# A Numerical Study of Boson Star Binaries

by

Bruno Coutinho Mundim

B.Sc., University of Brasilia, 1998
M.Sc., University of Brasilia, 2002

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

in

THE FACULTY OF GRADUATE STUDIES

(Department of Physics and Astronomy)

We accept this thesis as conforming
to the required standard

.....

.....

.....

.....

THE UNIVERSITY OF BRITISH COLUMBIA

January 2009

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

(Signature) _____

Department of Physics and Astronomy

The University Of British Columbia
Vancouver, Canada

Date _____

# ABSTRACT

This thesis reports on numerical studies of binary boson stars within the context of an approximation to general relativity. Boson stars, which are static, gravitationally bound configurations of a massive complex scalar field, can be made gravitationally compact. Astrophysically, the study of gravitationally compact binaries—in which each constituent is either a neutron star or a black hole—and especially the merger of the constituents that generically results from gravitational wave emission, continues to be of great interest. Such mergers are among the most energetic phenomena thought to occur in our universe. They typically emit copious amounts of gravitational radiation, and are thus excellent candidates for early detection by current and future gravitational wave experiments.

Detailed information concerning the general relativistic dynamics of compact binary systems relies on computer simulation involving the solution of very complicated systems of time-dependent partial differential equations in three spatial dimensions (Einstein's equations plus the equations of general relativistic hydrodynamics for the case of neutron stars). It is only over the past few years that reasonably accurate simulations of individual mergers has become possible, and the complexity and cost of the calculations in part motivates our use of an approximation of general relativity, as well as our choice of a type of matter that is easier to model than a fluid. The approximation that is adopted places certain restrictions on the dynamical variables of general relativity (conformal flatness of the 3-metric) and on the time-slicing of the spacetime (maximal slicing), and has been previously used in the modeling of neutron stars mergers. By studying a simplified, and more computationally tractable model within the context of the approximation, we hope to gain insight into the basic gravitational physics of compact-object interaction, as well as to eventually calibrate the fidelity of the approximation.

Mathematically, our modeling requires the solution of a coupled nonlinear system of two hyperbolic, and 5 elliptic partial differential equations (PDEs) in three space dimensions and time. We approximately solve this system as an initial-boundary value problem, using finite difference techniques and well known, computationally efficient numerical algorithms such as the multigrid method in the case of the elliptic equations. Careful attention is paid to the issue of code validation, and a key part of the thesis is the demonstration that, as the basic scale of finite difference discretization is reduced, our numerical code generates results that converge to a solution of the continuum system of PDEs as desired.

The thesis concludes with a discussion of results from some initial explorations of the orbital dynamics of boson star binaries. In particular, we describe calculations in which motion of such a binary is followed for more than two orbital periods, which is a significant advance over previous studies. We also present results from computations in which the boson stars merge, and where there is evidence for black hole formation.

# CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# LIST OF ABBREVIATIONS

ADM: refers to the authors of the formalism, R. Arnowitt, S. Deser and C. W. Misner.

AMR: Adaptive Mesh Refinement.

BH: Black Hole.

BS: Boson Star.

CFA: Conformally Flat Approximation.

CFC: Conformally Flat Condition.

EKG: Einstein-Klein-Gordon system.

FDA: Finite Difference Approximation.

GR: General Relativity.

KG: Klein-Gordon.

NS: Neutron Star.

NGS: Newton-Gauss-Seidel iterative technique.

ODE: Ordinary Differential Equation.

PDE: Partial Differential Equation.

WAT: Waveless Approximation Theories of Gravity.

# ACKNOWLEDGEMENTS

First and foremost I would like to express my profound gratitude for my supervisor, Matthew W. Choptuik. Thank you very much for all your support and wise guidance over the years this project took place. Second, I would like to thank all the members of my graduate committee: William G. Unruh, Jeremy S. Heyl and Ingrid Stairs for all their input on this project and for asking all those hard questions that made me think deeper about physics in general and general relativity in particular. Also I would like to thank Bill Unruh for fostering interesting discussions taking place in the gravity group meeting. I enjoyed being part of it and certainly will miss it a lot.

It was great being a member of the numerical relativity group at UBC. I learned a lot over the years with the innumerable interactions among different members of the group. In particular, I would like to thank the members of the "old crew": Inaki Olabarrieta and Kevin Lai for their patience in answering all my questions when I first started. My thanks goes to Roland Stevenson as well for sharing his programming expertise. Special thanks for the members of the "new crew", Jason Penner and Benjamin Gutierrez, for their promptness to help on the most diverse issues. Thank you guys for all your friendship.

I would like to thank my friends and colleagues from UBC for all their friendship and for making my stay at UBC so vibrant. Special thanks for my friend Sanaz Vafaei for all her support during the writing of this manuscript. I am deeply indebted to several of friends in Brazil that have actively encouraged me to pursue my dreams.

I am grateful to my wife, Janine Kurtz, for all her dedication, love, companionship and patience during my final steps to conclude this project. Thanks for helping me to get through this difficult time.

I wish to thank my entire family for providing me a loving and nurturing environment. Without their support and sacrificies during my education, I would not have made it so far. I would like to dedicate this thesis to my parents, Elson R. Mundim and Selmara C. Mundim, for all their love, support and encouragement during my entire life.

# CHAPTER 1

# INTRODUCTION

Binaries of gravitationally compact objects, such as binary black holes, binary neutron stars or binaries composed of a neutron star and a black hole, are among the most promising sources of gravitational waves. An international network of laser interferometer detectors (e.g. LIGO [1], VIRGO [2], TAMA300 [3] and GEO600 [4]) have been constructed to detect gravitational waves in the frequency range corresponding to collisions of such binaries, where the compact objects generally have masses of the order 1 to 10 solar. However, even for such strong sources, the terrestrial signal strength from a typical event (not likely in our own galaxy, or even in the local group of galaxies) is expected to be much less than the noise in the detectors. The usual strain amplitude on the detectors due to gravitational waves is of the order of $10^{-21}$m, several orders of magnitude smaller than atomic nuclei. In this case, the most promising technique to extract the signal from the noise is called matched filtering [5], which involves comparison of the measured signal against a known (precomputed) family of waveforms. Therefore, accurate theoretical modelling of compact binary inspiral is needed to build these waveform templates so as to help to extract the gravitational wave signal from the noise in the experimental apparatus.

The plunge and merger phase of compact objects in inspiral is characterized by a strong dynamical gravitational field. In this regime the gross, gravitationally-mediated dynamics of fluid stars may be similar to that of boson stars, which are static, gravitationally bound configurations of a (hypothetical) massive, complex scalar field, which satisfies the Klein-Gordon equation (possibly with some non-linear self-interaction potential). Indeed, fluid and boson stars show definite similarities when they are considered as models of isolated, static objects: both can be parametrized by their central density, $\rho_0$, and both exhibit qualitatively similar plots of total gravitational mass as a function of $\rho_0$. Since the details of the dynamics of the stars (e.g. shocks) tend not to be important gravitationally, boson star binaries may provide some insight into neutron star binaries, while avoiding the computational difficulties associated with general relativistic hydrodynamics.

Finding analytic solutions to the Einstein field equations for astrophysical scenarios such as binary collisions is a hopeless task and one is thus led to the construction of numerical solutions. The full Einstein field equations—where field quantities depend on all three spatial dimensions and time— are very complex and computationally expensive to solve. Despite some recent advances in the modelling of fully general relativistic interactions of boson stars [6, 7], a complete survey of the orbital parameter space is still lacking. Given this state, one is naturally led to consider approximations of the gravitational field equations that can reduce at least some of the computational difficulties associated with the full Einstein equations. One such approach is based on a constraint imposed on the spatial part of the metric together with a restriction on the specific choice of time coordinate, leading to an approximation of general relativity that has been used to study strong-gravity effects in several scenarios of astrophysical interest, including the interaction

of compact fluid stars. This thesis extends such studies to the case of interacting boson stars.

More specifically, and as will be discussed in detail in Chap. 2, this approximation is based on the *Conformally Flat Condition*, or CFC, for the spatial part of the metric, along with the *Maximal Slicing Condition* that fixes the time coordinatization of the spacetime. Although the resulting approximation to Einsteinian gravity has traditionally been referred to using the CFC acronym in the literature, we prefer to use CFA, for *Conformally Flat Approximation*. This stresses the fact that we *are* dealing with an approximation to general relativity. Also, we emphasize that although neither acronym makes explicit reference to the maximal slicing condition, that choice of time coordinatization has always been an essential ingredient of the approximation, and is so here.

The CFA is based on the heuristic assumption that the dynamical degrees of freedom of the gravitational field, i.e. the gravitational radiation, play a small role in at least some phases of the strong field interaction of a merging binary, and on the related fact that the gravitational radiation expressed as a fraction of the total mass-energy in the system, tends to be small. The CFA effectively eliminates the two dynamical degrees of freedom present in general relativity, but still allows the investigation of the same kinds of phenomena observed in the full general relativistic case. These include the description of compact objects, the dynamics of their interaction, and black hole formation. It is also worth mentioning that it is still possible to study gravitational wave generation within this approximation via a multipole expansion of the metric perturbation components. Briefly, the incorporation of radiation effects, although far from a trivial matter, can be realized through the introduction of a radiation reaction potential in the equations of motion for the matter model. We also note that for spherically symmetric systems the CFA is not an approximation, but can always be adopted through an appropriate choice of coordinates. Furthermore, use of the approximation for axially symmetric problems has indicated that the obtained results mimic those of general relativity quite accurately [8, 9, 10, 11]. There is thus considerable motivation to perform additional studies of strong-field gravity using this approach.

The CFA was first studied in a theoretical context by Jim Isenberg in the 1970's [12], and applied numerically (and independently) by Wilson *et al.* [13, 14, 15, 16, 17, 18, 19, 20] in the 1990's to the study of a coalescing neutron-star binary system. In these previous works, Wilson and collaborators presented evidence that, for a realistic neutron-star equation of state, general relativistic effects might cause the stars to individually collapse into black holes prior to merging. Furthermore, they observed that at least for some set of initial orbital parameters, strong-field effects caused the last innermost stable circular orbit, or ISCO, to occur at a larger separation distance, and thus at lower frequency, than was previously estimated by post-Newtonian methods. This result has significance for the possible detection of gravitational waves, since it places the frequency of radiation from coalescence closer to the maximum sensitivity range of current laser-interferometric detectors. However, the Wilson-Mathews compression effect was unexpected and controversial, and raised questions concerning the validity of the CFA.

Subsequently, Flanagan [21] identified an inconsistency in the derivation of some of the equations of motion used in the study, and suggested that use of the correct equations would reduce the crushing effect. A revised version of simulations was published shortly thereafter [22]: a key claim resulting from this work was that the crushing effect was still present, although the magnitude of

the observed effect was reduced in comparison to the previous calculations. Further, comparisons between a fully relativistic code and its CFA counterpart in the context of head-on collisions of neutron stars showed the presence of this effect and lended more credibility to the earlier calculations [23]. However, the most recent simulations [24] aimed at studying the possible crushing phenomenon actually indicate a *decompressing* effect on the neutron stars. Still, this result is not in direct contradiction to Wilson *et al*'s results since the initial data used for the two sets of simulations differ. For a more complete review of the history of this controversy, as well as a possible explanation for the neutron star crushing effect, the reader should refer to the work by Favata [25] and references therein. Also it is worth noticing that since the start of this project new "waveless" formalisms have been developed [26, 27] and used to improve the accuracy of the binary compact stars computations [28] in the inspiral phase with respect to the fully relativistic case.

One key goal of the work started in this thesis is to determine to what extent the CFA is a good approximation to model compact binaries. From this point of view, it is particularly interesting to study the CFA within the context of a simpler matter model than that commonly adopted, and this provides a key motivation for our use of boson stars. Some of the key questions we wish to address are as follows:

- Would boson stars collapse individually before merger, or is this phenomena strongly dependent on how the matter is modelled? Favata [25] posited a mechanism that would tend to compress the neutron stars given a particular set of conditions. Does that analysis apply to the binary boson star as well?

- How well does the CFA approximate the full general relativistic equations? Can we shed some light on the nature of the radiative degrees of freedom in the full theory from a detailed study of the differences between results obtained using the CFA and the full Einstein equations?

- What is the final result of the merger? Can we compare the results to those obtained from other techniques, including fully general relativistic calculations?

- Where is the ISCO? Do the results obtained match those seen in collisions of fluid stars, or can they at least be compared qualitatively?

In order to answer such questions, the Einstein equations (and ultimately the equations resulting from the adoption of the CFA) have to be cast in a form convenient for numerical calculations. What follows is a brief description our such modelling procedure; more details will be provided in subsequent chapters.

The current work makes use of the $3+1$ or ADM (Arnowitt-Deser-Misner) [29] decomposition of the equations is used. One of the main features of the $3+1$ approach, which underlies the majority of the work in numerical relativity, is that it provides a prescription for disentangling the dynamical field variables from those associated with the coordinate invariance of general relativity. Specifically, the spacetime metric components are grouped into 4 kinematical components which encode the coordinate freedom of the theory (the lapse function and three components of the shift vector), and 6 dynamical ones (the components of a 3-dimensional metric induced in a spacelike hypersurface). Now, for any specific calculation, the coordinate system must be completely fixed

by giving prescriptions for the lapse and shift. In particular, the time coordinate is fixed by a choice of the lapse function, and, as already mentioned, in this work we follow previous studies using the CFA and adopt so-called maximal slicing (a very commonly adopted slicing condition in numerical relativity that was originally proposed by Lichnerowicz [30]). A key property of maximal slicing is that its use inhibits the focusing of the world lines of observers that move orthogonally to the hypersurfaces: such focusing can result in the development of coordinate singularities, and is also associated with the formation of physical singularities. An important point is that this coordinate choice generally results in a well-posed elliptic equation for the lapse functions.

As the name suggests, the CFA requires the spatial 3-geometry to be conformally flat. This reduces the number of independent components of the spatial metric from 6 to 1, with the single, non-trivial function that then defines the spatial metric being known as the *conformal factor*. Given the maximal slicing condition, it transpires that this implies that all non-trivial components of the 4-metric are governed by elliptic equations. Specifically, the Hamiltonian constraint reduces to an elliptic equation for the conformal factor, and the momentum constraints yield elliptic equations for the shift vector components.

Within the CFA, the dynamics of the gravitational field is completely determined by the dynamics of the matter source(s), which in our case is a complex scalar field that satisfies the Klein-Gordon equation. Overall then, the model considered in this thesis is governed by a mixed system of 2 second-order hyperbolic and 5 elliptic partial differential equations (PDEs)—hereafter often referred to as "our model"—which is solved as an initial-boundary value problem. To generate numerical solutions we adopt finite difference techniques, and use approximations which are second-order accurate in the mesh spacings.

The starting point of a numerical evolution of our system is the specification of initial conditions for the complex scalar field. For example, for the relatively simple case of two identical boson stars that are initially at rest, the solution of the coupled Einstein/scalar field equations in spherical symmetry, and with a time-periodic ansatz for the scalar field, provides a static star profile that can be duplicated and interpolated onto a three dimensional domain. This sets initial values representing a binary system. The elliptic equations governing the geometric quantities are then solved on the initial slice by use of a numerical technique called multi-grid. The matter field values are then time advanced using a second order discretization of the Klein-Gordon equation, which, along with the elliptic equations for the advanced-time metric unknowns, is solved iteratively.

This thesis focuses on the development, testing and preliminary use of a code that approximately solves the system of equations sketched above. The code has been subjected to a thorough series of tests that assess convergence of the numerical results with respect to the basic discretization scale, as well as conservation of mass and Noether charge. Additional tests include the use of the technique of independent residual evaluation as described in detail in Chap. 4. Overall, results from these tests indicate that the code is correctly solving the equations of motion governing the model. Specific results calculated using the code include long-term evolution of an orbiting binary system, (more than two orbital periods), as well as high speed head-on collisions. These results are promising and suggest that, especially with enhancements such as the incorporation of adaptive mesh refinement techniques and capabilities for parallel execution, the code will be a powerful tool

for investigating strong-gravity effects in the interaction of boson stars.

We now proceed to two short overviews of subjects that are germane to our current study. First, in Sec. 1.1, the current status of the numerical modelling of gravitationally-compact binary systems is summarized. Second, some basic concepts and properties concerning boson stars are reviewed in Sec. 1.2.

## 1.1 An Overview of Compact Object Binaries

As mentioned previously, accurate modelling of compact star binaries is needed to help extract the gravitational wave signal from the noise in current and planned terrestrial gravitational wave interferometers. This section briefly describes the status of numerical relativistic modelling of these binaries by means of a quick survey of the pertinent literature. We first note that there are several good review papers on the general subject, including the following:

- Rasio & Shapiro's review of coalescing binary neutron stars [31]. This covers work up to 1999 on coalescing binary neutron stars—using Newtonian, post-Newtonian and semi-relativistic approximations—and is highly recommended for familiarizing the interested reader with earlier research in the field.
- Baumgarte & Shapiro's extensive review of the use of numerical relativity to model compact binaries [32].
- Pretorius' more recent review on black hole collisions [33].

