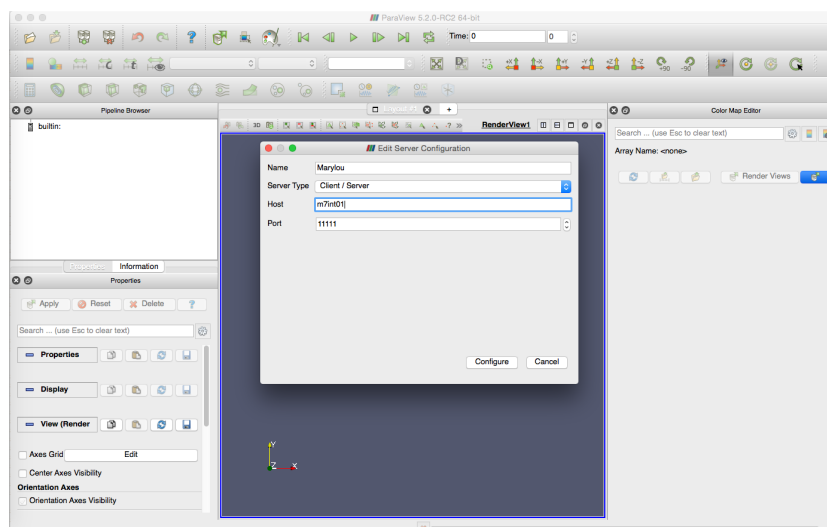
```
m7int01: hlim10$ pvserver
Waiting for client...
Connection URL: cs://m7int01:11111
Accepting connection(s): m7int01:11111
```

Note that you can run `pvserver` in parallel with the command `mpirun -np <number of cores> pvserver`.

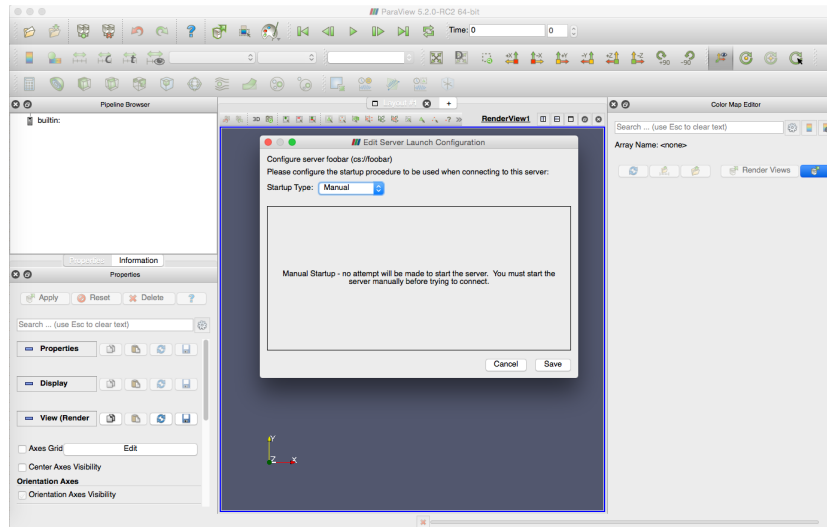
After this, run Paraview and then select File/Connect or click the button that is highlighted with the red box in the below figure (in the upper left portion of the screen). You should then see the Choose Server Configuration dialog box



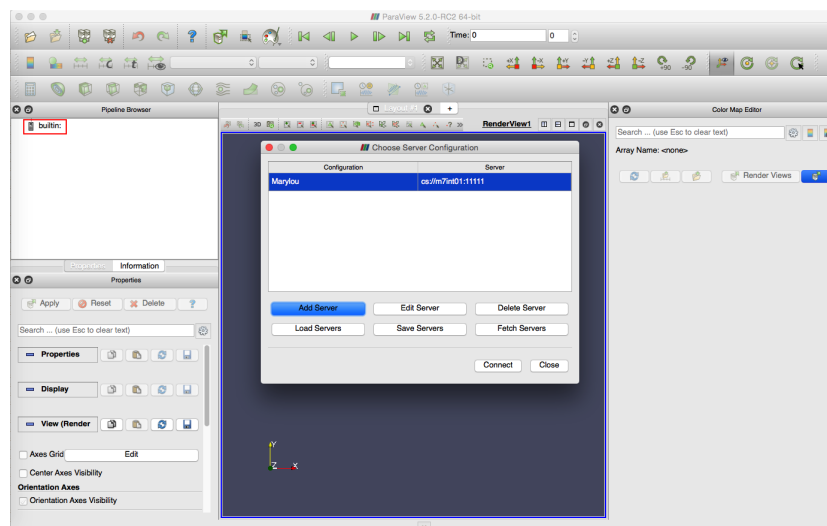
Now, an entry to the list must be added by clicking Add Server. At this point, you should see Edit Server Configuration. You will need to set a name for the connection, the server type, the DNS name of the host on which you just started the server, and the port.



When you are done, click the Configure button. Another dialog box, as shown in the figure below, will appear where you specify how to start the server. In this example, we started the server manually, we are not able to change this setting. Alternatively, we can optionally set the Startup Type to Command and specify an external shell command to launch a server.



After all of this, you should see a list similar to below. Clicking on Connect will connect to the server (in this example marylou). After a successful connection, you can use Paraview exactly the same as with a usual GUI environment. The only difference in user interfaces is the Pipeline Browser (red-boxed in left figure) which will reflect the name of the server to which you are connected.



Bibliography

- [1] L. S. Collaboration and V. collaboration, “Observation of Gravitational Waves from a Binary Black Hole Merger,” *Phys. Rev. Lett.* **116**, 061102 (2016).
- [2] B. P. Abbott *et al.*, “Observation of Gravitational Waves from a Binary Black Hole Merger,” *Phys. Rev. Lett.* **116**, 061102 (2016).
- [3] B. P. Abbott *et al.*, “GW151226: Observation of Gravitational Waves from a 22-Solar-Mass Binary Black Hole Coalescence,” *Phys. Rev. Lett.* **116**, 241103 (2016).
- [4] B. P. Abbott *et al.*, “GW170104: Observation of a 50-Solar-Mass Binary Black Hole Coalescence at Redshift 0.2,” *Phys. Rev. Lett.* **118**, 221101 (2017).
- [5] B. P. Abbott *et al.*, “GW170814: A Three-Detector Observation of Gravitational Waves from a Binary Black Hole Coalescence,” *Phys. Rev. Lett.* **119**, 141101 (2017).
- [6] The LIGO Scientific Collaboration and the Virgo Collaboration, “GWTC-1: A Gravitational-Wave Transient Catalog of Compact Binary Mergers Observed by LIGO and Virgo during the First and Second Observing Runs,” arXiv e-prints p. arXiv:1811.12907 (2018).
- [7] B. Abbott *et al.*, “GW170817: Observation of Gravitational Waves from a Binary Neutron Star Inspiral,” *Phys. Rev. Lett.* **119**, 161101 (2017).
- [8] S. Ghonge and K. JANI, “<https://www.ligo.org/detections/images/GWTC1-poster.png>”, <https://www.ligo.org/detections/images/GWTC1-poster.png>.

- [9] J. M. Weisberg, J. H. Taylor, and L. A. Fowler, “Gravitational waves from an orbiting pulsar,” *Scientific American* **245**, 74–82 (1981).
- [10] G. B. Cook, “Initial Data for Numerical Relativity,” *Living Reviews in Relativity* 3 (2000).
- [11] N. Stergioulas, “Rotating Stars in Relativity,” *Living Reviews in Relativity* 6 (2003).
- [12] J. M. Martí and E. Müller, “Numerical Hydrodynamics in Special Relativity,” *Living Reviews in Relativity* 6 (2003).
- [13] J. A. Font, “Numerical Hydrodynamics and Magnetohydrodynamics in General Relativity,” *Living Reviews in Relativity* 11 (2008).
- [14] M. Shibata and K. Taniguchi, “Coalescence of Black Hole-Neutron Star Binaries,” *Living Reviews in Relativity* 14 (2011).
- [15] J. A. Faber and F. A. Rasio, “Binary Neutron Star Mergers,” *Living Reviews in Relativity* 15 (2012).
- [16] T. W. Baumgarte and S. L. Shapiro, “Numerical integration of Einstein’s field equations,” *Phys. Rev. D* **59**, 024007 (1998).
- [17] M. Shibata and T. Nakamura, “Evolution of three-dimensional gravitational waves: Harmonic slicing case,” *Phys. Rev. D* **52**, 5428–5444 (1995).
- [18] D. Alic, C. Bona-Casas, C. Bona, L. Rezzolla, and C. Palenzuela, “Conformal and covariant formulation of the Z4 system with constraint-violation damping,” *Phys. Rev. D* **85**, 064040 (2012).
- [19] H. Lim, E. Hirschmann, L. Lehner, S. Liebling, and C. Palenzuela, “Black Hole Dynamics in the Einstein-Maxwell-Dilaton-Axion Theory,” in preparation .
- [20] H. Lim and A. Held, “Nonlinear Stability in Quadratic Gravity,” in preparation .

- [21] M. Fernando, D. Neilsen, H. Lim, E. Hirschmann, and H. Sundar, “Massively Parallel Simulations of Binary Black Hole Intermediate-Mass-Ratio Inspirals,” *SIAM Journal on Scientific Computing* **41**, C97–C138 (2019).
- [22] J. Loiseau, H. Lim, B. K. Bergen, N. D. Moss, and F. Alin, “FleCSPH: a Parallel and Distributed Smoothed Particle Hydrodynamics Framework Based on FleCSI,” 2018 International Conference on High Performance Computing & Simulation (HPCS) pp. 484–491 (2018).
- [23] H. Lim, J. Loiseau, O. Korobkin, I. Sagert, M. A. R. Kaltenborn, and W. Even, “Ejecta Morphologies from Binary Neutron Mergers using Smoothed Particle Hydrodynamics,” in preparation .
- [24] H. Lim and W. East, “Interactions of Primordial Black Holes and Neutron Stars: Spherical Symmetry Study,” in preparation .
- [25] R. P. Feynman, F. B. Morinigo, and W. G. Wagner, “Feynman Lectures on Gravitation,” *European Journal of Physics* **24**, 330–330 (2003).
- [26] T. Clifton, P. G. Ferreira, A. Padilla, and C. Skordis, “Modified gravity and cosmology,” *Physics Reports* **513**, 1 – 189 (2012), modified Gravity and Cosmology.
- [27] D. Psaltis, “Probes and Tests of Strong-Field Gravity with Observations in the Electromagnetic Spectrum,” *Living Reviews in Relativity* **11**, 9 (2008).
- [28] L. Scientific and V. Collaborations, “Tests of General Relativity with GW150914,” *Phys. Rev. Lett.* **116**, 221101 (2016).
- [29] N. Yunes, K. Yagi, and F. Pretorius, “Theoretical physics implications of the binary black-hole mergers GW150914 and GW151226,” *Phys. Rev. D* **94**, 084002 (2016).
- [30] N. Yunes and X. Siemens, “Gravitational-Wave Tests of General Relativity with Ground-Based Detectors and Pulsar-Timing Arrays,” *Living Reviews in Relativity* **16**, 9 (2013).