Efforts to accurately model the collision of compact objects using fully general relativistic calculations began with the work on black hole head-on collisions by Smarr and collaborators in the 1970's [34]. These calculations established numerical relativity as a sub-field of general relativity in its own right, and indicated that one could expect perhaps a few percent of the total mass-energy of the system to be emitted in gravitational waves. In the early 1990's, Anninos *et al.* [35] revisited and extended this work on head-on collisions. In particular, they showed that at late times the extracted gravitational waveform could be matched to black hole normal modes. In 1993 the Binary Black Hole Grand Challenge Alliance consisting of investigators of several different intitutions was founded with the mission of provide stable, convergent algorithms to compute the gravitational waveforms originated from black-hole collisions. The Alliance main result [36] consisted in the evolution of a single black-hole moving freely in the three dimensional computational domain. This result constituted an important step towards the binary black hole evolution, specially when contrasted with previous computations where the black-holes were kept fixed in the domain by specific choices of coordinates. However, it was not until 2004 that a significant amount of orbital motion for a black hole binary was successfully simulated. At that time Brügmann and colaborators published results [37] demonstrating a binary with wide separation completing a full orbital period.

2005 then saw the publication of breakthrough results by Pretorius [38, 39], and the beginning of the most productive period of binary black hole research. Pretorius presented calculations that tracked the evolution of a binary black hole spacetime through several orbits, the plunge-and-merge phase, and into the late-time stage where the final black hole quickly rings down to

a near-stationary (Kerr) configuration. Furthermore, Pretorius was able to extract a reasonably accurate gravitational wave signal from the whole evolution. Pretorius' work had an electrifying effect on research in the field, and other groups, led by the UTB [40, 41, 42] and NASA [43, 44] group efforts, were quickly able to produce comparable results using similar [45] or different techniques (moving punctures) [43, 44, 40, 41, 42, 46, 47]. Moreover, it was possible to compare the various simulations and to show that the results were generally consistent to within the level of numerical error [48].

Since that time, a large number of distinct simulations have been performed, and some interesting phenomenology has been unearthed. Of particular interest are the kicks that the final black holes that form from the merger can experience due to asymmetric emission of gravitational radiation during the colaescence [49, 50, 51, 52]. When two black-holes merge in a condition of no special symmetry, such as unequal black hole masses, the gravitational waves produced carry away net linear momentum and are responsible then for the kicks observed on the final merged object. The magnitude of the resulting recoil has important astrophysical implications. For example, depending on the mass of the host stellar cluster, it may not be possible to retain the coalesced black-hole after a major merger. Typical recoil speeds are reported to be of the order of $100km/s$. The early calculations concerning kicks—which used non-spinning black holes (no intrinsic angular momentum)—have now been extended to investigate the effects of the black hole spins on the recoil velocity of the final black hole. [53, 54, 55, 56, 57, 58] These studies have indicated that impressively large kicks ($\sim 4000km/s$) can result from misalignment of the black holes' intrinsic spins and the orbital angular momentum, as well as from large initial spins. In addition, pioneering studies by the UTB/RIT group [59, 60, 61, 62] have shown that the black holes' initial spins can have a significant impact on the orbital dynamics, as well as on the emitted gravitational waveform at late times. Another interesting effect that is still being studied concerns the effect of orbital eccentricity on the gravitational radiation waveforms; summaries of recent progress on this problem are given in [49, 63, 64, 65, 66]. Comparison of the waveforms resulting from numerical simulations on the one hand and post-Newtonian calculations on the other is also underway [67, 68, 69, 70, 71]. Especially notable in this regard are the the very high accuracy simulations, recently reported by the Cornell-Caltech group, that use pseudo-spectral techniques [72, 73]. Gravitational waveforms from some of these studies have now been compared to those calculated from post-Newtonian techniques [74], demonstrating for the first time that numerical relativity calculations can provide a more accurate waveform for the late phases of inspiral than that provided by post-Newtonian methods. Finally, it is worth mentioning two studies of relativistic (high speed) collision of black holes [75, 76]. This type of collision is of particular current interest since it ties in with the phenomenology of possible mini-black hole formation in new generations of particle accelerators such as the Large Hadron Collider at CERN. Moreover, ultrarelativistic black-hole collisions are still an unexplored scenario where interesting theoretical limits could be tested. For example, the investigation of classes of initial conditions that could lead to naked singularity formation. So far preliminary results of such collisions have shown no evidence of cosmic censorship conjecture violation. Their results also report on a copious amount of energy emitted as gravitation radiation, ranging from 14% to 25% of the initial rest mass energy and approaching thus the upper bound of 29% estimated by Penrose.

The first fully relativistic simulation of binary neutron star inspiral and merger was performed by Shibata and Uryū [77] in 2000. They paid particular attention to the final product of the coalescence for a simple equation of state. Their work indicated that the nature of the final configuration would depend on the initial compactness of the neutron star: more compact neutron stars would lead to black hole formation, while those less compact would lead to a massive, differentially rotating neutron star. In 2002, subsequent work by the same authors, using increased resolution in the simulations, produced gravitational waveforms for the mergers [78]. Additional research by Shibata and his collaborators [79] has focused on unequal mass binaries, as well as the use of several different approaches to mimic realistic equations of state [80, 81]. Recent stable longterm simulations extending over a few inspiraling orbits through to merger (the end state is typically a black hole) have now been performed by several different groups [82, 83, 84, 85]. It is worth noting that in all of the work mentioned this far, the simulations ignore any effects due to magnetic fields, and it is only very recently that the first results involving magnetized neutron star binaries have been published [86, 87]. Finally, results from three dimensional general relativistic simulations of binaries in which one constitutent is a black hole and the other is a neutron-star binaries, have also just started to appear; see [88, 89, 90, 91, 92] and references therein.

Despite the tremendous advances in the simulation of compact star binaries over the last few years, relatively small regions of the multi-dimensional parameter spaces associated with the problems have been surveyed. In addition, these fully general relativistic calculations still present considerable computational challenges, both in terms of their inherent complexity, and in their typical computationsl cost. These challenges are only enhanced by the use of fluid matter sources. In this context, and as mentioned previously, the use of scalar-field matter (i.e. boson stars) provides a toy model with which to investigate at least some of the strong gravitational field phenomenology associated with compact binaries. The key observation here is that the addition of the evolution equations for a massive scalar field to the set of equations for the dynamical, relativistic gravitational field does not significantly increase the overall difficulty in generating a numerical solution, relative to a pure-gravity problem. The same cannot be said for fluid matter, where considerable effort and finesse must be used to accurately deal with solution features such as shocks, turbulence and vacuum or near-vacuum regions.

Thus, assuming that boson star binaries can serve as reasonable (gravitational) facsimiles of neutron star binaries, it may be possible to use the boson star model to more quickly survey parameter space to determine regions where particularly interesting phenomena might be expected to arise in the neutron star case. Moreover, and more speculatively, there is still a chance that boson stars may turn out to have real astrophysical relevance, in which case the computation of gravitational waveforms from their interaction could be relevant to the ongoing efforts to detect gravitational waves. For these and other reasons then, the numerical study of general relativistic boson star binaries continues to be of significant interest.

In this thesis we take the "toy model" approach one step further. Thus, although it is certainly possible to simulate boson stars within the context of full general relativity [93, 94, 95, 6, 7], we opt to study these bosonic compact objects within the context of a relatively simple approximation to general relativity.

The next section provides a short overview of the basic physics of boson stars as well as a survey of some of the key literature related to boson star research.

## 1.2 An Overview of Boson Stars

The concept of a boson star can be traced to the pioneering work by Wheeler [96] and collaborators who considered self-gravitating "lumps" of massless fields (electromagnetic or gravitational) called *geons*. Intuitively at least, such configurations are unlikely to be stable, unless perhaps they have a large amount of angular momentum that could counteract gravitational collapse.

Kaup [97], based on the geon concept, introduced a massive complex scalar field minimally coupled to general relativity rather than the electromagnetic field. Assuming a static spherically symmetric solution, he found solutions to the coupled equations, which he called Klein-Gordon geons. These solutions were later renamed boson stars. The gravitational stability of such stars have been studied in several different contexts and the interested reader should refer to the review work by Jetzer [98].

Following the work by Kaup, Ruffini and Bonazzola [99] showed that the classical limit for the boson star stress-energy tensor could be obtained by the mean value of its quantum counterpart over the ground state vector for a system of many particles. All the bosons in the system will occupy this ground state at zero temperature, forming a Bose-Einstein condensate. Then a boson star is a self-gravitating compact object (compact in the sense that its radius is of the same order as the Schwarzschild radius) composed of a large number of scalar particles in their ground state, and described classically by a complex scalar field minimally coupled to gravity.

To date there exists no known fundamental scalar particle. Scalar fields, however, hardly represent a revolution in cosmology as their existence has been studied for quite some time. Examples are the inflation field, proposed by Guth in 1981 [100] or the dilaton field which is a fundamental field in bosonic string theory. There is of course a demand for the existence of the massive Higgs boson, which is currently being sought by the particle physicists[1]. Scalar particles have been proposed as a good candidate for, or at least as a component making up a good fraction of, the dark matter in the universe.

Certainly, the study of the collapse of such a boson cloud of scalar particles into a boson star could lead to a better understanding of astrophysical phenomena. The field of gravitational lensing has achieved quite some maturity in recent years and could be helpful in its detection as well as in the determination of its properties. Boson stars should exhibit distinct lensing effects, some of which have already been determined. Finally, as another example of astrophysical speculation, boson stars could offer an alternative to super black holes in galactic centers, since they could achieve a very large size.

Despite all their possible astrophysical/particle physics applications, studies on boson stars are strongly motivated by the simplification that this matter model introduces in the system of equations when compared with their fermionic counterparts. The dynamics of the scalar field is governed by a partial differential equation, viz, the Klein-Gordon equation, that has a simpler

[1] See the paper by Schunck and Mielke for a review [101] from particle physics point of view.

mathematical structure than the simplest equation describing fluids, the Euler equation. The simplicity comes from the absence of the main non-linear term present in the fluid dynamics equation. For example, non-linear terms of the kind $uu_x$, for $u = u(x)$, is known to be responsible to develop singularities in the evolution of matter from a smooth initial data. With the choice of the Klein-Gordon equation, no problems with shocks, low density regions, ultrarelativistic flows, etc in the evolution of this kind of matter (as opposed to fluids or fermionic matter) is expected. Modelling the matter fields with a scalar field becomes then a promising model to investigate the strong-field dynamics of gravitationally compact objects and it is expected to shed some insight about the dynamics of its fermionic counterpart.

Note that some features of both the fermionic and bosonic system can easily be noticed from the start. For example, in spherical symmetry the family of solutions can be parameterized by the modulus of the field at $r = 0$, the central field, $\phi_0$, which is analogous to the central density for perfect fluid stars, governed by the TOV equations and the ideal equation of state $P = K\rho_o^\Gamma$. A noticeable difference between the models though, shows up in the configuration of a boson star in equilibrium. Its tail expands, in principle to infinity, unlike the tail on fluid model stars which have sharp edges.

The gravitational equilibrium of such gravitational solitons has already been investigated but still raises lots of interest. Boson stars are prevented from collapsing gravitationally by the pressure that stems from the Heisenberg uncertainty principle. Like their fermionic counterparts, neutron stars and white dwarfs, boson stars also have a limiting ADM mass below which the star is stable against complete gravitational collapse into a black hole. As for the neutron star case (where the degeneracy pressure provided by the Pauli exclusion principle provides the repulsive force), an expression for the maximum possible mass for the boson star can be also derived. The maximum mass turns out to be $\sim M_{pl}^3/m^2$ for the fermionic matter star and $\sim M_{pl}^2/m$ for a boson star with no self-interaction potential energy terms. In both expressions $M_{pl}$ is the planck mass and $m$ represents the respective constituent particle mass. This result comes from the fact that the boson particles within the star are confined to a region R, and thus, via the uncertainty principle, the following is valid $p \cdot R \sim \hbar$. For moderately relativistic boson stars, $p = mc$, a result obtained by considering the relativistic kinetic energy of the order of the rest energy. Combining both results above: $R \sim \hbar/mc$. Equating this to the Schwarzschild condition, the mass formula finally arises:

$$R \sim \frac{\hbar}{mc} = \frac{2M_{max}G}{c^2} \quad \Rightarrow \quad M_{max} \sim \frac{M_{Pl}^2}{m} \tag{1.1}$$

The simplest variation of the standard boson star model generally consists of adding self-interaction terms to the usual massive Klein-Gordon Lagrangian, such as $\lambda|\phi^4|$, studied by Colpi *et al.* [102] and several others. As mentioned before, stable stars are formed when there is a balance between all the forces acting on its constituent matter. In this case it is a balance between the gravitational force, Heisenberg's uncertainty principle, and the attractive/repulsive self interaction between constituent particles. So the size and mass of the star greatly depends on the mass of the individual bosons, and on the effect of self interaction terms. In principle, then boson stars can exist in a very wide size range, from microscopic to cosmologically significant scales.

## 1.3 Thesis Layout

The rest of this thesis is organized as follows:

Chap. 2 discusses the key formalisms and (continuum) approximations used in this thesis. A quick overview of general relativity is used primarily to fix notation, and to introduce key geometrical quantities (metric, curvature etc.). This is followed by a discussion of the $3+1$ decomposition of spacetime, in which the 4-geometry of spacetime is viewed as the time history of the 3-geometry of a spacelike hypersurface. We then proceed to a discussion of our matter model (a massive complex scalar field) including a derivation of the associated equations of motion. Sec. 2.4 is devoted to the issue of a conformal decomposition of the equations. The CFA is imposed on these equations and the final form of the equations of motion is derived and discussed. Related issues such as boundary conditions and initial data are also discussed in this chapter.

Chap. 3 begins with a review the basics concepts related to the finite difference approximation (FDA) of partial differential equations (PDE) that are used in the thesis. Some strategies and techniques used to evaluate code correctness are then presented and are followed by a discussion of the methods used to solve the coupled system of elliptic-hyperbolic equations.

The majority of the original research contributions in this thesis are contained in Chap. 4. The chapter starts with a discussion of the construction of the numerical code, including a descripition of the overall algorithmic flow. This is followed by the results from a thorough series of code tests that employed several different types of initial data. The dynamics of head-on collision of boson stars is then considered. The last part of the chapter discusses results related to orbital dynamics. Three different cases are identified, according to the nature of the final merged object: 1) no merger and longterm orbital dynamics, 2) rotating boson star and 3) (possible) black hole formation.

App. A documents the use of a publically-available FORTRAN subroutine that generates initial data for a spherically-symmetric boson star, and which implements a general polynomial self-interaction term for the scalar field. We feel that this routine should be of significant utility to others who wish to study the dynamics of general relativistic boson stars.

App. B discusses the derivation of finite difference operator formulas. All the finite difference operators used in this thesis is tabulated here. We note that the derivation process introduced here is not new. It is only included in this appendix for quick reference and for being a key element on the development of the MAPLE routines presented in the next appendix.

App. C contains a detailed description of the design and use of a MAPLE procedure which facilitates the construction and coding of finite-difference discretizations of generic systems of PDEs. Again, some effort has been expended in designing the routine so that it is potentially useful in other contexts, and for other researchers.

App. D contains a short review of the iterative methods for solving either linear or non-linear systems of algebraic equations. Since these techniques are at the heart of the numerical techniques to solve our model equations, we feel that it would be useful to include this review here as a quick reference.

## 1.4 Conventions, Notation and Units

We adopt the abstract index notation for tensors as defined and discussed in Wald'[103]. In particular, letters from the beginning of the Latin alphabet, $\{a, b, c, ...\}$, denote abstract indices. We then use two sets of indices for tensor *components*: Greek indices $\{\mu, \nu, ...\}$ range over the spacetime values $0, 1, 2, 3$ (where 0 is the time index), while the subset of Latin indices $\{i, j, k, l, m, n\}$ range over the spatial values $1, 2, 3$. The Einstein summation convention applies to both of these component index types. We also adopt Wald's sign conventions, so in particular, the metric signature is $(-, +, +, +)$. The totally symmetric and totally antisymmetric parts of a tensor of type $(0, 2)$ are defined as

$$T_{(ab)} = \frac{1}{2}(T_{ab} + T_{ba}) \tag{1.2}$$

and

$$T_{[ab]} = \frac{1}{2}(T_{ab} - T_{ba}). \tag{1.3}$$

respectively.

We also adopt a common terminology from computational science in which the sets of time-dependent PDEs with dependence on 1, 2 and 3 spatial dimensions (independent variables) are referred to as 1D, 2D and 3D, respectively. In particular, PDEs describing spherically symmetric systems in which the field variables depend on time and a radial coordinate, $r$, are referred to as 1D.