- [31] K. Hinterbichler, “Theoretical aspects of massive gravity,” *Rev. Mod. Phys.* **84**, 671–710 (2012).
- [32] S. S. Gubser, “AdS/CFT and gravity,” *Phys. Rev. D* **63**, 084017 (2001).
- [33] F. Pretorius and M. W. Choptuik, “Adaptive mesh refinement for coupled elliptic-hyperbolic systems,” *Journal of Computational Physics* **218**, 246 – 274 (2006).
- [34] E. W. Hirschmann, L. Lehner, S. L. Liebling, and C. Palenzuela, “Black hole dynamics in Einstein-Maxwell-dilaton theory,” *Phys. Rev. D* **97**, 064032 (2018).
- [35] R. Arnowitt, S. Deser, and C. W. Misner, “Dynamical Structure and Definition of Energy in General Relativity,” *Phys. Rev.* **116**, 1322–1330 (1959).
- [36] C.-D. Munz, P. Omnes, R. Schneider, E. Sonnendrüjcker, and U. Voigt, “Divergence Correction Techniques for Maxwell Solvers Based on a Hyperbolic Model,” *Journal of Computational Physics* **161**, 484 – 511 (2000).
- [37] L. Lehner, “Numerical relativity: a review,” *Classical and Quantum Gravity* **18**, R25–R86 (2001).
- [38] A. Sen, “Rotating charged black hole solution in heterotic string theory,” *Phys. Rev. Lett.* **69**, 1006–1009 (1992).
- [39] A. García, D. Galtsov, and O. Kechkin, “Class of Stationary Axisymmetric Solutions of the Einstein-Maxwell-Dilaton-Axion Field Equations,” *Phys. Rev. Lett.* **74**, 1276–1279 (1995).
- [40] D. Garfinkle, G. T. Horowitz, and A. Strominger, “Charged black holes in string theory,” *Phys. Rev. D* **43**, 3140–3143 (1991).
- [41] J. H. Horne and G. T. Horowitz, “Rotating dilaton black holes,” *Phys. Rev. D* **46**, 1340–1346 (1992).

- [42] F. Larsen, “Rotating Kaluza-Klein black holes,” *Nuclear Physics B* **575**, 211 – 230 (2000).
- [43] D. Rasheed, “The rotating dyonic black holes of Kaluza-Klein theory,” *Nuclear Physics B* **454**, 379 – 401 (1995).
- [44] M. J. Berger and J. Olinger, “Adaptive mesh refinement for hyperbolic partial differential equations,” *J. Comp. Phys.* **53**, 484 (1984).
- [45] K. S. Stelle, “Renormalization of Higher Derivative Quantum Gravity,” *Phys. Rev.* **D16**, 953–969 (1977).
- [46] J. F. Donoghue, “The effective field theory treatment of quantum gravity,” *AIP Conf. Proc.* **1483**, 73–94 (2012).
- [47] D. R. Noakes, “The initial value formulation of higher derivative gravity,” *J. Math. Phys.* **24**, 7 (1983).
- [48] C. D. Hoyle, D. J. Kapner, B. R. Heckel, E. G. Adelberger, J. H. Gundlach, U. Schmidt, and H. E. Swanson, “Sub-millimeter tests of the gravitational inverse-square law,” *Phys. Rev.* **D70**, 042004 (2004).
- [49] D. J. Kapner, T. S. Cook, E. G. Adelberger, J. H. Gundlach, B. R. Heckel, C. D. Hoyle, and H. E. Swanson, “Tests of the gravitational inverse-square law below the dark-energy length scale,” *Phys. Rev. Lett.* **98**, 021101 (2007).
- [50] X. Calmet, S. Capozziello, and D. Pryer, “Gravitational Effective Action at Second Order in Curvature and Gravitational Waves,” *Eur. Phys. J.* **C77**, 589 (2017).
- [51] C. P. L. Berry and J. R. Gair, “Linearized $f(R)$ Gravity: Gravitational Radiation and Solar System Tests,” *Phys. Rev.* **D83**, 104022 (2011), [Erratum: *Phys. Rev.* **D85**, 089906(2012)].

- [52] Z. Cao, P. Galaviz, and L.-F. Li, “Binary black hole mergers in $f(R)$ theory,” *Phys. Rev. D* **87**, 104029 (2013).
- [53] S. Weinberg, “ULTRAVIOLET DIVERGENCES IN QUANTUM THEORIES OF GRAVITATION,” in *General Relativity: An Einstein Centenary Survey* (1980), pp. 790–831.
- [54] M. Reuter, “Nonperturbative evolution equation for quantum gravity,” *Phys. Rev. D* **57**, 971–985 (1998).
- [55] H. Ringström, *The Cauchy Problem in General Relativity, ESI lectures in mathematics and physics* (European Mathematical Society, 2009).
- [56] C. Gundlach and J. M. Martín-García, “Well-posedness of formulations of the Einstein equations with dynamical lapse and shift conditions,” *Phys. Rev. D* **74**, 024016 (2006).
- [57] O. A. Reula, “Hyperbolic Methods for Einstein’s Equations,” *Living Reviews in Relativity* **1**, 3 (1998).
- [58] J. Leray and N. Institute for Advanced Study (Princeton, *Hyperbolic differential equations* (Princeton Institute for Advanced Study, 1953).
- [59] K. S. Stelle, “Classical Gravity with Higher Derivatives,” *Gen. Rel. Grav.* **9**, 353–371 (1978).
- [60] A. Salvio, “Metastability in quadratic gravity,” *Phys. Rev. D* **99**, 103507 (2019).
- [61] H. Lü, A. Perkins, C. N. Pope, and K. S. Stelle, “Lichnerowicz modes and black hole families in Ricci quadratic gravity,” *Phys. Rev. D* **96**, 046006 (2017).
- [62] H. Lü, A. Perkins, C. N. Pope, and K. S. Stelle, “Spherically symmetric solutions in higher-derivative gravity,” *Phys. Rev. D* **92**, 124019 (2015).