A variety of differentiation operators are used below. $\nabla_a$ and $D_a$ denote the covariant derivative operators compatible with the spacetime metric $g_{ab}$ and induced hypersurface metric $\gamma_{ab}$, respectively. The Lie derivative along a vector field $v^a$ is denoted by $\mathcal{L}_v$. Ordinary derivatives are represented by different notations according to the context. For example the ordinary derivative of the function $f = f(x, y, z)$ with respect to the coordinate $x$ will be denoted in one of the following ways: $\partial f/\partial x \equiv \partial_x f \equiv f_{,x}$. For time dependent functions, $g = g(t, x, y, z)$ it is usual to temporal differentiation using an over-dot, but the following are equivalent: $\partial g/\partial t \equiv \partial_t g \equiv \dot{g}$. Finally, Planck units are used throughout this thesis. Thus Newton's gravitational constant $G$, the reduced Planck constant, $\hbar$ and the speed of light in vacuum $c$ are all set to unity: $G = \hbar = c = 1$.

# CHAPTER 2

# FORMALISM

This chapter discusses the mathematical formalism employed to model a binary system of compact stars in coalescence, the central topic of this dissertation. Due to the compactness of the stars, newtonian theory of gravity fails to accurately describe this system. The most accurate theory of gravity to date is the Einstein's theory of General Relativity and it is in the context of an approximation to General Relativity that the numerical study of two compact stars (boson stars) orbiting each other was conducted. Sec. 2.1 briefly review the general relativistic field equations. Sec. 2.2 discusses the mathematical formalism to cast the Einstein equations into a form amenable to numerical treatment, the so called $3+1$ or ADM formalism. Sec. 2.3 discusses the choice of a complex scalar field as the matter model. The derivation of the scalar field equations of motion, the Klein-Gordon equations, is presented along with related derived quantities. Sec. 2.4 exposes in detail the derivation of the equations of motion for the coupled gravity-scalar-field system under the key assumptions for this thesis: conformally flat and maximal slicing conditions. The $3+1$ form of the Einstein equations is first conformally decomposed. This conformal decomposition is not crucial per se for the implementation of the approximation (CFA), however it provides a context in which the CFA can be easily compared to the full Einstein equations. It is therefore discussed here for the sake of completeness and clarity of exposition. The conformally flat condition along with the maximal slicing choice is then imposed on these conformally decomposed field equations. As a result a system of partial differential equations in cartesian coordinates is obtained. The several strategies for implementing boundary conditions for this system is, at last, discussed. The reader interested only on the final form of the equations may consider to skip the previous sections of lengthy derivations and approach Sec. 2.5 directly. Sec. 2.5 summarizes the equations derived previously and points out their main features besides indicating how they were solved. This chapter ends by discussing, also in detail, how the initial data for the binary boson star system was calculated. Note that the binary boson star is only a particular scenario chosen to be studied under CFA. Sec. 2.6 derives the equations necessary for modelling a boson star in spherical symmetry; adopts the static ansatz to describe a spherically symmetric static star and describes how each star is then interpolated into the three dimensional spacelike hypersurface in order to consist the initial data set describing a boson star binary.

## 2.1 General Relativity Field Equation

Space and time in General Relativity are modeled as a four dimensional Lorentzian manifold $\mathcal{M}$. Each point in the spacetime manifold corresponds to a physical event. The notion of distance between two points or, in the spacetime case, the interval between two events, is encoded in a symmetric, non-degenerate, tensor field of type $(0,2)$, the *metric* $g_{ab}$. The metric is thus the

fundamental entity used to quantify the geometry of spacetime, and gravitational effects arise due to the fact that this geometry is, in general, curved.

Let $\{x^\mu\}$ be a coordinate system that we assume covers that part of the manifold, $\mathcal{M}$, which is of interest. The vector and dual vector bases associated with this coordinate system are called coordinate bases and are usually written as $\{(\partial/\partial x^\mu)^a\}$ and $\{(dx^\mu)_a\}$, respectively. Therefore the metric components (or the line element corresponding intuitively to infinitesimal squared distance) in this coordinate basis can be read as:

$$g_{ab} \equiv ds^2 = g_{\mu\nu} \left(dx^\mu\right)_a \left(dx^\nu\right)_b . \tag{2.1}$$

The covariant derivative map, $\nabla_a$ is chosen in terms of the ordinary (partial) derivative map, $\partial_a$ via

$$\nabla_a v^b = \partial_a v^b + \Gamma^b_{ac} v^c, \tag{2.2}$$

where $\Gamma^b_{ac}$ is the *Christoffel symbol* and $v^b$ is an arbitrary vector. Throughout this thesis we use a covariant derivative that is compatible with the metric, i.e. that satisifies $\nabla_a g_{bc} = 0$. The Christoffel symbol can then be calculated in terms of the ordinary derivative operator:

$$\Gamma^c_{ab} = \frac{1}{2} g^{cd} \left(\partial_a g_{bd} + \partial_b g_{ad} - \partial_d g_{ab}\right). \tag{2.3}$$

The intrinsic notion of curvature of the spacetime can be made mathematically precise by considering the parallel transport of vectors along curves in the spacetime. For example, a spacetime is curved in a region if an arbitrary vector that is parallel transported along a closed path that bounds that region experiences an overall rotation. Another consequence of curvature is that the result of parallel transporting a vector is, in general, path-dependent. Parallel transport can be made mathematically precise in terms of the covariant derivative operator [1] The failure of successive applications of the covariant derivative to commute captures the path-dependence of parallel transport (along with the notion of curvature) is quantified by the so called intrinsic curvature tensor, the *Riemann curvature tensor*, defined by:

$$2\nabla_{[a} \nabla_{b]} w_c = R_{abc}{}^d \, w_d. \tag{2.4}$$

Contractions of the Riemann tensor give rise to the *Ricci tensor*:

$$R_{ac} = R_{adc}{}^d \tag{2.5}$$

and to the *Ricci scalar* or *scalar curvature*:

$$R = g^{ab} R_{ab}. \tag{2.6}$$

A key feature of the Riemann curvature tensor is that it satisfies the contracted *Bianchi identity* which can be expressed as the fact that the *Einstein tensor*, defined in terms of the Ricci tensor

[1] See Chap. 3 of [103] for a complete and detailed discussion of this material

and scalar curvature as

$$G_{ab} = R_{ab} - \frac{1}{2} g_{ab} R \tag{2.7}$$

has vanishing divergence:

$$\nabla^a G_{ab} = 0. \tag{2.8}$$

Given a metric $g_{ab}$, the Riemann curvature tensor can be written in terms of the Christoffel symbols through the following tensorial equation:

$$R_{abc}{}^{d} = -2\left(\partial_{[a}\Gamma^{d}_{b]c} - \Gamma^{e}_{c[a}\Gamma^{d}_{b]e}\right). \tag{2.9}$$

Once a coordinate basis is chosen, the Riemann tensor components can be computed from the metric components and its various derivatives in that basis. In particular, since from (2.3) the Christoffel symbol components involve first derivatives of the metric components, the components of the curvature tensor generally involve second derivatives of the metric.

The curvature of spacetime arises in response to the distribution of matter-energy in the spacetime, which includes contributions from the gravitational field itself, as well as those from any matter fields present in the spacetime. For any given matter field, the coupling to gravity is governed by the stress-energy-momentum tensor (stress-energy for short): $T_{ab}$. In many cases, including that considered in this thesis, the equation of motion for the matter field can be derived from the vanishing of the divergence of the stress tensor,

$$\nabla^a T_{ab} = 0. \tag{2.10}$$

The *Einstein field equation* expresses the intimate relationship between the physical phenomena taking place in the spacetime and the curvature of the spacetime geometry itself. In its most compact and elegant form the field equation is:

$$G_{ab} \equiv R_{ab} - \frac{1}{2} g_{ab} R = \kappa T_{ab}, \tag{2.11}$$

where the constant $\kappa$ depends on the system of units, but is also chosen so that in the weak-field limit, the Newtonian description of gravity is recovered. In the geometric units used in this thesis we have $\kappa = 8\pi$.

Once a coordinate basis is chosen, the Einstein equations can be cast as a system of 10 non-linear second order partial differential equations for the metric components, $g_{\mu\nu}$, in the chosen coordinate system. Since the metric signature is Lorentzian, i.e. $(-,+,+,+)$, this system of partial differential equations has a hyperbolic or wave-like character. Physically relevant solutions of this system of equations are quite hard to obtain. Only a few of them are possible to be obtained *exactly* by restricting to spacetimes with high degree of symmetry as for example the Schwarzschild solution describing the exterior gravitational field of a static, spherically symmetric body or the Kerr solution describing stationary rotating black-holes. Many other exact solutions of the Einstein equations have been found with the discovery of Killing vector fields or the use of null tetrads (Newman-Penrose formalism). However, most of them are of little astrophysical interest.

The tensorial nature of the Einstein field equations gives rise to several different formalisms.

Formalisms are prescriptions on how to decompose the spacetime and how to choose fundamental and auxiliary variables. It provides thereby the framework to choose coordinates. Ultimately a coordinate system that is adapted to the symmetries or features of the solution would be the ideal coordinates for the problem. Since no one is able predict in advance how the field solution for a highly dynamical and strong field spacetime would be, the choice of coordinate becomes a hard craft. One way to generate solutions of Einstein equations involves treating the equations as a dynamical system. Loosely speaking, in this approach one of the spacetime coordinates is singled out as a "time" function with respect to which the system evolves. One then considers some collection of quantities (e.g. metric components, and derivatives thereof) that defines the state of the geometry at any specific time. Initial values for these quantities are to be prescribed at some initial value of the "time" function, and then equations of motion generate "future" (or "past") values. This, in a nutshell, is the initial value formulation of Einstein equations and the next section presents details of a particular approach—the ADM or $3+1$ formalism—that casts Einstein equations as an initial value problem (or Cauchy problem) on a spacelike hypersurface. This formalism (as well as others based on it) has proved to be very convenient in obtaining *numerical* solutions of the field equations that are of astrophysical interest.

## 2.2 3+1 Decomposition - ADM Formalism

In the ADM formalism spacetime is decomposed into space and time. Fundamental to this approach is the choice of a timelike unit vector field $t^a$ in the spacetime and a foliation of spacelike hypersurfaces, $\Sigma_t$, parametrized by the time function. The timelike vector field is chosen such that its integral curves represent the time function coordinate $t$ throughout the spacetime, i.e. such that $t^a \nabla_a t = 1$. In essence the vector field and the time function are chosen to represent the development in time of quantities defined on the hypersurface. In order to make the last statement precise, it is necessary to decompose spacetime vector fields into (vector) parts defined either exclusively on the hypersurface or in a normal direction to the hypersurface. We thus introduce the *lapse function*, $\alpha$, as the projection component of the vector field $t^a$ along a unit vector field, $n^a$, which is normal to the hypersurface and the *shift vector*, $\beta^a$, the projection of $t^a$ along the hypersurface, and thus satisfying $g_{ab}\beta^a n^b = 0$. Therefore $t^a$ can be expressed as:

$$t^a = \alpha n^a + \beta^a. \tag{2.12}$$

Here, the lapse function is obtained from $t^a$ by the formula:

$$\alpha = -g_{ab} t^a n^b, \tag{2.13}$$

since $g_{ab} n^a n^b = -1$. Introducing the projection tensor, $\gamma^a{}_b$, defined by

$$\gamma^a{}_b = \delta^a{}_b + n^a n_b, \tag{2.14}$$

the shift vector can be expressed as

$$\beta^a = \gamma^a{}_b t^b, \tag{2.15}$$

as can be verified from equation (2.12) and the definition of the $\gamma^a{}_b$.

Also note that the relation $t^a \nabla_a t = 1$ is satisfied from equation (2.13) if the dual normal vector is chosen such that:

$$n_a = -\alpha \nabla_a t. \tag{2.16}$$

The decomposition of vectors into "temporal" and "spatial" parts can be easily generalized to tensors. The tensor part normal to the hypersurface can be obtained by contraction of the tensor indices with the unit norm vector $n^a$. Its hypersurface tangent part, on the other hand, comes from applying the projection tensor to each tensor index to be projected into the hypersuface:

$$\perp T^{ab\ldots}{}_{cd\ldots} \equiv \gamma^a{}_e \, \gamma^b{}_f \ldots \gamma^g{}_c \, \gamma^h{}_d \ldots T^{ef\ldots}{}_{gh\ldots} \tag{2.17}$$

A tensor which has had all of its indices projected onto the hypersurface is known as a spatial tensor. One such tensor of special interest is the induced three dimensional metric, $\gamma_{ab}$, of the hypersurface:

$$\gamma_{ab} = g_{ab} + n_a n_b. \tag{2.18}$$

In order to describe parallel transport of spatial tensors and curvature within a hypersurface, a covariant derivative operator must be defined. A natural choice is to project the four dimensional covariant derivative into the spacelike hypersurface, leading to the definition of a three dimensional covariant derivative operator, $D_a$

$$D_a \equiv \perp \nabla_a. \tag{2.19}$$

The Riemann curvature tensor on the hypersurface is defined analogously to its four dimensional counterpart:

$$2D_{[a}D_{b]}w_c = \mathcal{R}_{abc}{}^{d}w_d, \tag{2.20}$$

while the Ricci tensor and Ricci scalar are obtained by the usual contractions of the Riemann curvature tensor:

$$\mathcal{R}_{ab} = \mathcal{R}_{acb}{}^{c} \qquad \text{and} \qquad \mathcal{R} = \mathcal{R}^{a}{}_{a}\,. \tag{2.21}$$

As mentioned previously, the Riemann curvature tensor describes the curvature *intrinsic* to a manifold. In the current case, which involves the embedding of three-dimensional hypersurfaces in a four-dimensional spacetime, there is a second type of curvature, known as the *extrinsic* curvature, that quantifies the embedding. Since the orientation of the hypersurface within the spacetime is related to the unit normal vector, $n^a$, the covariant derivative of $n^a$ thus characterizes nearby changes in the orientation. The extrinsic curvature tensor, $K_{ab}$, can thus be defined as the projection of the covariant derivative of the dual vector field associated to the normal vector field:

$$K_{ab} = -\perp \nabla_a n_b = -\frac{1}{2} \perp \mathcal{L}_n g_{ab} = -\frac{1}{2}\mathcal{L}_n \gamma_{ab}. \tag{2.22}$$

Since both sides of the Einstein equations must be decomposed in the 3+1 approach, we must also consider various projections of the stress-energy tensor along the normal $n^a$ and into the hypersurface $\Sigma_t$. First note that the stress-energy tensor $T_{ab}$ is a type $(0,2)$ symmetric tensor. A generic tensor of this type can be decomposed in the following way in the 3+1 formalism:

$$T_{ab} = \perp T_{ab} - 2n_{(a} \perp T_{b)\hat{n}} + n_a n_b T_{\hat{n}\hat{n}}. \tag{2.23}$$

We rewrite the above as

$$T_{ab} = S_{ab} - 2J_{(a}n_{b)} + \rho n_a n_b, \tag{2.24}$$

where the quantities, $\rho$, $J_a$ and $S_{ab}$ are defined by

$$\rho \equiv T_{\hat{n}\hat{n}} = T_{ab}n^a n^b = T^{ab}n_a n_b, \tag{2.25}$$

$$J_a \equiv \perp T_{a\hat{n}} = \perp (T_{ab}n^b), \tag{2.26}$$

$$J^a \equiv \perp T^{a\hat{n}} = - \perp (T^{ab}n_b), \tag{2.27}$$

$$S_{ab} \equiv \perp T_{ab}, \tag{2.28}$$

Physically, $\rho$ is interpreted as the local energy density, $J^a$ as the momentum density and $S_{ab}$ as the spatial stress tensor, all measured by observers moving orthogonally to the slices.

The several possible combinations of projections of Einstein equations along the "temporal" and "spatial" directions give rise to the equations of motion in the $3+1$ form. Projecting both indices with $n^a$ we find

$$\mathcal{R} + K^2 - K_{ab}K^{ab} = 16\pi\rho, \tag{2.29}$$

where $K \equiv K^a{}_a$ is the trace of the extrinsic curvature. Equation (2.29) is also known as *Hamiltonian constraint*. On the other hand, if only one index is contracted along $n^a$ while the other is projected onto the hypersurface, we derive a three-vector equation known as the *Momentum constraint*:

$$D_b K^{ab} - D^a K = 8\pi J^a. \tag{2.30}$$

We note that care must be exercised in using the covariant form of this equation, since due to the relative sign in the definitions of $J_a$ and $J^a$ in equations (2.26) and (2.27) we have

$$\perp G_{a\hat{n}} = -D^b K_{ab} + D_a K = 8\pi J_a, \tag{2.31}$$

$$\perp G^{a\hat{n}} = D_b K^{ab} - D^a K = 8\pi J^a. \tag{2.32}$$

Key features of the constraint equations are the presence of *only* spatial tensors and the absence of explicit time derivative of these tensors. They must be satisfied by $\{\gamma_{ab}, K_{ab}\}$ on all slices, including the initial slice.

The 3+1 equations that *do* involve time derivatives of the spatial tensors $\{\gamma_{ab}, K_{ab}\}$, are thus called evolution equations. For the spatial metric, an evolution equation follows from the definition of the extrinsic curvature (2.22):

$$\mathcal{L}_t \gamma_{ab} = -2\alpha K_{ab} + \mathcal{L}_\beta \gamma_{ab}. \tag{2.33}$$

where $\mathcal{L}_t$ is the Lie derivative with respect to the vector field $t^a$. The evolution equation for the extrinsic curvature can be derived by considering the projection of both indices of Einstein equations, which involves computation of $\perp R_{a\hat{n}a\hat{n}}$. After some manipulation, we find

$$\mathcal{L}_t K^a{}_b = \mathcal{L}_\beta K^a{}_b - D^a D_b \alpha + \alpha \left[ \mathcal{R}^a{}_b + K K^a{}_b + 8\pi \left( \frac{1}{2}\gamma^a{}_b (S - \rho) - S^a{}_b \right) \right]. \tag{2.34}$$

All the definitions and decompositions done so far were coordinate independent ones. Operationally however, a coordinate system $x^\mu \equiv \{t, x^i\}$ needs to be introduced in order to generate solutions of the equations (we remind the reader that Greek indices such as $\mu$ range over the spacetime values, 0,1,2,3, while Latin indices such as $i$ are restricted to spatial values, 1,2 and 3). Having adopted such a basis, the spacetime displacement can be written as

$$\begin{aligned} ds^2 &= g_{\mu\nu} dx^\mu dx^\nu \\ &= -\alpha^2 dt^2 + \gamma_{ij} \left(dx^i + \beta^i dt\right) \left(dx^j + \beta^j dt\right), \end{aligned} \tag{2.35}$$

where the lapse is a function of time and the spatial coordinates, $\alpha \equiv \alpha(t, x^i)$, while the shift is a spatial vector function of time and spatial coordinates, $\beta^i \equiv \beta^i(t, x^i)$. The lapse function can be interpreted as the lapse of proper time, with respect to coordinate time, for an observer travelling normally to the hypersurface. The shift vector encodes the translation of spatial coordinates from one slice to the other, again relative to propagation in the normal direction. All spatial tensors indices are lowered and raised with the 3-metric $\gamma_{ij}$ and its inverse $\gamma^{ij}$, defined by $\gamma^{ik}\gamma_{kj} = \delta^i{}_j$. Both the spatial metric and its inverse are functions of the coordinates $x^\mu$.

As discussed above, the decomposition of the stress-energy tensor gives rises to a variety of energy-momentum quantities defined by equations (2.25-2.28). Using the relations $n^\mu = (1/\alpha; -\beta^i/\alpha)$ and $n_\mu = (-\alpha; 0)$, these quantities become

$$\rho = \frac{T_{00}}{\alpha^2} - 2\frac{\beta^i T_{0i}}{\alpha^2} + \frac{\beta^i \beta^j T_{ij}}{\alpha^2} = \alpha^2 T^{00}, \tag{2.36}$$

$$J_i = \frac{T_{i0}}{\alpha} - \frac{T_{ij}\beta^j}{\alpha}, \tag{2.37}$$

$$J^i = \alpha(T^{i0} + T^{00}\beta^i), \tag{2.38}$$

$$S_{ij} = T_{ij}. \tag{2.39}$$

The component form of the evolution equations can then be written as

$$\partial_t \gamma_{ij} = -2\alpha\gamma_{ik}K^k{}_j + 2D_{(i}\beta_{j)} \tag{2.40}$$

and

$$\begin{aligned} \partial_t K^i{}_j &= \beta^k \partial_k K^i{}_j - \partial_k \beta^i K^k{}_j + \partial_j \beta^k K^i{}_k - D^i D_j \alpha \\ &+ \alpha \left[\mathcal{R}^i{}_j + K K^i{}_j + 8\pi \left(\frac{1}{2}\delta^i{}_j (S - \rho) - S^i{}_j\right)\right], \end{aligned} \tag{2.41}$$

while the constraint equations become:

$$\mathcal{R} + K^2 - K_{ij}K^{ij} = 16\pi\rho, \tag{2.42}$$

$$D_j K^{ij} - D^i K = 8\pi J^i. \tag{2.43}$$

Even within a specific formulation of the Einstein equations, such as the 3+1 approach sketched above, the coordinate invariance of general relativity ensures that there are generally many distinct possibilities to solve the specific set of PDEs that results once the coordinate system has been fully fixed (full specification of the lapse and shift). Here we are referring to the fact that we have more equations (4 second-order "elliptic" constraints + 12 first-order-in-time evolution for a total of 16 equations) than fundamental dynamical unknowns (6 $g_{ij}$ + 6 $K_{ij}$ = 12 unknowns). The interested reader is referred to the classic paper by Piran [104] in which nomenclature, such as free evolution, constrained evolution and partially-constrained evolution is defined and discussed. Here, the key thing to note is that the approximation (CFA) that is adopted in this thesis has the advantage of providing a single, well defined set of 5 elliptic PDEs for 5 well defined functions that completely fix the spacetime geometry. In this sense, and in an abuse of Piran's original classification, we implement a fully constrained evolution for the geometrical field and, further, in contrast to the full general-relativistic situation there are *no* purely gravitational degrees of freedom. That means that in the model considered here, as is the case for any model that adopts the CFA with maximal slicing condition, all dynamics is linked to the dynamics of the matter. This has a host of ramifications, physically, mathematically and computationally, but particularly given the efforts that have been expended on taming instabilities in free evolution approaches for the full Einstein equations, is one of the most attractive features of Isenberg's proposal.

## 2.3 Matter Model: Massive Complex Scalar Field

The matter model adopted in this thesis is a complex *Klein-Gordon field*, which satisfies a Klein-Gordon equation as discussed in detail below. This field represents a simple type of matter that when coupled to Einstein gravity, or in the context of the approximation adopted in this thesis, admits star-like solutions. Studies focusing on such solutions—known as boson stars—using a variety of techniques including numerical analysis, have a rich history and we refer the reader to the paper by Schunck and Mielke, [101] (and references therein), for an excellent and thorough review of the subject as of about five years ago.

General relativists have studied Klein-Gordon fields for many purposes over the years. As either 1) a classical field or 2) a quantum-theory of spin 0 particles, scalar fields have been widely exploited for exploratory theoretical studies. A key point is that the simplicity of scalar matter (in terms, e.g. of physical interpretation as well as complexity of the equations of motion), often allows one to investigate and understand basic theoretical issues in Einstein gravity relatively free of the complications a more realistic matter model could bring in. This is a chief motivation for the use of a scalar field in the current work.

In the discussion below, we will refer to the system of a single Klein-Gordon field coupled to Einstein gravity as the Einstein-Klein-Gordon (EKG) system. We note that we adopt a complex scalar field, rather than a real one, since it has been long known that there are no regular, static solutions (i.e. star-like solutions) for a real scalar field in general relativity. Interestingly, for us this turns out to be something of a technical point, since for a real field coupled to Einstein gravity there are quasi-static solutions known as "oscillons" which have decay times that can be *much* longer than the intrinsic dynamical time [105]. Thus in principle one could use a real scalar field to study some of the effects we wish to examine in this thesis and follow-up work. However, for a variety of reasons, not least including the ease with which one can generate star-like solutions, we prefer to work with the complex field.

Additionally, the complex field must interact in an non-trivial potential, which we define to include a mass term. The possibilities for potential choice are endless, and have formed the basis for much previous work. Again, we choose the simplest approach and, at least initially, adopt only a mass term. The boson stars modelled by scalar fields with this self-interaction potential are also known in the literature as *mini-boson stars*.

Since mathematically we are ultimately interested in solving an approximate EKG system as an initial value problem, we note in passing that a scalar field is known to admit a well-posed initial value formulation in the following sense [103]:

- For an initial data in a spacelike Cauchy surface $\Sigma$ in a globally hyperbolic spacetime [2] $(\mathcal{M}, g_{ab})$, there is an open neighborhood $O$ of $\Sigma$ such that the Klein-Gordon equation has a solution in $O$ and $(O, g_{ab})$ is globally hyperbollic.
- The solution in $O$ is unique and propagates causally.
- The solution depends continuously on the initial data.

[2] When the domain of dependence of a Cauchy surface is the whole spacetime (region of interest) then this spacetime (region) is said to be globally hyperbolic

In addition to possessing star-like solutions, a key advantage of scalar matter relative to the more-astrophysically relevant perfect fluid case, is that the solutions do not tend to develop shocks or rarefraction regions. Rather, as is expected from the structure of the equations, and has been born out my many [106, 107, 108] previous numerical studies, solutions tend to remain as smooth as the initial data, except at actual physical singularities (produced, for example, by gravitational collapse).

### 2.3.1 Einstein-Klein-Gordon System

One route to study matter models in general relativity is to postulate equations of motion for the matter, derive a suitable stress-energy tensor $T_{ab}$ compatible with those equations and then use the Einstein equations to relate the matter distribution to the spacetime curvature through this stress-energy tensor. However, and as already mentioned, in many cases local conservation of the stress-tensor, implies the matter equation of motion. In such cases it is essentially sufficient to postulate the stress-energy tensor $T_{ab}$ for the matter model in order to study the coupled system of matter distribution and spacetime geometry.

Additionally, and for a variety of reasons, it is often useful to adopt a Lagrangian (or variational) approach to Einstein equations and we will do so here. Here, a basic observation is that the vacuum Einstein equations can be obtained from the functional derivative of the so called Hilbert action functional:

$$S_G[g^{ab}] = \int_{\mathcal{M}} \mathcal{L}_G = \int_{\mathcal{M}} \sqrt{-g}R, \tag{2.44}$$

where $\mathcal{L}_G = \sqrt{-g}R$ is the Einstein lagrangian density and $R$ the Ricci scalar. It is a standard exercise to show that the functional derivative of the action with respect to the inverse metric $g^{ab}$ is

$$\frac{\delta S_G}{\delta g^{ab}} = \sqrt{-g}G_{ab}, \tag{2.45}$$

which then clearly yields the vacuum Einstein equations $G_{ab} = 0$ when the field configuration satisfies the action extremization condition:

$$\frac{\delta S_G}{\delta g^{ab}} = 0. \tag{2.46}$$

In order to obtain a coupled matter-gravity system, one then simply adds to the matter lagrangian density to the Hilbert term (this is the so-called minimal coupling prescription). We thus have

$$\mathcal{L} = \mathcal{L}_G + \alpha_M \mathcal{L}_M, \tag{2.47}$$

where $\alpha_M$ is a coupling constant that can typically be rescaled through a redefinition of the matter fields. In the case of the Einstein-Klein-Gordon system one conventional choice that we adopt here is $\alpha_{KG} = 16\pi$.

The stress-energy tensor can now be calculated as the variation of the matter action with respect to the inverse metric field $g^{ab}$. Specifically, one has

$$T_{ab} = -\frac{\alpha_M}{8\pi}\frac{1}{\sqrt{-g}}\frac{\delta S_M}{\delta g^{ab}}, \tag{2.48}$$

where $S_M$ is the action functional for the matter field $M$ (understood here as a generic collection of matter fields and their higher order covariant derivatives):

$$S_M[g^{ab}, M] = \int_{\mathcal{M}} \mathcal{L}_M. \tag{2.49}$$

Finally, variations of the action $S_M$ with respect to the matter fields themselves generate the equations of motion for the matter.

For the reasons discussed above, we now restrict attention to matter consisting of a single complex scalar field, $\Phi$. We write the field as

$$\Phi = \phi_1 + i\phi_2 = \phi_0 \exp(i\theta), \tag{2.50}$$

where $\phi_1$, $\phi_2$, $\phi_0$ and $\theta$ are real-valued functions of the spacetime coordinates $x^\mu$. The Lagrangian density associated with this field is

$$\mathcal{L}_\Phi \quad = \quad -\frac{1}{2}\sqrt{-g}\left(g^{ab}\nabla_a\Phi\nabla_b\Phi^* + U(|\Phi|^2)\right), \tag{2.51}$$

where $U(|\Phi|^2)$ is the scalar field self-interaction potential. As also discussed above, we will eventually specialize to the case where $U$ contains only a mass term

$$U(|\Phi|^2) = m^2\Phi\Phi^* = m^2\phi_0^2 = m^2(\phi_1^2 + \phi_2^2), \tag{2.52}$$

but for the time being we will continue the discussion in terms of general potentials.

We now rewrite the lagrangian (2.51) in terms of the real-valued quantities defined in (2.50):

$$\mathcal{L}_\Phi \quad = \quad -\frac{1}{2}\sqrt{-g}\left(g^{ab}\nabla_a\phi_1\nabla_b\phi_1 + g^{ab}\nabla_a\phi_2\nabla_b\phi_2 + U(\phi_0^2)\right). \tag{2.53}$$

Klein-Gordon equations for each real valued component ($\phi_A \in \{\phi_1, \phi_2\}$) can then be obtained by the usual variational procedure, yielding

$$\Box\phi_A - \frac{dU(\phi_0^2)}{d\phi_0^2}\phi_A = 0 \qquad \text{or} \qquad g^{ab}\nabla_a\nabla_b\phi_A - \frac{dU(\phi_0^2)}{d\phi_0^2}\phi_A = 0, \qquad A = 1, 2, \tag{2.54}$$

where $\Box \equiv g^{ab}\nabla_a\nabla_b$ is the general relativistic D'Alembertian operator.

Once a coordinate system is chosen, each of the above scalar Klein-Gordon equations is a second-order-in-time PDE. In keeping with the 3+1 spirit, it is often conventional to recast these equations in first-order-in-time form, and we do so here. One specific way of doing this is to pass to the

Hamiltonian description of the system in the standard fashion. Namely, we consider the lagrangian as a function of the field and its spatial and time derivatives; we define a conjugate momentum associated with the field; we write down the Hamiltonian functional from the langrangian by performing a Legendre transformation for the conjugate momentum; and we then evaluate the Hamilton evolution equations from the lagrangian. Full details of this procedure can be found in standard texts such as Wald [103], and here we simply summarize the results for the scalar field.

Since the scalar-field Lagrangian (2.51) does not contain time derivatives higher than first order, the conjugate momentum associated with each component of the scalar field can be defined as:

$$\Pi_A \equiv \frac{\delta(\sqrt{-g}L_{\phi_A})}{\delta\dot{\phi}_A}, \tag{2.55}$$

or, more explicitly

$$\Pi_A = \frac{\sqrt{-g}}{\alpha^2}\left[\dot{\phi}_A - \beta^i\partial_i\phi_A\right]. \tag{2.56}$$

The dynamical equations of motion (2.54) can be rewritten in terms of these conjugate fields, leading to four first-order-in-time partial differential equations for the two conjugate pairs of variables $\{\phi_A, \Pi_A\}$ (where $A = 1, 2$):

$$\partial_t\phi_A = \frac{\alpha^2}{\sqrt{-g}}\Pi_A + \beta^i\partial_i\phi_A, \tag{2.57}$$

$$\partial_t\Pi_A = \partial_i(\beta^i\Pi_A) + \partial_i(\sqrt{-g}\gamma^{ij}\partial_j\phi_A) - \sqrt{-g}\frac{dU(\phi_0^2)}{d\phi_0^2}\phi_A. \tag{2.58}$$

These last equations can be further manipulated using the following relationship between the determinants of the spacetime and spatial metrics: [3]

$$\sqrt{-g} \equiv \alpha\sqrt{\gamma}, \tag{2.59}$$

yielding

$$\partial_t\phi_A = \frac{\alpha}{\sqrt{\gamma}}\Pi_A + \beta^i\partial_i\phi_A, \tag{2.60}$$

$$\partial_t\Pi_A = \partial_i(\beta^i\Pi_A) + \partial_i(\alpha\sqrt{\gamma}\gamma^{ij}\partial_j\phi_A) - \alpha\sqrt{\gamma}\frac{dU(\phi_0^2)}{d\phi_0^2}\phi_A. \tag{2.61}$$

Having obtained equations of motion for the scalar field, we now consider computation of the stress-energy tensor and the 3+1 quantities derived from it. Using the variational prescription sketched above we find

$$T_{ab} = \frac{1}{2}\left[\nabla_a\Phi\nabla_b\Phi^* + \nabla_b\Phi\nabla_a\Phi^* - g_{ab}(g^{cd}\nabla_c\Phi\nabla_d\Phi^* + U(\Phi\Phi^*))\right], \tag{2.62}$$

[3] This relation is derived from the definition of inverse metric: $g^{00} = -\alpha^{-2} = (-1)^{0+0}det(\gamma_{ij})/det(g_{\mu\nu}) = \gamma/g$.

which, adopting a coordinate basis, and working with the real-valued field components becomes

$$T_{\alpha\beta} = \sum_{A=1}^{2} \frac{1}{2} \left[ \partial_\alpha \phi_A \partial_\beta \phi_A + \partial_\beta \phi_A \partial_\alpha \phi_A - g_{\alpha\beta} g^{\mu\nu} \partial_\mu \phi_A \partial_\nu \phi_A \right] - \frac{1}{2} g_{\alpha\beta} U(\phi_0^2). \tag{2.63}$$