- [63] <http://flash.uchicago.edu/website/home/> (unpublished).
- [64] D. Shoemaker *et al.*, <https://dcc.ligo.org/LIGO-M060056-v2/public> (unpublished).
- [65] B. P. Abbott *et al.*, “Observation of Gravitational Waves from a Binary Black Hole Merger,” *Phys. Rev. Lett.* **116**, 061102 (2016).
- [66] <http://http://www.virgo-gw.eu> (unpublished).
- [67] F. Acernese *et al.*, “Advanced Virgo: a second-generation interferometric gravitational wave detector,” *Class. Quant. Grav.* **32**, 024001 (2015).
- [68] A. Goldstein *et al.*, “An Ordinary Short Gamma-Ray Burst with Extraordinary Implications: Fermi-GBM Detection of GRB 170817A,” *Astrophys. J.* **848**, L14 (2017).
- [69] V. Savchenko *et al.*, “INTEGRAL Detection of the First Prompt Gamma-Ray Signal Coincident with the Gravitational-wave Event GW170817,” *Astrophys. J.* **848**, L15 (2017).
- [70] B. P. Abbott *et al.*, “Gravitational Waves and Gamma-rays from a Binary Neutron Star Merger: GW170817 and GRB 170817A,” *Astrophys. J.* **848**, L13 (2017).
- [71] B. P. Abbott *et al.*, “Multi-messenger Observations of a Binary Neutron Star Merger,” *Astrophys. J.* **848**, L12 (2017).
- [72] A. Albert *et al.*, “Search for High-energy Neutrinos from Binary Neutron Star Merger GW170817 with ANTARES, IceCube, and the Pierre Auger Observatory,” *Astrophys. J.* **850**, L35 (2017).
- [73] B. P. Abbott *et al.*, “Estimating the Contribution of Dynamical Ejecta in the Kilonova Associated with GW170817,” *Astrophys. J.* **850**, L39 (2017).
- [74] S. Babak, A. Taracchini, and A. Buonanno, “Validating the effective-one-body model of spinning, precessing binary black holes against numerical relativity,” (2016).

- [75] P. B. Graff, A. Buonanno, and B. S. Sathyaprakash, “Missing Link: Bayesian detection and measurement of intermediate-mass black-hole binaries,” *Phys. Rev. D* **92**, 022002 (2015).
- [76] D. Keitel *et al.*, “The most powerful astrophysical events: Gravitational-wave peak luminosity of binary black holes as predicted by numerical relativity,” *Phys. Rev. D* **96**, 024006 (2017).
- [77] L. Lehner and F. Pretorius, “Numerical Relativity and Astrophysics,” *Ann.Rev.Astron.Astrophys.* **52**, 661–694 (2014).
- [78] U. Sperhake, “The numerical relativity breakthrough for binary black holes,” *Class. Quant. Grav.* **32**, 124011 (2015).
- [79] J. Abadie *et al.*, “Predictions for the Rates of Compact Binary Coalescences Observable by Ground-based Gravitational-wave Detectors,” *Class.Quant.Grav.* **27**, 173001 (2010).
- [80] J. M. Fregeau, S. L. Larson, M. C. Miller, R. W. O’Shaughnessy, and F. A. Rasio, “Observing IMBH-IMBH Binary Coalescences via Gravitational Radiation,” *Astrophys. J.* **646**, L135–L138 (2006).
- [81] C. O. Lousto and Y. Zlochower, “Orbital Evolution of Extreme-Mass-Ratio Black-Hole Binaries with Numerical Relativity,” *Phys. Rev. Lett.* **106**, 041101 (2011).
- [82] C. O. Lousto, H. Nakano, Y. Zlochower, and M. Campanelli, “Intermediate Mass Ratio Black Hole Binaries: Numerical Relativity meets Perturbation Theory,” *Phys. Rev. Lett.* **104**, 211101 (2010).
- [83] U. Sperhake, V. Cardoso, C. D. Ott, E. Schnetter, and H. Witek, “Extreme black hole simulations: collisions of unequal mass black holes and the point particle limit,” *Phys. Rev. D* **84**, 084038 (2011).

- [84] H. Sundar, R. Sampath, and G. Biros, “Bottom-up construction and 2:1 balance refinement of linear octrees in parallel,” *SIAM Journal on Scientific Computing* **30**, 2675–2708 (2008).
- [85] R. S. Sampath, S. S. Adavani, H. Sundar, I. Lashuk, and G. Biros, “Dendro: Parallel algorithms for multigrid and AMR methods on 2:1 balanced octrees,” In *SC’08: Proceedings of the International Conference for High Performance Computing, Networking, Storage, and Analysis*, (ACM/IEEE, 2008).
- [86] S. Paolucci, Z. J. Zikoski, and D. Wirasaet, “WAMR: An adaptive wavelet method for the simulation of compressible reacting flow. Part I. Accuracy and efficiency of algorithm,” *J. Comput. Phys.* **272**, 814 – 841 (2014).
- [87] S. Paolucci, Z. J. Zikoski, and T. Grenga, “WAMR: An adaptive wavelet method for the simulation of compressible reacting flow. Part II. The parallel algorithm,” *J. Comput. Phys.* **272**, 842 – 864 (2014).
- [88] J. DeBuhr, B. Zhang, M. Anderson, D. Neilsen, and E. W. Hirschmann, “Relativistic Hydrodynamics with Wavelets,” (2015).
- [89] M. Holmström, “Solving hyperbolic PDEs using interpolating wavelets,” *SIAM J. Sci. Comput.* **21**, 405–420 (1999).
- [90] D. L. Donoho, “Interpolating wavelet transforms,” Preprint, Department of Statistics, Stanford University 2 (1992).
- [91] A. Akansu and R. Haddad, *Multiresolution Signal Decomposition: Transforms, Subbands, and Wavelets* (Academic Press, 1992).
- [92] Y. A. Rastigejev and S. Paolucci, “Wavelet-based adaptive multiresolution computation of viscous reactive flows,” *International Journal for Numerical Methods in Fluids* **52**, 749–784 (2006).