From the above, and using (2.36)-(2.39), we compute the 3+1 stress-energy quantities and find:

$$\rho = \frac{1}{2} \sum_{A=1}^{2} \left[ \frac{\Pi_A^2}{\gamma} + \gamma^{ij} \partial_i \phi_A \partial_j \phi_A \right] + \frac{1}{2} U(\phi_0^2), \tag{2.64}$$

$$J_i = \sum_{A=1}^{2} \left[ \frac{\Pi_A}{\sqrt{\gamma}} \partial_i \phi_A \right], \tag{2.65}$$

$$J^i = \sum_{A=1}^{2} \left[ -\frac{\Pi_A}{\sqrt{\gamma}} \gamma^{ij} \partial_j \phi_A \right], \tag{2.66}$$

$$S_{ij} = \frac{1}{2} \sum_{A=1}^{2} \left\{ 2 \partial_i \phi_A \partial_j \phi_A + \gamma_{ij} \left[ \frac{\Pi_A^2}{\gamma} - \gamma^{pq} \partial_p \phi_A \partial_q \phi_A \right] \right\} - \frac{1}{2} \gamma_{ij} U(\phi_0^2). \tag{2.67}$$

Additionally, we need to compute the trace of the spatial stress tensor, $S \equiv S^i{}_i$ as well as the combination $\rho + S$. These are given by

$$S^i{}_i = \frac{1}{2} \sum_{A=1}^{2} \left[ 3 \frac{\Pi_A^2}{\gamma} - \gamma^{ij} \partial_i \phi_A \partial_j \phi_A \right] - \frac{3}{2} U(\phi_0^2), \tag{2.68}$$

$$\rho + S = \sum_{A=1}^{2} \left[ 2 \frac{\Pi_A^2}{\gamma} \right] - U(\phi_0^2). \tag{2.69}$$

### 2.3.2 Noether Charge

The invariance of the Klein-Gordon lagrangian density, equation (2.51), under a global $U(1)$ symmetry transformation $\Phi \to \Phi e^{i\epsilon}$ gives rise to a conserved current density according to Noether's theorem. Roughly, this result can be obtained as follows:

First, consider the Klein-Gordon action as a functional of the inverse metric, the scalar field and its first covariant derivative, instead of the inverse metric and the scalar field alone, as in the last subsection:

$$S_{KG}[g^{ab}, \Phi, \nabla_a \Phi] = \int_{\mathcal{M}} \mathcal{L}_{KG}(g^{ab}, \Phi, \nabla_a \Phi). \tag{2.70}$$

Also note that for a scalar field we have $\nabla_a \Phi = \partial_a \Phi$. Defining the variation of a functional or function with respect to a parameter $\epsilon$ as:

$$\delta S[\Phi_\epsilon] \equiv \left. \frac{dS}{d\epsilon} \right|_{\epsilon=0} \qquad \text{and} \qquad \delta\Phi \equiv \left. \frac{d\Phi}{d\epsilon} \right|_{\epsilon=0}, \tag{2.71}$$

the variation of the Klein-Gordon action functional can then be expanded as:

$$\delta S_{KG} = \int_{\mathcal{M}} \frac{\delta S_{KG}}{\delta g^{ab}} \delta g^{ab} + \frac{\delta S_{KG}}{\delta \Phi} \delta \Phi + \frac{\delta S_{KG}}{\delta(\partial_a \Phi)} \delta(\partial_a \Phi). \tag{2.72}$$

Our interest here is in variations that keep the action functional constant; that is, variations such that $\delta S_{KG} = 0$. The inverse metric $g^{ab}$ is invariant under the action of a $U(1)$ transformation: $\delta g^{ab} = \frac{dg^{ab}}{d\epsilon}|_{\epsilon=0} = 0$, and the first term in the equation (2.72) drops out. Further simplification of equation (2.72) results from noting that the variation with respect to the field derivative can be rewritten as:

$$\delta(\partial_a \Phi) = \frac{d(\partial_a \Phi)}{d\epsilon}\bigg|_{\epsilon=0} = \partial_a \left( \frac{d\Phi}{d\epsilon} \right) \bigg|_{\epsilon=0} = \partial_a(\delta\Phi), \tag{2.73}$$

since ordinary derivatives commute. Inserting the above relationship in equation (2.72), we have after some simple algebraic manipulation:

$$\delta S_{KG} = \int_{\mathcal{M}} \left\{ \left[ \frac{\delta S_{KG}}{\delta \Phi} - \partial_a \left( \frac{\delta S_{KG}}{\delta(\partial_a \Phi)} \right) \right] \delta\Phi + \partial_a \left( \frac{\delta S_{KG}}{\delta(\partial_a \Phi)} \delta\Phi \right) \right\}. \tag{2.74}$$

The first term of the equation above is simply the Klein-Gordon equation of motion which vanishes identically. The second term is a total divergence that can be converted to a surface term using Stokes theorem, and which also has to vanish if the action is supposed to be invariant under the field variation $\delta\Phi$. This then implies that the current density, $j^a$, associated with the $U(1)$ symmetry and defined by

$$j^a \equiv \frac{\delta S_{KG}}{\delta(\partial_a \Phi)} \delta\Phi, \tag{2.75}$$

is conserved

$$\partial_a j^a = 0. \tag{2.76}$$

A conserved (Noether) charge, $Q_N$, is associated with the "time" component of the current density:

$$Q_N = \int_{\Sigma_t} j^t, \tag{2.77}$$

where $\Sigma_t$ is a spacelike hypersurface as previously, and a fixed volume element, $\mathbf{e}$, $\Sigma_t$ is understood in the integration.

To compute the explicit form of the Noether current we apply eq. (2.76) to the Klein-Gordon lagrangian (2.51), obtaining

$$\delta\mathcal{L}_{KG} \equiv \partial_a \left( \frac{\delta S_{KG}}{\delta(\partial_a \Phi)} \delta\Phi + \frac{\delta S_{KG}}{\delta(\partial_a \Phi^\star)} \delta\Phi^\star \right) = \partial_a \left( -\frac{1}{2}\sqrt{-g} g^{ab} (\partial_b \Phi^\star \delta\Phi + \partial_b \Phi \delta\Phi^\star) \right), \tag{2.78}$$

so the current density is

$$j^a = -\frac{1}{2}\sqrt{-g} g^{ab} (\partial_b \Phi^\star \delta\Phi + \partial_b \Phi \delta\Phi^\star). \tag{2.79}$$

For an infinitesimal $U(1)$ transformation, $\Phi \to \Phi + i\epsilon\Phi$, we have

$$\delta\Phi = i\epsilon\Phi \qquad \text{and} \qquad \delta\Phi^\star = -i\epsilon\Phi^\star, \tag{2.80}$$

and we have

$$j^a = -i\frac{1}{2}\sqrt{-g}g^{ab}(\Phi\partial_b\Phi^\star - \Phi^\star\partial_b\Phi), \tag{2.81}$$

where the constant $\epsilon$ has been factored out. Using the component form of the field, $\Phi = \phi_1 + i\phi_2$, this can also be expressed as:

$$j^a = \sqrt{-g}g^{ab}(\phi_2\partial_b\phi_1 - \phi_1\partial_b\phi_2). \tag{2.82}$$

Finally the time component of the current density in a 3+1 coordinate basis assumes the following form:

$$\begin{aligned} j^t &= \sqrt{-g}\left[g^{tt}(\phi_2\partial_t\phi_1 - \phi_1\partial_t\phi_2) + g^{ti}(\phi_2\partial_i\phi_1 - \phi_1\partial_i\phi_2)\right] \\ &= \sqrt{-g}\left[-\frac{1}{\alpha^2}(\phi_2\partial_t\phi_1 - \phi_1\partial_t\phi_2) + \frac{\beta^i}{\alpha^2}(\phi_2\partial_i\phi_1 - \phi_1\partial_i\phi_2)\right] \\ &= \phi_1\Pi_2 - \phi_2\Pi_1, \end{aligned} \tag{2.83}$$

where equation (2.57) was used to simplify the second line of the above, and to express $\partial_t\phi_A$ in terms of their respective conjugate momenta $\Pi_A$. Choosing the fixed volume element $\mathbf{e}$ on $\Sigma_t$ to be the coordinate volume element $d^3x$, the Noether charge can be written as:

$$Q_N = \int_{\Sigma_t}(\phi_1\Pi_2 - \phi_2\Pi_1)\, d^3x \tag{2.84}$$

and can be expected to be conserved: i.e. to have the same value on each slice $\Sigma_t$ of the spacetime foliation. This expression is used in Chap. 4 as one diagnostic to ensure that the numerical code used to solve our model system is producing sensible results.

## 2.4 Conformally Flat and Maximal Slicing Conditions

This section discusses in details how the CFA with maximal slicing condition are applied to the EKG system and what the resulting equations of motion are. Sec. 2.4.1 briefly reviews the maximal slicing condition as a way to fix the time coordinate. The reader should notice however that this slicing condition is actually a necessary condition to be applied in addition to the CFA. As it will become clearer later, when both conditions are applied together to the conformal Einstein equations, the extrinsic curvature evolution equations can be neglected (except for its trace part). Sec. 2.4.2 shows in details how to conformally decompose the $3+1$ Einstein equations. Despite this decomposition not being used in its full form later in this thesis, it is included here for sake of completeness and to provide a framework in which the effects of the CFA on the Einstein equations would be more evident. The key approximation used in this thesis is then exposed in Sec. 2.4.3 while Sec. 2.4.4 presents in cartesian coordinates the whole set of equations for our model, the Klein-Gordon field within the conformally flat approximation to general relativity. Sec. 2.4.5 discussed several different strategies for supplementing the equations of motion with boundary conditions. At last, Sec. 2.4.6 discuss the calculation of the ADM mass under these conditions.

### 2.4.1 Time Coordinate Choice: Maximal Slicing

In the initial value formulation or Cauchy formulation of the Einstein equations, the choice of time coordinate is related to the choice of the spacelike hypersurface since the hypersurfaces are level surfaces of the time coordinate. As briefly discussed in Sec. 2.2, the embedding of these three dimensional hypersurfaces in the four dimensional spacetime is controlled by the extrinsic curvature $K_{ab}$. It is natural therefore to choose the time coordinate by imposing a condition on the extrinsic curvature. One particular choice widely used in numerical relativity consists of imposing the following condition on the trace of the extrinsic curvature tensor:

$$K \equiv K^a{}_a = 0 \qquad \text{and} \qquad \partial_t K = 0. \tag{2.85}$$

This choice is called *maximal slicing* and it is a particularly useful slicing for numerical computation since it tends to avoid any sort of spacetime singularity by freezing the slice time evolution in regions close to singular points [109]. For a congruence of normal geodesics – observers travelling normal to the hypersurface – the maximal slicing condition imposes that the expansion of this congruence vanishes therefore avoiding them to focus and form caustics. This demand also causes the volume of the hypersurface to be maximal. For a more detailed discussion of this slicing condition and a proof of its properties, the reader should refer to *Gravitation* [110].

### 2.4.2 Conformal Decomposition

Conformal decompositions of the Einstein equations in the context of $3+1$ formalism were first introduced by Lichnerowicz in 1944 [30], with a proposal for conformal decomposition of the Hamiltonian constraint. The idea was to cast the Hamiltonian constraint in a pure elliptic form and prove existence and uniqueness of the constraint solutions. Later on in the seventies, James York

[111, 112] worked on the problem of initial data by posing the question of which field variables are freely specifiable on the initial slice and which one are constrained. In analogy to electromagnetism where the field can be decomposed in longitudinal and transverse parts and thereby the gauge and physical parts separated, a strong interest arose in tensor decompositions as well [113, 114, 115]. The main idea was that the covariant decomposition of symmetric tensors would provide at least approximately or formally a solution to the constraint equations (and show that the solutions are unique) besides characterizing the dynamical (unconstrained) variables covariantly. Of a particular interest was the so called conformal transverse-traceless decomposition. This decomposition was able to cast the constraints in an explicit elliptic form and consequently to prove the existence and uniqueness of the constrained solutions. From these studies and the analysis of the asymptotic behaviour of the Riemann curvature tensor by Newman-Penrose [116] emerged the notion that the gravitational radiation may be represented by the transverse-traceless part of the pair 3-metric and extrinsic curvature $\{\gamma_{ab}, K_{ab}\}$ besides representing a pure spin 2 part of the tensors. This interpretation is accepted in this section with no further discussion. For a comparison between the formalisms the reader is refered to the paper by Thorne [117].

The main purpose of this section is to conformally decompose the $3+1$ Einstein equations. It is expected then to separate the coordinate from the dynamical part. This decomposition becomes the basis for the approximation to be introduced in the next section: conformally flat condition on a maximal slice. The discussion here follows approximately the lecture notes on $3+1$ formalism by Gourgoulhon [118] and, as mentioned before, it is included here for completeness.

The first step towards the conformal decomposition of the Einstein equations is to investigate how each of the elements entering the equation changes under a conformal transformation. These elements are basically the 3-metric $\gamma_{ij}$, the connection $C^i_{\ jk}$ associated with the choices of the spatial covariant derivatives $D_i$, the Ricci tensor for the $\Sigma_t$ slicing geometry and finally the extrinsic curvature $K_{ij}$. Each of them is discussed below and all of them are put together in the Einstein equations afterwards.

A conformal decomposition of the induced spatial metric $\gamma_{ij}$ means that a base, unphysical or also called conformal metric $\tilde{\gamma}_{ij}$ is associated to the physical one through a strictly positive function of the coordinates, the conformal factor $\psi = \psi(x^\mu)$. This relationship is expressed as:

$$\gamma_{ij} = \psi^4 \tilde{\gamma}_{ij} \tag{2.86}$$

and, since $\gamma_{ij}\gamma^{jk} = \tilde{\gamma}_{ij}\tilde{\gamma}^{jk} = \delta^k_i$, the inverse relationship becomes:

$$\gamma^{ij} = \psi^{-4} \tilde{\gamma}^{ij}. \tag{2.87}$$

The covariant derivative can be defined in terms of the ordinary derivative operator:

$$D_i v^j = \partial_i v^j + \Gamma^j_{\ ik} v^k, \tag{2.88}$$

where $v^j$ is a spatial vector and $\Gamma^j_{\ ik}$ is the Christoffel symbol. If the covariant derivative is associated to the spatial metric, as it is going to be the case here, then $D_i \gamma_{jk} = 0$ and the Christoffel symbol is calculated in terms of the ordinary derivatives by the formula:

$$\Gamma^{j}_{\ ik} = \frac{1}{2}\gamma^{jl}\left(\partial_i\gamma_{kl} + \partial_k\gamma_{il} - \partial_l\gamma_{ik}\right) \tag{2.89}$$

and the same is true for the covariant derivative associated to the conformal metric, $\tilde{D}_i\tilde{\gamma}_{jk} = 0$. Both covariant derivatives can be related to each other through the connection tensor $C^{j}_{\ ik}$:

$$D_i v^j = \tilde{D}_i v^j + C^{j}_{\ ik} v^k, \tag{2.90}$$

where the connection is calculated in terms of the conformal covariant derivatives instead of ordinary derivative operator (where it has the special name Christoffel symbol) for a covariant derivative associated to the spatial metric $D_i\gamma_{jk} = 0$:

$$C^{j}_{\ ik} = \frac{1}{2}\gamma^{jl}\left(\tilde{D}_i\gamma_{kl} + \tilde{D}_k\gamma_{il} - \tilde{D}_l\gamma_{ik}\right) \tag{2.91}$$

and can be expressed in terms of the conformal metric by substituting equations (2.86)and(2.87) into equation (2.91) and using the fact that the conformal covariant derivative is metric compatible $\tilde{D}_i\tilde{\gamma}_{jk} = 0$:

$$C^{j}_{\ ik} = 2\tilde{\gamma}^{jl}\left[\tilde{D}_i\left(\ln\psi\right)\tilde{\gamma}_{kl} + \tilde{D}_k\left(\ln\psi\right)\tilde{\gamma}_{il} - \tilde{D}_l\left(\ln\psi\right)\tilde{\gamma}_{ik}\right], \tag{2.92}$$

while tracing the equation above gives an useful expression for the divergence of a vector in terms of the conformal covariant derivative and the conformal factor:

$$D_i v^i = \tilde{D}_i v^i + C^{i}_{\ ik} v^k = \tilde{D}_i v^i + 6\tilde{D}_i\left(\ln\psi\right) v^i = \frac{1}{\psi^6}\tilde{D}_i\left(\psi^6 v^i\right). \tag{2.93}$$