- [93] S. Bertoluzza and G. Naldi, “A wavelet collocation method for the numerical solution of partial differential equations,” *Appl. Comput. Harmon. A.* **3**, 1–9 (1996).
- [94] O. V. Vasilyev and C. Bowman, “Second-generation wavelet collocation method for the solution of partial differential equations,” *J. Comput. Phys.* **165**, 660–693 (2000).
- [95] J. D. Regele and O. V. Vasilyev, “An adaptive wavelet-collocation method for shock computations,” *Int. J. Comput. Fluid D.* **23**, 503–518 (2009).
- [96] O. V. Vasilyev, S. Paolucci, and M. Sen, “A multilevel wavelet collocation method for solving partial differential equations in a finite domain,” *J. Comput. Phys.* **120**, 33 – 47 (1995).
- [97] O. V. Vasilyev and S. Paolucci, “A dynamically adaptive multilevel wavelet collocation method for solving partial differential equations in a finite domain,” *J. Comput. Phys.* **125**, 498–512 (1996).
- [98] O. V. Vasilyev and S. Paolucci, “A fast adaptive wavelet collocation algorithm for multidimensional PDEs,” *J. Comput. Phys.* **138**, 16–56 (1997).
- [99] S. A. Teukolsky, “Formulation of discontinuous Galerkin methods for relativistic astrophysics,” (2015).
- [100] M. Bugner, T. Dietrich, S. Bernuzzi, A. Weyhausen, and B. Bruegmann, “Solving 3D relativistic hydrodynamical problems with WENO discontinuous Galerkin methods,” (2015).
- [101] H. Sundar, G. Biros, C. Burstedde, J. Rudi, O. Ghattas, and G. Stadler, “Parallel Geometric-algebraic Multigrid on Unstructured Forests of Octrees,” In *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis, SC '12* pp. 43:1–43:11 (IEEE Computer Society Press, Los Alamitos, CA, USA, 2012).

- [102] M. Fernando, D. Duplyakin, and H. Sundar, “Machine and Application Aware Partitioning for Adaptive Mesh Refinement Applications,” In *Proceedings of the 26th International Symposium on High-Performance Parallel and Distributed Computing*, HPDC '17 pp. 231–242 (ACM, New York, NY, USA, 2017).
- [103] C. Burstedde, L. C. Wilcox, and O. Ghattas, “p4est: Scalable Algorithms for Parallel Adaptive Mesh Refinement on Forests of Octrees,” *SIAM Journal on Scientific Computing* **33**, 1103–1133 (2011).
- [104] A. Ahimian *et al.*, “Petascale direct numerical simulation of blood flow on 200K cores and heterogeneous architectures,” In *SC10: Proceedings of the International Conference for High Performance Computing, Networking, Storage, and Analysis*, (ACM/IEEE, 2010), (Gordon Bell Prize).
- [105] T. Weinzierl, “The Peano software—parallel, automaton-based, dynamically adaptive grid traversals.,” *ACM Transactions on Mathematical Software*. (2019).
- [106] J. Bédorf, E. Gaburov, and S. Portegies Zwart, “Bonsai: A GPU Tree-Code,” In *Advances in Computational Astrophysics: Methods, Tools, and Outcome*, R. Capuzzo-Dolcetta, M. Limongi, and A. Tornambè, eds., *Astronomical Society of the Pacific Conference Series* **453**, 325 (2012).
- [107] L. Valgaerts, “Space-Filling Curves An Introduction,” *Technical University Munich* (2005).
- [108] W. D. Frazer and A. C. McKellar, “Samplesort: A Sampling Approach to Minimal Storage Tree Sorting,” *J. ACM* **17**, 496–507 (1970).
- [109] M. F. Ionescu and K. E. Schauer, “Optimizing parallel bitonic sort,” In *Proceedings 11th International Parallel Processing Symposium*, pp. 303–309 (1997).
- [110] H. Sundar, R. S. Sampath, S. S. Adavani, C. Davatzikos, and G. Biros, “Low-constant Parallel Algorithms for Finite Element Simulations using Linear Octrees,” (2007).

- [111] M. Bern, D. Eppstein, and S.-H. Teng, “Parallel construction of quadtrees and quality triangulations,” *International Journal of Computational Geometry & Applications* **9**, 517–532 (1999).
- [112] L. B. N. L. Applied Numerical Algorithms Group, <http://seesar.lbl.gov/anag/chombo/> (unpublished).
- [113] Carpet, an AMR driver for Cactus, <http://www.carpetcode.org> (unpublished).
- [114] S. Husa, I. Hinder, and C. Lechner, “Kranc: A Mathematica application to generate numerical codes for tensorial evolution equations,” *Comput. Phys. Commun.* **174**, 983–1004 (2006).
- [115] I. Ruchlin, Z. B. Etienne, and T. W. Baumgarte, “SENr/NRPy+: Numerical Relativity in Singular Curvilinear Coordinate Systems,” (2017).
- [116] J. Cocke, “Global Common Subexpression Elimination,” In *Proceedings of a Symposium on Compiler Optimization*, pp. 20–24 (ACM, New York, NY, USA, 1970).
- [117] A. Meurer *et al.*, “SymPy: symbolic computing in Python,” *PeerJ Computer Science* **3**, e103 (2017).
- [118] T. W. Baumgarte and S. L. Shapiro, “On the numerical integration of Einstein’s field equations,” *Phys.Rev.* **D59**, 024007 (1999).
- [119] M. Alcubierre, *Introduction to 3+1 numerical relativity, International series of monographs on physics* (Oxford Univ. Press, Oxford, 2008).
- [120] M. Shibata, *Numerical Relativity* (World Scientific Publishing Co., Inc., River Edge, NJ, USA, 2015).
- [121] L. Rezzolla and O. Zanotti, *Relativistic Hydrodynamics* (Oxford Univ. Press, Oxford, 2013).