The next object of interest to appear in the $3+1$ equations is the Ricci tensor. The main goal is to express the Ricci tensor associated with the physical metric $\mathcal{R}_{ik}$ to the Ricci tensor associated with the conformal metric $\tilde{\mathcal{R}}_{ik}$ (and its covariant derivatives, connections, conformal factor, etc...). In order to do that, one should start with the definition of the Riemann tensor for the three dimensional spacelike hypersurface, equation (2.20), and express it in terms of the connection tensor $C^{j}_{\ ik}$. From its definition:

$$\mathcal{R}_{ijk}^{\ \ \ l} w_l = 2D_{[i}D_{j]}w_k \tag{2.94}$$

and from the generalization of the covariant derivative of a tensor of type $(p,q)$:

$$D_k T^{i_1\ldots i_p}_{\ \ \ \ \ j_1\ldots j_q} = \tilde{D}_k T^{i_1\ldots i_p}_{\ \ \ \ \ j_1\ldots j_q} + \sum_{l=1}^{p} C^{il}_{\ \ ks} T^{i_1\ldots s\ldots i_p}_{\ \ \ \ \ \ \ \ j_1\ldots j_q} - \sum_{l=1}^{q} C^{s}_{\ kjl} T^{i_1\ldots i_p}_{\ \ \ \ \ j_1\ldots s\ldots j_q}, \tag{2.95}$$

it can be shown, after a few algebraic simplification, that the Riemann tensor becomes:

$$\mathcal{R}_{ijk}^{\ \ \ l} = \tilde{\mathcal{R}}_{ijk}^{\ \ \ l} - 2\tilde{D}_{[i}C^{l}_{\ j]k} + 2C^{m}_{\ k[i}C^{l}_{\ j]m}, \tag{2.96}$$

where by definition:

$$\tilde{\mathcal{R}}_{ijk}{}^{l} w_l = 2\tilde{D}_{[i}\tilde{D}_{j]} w_k. \tag{2.97}$$

On the other hand the Ricci tensor is obtained by tracing the second and forth indices of the Riemann tensor, equation (2.96):

$$\mathcal{R}_{ik} = \tilde{\mathcal{R}}_{ik} - 2\tilde{D}_{[i}C^{l}_{\ l]k} + 2C^{m}_{\ k[i}C^{l}_{\ l]m}, \tag{2.98}$$

that can be rewritten in terms of the conformal factor as:

$$\begin{aligned} \mathcal{R}_{ik} &= \tilde{\mathcal{R}}_{ik} - 2\tilde{D}_i\tilde{D}_k\left(\ln\psi\right) - 2\tilde{\gamma}_{ik}\tilde{\gamma}^{lm}\tilde{D}_l\tilde{D}_m\left(\ln\psi\right) \\ &+ 4\tilde{D}_i\left(\ln\psi\right)\tilde{D}_k\left(\ln\psi\right) - 4\tilde{\gamma}_{ik}\tilde{\gamma}^{lm}\tilde{D}_l\left(\ln\psi\right)\tilde{D}_m\left(\ln\psi\right). \end{aligned} \tag{2.99}$$

The Ricci scalar is obtained by tracing the Ricci tensor, equation (2.99). That results into an expression in terms of the conformal factor, conformal covariant derivatives and the Ricci scalar associated with the conformal metric:

$$\begin{aligned} \mathcal{R} \equiv \gamma^{ik}\mathcal{R}_{ik} &= \psi^{-4}\left[\tilde{\mathcal{R}} - 8\tilde{\gamma}^{ik}\tilde{D}_i\tilde{D}_k\left(\ln\psi\right) - 8\tilde{\gamma}^{ik}\tilde{D}_i\left(\ln\psi\right)\tilde{D}_k\left(\ln\psi\right)\right] \\ &= \psi^{-4}\tilde{\mathcal{R}} - 8\psi^{-5}\tilde{\gamma}^{ik}\tilde{D}_i\tilde{D}_k\psi, \end{aligned} \tag{2.100}$$

where $\tilde{\mathcal{R}} \equiv \tilde{\gamma}^{ik}\tilde{\mathcal{R}}_{ik}$.

The last element to enter the Einstein equations that needs to be decomposed is the extrinsic curvature $K_{ij}$. In order to conform with the maximal slicing choice of time coordinate and make the approximation to be applied later to the equations explicitly evident, it is convenient to decompose the extrinsic curvature first in its trace and traceless part:

$$K_{ij} = A_{ij} + \frac{1}{3}K\gamma_{ij} \qquad \text{and} \qquad K^{ij} = A^{ij} + \frac{1}{3}K\gamma^{ij}, \tag{2.101}$$

where $A_{ij}$ is by definition the extrinsic curvature traceless part, $\gamma^{ij}A_{ij} = 0$, and $K$ is its trace with respect to the spatial metric, $K \equiv \gamma^{ij}K_{ij}$.

The evolution equation for the spatial metric components, equation (2.40), provides a natural framework to choose the scaling exponent $s$ in the conformal relation for the traceless part of the extrinsic curvature, $A^{ij} = \psi^s\tilde{A}^{ij}$:

$$\mathcal{L}_m\gamma_{ij} \equiv \mathcal{L}_{(t-\beta)}\gamma_{ij} = -2\alpha\gamma_{ik}K^{i}_{\ j}, \tag{2.102}$$

where $m^a = t^a - \beta^a = \alpha n^a$. Inserting the extrinsic curvature decomposition, equation (2.101), and the conformal metric decomposition, the equation above becomes:

$$\mathcal{L}_m\tilde{\gamma}_{ij} = -2\alpha\psi^{-4}A_{ij} - \frac{2}{3}\alpha K\tilde{\gamma}_{ij} - 4\tilde{\gamma}_{ij}\mathcal{L}_m\left(\ln\psi\right). \tag{2.103}$$

Note that the equation above has time derivatives (through the lie derivatives along $m^a$) for both

the conformal metric and conformal factor. One could say that it is a coupled evolution equation for both quantities. Fortunately, tracing this equation with respect to the conformal metric $\tilde{\gamma}_{ij}$ does the job of decoupling the evolution equation for the conformal metric from one for the conformal factor:

$$\tilde{\gamma}^{ij}\mathcal{L}_m\tilde{\gamma}_{ij} = -2\alpha K - 12\mathcal{L}_m\left(\ln\psi\right). \tag{2.104}$$

From a well known relationship between the variation or derivative of the logarithm of the determinant of a invertible matrix $M$ and the trace of the matrix product of its inverse $M^{-1}$ by its variation or derivative, i.e. $\delta(\ln\det M) \equiv \mathrm{tr}\ (M^{-1}\times\delta M)$, the left-hand side of the equation above can be expressed as:

$$\tilde{\gamma}^{ij}\mathcal{L}_m\tilde{\gamma}_{ij} = \mathcal{L}_m\left(\ln\det\tilde{\gamma}_{ij}\right) = \mathcal{L}_m\left(\ln\tilde{\gamma}\right). \tag{2.105}$$

It is usual to require in the conformal decomposition formalism that the determinant of the conformal spatial metric be Lie dragged from one slice to the other, i.e. $\mathcal{L}_t\tilde{\gamma} = 0$, preserving this way a unit of volume between the hypersurfaces. After substituting this requirement, the expression above becomes:

$$\tilde{\gamma}^{ij}\mathcal{L}_m\tilde{\gamma}_{ij} = -\mathcal{L}_\beta\left(\ln\tilde{\gamma}\right). \tag{2.106}$$

Moreover, the left-hand side of the equation above can be reexpressed in terms of the trace of the Lie derivative along the shift vector $\beta^i$ by the same sort of argument used to express determinants in terms of traces. Therefore the expression above becomes:

$$\tilde{\gamma}^{ij}\mathcal{L}_m\tilde{\gamma}_{ij} = -\tilde{\gamma}^{ij}\mathcal{L}_\beta\tilde{\gamma}_{ij} = -2\tilde{D}_i\beta^i. \tag{2.107}$$

Once this relationship is inserted back into equation (2.104), it clearly provides an evolution equation only for the conformal factor:

$$\left(\partial_t - \mathcal{L}_\beta\right)\ln\psi = \frac{1}{6}\left(\tilde{D}_i\beta^i - \alpha K\right), \tag{2.108}$$

while the same relationship inserted back on equation (2.103) gives an evolution equation for the conformal metric:

$$\left(\partial_t - \mathcal{L}_\beta\right)\tilde{\gamma}_{ij} = -2\alpha\tilde{A}_{ij} - \frac{2}{3}\tilde{\gamma}_{ij}\tilde{D}_k\beta^k, \tag{2.109}$$

where $\tilde{A}_{ij} = \psi^{-4}A_{ij}$. Note that $\tilde{A}_{ij}$ is still traceless: $\tilde{\gamma}^{ij}\tilde{A}_{ij} = \psi^4\gamma^{ij}\psi^{-4}A_{ij} = 0$ and that its bilinear dual form is obtained by raising the indexes with the inverse conformal factor, resulting into:

$$A^{ij} = \psi^{-4}\tilde{A}^{ij}, \tag{2.110}$$

what clearly suggests that the conformal exponent $s$ for the traceless extrinsic curvature be adopted as $s = -4$. This choice arose naturally from the decomposition of the evolution equation for the

spatial metric. It is worth noting though that this choice is not unique. Another common choice that was first adopted by Lichnerowicz [30] is $s = -10$. It also arises naturally from a decomposition of one the Einstein equations but this time it comes from the decomposition of the momentum constraint equations. As this particular decomposition is not applied in the approximation to be presented in the next section, the reader is recomended to look at Gourgoulhon's lectures [118] for a more detailed discussion.

The decomposition of the extrinsic curvature allows to express its evolution equation (equation (2.41) in the 3+1 formalism) into two parts: one evolution equation for its conformal traceless part and another for its trace part. The first step to rewrite equation (2.41) is substituting the extrinsic curvature decomposition $K_{ij} = A_{ij} + \frac{1}{3}K\gamma_{ij}$ in the left-hand side of its lower index version [4]:

$$\mathcal{L}_m K_{ij} = -D_i D_j \alpha + \alpha \left[ \mathcal{R}_{ij} + K K_{ij} - 2K_{ik}K^k_{\ j} + 8\pi \left( \frac{1}{2}\gamma_{ij}(S-\rho) - S_{ij} \right) \right]. \tag{2.111}$$

Thus the left-hand side of the equation above becomes:

$$\mathcal{L}_m K_{ij} = \mathcal{L}_m A_{ij} + \frac{1}{3}\gamma_{ij}\mathcal{L}_m K + \frac{1}{3}K\mathcal{L}_m\gamma_{ij} = \mathcal{L}_m A_{ij} + \frac{1}{3}\gamma_{ij}\mathcal{L}_m K - \frac{2}{3}\alpha K K_{ij}, \tag{2.112}$$

where the equation (2.102) was used in the last equality. Note also that the lie derivative of the extrinsic curvature trace can be rewritten as:

$$\mathcal{L}_m K = \mathcal{L}_m \left( \gamma^{ij} K_{ij} \right) = \gamma^{ij}\mathcal{L}_m K_{ij} + K_{ij}\mathcal{L}_m\gamma^{ij} = \gamma^{ij}\mathcal{L}_m K_{ij} + 2\alpha K_{ij}K^{ij}, \tag{2.113}$$

where the upper index version of equation (2.102) was used in the last equality. One last piece that can be simplified in the equation above corresponds to the trace of equation (2.111):

$$\gamma^{ij}\mathcal{L}_m K_{ij} = -\gamma^{ij} D_i D_j \alpha + \alpha \left[ \mathcal{R} + K^2 - 2K_{ij}K^{ij} + 4\pi\left(S - 3\rho\right) \right]. \tag{2.114}$$

Substituting the equation (2.114) into equation (2.113), results into:

$$\mathcal{L}_m K = -\gamma^{ij} D_i D_j \alpha + \alpha \left[ \mathcal{R} + K^2 + 4\pi\left(S - 3\rho\right) \right]. \tag{2.115}$$

By making use of the hamiltonian equation, equation (2.42), the equation above can be simplified even further and be considered as the evolution equation for the trace part of the extrinsic curvature tensor:

$$\mathcal{L}_m K \equiv \left(\partial_t - \mathcal{L}_\beta\right) K = -\gamma^{ij} D_i D_j \alpha + \alpha \left[ K_{ij}K^{ij} + 4\pi\left(S + \rho\right) \right]. \tag{2.116}$$

Replacing the evolution equation for the trace part, equation (2.115)[5], and the evolution equation

[4] Note the presence of the term $-2K_{ik}K^k_{\ j}$ in the lower index version that is absent in the mixed index version, equation (2.41).

[5] Using equation (2.115) doesn't make use of the hamiltonian constraint yet as in equation (2.116) besides resulting in the elimination of the density $\rho$ in the subsequent equation.

for the extrinsic curvature itself, equation (2.111), into the equation (2.112) provides an evolution equation for the traceless part of the extrinsic curvature tensor, $A_{ij}$:

$$\begin{aligned}\mathcal{L}_m A_{ij} = -D_i D_j \alpha &+ \alpha\left[\mathcal{R}_{ij} + \frac{5}{3}KK_{ij} - 2K_{ik}K^k_{\ j} - 8\pi\left(S_{ij} - \frac{1}{3}\gamma_{ij}S\right)\right] \\ &+ \frac{1}{3}\gamma_{ij}\left[\gamma^{lk}D_l D_k\alpha - \alpha\left(\mathcal{R} + K^2\right)\right], \end{aligned} \tag{2.117}$$

that can be simplified further by introducing the expression for $K_{ij}$ decomposition into the right-hand side of the expression above. After a little bit of manipulation that results into:

$$\begin{aligned}\mathcal{L}_m A_{ij} = -D_i D_j \alpha &+ \alpha\left[\mathcal{R}_{ij} + \frac{1}{3}KA_{ij} - 2A_{ik}A^k_{\ j} - 8\pi\left(S_{ij} - \frac{1}{3}\gamma_{ij}S\right)\right] \\ &+ \frac{1}{3}\gamma_{ij}\left(\gamma^{lk}D_l D_k\alpha - \alpha\mathcal{R}\right). \end{aligned} \tag{2.118}$$

At this point one can see that the evolution equation for the extrinsic curvature $K_{ij}$ were split into its trace part, equation (2.116), and its traceless part, equation (2.118). It is still left to express them in terms of the conformal quantities. The left-hand side of equation (2.118) can be rewritten as:

$$\mathcal{L}_m A_{ij} = \mathcal{L}_m\left(\psi^4\tilde{A}_{ij}\right) = \psi^4\mathcal{L}_m\tilde{A}_{ij} + 4\psi^3\mathcal{L}_m\psi\tilde{A}_{ij}. \tag{2.119}$$

The expression above can be simplified even further by substituting equation (2.108):

$$\mathcal{L}_m A_{ij} = \psi^4\left[\mathcal{L}_m\tilde{A}_{ij} + \frac{2}{3}\tilde{A}_{ij}\left(\tilde{D}_l\beta^l - \alpha K\right)\right]. \tag{2.120}$$

After substituting back this last expression, equation (2.118) becomes then an evolution equation for $\tilde{A}_{ij}$:

$$\begin{aligned}\mathcal{L}_m\tilde{A}_{ij} &= \psi^{-4}\left\{-D_i D_j\alpha + \alpha\left[\mathcal{R}_{ij} + \frac{1}{3}KA_{ij} - 2A_{ik}A^k_{\ j} - 8\pi\left(S_{ij} - \frac{1}{3}\gamma_{ij}S\right)\right]\right. \\ &+ \left.\frac{1}{3}\gamma_{ij}\left(\gamma^{lk}D_l D_k\alpha - \alpha\mathcal{R}\right)\right\} - \frac{2}{3}\tilde{A}_{ij}\left(\tilde{D}_l\beta^l - \alpha K\right). \end{aligned} \tag{2.121}$$

In order to keep up with the program of reexpressing equation (2.121) in terms of its conformal quantities, the double covariant derivative of the lapse needs to be expressed in terms of the conformal counterparts. That is realized as:

$$\begin{aligned} D_iD_j\alpha &= \tilde{D}_i\left(D_j\alpha\right) - C^k_{ij}\left(D_k\alpha\right) \\ &= \tilde{D}_i\tilde{D}_j\alpha - C^k_{ij}\tilde{D}_k\alpha \\ &= \tilde{D}_i\tilde{D}_j\alpha - 2\tilde{D}_k\alpha\left[\tilde{D}_i\left(\ln\psi\right)\delta^k_j + \tilde{D}_j\left(\ln\psi\right)\delta^k_i - \tilde{\gamma}_{ij}\tilde{\gamma}^{kl}\tilde{D}_l\left(\ln\psi\right)\right] \\ &= \tilde{D}_i\tilde{D}_j\alpha - 2\tilde{D}_i\left(\ln\psi\right)\tilde{D}_j\alpha - 2\tilde{D}_j\left(\ln\psi\right)\tilde{D}_i\alpha + 2\tilde{\gamma}_{ij}\tilde{\gamma}^{kl}\tilde{D}_l\left(\ln\psi\right)\tilde{D}_k\alpha, \end{aligned} \quad (2.122)$$

where the definition given by equation (2.95) was used on the first line while equation (2.92) was used in the third line. Tracing equation (2.122) one also gets another important relation entering the evolution equations:

$$\gamma^{ij}D_iD_j\alpha = \psi^{-4}\tilde{\gamma}^{ij}D_iD_j\alpha = \psi^{-4}\left[\tilde{\gamma}^{ij}\tilde{D}_i\tilde{D}_j\alpha + 2\tilde{\gamma}^{ij}\tilde{D}_i\left(\ln\psi\right)\tilde{D}_j\alpha\right]. \quad (2.123)$$

Finally the last piece to be expressed in terms of its conformal counterparts is the contraction between the extrinsic curvature tensor:

$$K_{ij}K^{ij} = \left(A_{ij} + \frac{1}{3}K\gamma_{ij}\right)\left(A^{ij} + \frac{1}{3}K\gamma^{ij}\right) = \tilde{A}_{ij}\tilde{A}^{ij} + \frac{1}{3}K^2, \quad (2.124)$$

where equation (2.110), its lower index version and the fact that $A_{ij}$ is traceless have been used.

With all the elements decomposed into their conformal parts, the evolution equation for the conformal traceless part of the extrinsic curvature, equation (2.121), finally becomes after quite a bit of algebraic simplification:

$$\begin{aligned} \left(\partial_t - \mathcal{L}_\beta\right)\tilde{A}_{ij} &= \psi^{-4}\left\{-\tilde{D}_i\tilde{D}_j\alpha + 4\tilde{D}_{(i}\left(\ln\psi\right)\tilde{D}_{j)}\alpha + \frac{1}{3}\tilde{\gamma}_{ij}\tilde{\gamma}^{kl}\left(\tilde{D}_k\tilde{D}_l\alpha - 4\tilde{D}_k\left(\ln\psi\right)\tilde{D}_l\alpha\right)\right. \\ &+ \alpha\left[\tilde{\mathcal{R}}_{ij} - \frac{1}{3}\tilde{\gamma}_{ij}\tilde{\mathcal{R}} - 2\tilde{D}_i\tilde{D}_j\left(\ln\psi\right) + 4\tilde{D}_i\left(\ln\psi\right)\tilde{D}_j\left(\ln\psi\right)\right. \\ &+ \left.\left.\frac{2}{3}\tilde{\gamma}_{ij}\tilde{\gamma}^{kl}\left(\tilde{D}_k\tilde{D}_l\left(\ln\psi\right) - 2\tilde{D}_k\left(\ln\psi\right)\tilde{D}_l\left(\ln\psi\right)\right)\right]\right\} \\ &+ \alpha\left[K\tilde{A}_{ij} - 2\tilde{\gamma}^{kl}\tilde{A}_{ik}\tilde{A}_{lj} - 8\pi\left(\psi^{-4}S_{ij} - \frac{1}{3}\tilde{\gamma}_{ij}S\right)\right] - \frac{2}{3}\tilde{A}_{ij}\tilde{D}_l\beta^l, \end{aligned} \quad (2.125)$$

where equations (2.99) and (2.100) were used for the Ricci tensor and scalar respectively, the lower index version of equation (2.110) for $A_{ij}$ and equations (2.122) and (2.123) for the lapse double covariant derivatives expressions.

On the other hand, the evolution equation for the trace part of the extrinsic curvature, equation (2.116), can also be expressed in terms of conformal quantities as:

$$\left(\partial_t - \mathcal{L}_\beta\right)K = -\psi^{-4}\tilde{\gamma}^{kl}\left[\tilde{D}_k\tilde{D}_l\alpha + 2\tilde{D}_k\left(\ln\psi\right)\tilde{D}_l\alpha\right] + \alpha\left[\tilde{A}_{kl}\tilde{A}^{kl} + \frac{1}{3}K^2 + 4\pi\left(S+\rho\right)\right], \quad (2.126)$$

where equations(2.123) and (2.124) have been used.

The hamiltonian constraint, equation (2.42), can be expressed in terms of conformal quantities by use of equations (2.100) and (2.124):

$$\tilde{\gamma}^{kl}\tilde{D}_k\tilde{D}_l\psi - \frac{1}{8}\psi\tilde{\mathcal{R}} + \left(\frac{1}{8}\tilde{A}_{kl}\tilde{A}^{kl} - \frac{1}{12}K^2 + 2\pi\rho\right)\psi^5 = 0. \tag{2.127}$$

In order to rewrite the momentum constraint, first note that:

$$\begin{aligned} D_jK^{ij} &= D_jA^{ij} + \frac{1}{3}\gamma^{ij}D_jK \\ &= \tilde{D}_jA^{ij} + C^i{}_{jk}A^{jk} + C^j{}_{jk}A^{ik} + \frac{1}{3}\psi^{-4}\tilde{\gamma}^{ij}\tilde{D}_jK \\ &= \tilde{D}_jA^{ij} + 4\tilde{D}_j\left(\ln\psi\right)A^{ij} + 6\tilde{D}_j\left(\ln\psi\right)A^{ij} + \frac{1}{3}\psi^{-4}\tilde{D}^iK \\ &= \tilde{D}_jA^{ij} + \psi^{-10}\tilde{D}_j\left(\psi^{10}A^{ij}\right) + \frac{1}{3}\psi^{-4}\tilde{D}^iK, \end{aligned} \tag{2.128}$$

where equation (2.101) is used in the first line; on the second line, equation (2.95) along with the fact that the derivatives are identical on scalar functions, i.e. $D_iK \equiv \tilde{D}_iK$; the expressions for the connection, equation (2.92), and the fact that $A^{ij}$ is traceless are used to simplify the third line. Inserting equation (2.128) into the momentum constraint, equation (2.43), results into:

$$\tilde{D}_jA^{ij} + \psi^{-10}\tilde{D}_j\left(\psi^{10}A^{ij}\right) - \frac{2}{3}\psi^{-4}\tilde{D}^iK = 8\pi J^i, \tag{2.129}$$

that finally can be expressed in terms of the conformal quantities as:

$$\tilde{D}_j\tilde{A}^{ij} + 6\tilde{D}_j\left(\ln\psi\right)\tilde{A}^{ij} - \frac{2}{3}\tilde{D}^iK = 8\pi\psi^4J^i. \tag{2.130}$$

This section has reached its goal of conformally decomposing the Einstein equations in the $3+1$ formalism. To gain some perspective on how to solve this set of equations and on how a possible approximation modify them, it is convenient to gather all of them together. Therefore, equations (2.108), (2.109), (2.126), (2.125), (2.127) and (2.130) are rewritten below as the evolution equation for the conformal factor $\psi$, or the trace part of equation (2.40):

$$\left(\partial_t - \mathcal{L}_\beta\right)\psi = \frac{1}{6}\psi\left(\tilde{D}_i\beta^i - \alpha K\right), \tag{2.131}$$

the evolution equation for the conformal spatial metric $\tilde{\gamma}_{ij}$, or the traceless part of equation (2.40):

$$\left(\partial_t - \mathcal{L}_\beta\right)\tilde{\gamma}_{ij} = -2\alpha\tilde{A}_{ij} - \frac{2}{3}\tilde{\gamma}_{ij}\tilde{D}_k\beta^k, \tag{2.132}$$

the evolution equation for the extrinsic curvature trace $K$, trace part of equation (2.41):

$$\left(\partial_t - \mathcal{L}_\beta\right)K = -\psi^{-4}\tilde{\gamma}^{kl}\left[\tilde{D}_k\tilde{D}_l\alpha + 2\tilde{D}_k\left(\ln\psi\right)\tilde{D}_l\alpha\right] + \alpha\left[\tilde{A}_{kl}\tilde{A}^{kl} + \frac{1}{3}K^2 + 4\pi\left(S+\rho\right)\right], \tag{2.133}$$

the evolution equation for the extrinsic curvature traceless part $\tilde{A}_{ij}$, traceless part of equation (2.41):

$$\begin{aligned}(\partial_t - \mathcal{L}_\beta)\,\tilde{A}_{ij} &= \psi^{-4}\left\{-\tilde{D}_i\tilde{D}_j\alpha + 4\tilde{D}_{(i}\left(\ln\psi\right)\tilde{D}_{j)}\alpha + \frac{1}{3}\tilde{\gamma}_{ij}\tilde{\gamma}^{kl}\left(\tilde{D}_k\tilde{D}_l\alpha - 4\tilde{D}_k\left(\ln\psi\right)\tilde{D}_l\alpha\right)\right. \\ &+ \alpha\left[\tilde{\mathcal{R}}_{ij} - \frac{1}{3}\tilde{\gamma}_{ij}\tilde{\mathcal{R}} - 2\tilde{D}_i\tilde{D}_j\left(\ln\psi\right) + 4\tilde{D}_i\left(\ln\psi\right)\tilde{D}_j\left(\ln\psi\right)\right. \\ &+ \left.\left.\frac{2}{3}\tilde{\gamma}_{ij}\tilde{\gamma}^{kl}\left(\tilde{D}_k\tilde{D}_l\left(\ln\psi\right) - 2\tilde{D}_k\left(\ln\psi\right)\tilde{D}_l\left(\ln\psi\right)\right)\right]\right\} \\ &+ \alpha\left[K\tilde{A}_{ij} - 2\tilde{\gamma}^{kl}\tilde{A}_{ik}\tilde{A}_{lj} - 8\pi\left(\psi^{-4}S_{ij} - \frac{1}{3}\tilde{\gamma}_{ij}S\right)\right] - \frac{2}{3}\tilde{A}_{ij}\tilde{D}_l\beta^l, \end{aligned} \tag{2.134}$$

the hamiltonian constraint equation:

$$\tilde{\gamma}^{kl}\tilde{D}_k\tilde{D}_l\psi - \frac{1}{8}\psi\tilde{\mathcal{R}} + \left(\frac{1}{8}\tilde{A}_{kl}\tilde{A}^{kl} - \frac{1}{12}K^2 + 2\pi\rho\right)\psi^5 = 0, \tag{2.135}$$

and the momentum constraint equation:

$$\tilde{D}_j\tilde{A}^{ij} + 6\tilde{D}_j\left(\ln\psi\right)\tilde{A}^{ij} - \frac{2}{3}\tilde{D}^iK = 8\pi\psi^4J^i. \tag{2.136}$$

The system of equations (2.131)–(2.136) consists in the conformal $3+1$ Einstein equations. It can be solved in terms of the conformal unknowns $\tilde{\gamma}_{ij}$, $\tilde{A}_{ij}$, $\psi$ and $K$. In order to recover the physical quantities, spatial metric and extrinsic curvature, the equations (2.86) and (2.101) are applied afterwards:

$$\gamma_{ij} = \psi^4\tilde{\gamma}_{ij}, \tag{2.137}$$

$$K_{ij} = \psi^4\left(\tilde{A}_{ij} + \frac{1}{3}K\tilde{\gamma}_{ij}\right). \tag{2.138}$$

### 2.4.3 Conformally Flat Metric

The conformally flat condition fixes the conformal metric as a flat metric, $\tilde{\gamma}_{ij} \equiv \hat{\gamma}_{ij}$. Moreover, this choice is taken along with the coordinate freedom to choose the slicing. In particular the CFA is applied along with the maximal slicing condition or $K = 0$. Both conditions together simplify considerably the conformal $3+1$ set of equations derived in the last section. Taken into account that the conformal spatial covariant derivative is now associated to the flat metric, i.e. $\tilde{D}_i$ becomes simply $\hat{D}_i$ and that the Ricci tensors and scalars vanish for flat metric, i.e. $\hat{\mathcal{R}}_{ij} = 0$ and $\hat{\mathcal{R}} = 0$, the system of equations (2.131)–(2.136) immediately reduce to the evolution equation for the conformal factor $\psi$, or the trace part of equation (2.40):

$$(\partial_t - \mathcal{L}_\beta)\,\psi = \frac{1}{6}\psi\hat{D}_i\beta^i, \tag{2.139}$$

the evolution equation for the conformal spatial flat metric $\hat{\gamma}_{ij}$, or the traceless part of equation (2.40):

$$\left(\partial_t - \mathcal{L}_\beta\right)\hat{\gamma}_{ij} = -2\alpha\tilde{A}_{ij} - \frac{2}{3}\hat{\gamma}_{ij}\hat{D}_k\beta^k, \tag{2.140}$$

the evolution equation for the extrinsic curvature trace $K$, trace part of equation (2.41), now vanishes due to the maximal slicing condition and gives rise to an elliptic equation for the lapse function:

$$0 = -\psi^{-4}\hat{\gamma}^{kl}\left[\hat{D}_k\hat{D}_l\alpha + 2\hat{D}_k\left(\ln\psi\right)\hat{D}_l\alpha\right] + \alpha\left[\tilde{A}_{kl}\tilde{A}^{kl} + 4\pi\left(S+\rho\right)\right], \tag{2.141}$$

the evolution equation for the extrinsic curvature traceless part $\tilde{A}_{ij}$, traceless part of equation (2.41):

$$\begin{aligned}
\left(\partial_t - \mathcal{L}_\beta\right)\tilde{A}_{ij} &= \psi^{-4}\left\{-\hat{D}_i\hat{D}_j\alpha + 4\hat{D}_{(i}\left(\ln\psi\right)\hat{D}_{j)}\alpha + \frac{1}{3}\hat{\gamma}_{ij}\hat{\gamma}^{kl}\left(\hat{D}_k\hat{D}_l\alpha - 4\hat{D}_k\left(\ln\psi\right)\hat{D}_l\alpha\right)\right. \\
&+ \alpha\left[-2\hat{D}_i\hat{D}_j\left(\ln\psi\right) + 4\hat{D}_i\left(\ln\psi\right)\hat{D}_j\left(\ln\psi\right)\right. \\
&+ \left.\left.\frac{2}{3}\hat{\gamma}_{ij}\hat{\gamma}^{kl}\left(\hat{D}_k\hat{D}_l\left(\ln\psi\right) - 2\hat{D}_k\left(\ln\psi\right)\hat{D}_l\left(\ln\psi\right)\right)\right]\right\} \\
&+ \alpha\left[-2\hat{\gamma}^{kl}\tilde{A}_{ik}\tilde{A}_{lj} - 8\pi\left(\psi^{-4}S_{ij} - \frac{1}{3}\hat{\gamma}_{ij}S\right)\right] - \frac{2}{3}\tilde{A}_{ij}\hat{D}_l\beta^l,
\end{aligned} \tag{2.142}$$

the hamiltonian constraint equation:

$$\hat{\gamma}^{kl}\hat{D}_k\hat{D}_l\psi + \left(\frac{1}{8}\tilde{A}_{kl}\tilde{A}^{kl} + 2\pi\rho\right)\psi^5 = 0, \tag{2.143}$$

and the momentum constraint equation:

$$\hat{D}_j\tilde{A}^{ij} + 6\hat{D}_j\left(\ln\psi\right)\tilde{A}^{ij} = 8\pi\psi^4 J^i. \tag{2.144}$$

Equation (2.140) deserves special attention. By hypothesis the flat metric is fixed on each slice, what results into $\partial_t\hat{\gamma}_{ij} = 0$. The equation then becomes an expression for the traceless part of the extrinsic curvature $\tilde{A}_{ij}$ in terms of the covariant derivatives of the shift vector:

$$\tilde{A}_{ij} = \frac{1}{2\alpha}\left(\hat{\gamma}_{kj}\hat{D}_i\beta^k + \hat{\gamma}_{ik}\hat{D}_j\beta^k - \frac{2}{3}\hat{\gamma}_{ij}\hat{D}_k\beta^k\right) \tag{2.145}$$

or, using the flat metric to raise the indexes, the upper index version reads as:

$$\tilde{A}^{ij} = \frac{1}{2\alpha}\left(\hat{\gamma}^{ik}\hat{D}_k\beta^j + \hat{\gamma}^{jk}\hat{D}_k\beta^i - \frac{2}{3}\hat{\gamma}^{ij}\hat{D}_k\beta^k\right). \tag{2.146}$$

The 5 components of the traceless part of the extrinsic curvature become therefore 5 dependent variables intead of independent ones. The contraction of the two equations above with each other gives rise to a common expression entering the equations of motion in terms of the covariant derivatives of the shift vector:

$$\tilde{A}_{ij}\tilde{A}^{ij} = \frac{1}{2\alpha}\left(\hat{\gamma}_{kj}\hat{\gamma}^{il}\hat{D}_i\beta^k\hat{D}_l\beta^j + \hat{D}_k\beta^i\hat{D}_i\beta^k - \frac{2}{3}\hat{D}_i\beta^i\hat{D}_j\beta^j\right). \tag{2.147}$$

Note also that according to equation (2.124) and the maximal slicing choice, the contraction above actually corresponds to the contraction of the extrinsic curvature itself, i.e. $K_{ij}K^{ij} = \tilde{A}_{ij}\tilde{A}^{ij}$.