- [122] M. Campanelli, C. O. Lousto, P. Marronetti, and Y. Zlochower, “Accurate evolutions of orbiting black-hole binaries without excision,” *Phys. Rev. Lett.* **96**, 111101 (2006).
- [123] D. Neilsen *et al.*, “Magnetized Neutron Stars With Realistic Equations of State and Neutrino Cooling,” *Phys.Rev.* **D89**, 104029 (2014).
- [124] Y. Zlochower, J. G. Baker, M. Campanelli, and C. O. Lousto, “Accurate black hole evolutions by fourth-order numerical relativity,” *Phys. Rev.* **D72**, 024021 (2005).
- [125] B. Bruegmann, J. A. Gonzalez, M. Hannam, S. Husa, U. Sperhake, and W. Tichy, “Calibration of Moving Puncture Simulations,” *Phys. Rev.* **D77**, 024027 (2008).
- [126] R. Courant, K. Friedrichs, and H. Lewy, “On the Partial Difference Equations of Mathematical Physics,” *IBM Journal of Research and Development* **11**, 215–234 (1967).
- [127] H.-O. Kreiss and J. Olinger, *Methods for Approximate Solution of Time Dependent Problems* (GARP Publication Series, Geneva, 1973).
- [128] E. Evans, S. Iyer, E. Schnetter, W.-M. Suen, J. Tao, R. Wolfmeyer, and H.-M. Zhang, “Computational relativistic astrophysics with adaptive mesh refinement: Testbeds,” *Phys. Rev. D* **71**, 081301 (2005).
- [129] T. Goodale, G. Allen, G. Lanfermann, J. Masso, T. Radke, E. Seidel, and J. Shalf, “The Cactus Framework and Toolkit: Design and Applications,” In *Vector and Parallel Processing - VECPAR '2002, 5th International Conference*, (Springer, 2003).
- [130] M. Boyle, “Transformations of asymptotic gravitational-wave data,” *Phys. Rev. D* **93**, 084031 (2016).
- [131] J. Healy, C. O. Lousto, Y. Zlochower, and M. Campanelli, “The RIT binary black hole simulations catalog,” *Classical and Quantum Gravity* **34**, 224001 (2017).

- [132] K. Jani, J. Healy, J. A. Clark, L. London, P. Laguna, and D. Shoemaker, “Georgia tech catalog of gravitational waveforms,” *Classical and Quantum Gravity* **33**, 204001 (2016).
- [133] E. R. Most, L. R. Weih, L. Rezzolla, and J. Schaffner-Bielich, “New Constraints on Radii and Tidal Deformabilities of Neutron Stars from GW170817,” *Phys. Rev. Lett.* **120**, 261103 (2018).
- [134] M. Shibata, S. Fujibayashi, K. Hotokezaka, K. Kiuchi, K. Kyutoku, Y. Sekiguchi, and M. Tanaka, “Modeling GW170817 based on numerical relativity and its implications,” *Phys. Rev.* **D96**, 123012 (2017).
- [135] B. Côté *et al.*, “The Origin of r -process Elements in the Milky Way,” *The Astrophysical Journal* **855**, 99 (2018).
- [136] B. P. Abbott *et al.*, “Astrophysical Implications of the Binary Black-Hole Merger GW150914,” *Astrophys. J.* **818**, L22 (2016).
- [137] S. Kulkarni, “Modeling supernova-like explosions associated with gamma-ray bursts with short durations,” (2005).
- [138] B. D. Metzger, G. Martinez-Pinedo, S. Darbha, E. Quataert, A. Arcones, D. Kasen, R. Thomas, P. Nugent, I. V. Panov, and N. T. Zinner, “Electromagnetic counterparts of compact object mergers powered by the radioactive decay of r-process nuclei,” *Mon. Not. Roy. Astron. Soc.* **406**, 2650–2662 (2010).
- [139] B. Metzger and E. Berger, “What is the Most Promising Electromagnetic Counterpart of a Neutron Star Binary Merger?,” *Astrophys. J.* **746**, 48 (2012).
- [140] J. Barnes and D. Kasen, “Effect of a High Opacity on the Light Curves of Radioactively Powered Transients from Compact Object Mergers,” *Astrophys. J.* **775**, 18 (2013).

- [141] T. Piran, O. Korobkin, and S. Rosswog, “Implications of GRB 130603B and its macronova for r-process nucleosynthesis,” (2014).
- [142] D. Kasen, R. Fernandez, and B. Metzger, “Kilonova Light Curves from the Disk Wind Outflows of Compact Object Mergers,” (2014).
- [143] N. R. Tanvir, A. J. Levan, A. S. Fruchter, J. Hjorth, R. A. Hounsell, K. Wiersema, and R. L. Tunnicliffe, “A ‘kilonova’ associated with the short-duration γ -ray burst GRB 130603B,” *Nature* **500**, 547–549 (2013).
- [144] E. Berger, W. Fong, and R. Chornock, “An r-process Kilonova Associated with the Short-hard GRB 130603B,” *Astrophys.J.* **774**, L23 (2013).
- [145] R. A. Gingold and J. J. Monaghan, “Smoothed Particle Hydrodynamics: Theory and Application to Non-Spherical Stars,” *Monthly Notices of the Royal Astronomical Society* **181**, 375–389 (1977).
- [146] L. B. Lucy, “A Numerical Approach to the Testing of the Fission Hypothesis,” *Astrophysical Journal* **82**, 1013–1024 (1977).
- [147] R. Davé, J. Dubinski, and L. Hernquist, “Parallel TreeSPH,” *New Astronomy* **2**, 277–297 (1997).
- [148] G. Carraro, C. Lia, and C. Chiosi, “Galaxy Formation and Evolution - I. The Padua TREE-SPH code (PD-SPH),” *Monthly Notices of the Royal Astronomical Society* **297**, 1021–1040 (1998).
- [149] V. Springel, “The Cosmological Simulation Code GADGET-2,” *Monthly Notices of the Royal Astronomical Society* **364**, 1105–1134 (2005).
- [150] M. Wetzstein, A. F. Nelson, T. Naab, and A. Burkert, “VINE-ÅTA Numerical Code for Simulating Astrophysical Systems Using Particles. I. Description of the Physics and the Numerical Methods,” *The Astrophysical Journal Supplement Series* **184**, 298 (2009).

- [151] A. F. Nelson, M. Wetzstein, and T. Naab, “VINE- A Numerical Code for Simulating Astrophysical Systems Using Particles. II. Implementation and Performance Characteristics,” *The Astrophysical Journal Supplement Series* **184**, 326 (2009).
- [152] E. Gafton and S. Rosswog, “A Fast Recursive Coordinate Bisection Tree for Neighbor Search and Gravity,” *Monthly Notices of the Royal Astronomical Society* **418**, 770–781 (2011).
- [153] W. Dehnen, “A Fast Multipole Method for Stellar Dynamics,” *Computational Astrophysics and Cosmology* **1**, 1–23 (2014).
- [154] S. Rosswog, “SPH Methods in the Modelling of Compact Objects,” *Living Reviews in Computational Astrophysics* **1**, 1–109 (2015).
- [155] S. Rosswog, “Astrophysical Smooth Particle Hydrodynamics,” *New Astronomy Reviews* **53**, 78 – 104 (2009).
- [156] J. J. Monaghan and R. A. Gingold, “Shock Simulation by the Particle Method SPH,” *Journal of Computational Physics* **52**, 374–389 (1983).
- [157] D. J. Price, “Smoothed particle hydrodynamics and magnetohydrodynamics,” *Journal of Computational Physics* **231**, 759 – 794 (2012), special Issue: Computational Plasma Physics.
- [158] M. B. Liu and G. R. Liu, “Smoothed Particle Hydrodynamics (SPH): an Overview and Recent Developments,” *Archives of Computational Methods in Engineering* **17**, 25–76 (2010).
- [159] I. Hachisu, “A versatile method for obtaining structures of rapidly rotating stars,” *Astrophysical Journal Supplement Series* **61**, 479–507 (1986).
- [160] I. Hachisu, “A versatile method for obtaining structures of rapidly rotating stars. II - Three-dimensional self-consistent field method,” *Astrophysical Journal Supplement Series* **62**, 461–499 (1986).

- [161] S. Chandrasekhar, “The highly collapsed configurations of a stellar mass (Second paper),” *Monthly Notices of the Royal Astronomical Society* **95**, 207–225 (1935).
- [162] M. Nauenberg, “Analytic Approximations to the Mass-Radius Relation and Energy of Zero-Temperature Stars,” *Astrophysical Journal* **175**, 417 (1972).
- [163] F. Verbunt and S. Rappaport, “Mass transfer instabilities due to angular momentum flows in close binaries,” *Astrophysical Journal* **332**, 193–198 (1988).
- [164] J. M. Lattimer and F. D. Swesty, “A generalized equation of state for hot, dense matter,” *Nuclear Physics A* **535**, 331 – 376 (1991).
- [165] H. Shen, H. Toki, K. Oyamatsu, and K. Sumiyoshi, “Relativistic equation of state of nuclear matter for supernova and neutron star,” *Nuclear Physics A* **637**, 435 – 450 (1998).
- [166] E. O'Connor and C. D. Ott, “A new open-source code for spherically symmetric stellar collapse to neutron stars and black holes,” *Classical and Quantum Gravity* **27**, 114103 (2010).
- [167] S. Diehl, G. Rockefeller, C. L. Fryer, D. Riethmiller, and T. S. Statler, “Generating Optimal Initial Conditions for Smoothed Particle Hydrodynamics Simulations,” *Publications of the Astronomical Society of Australia* **32**, e048 (2015).
- [168] F. A. Rasio and S. L. Shapiro, “Hydrodynamical evolution of coalescing binary neutron stars,” *Astrophysical Journal* **401**, 226–245 (1992).
- [169] S. Rosswog and D. Price, “magma: a three-dimensional, Lagrangian magnetohydrodynamics code for merger applications,” *Monthly Notices of the Royal Astronomical Society* **379**, 915–931 (2007).
- [170] R. Beatson and L. Greengard, in *Wavelets, multilevel methods, and elliptic PDEs, Numerical Mathematics and Scientific Computation*, M. Ainsworth and al, eds., (Oxford University Press, 1997), pp. 1–37.

- [171] R. Yokota and L. Barba, “Treecode and fast multipole method for N-body simulation with CUDA,” arXiv e-prints p. arXiv:1010.1482 (2010).
- [172] H. Goldstein, *Classical Mechanics* (Pearson Education, 2002).
- [173] X. Zhuge, J. M. Centrella, and S. L. W. McMillan, “Gravitational radiation from coalescing binary neutron stars,” *Phys. Rev. D* **50**, 6247–6261 (1994).
- [174] L. S. Finn and C. R. Evans, “Determining gravitational radiation from Newtonian self-gravitating systems,” *Astrophysical Journal* **351**, 588–600 (1990).
- [175] J. M. Centrella and S. L. W. McMillan, “Gravitational Radiation from Nonaxisymmetric Collisions of Neutron Stars,” *Astrophysical Journal* **416**, 719 (1993).
- [176] L. Blanchet, “Gravitational radiation from post-Newtonian sources and inspiralling compact binaries,” *Living Reviews in Relativity* **9**, 4 (2006).
- [177] L. Blanchet, “Gravitational Radiation from Post-Newtonian Sources and Inspiralling Compact Binaries,” *Living Reviews in Relativity* **17** (2014).
- [178] G. A. Sod, “A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws,” *Journal of Computational Physics* **27**, 1 – 31 (1978).
- [179] L. I. Sedov, “Propagation of strong shock waves,” *Journal of Applied Mathematics and Mechanics* **10**, 241–250 (1946).
- [180] M. Gomez-Gesteira, B. D. Rogers, A. J. Crespo, R. A. Dalrymple, M. Narayanaswamy, and J. M. Dominguez, “SPHysics—development of a free-surface fluid solver—Part 1: Theory and formulations,” *Computers & Geosciences* **48**, 289–299 (2012).
- [181] L. D. Libersky and A. Petschek, “Smooth particle hydrodynamics with strength of materials,” in *Advances in the free-Lagrange method including contributions on*

- adaptive gridding and the smooth particle hydrodynamics method* (Springer, 1991), pp. 248–257.
- [182] S. Adami, X. Hu, and N. Adams, “A generalized wall boundary condition for smoothed particle hydrodynamics,” *Journal of Computational Physics* **231**, 7057–7075 (2012).
- [183] D. J. Price, “SPLASH: An interactive visualisation tool for Smoothed Particle Hydrodynamics simulations,” *Publications of the Astronomical Society of Australia* **24**, 159–173 (2007).
- [184] M. R. S. Hawkins, “The case for primordial black holes as dark matter,” *Monthly Notices of the Royal Astronomical Society* **415**, 2744–2757 (2011).
- [185] P. H. Frampton, M. Kawasaki, F. Takahashi, and T. T. Yanagida, “Primordial black holes as all dark matter,” *Journal of Cosmology and Astroparticle Physics* **2010**, 023–023 (2010).
- [186] A. de Lavallaz and M. Fairbairn, “Neutron stars as dark matter probes,” *Phys. Rev. D* **81**, 123521 (2010).
- [187] I. Goldman and S. Nussinov, “Weakly interacting massive particles and neutron stars,” *Phys. Rev. D* **40**, 3221–3230 (1989).
- [188] J. Bramante and F. Elahi, “Higgs portals to pulsar collapse,” *Phys. Rev. D* **91**, 115001 (2015).
- [189] C. Kouvaris and P. Tinyakov, “Constraining asymmetric dark matter through observations of compact stars,” *Phys. Rev. D* **83**, 083512 (2011).
- [190] J. Bramante and T. Linden, “ON THE r-PROCESS ENRICHMENT OF DWARF SPHEROIDAL GALAXIES,” *The Astrophysical Journal* **826**, 57 (2016).

- [191] J. Fuller and C. D. Ott, “Dark matter-induced collapse of neutron stars: a possible link between fast radio bursts and the missing pulsar problem,” *Monthly Notices of the Royal Astronomical Society: Letters* **450**, L71–L75 (2015).
- [192] G. M. Fuller, A. Kusenko, and V. Takhistov, “Primordial Black Holes and r -Process Nucleosynthesis,” *Phys. Rev. Lett.* **119**, 061101 (2017).
- [193] J. Bramante, T. Linden, and Y.-D. Tsai, “Searching for dark matter with neutron star mergers and quiet kilonovae,” *Phys. Rev. D* **97**, 055016 (2018).
- [194] F. Capela, M. Pshirkov, and P. Tinyakov, “Constraints on primordial black holes as dark matter candidates from capture by neutron stars,” *Phys. Rev. D* **87**, 123524 (2013).
- [195] J. Bramante and T. Linden, “Detecting Dark Matter with Imploding Pulsars in the Galactic Center,” *Phys. Rev. Lett.* **113**, 191301 (2014).
- [196] S. C. Noble and M. W. Choptuik, “Type II critical phenomena of neutron star collapse,” *Phys. Rev. D* **78**, 064059 (2008).
- [197] T. W. Baumgarte and S. L. Shapiro, *Numerical Relativity: Solving Einstein’s Equations on the Computer* (Cambridge University Press, 2010).
- [198] M. Alcubierre, *Introduction to 3+1 Numerical Relativity, International Series of Monographs on Physics* (OUP Oxford, 2008).
- [199] G. Deslauriers and S. Dubuc, “Symmetric iterative interpolation processes,” *Constructive Approximation* **5**, 49–68 (1989).
- [200] D. L. Donoho, “Smooth Wavelet Decompositions with Blocky Coefficient Kernels,” 1993.
- [201] M. M. H. Holmström, “Solving Hyperbolic PDEs Using Interpolating Wavelets,” *SIAM Journal on Scientific Computing* **21**, 405–420 (1999).

- [202] B. Bergen, N. Moss, and M. R. J. Charest, “Flexible Computational Science Infrastructure,” Technical report, Los Alamos National Laboratory (LANL), Los Alamos, NM (United States) (2016) .
- [203] M. Bauer, S. Treichler, E. Slaughter, and A. Aiken, “Legion: Expressing locality and independence with logical regions,” In *Proceedings of the international conference on high performance computing, networking, storage and analysis*, p. 66 (2012).
- [204] L. V. Kale and S. Krishnan, “CHARM++: a portable concurrent object oriented system based on C++,” In *ACM Sigplan Notices*, **28**, 91–108 (1993).
- [205] M. S. Warren, “2HOT: an improved parallel hashed oct-tree n-body algorithm for cosmological simulation,” In *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*, p. 72 (2013).
- [206] V. Springel, “The cosmological simulation code GADGET-2,” *Monthly notices of the royal astronomical society* **364**, 1105–1134 (2005).
- [207] J. Bédorf, E. Gaburov, M. S. Fujii, K. Nitadori, T. Ishiyama, and S. P. Zwart, “24.77 Pflops on a gravitational tree-code to simulate the Milky Way Galaxy with 18600 GPUs,” In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pp. 54–65 (2014).
- [208] G. M. Morton, *A computer oriented geodetic data base and a new technique in file sequencing* (International Business Machines Company New York, 1966).
- [209] H. Sagan, *Space-filling curves* (Springer Science & Business Media, 2012).
- [210] J. Alber and R. Niedermeier, “On Multi-dimensional Hilbert Indexings,” In *Computing and Combinatorics*, W.-L. Hsu and M.-Y. Kao, eds., pp. 329–339 (Springer Berlin Heidelberg, Berlin, Heidelberg, 1998).
- [211] J. Barnes and P. Hut, “A hierarchical $O(N \log N)$ force-calculation algorithm,” *nature* **324**, 446–449 (1986).

- [212] J. E. Barnes, “A modified tree code: don’t laugh; it runs,” *Journal of Computational Physics* **87**, 161–170 (1990).
- [213] M. Folk, A. Cheng, and K. Yates, “HDF5: A file format and I/O library for high performance computing applications,” In *Proceedings of Supercomputing*, **99**, 5–33 (1999).
- [214] M. Howison *et al.*, “H5hut: A high-performance I/O library for particle-based simulations,” In *Cluster Computing Workshops and Posters (CLUSTER WORKSHOPS), 2010 IEEE International Conference on*, pp. 1–8 (2010).
- [215] W. Tichy and P. Marronetti, “A Simple method to set up low eccentricity initial data for moving puncture simulations,” *Phys. Rev.* **D83**, 024012 (2011).