Another useful result entering the equations of motion for the conformally flat condition on the conformal $3+1$ equations is the divergence of $\tilde{A}^{ij}$. The divergence of equation (2.146) then becomes:

$$\begin{aligned}\hat{D}_j\tilde{A}^{ij} &= \frac{1}{2\alpha}\hat{D}_j\left(\hat{\gamma}^{ik}\hat{D}_k\beta^j + \hat{\gamma}^{jk}\hat{D}_k\beta^i - \frac{2}{3}\hat{\gamma}^{ij}\hat{D}_k\beta^k\right) - \frac{1}{\alpha}\hat{D}_j\alpha\tilde{A}^{ij} \\ &= \frac{1}{2\alpha}\left(\hat{\gamma}^{jk}\hat{D}_k\hat{D}_j\beta^i + \frac{1}{3}\hat{\gamma}^{ik}\hat{D}_k\hat{D}_j\beta^j - 2\tilde{A}^{ij}\hat{D}_j\alpha\right). \qquad (2.148)\end{aligned}$$

Inserting equations (2.148) and (2.146) into the momentum constraint, equation (2.144), one obtains the first equation used in the approximation: an elliptic equation for the shift vector given by:

$$\begin{aligned}\hat{\gamma}^{jk}\hat{D}_k\hat{D}_j\beta^i &= -\frac{1}{3}\hat{\gamma}^{ik}\hat{D}_k\hat{D}_j\beta^j + 2\tilde{A}^{ij}\left[\hat{D}_j\alpha - 6\alpha\hat{D}_j\left(\ln\psi\right)\right] + 16\pi\alpha\psi^4 J^i \\ &= -\frac{1}{3}\hat{\gamma}^{ik}\hat{D}_k\hat{D}_j\beta^j + \left[\hat{\gamma}^{ik}\hat{D}_k\beta^j + \hat{\gamma}^{jk}\hat{D}_k\beta^i - \frac{2}{3}\hat{\gamma}^{ij}\hat{D}_k\beta^k\right]\hat{D}_j\left[\ln\left(\frac{\alpha}{\psi^6}\right)\right] \\ &+ 16\pi\alpha\psi^4 J^i. \qquad (2.149)\end{aligned}$$

The hamiltonian constraint, on the other hand, provides an elliptic equation for the conformal factor $\psi$:

$$\hat{\gamma}^{kl}\hat{D}_k\hat{D}_l\psi = -\left[\frac{1}{16\alpha}\left(\hat{\gamma}_{kj}\hat{\gamma}^{il}\hat{D}_i\beta^k\hat{D}_l\beta^j + \hat{D}_k\beta^i\hat{D}_i\beta^k - \frac{2}{3}\hat{D}_i\beta^i\hat{D}_j\beta^j\right) - 2\pi\rho\right]\psi^5. \qquad (2.150)$$

The evolution equation for mean extrinsic curvature $K$ along with the maximal slicing choice of time coordinate gives an elliptic equation for the lapse function:

$$\begin{aligned}\hat{\gamma}^{kl}\hat{D}_k\hat{D}_l\alpha &= -2\hat{\gamma}^{kl}\hat{D}_k\left(\ln\psi\right)\hat{D}_l\alpha + \psi^4\alpha\left[\tilde{A}_{kl}\tilde{A}^{kl} + 4\pi\left(S+\rho\right)\right] \\ &= -2\hat{\gamma}^{kl}\hat{D}_k\left(\ln\psi\right)\hat{D}_l\alpha + \frac{1}{2}\psi^4\left[\hat{\gamma}_{kj}\hat{\gamma}^{il}\hat{D}_i\beta^k\hat{D}_l\beta^j + \hat{D}_k\beta^i\hat{D}_i\beta^k - \frac{2}{3}\hat{D}_i\beta^i\hat{D}_j\beta^j\right] \\ &+ 4\pi\psi^4\alpha\left(S+\rho\right). \qquad (2.151)\end{aligned}$$

Note that equations (2.149), (2.150) and (2.151) consist of a sufficient set of equations to determine the 4-dimensional metric $g_{\mu\nu}$. Once they are solved, the 4-dimensional metric is simply built from the $3+1$ line element:

$$\begin{aligned} ds^2 &= g_{\mu\nu} dx^\mu dx^\nu \\ &= -\alpha^2 dt^2 + \gamma_{ij} \left(dx^i + \beta^i dt\right) \left(dx^j + \beta^j dt\right) \\ &= -\alpha^2 dt^2 + \psi^4 \hat{\gamma}_{ij} \left(dx^i + \beta^i dt\right) \left(dx^j + \beta^j dt\right). \end{aligned} \tag{2.152}$$

One last topic is missing to discuss in order to cast the equations above into a form convenient for numerical computation. The choice of spatial coordinates for the flat metric allows finally to specialize these set of equations into a set of partial differential equations ready for computation. Next subsection discuss this last topic and write down the equations in the cartesian coordinate system.

### 2.4.4 Specialization to Cartesian Coordinates

The choice of spacial coordinates represent another crucial step towards the solution of the Einstein's equation coupled to matter. Usually when the spacetime is known to hold some sort of symmetry, the spatial coordinate system is chosen such as to adapt to the symmetry. The main advantage is a considerable simplification of the equations of motion. Usually though this adapted coordinate system does not cover the entire manifold of events. For example, a spherical polar coordinate system is very useful to label the manifold events for spherically symmetric spacetimes. However, this system does not cover the "$z$ axis", i.e. this system cannot label this axis uniquely – a necessary condition to define a coordinate system (or a chart) in a particular region of the manifold. Most of the metric components along with the matter source field components become ill-behaved in the neighbourhood of these points or regions of the spacetime where the coordinate system does not cover. Special treatments to regularize those components in that neighbourhood are necessary although not always sufficient to cure the function pathologies in that neighbourhood.

An ultimate goal of the work described here is the study of boson star binaries, and this type of solution will not have any specific simple symmetry that would motivate the adoption of some special curvilinear coordinates. We thus adopt (topologically) cartesian coordinates, $(x, y, z)$, which have the advantage of covering the spatial hypersurfaces in a smooth fashion (i.e. without any coodinate singularities or other pathologies) provided that there are no physical singularities on the slices.

In addition to having cartesian topology, and in contrast to some previous related work (most notably that of Wilson, Mathews, Marronetti, ..., [14]) we require our coordinate system to be "asymptotically inertial", i.e. we demand that at large distances from the matter sources the metric components approach those of flat spacetime in an inertial frame, $g_{\mu\nu} \to \eta_{\mu\nu} = \mathrm{diag}(-1, 1, 1, 1)$. In the work of Wilson and collaborators the coordinate system was typically chosen to be in corotation with the binary system being studied. Finally, we emphasize that the topology of the hypersurfaces is taken to be $\mathbb{R}^3$, so that the slices are infinite in extent in all three directions. Naturally, this has significant implications for the numerical treatment of boundary conditions in our model as will be discussed in following sections.

Thus, from this point on, the set of field variables defining our model system, namely $\alpha$,$\psi$,$\beta^i$, $\phi_A$, $\Pi_A$, $i = 1, 2, 3$, $A = 1, 2$, are all to be understood to be functions of $(t, x, y, z)$, and we now have the following for the conformally flat 3-dimensional line element is simply:

$$^{(3)}ds^2 = \psi(t, x, y, z)^4(dx^2 + dy^2 + dz^2). \tag{2.153}$$

We can now display the equations of motion in essentially the form that will be used for our numerical computations. We start with the evolution equations for the complex scalar field, (2.60) and (2.61), which become

$$\begin{aligned}
\partial_t\phi_A &= \frac{\alpha}{\psi^6}\Pi_A + \beta^i\partial_i\phi_A, && (2.154)\\
\partial_t\Pi_A &= \partial_x\left(\beta^x\Pi_A + \alpha\psi^2\partial_x\phi_A\right) + \partial_y\left(\beta^y\Pi_A + \alpha\psi^2\partial_y\phi_A\right) && (2.155)\\
&+ \partial_z\left(\beta^z\Pi_A + \alpha\psi^2\partial_z\phi_A\right) - \alpha\psi^6\frac{dU(\phi_0^2)}{d\phi_0^2}\phi_A.
\end{aligned}$$

Continuing, the energy-momentum quantities defined by (2.64)-(2.69) are

$$\begin{aligned}
\rho &= \frac{1}{2}\sum_{A=1}^{2}\left[\frac{\Pi_A^2}{\psi^{12}} + \frac{1}{\psi^4}\left[(\partial_x\phi_A)^2 + (\partial_y\phi_A)^2 + (\partial_z\phi_A)^2\right]\right] + \frac{1}{2}U(\phi_0^2), && (2.156)\\
J_i &= \sum_{A=1}^{2}\left[\frac{\Pi_A}{\psi^6}\partial_i\phi_A\right], && (2.157)\\
J^i &= \sum_{A=1}^{2}\left[-\frac{\Pi_A}{\psi^{10}}\partial_i\phi_A\right], && (2.158)\\
S_{ij} &= \frac{1}{2}\sum_{A=1}^{2}\left\{2\partial_i\phi_A\partial_j\phi_A + \psi^4\delta_{ij}\left[\frac{\Pi_A^2}{\psi^{12}}\right.\right.\\
&\left.\left. -\frac{1}{\psi^4}\left[(\partial_x\phi_A)^2 + (\partial_y\phi_A)^2 + (\partial_z\phi_A)^2\right]\right]\right\} - \frac{1}{2}\psi^4\delta_{ij}U(\phi_0^2), && (2.159)\\
S_i^i &= \frac{1}{2}\sum_{A=1}^{2}\left[3\frac{\Pi_A^2}{\psi^{12}} - \frac{1}{\psi^4}\left[(\partial_x\phi_A)^2 + (\partial_y\phi_A)^2 + (\partial_z\phi_A)^2\right]\right] - \frac{3}{2}U(\phi_0^2), && (2.160)\\
\rho + S &= \sum_{A=1}^{2}\left[2\frac{\Pi_A^2}{\psi^{12}}\right] - U(\phi_0^2). && (2.161)
\end{aligned}$$

Next, we have the equations that constrain the geometric variables. The maximal slicing condition for the lapse function becomes

$$\frac{\partial^2\alpha}{\partial x^2} + \frac{\partial^2\alpha}{\partial y^2} + \frac{\partial^2\alpha}{\partial z^2} = -\frac{2}{\psi}\left[\frac{\partial\psi}{\partial x}\frac{\partial\alpha}{\partial x} + \frac{\partial\psi}{\partial y}\frac{\partial\alpha}{\partial y} + \frac{\partial\psi}{\partial z}\frac{\partial\alpha}{\partial z}\right] + \alpha\psi^4\left(\tilde{A}_{ij}\tilde{A}^{ij} + 4\pi\left(\rho + S\right)\right), \tag{2.162}$$

while the Hamiltonian constraint (2.150) for $\psi$ is

$$\frac{\partial^2\psi}{\partial x^2}+\frac{\partial^2\psi}{\partial y^2}+\frac{\partial^2\psi}{\partial z^2}=-\frac{\psi^5}{8}\left(\tilde{A}_{ij}\tilde{A}^{ij}+16\pi\rho\right). \tag{2.163}$$

where the source term $\tilde{A}_{ij}\tilde{A}^{ij}$ is given by

$$\begin{aligned}\tilde{A}_{ij}\tilde{A}^{ij}=&\frac{1}{2\alpha^2}\Bigg[\left(\frac{\partial\beta^x}{\partial x}\right)^2+\left(\frac{\partial\beta^x}{\partial y}\right)^2+\left(\frac{\partial\beta^x}{\partial z}\right)^2+\left(\frac{\partial\beta^y}{\partial x}\right)^2+\left(\frac{\partial\beta^y}{\partial y}\right)^2+\left(\frac{\partial\beta^y}{\partial z}\right)^2\\&+\left(\frac{\partial\beta^z}{\partial x}\right)^2+\left(\frac{\partial\beta^z}{\partial y}\right)^2+\left(\frac{\partial\beta^z}{\partial z}\right)^2+\left(\frac{\partial\beta^x}{\partial x}\frac{\partial\beta^x}{\partial x}+\frac{\partial\beta^y}{\partial x}\frac{\partial\beta^x}{\partial y}+\frac{\partial\beta^z}{\partial x}\frac{\partial\beta^x}{\partial z}\right)\\&+\left(\frac{\partial\beta^x}{\partial y}\frac{\partial\beta^y}{\partial x}+\frac{\partial\beta^y}{\partial y}\frac{\partial\beta^y}{\partial y}+\frac{\partial\beta^z}{\partial y}\frac{\partial\beta^y}{\partial z}\right)+\left(\frac{\partial\beta^x}{\partial z}\frac{\partial\beta^z}{\partial x}+\frac{\partial\beta^y}{\partial z}\frac{\partial\beta^z}{\partial y}+\frac{\partial\beta^z}{\partial z}\frac{\partial\beta^z}{\partial z}\right)\\&-\frac{2}{3}\left(\frac{\partial\beta^x}{\partial x}+\frac{\partial\beta^y}{\partial y}+\frac{\partial\beta^z}{\partial z}\right)^2\Bigg].\end{aligned} \tag{2.164}$$

Finally, from the momentum constraints (2.149), we have the three equations that fix the components of the shift vector:

$$\begin{aligned}\frac{\partial^2\beta^x}{\partial x^2}+\frac{\partial^2\beta^x}{\partial y^2}+\frac{\partial^2\beta^x}{\partial z^2}&=-\frac{1}{3}\frac{\partial}{\partial x}\left(\frac{\partial\beta^x}{\partial x}+\frac{\partial\beta^y}{\partial y}+\frac{\partial\beta^z}{\partial z}\right)+\alpha\psi^4\,16\pi J^x\\&-\frac{\partial}{\partial x}\left[\ln\left(\frac{\psi^6}{\alpha}\right)\right]\left[\frac{4}{3}\frac{\partial\beta^x}{\partial x}-\frac{2}{3}\left(\frac{\partial\beta^y}{\partial y}+\frac{\partial\beta^z}{\partial z}\right)\right]\\&-\frac{\partial}{\partial y}\left[\ln\left(\frac{\psi^6}{\alpha}\right)\right]\left[\frac{\partial\beta^x}{\partial y}+\frac{\partial\beta^y}{\partial x}\right]-\frac{\partial}{\partial z}\left[\ln\left(\frac{\psi^6}{\alpha}\right)\right]\left[\frac{\partial\beta^x}{\partial z}+\frac{\partial\beta^z}{\partial x}\right],\end{aligned} \tag{2.165}$$

$$\begin{aligned}\frac{\partial^2\beta^y}{\partial x^2}+\frac{\partial^2\beta^y}{\partial y^2}+\frac{\partial^2\beta^y}{\partial z^2}&=-\frac{1}{3}\frac{\partial}{\partial y}\left(\frac{\partial\beta^x}{\partial x}+\frac{\partial\beta^y}{\partial y}+\frac{\partial\beta^z}{\partial z}\right)+\alpha\psi^4\,16\pi J^y\\&-\frac{\partial}{\partial y}\left[\ln\left(\frac{\psi^6}{\alpha}\right)\right]\left[\frac{4}{3}\frac{\partial\beta^y}{\partial y}-\frac{2}{3}\left(\frac{\partial\beta^x}{\partial x}+\frac{\partial\beta^z}{\partial z}\right)\right]\\&-\frac{\partial}{\partial x}\left[\ln\left(\frac{\psi^6}{\alpha}\right)\right]\left[\frac{\partial\beta^x}{\partial y}+\frac{\partial\beta^y}{\partial x}\right]-\frac{\partial}{\partial z}\left[\ln\left(\frac{\psi^6}{\alpha}\right)\right]\left[\frac{\partial\beta^y}{\partial z}+\frac{\partial\beta^z}{\partial y}\right],\end{aligned} \tag{2.166}$$

$$\begin{aligned}\frac{\partial^2\beta^z}{\partial x^2}+\frac{\partial^2\beta^z}{\partial y^2}+\frac{\partial^2\beta^z}{\partial z^2}&=-\frac{1}{3}\frac{\partial}{\partial z}\left(\frac{\partial\beta^x}{\partial x}+\frac{\partial\beta^y}{\partial y}+\frac{\partial\beta^z}{\partial z}\right)+\alpha\psi^4\,16\pi J^z\\&-\frac{\partial}{\partial z}\left[\ln\left(\frac{\psi^6}{\alpha}\right)\right]\left[\frac{4}{3}\frac{\partial\beta^z}{\partial z}-\frac{2}{3}\left(\frac{\partial\beta^x}{\partial x}+\frac{\partial\beta^y}{\partial y}\right)\right]\\&-\frac{\partial}{\partial y}\left[\ln\left(\frac{\psi^6}{\alpha}\right)\right]\left[\frac{\partial\beta^z}{\partial y}+\frac{\partial\beta^y}{\partial z}\right]-\frac{\partial}{\partial x}\left[\ln\left(\frac{\psi^6}{\alpha}\right)\right]\left[\frac{\partial\beta^x}{\partial z}+\frac{\partial\beta^z}{\partial x}\right].\end{aligned} \tag{2.167}$$

The four hyperbolic scalar field evolution equations, equations (2.154) and (2.156) along with

the five elliptic equations, equations (2.162)-(2.167), constitute the basic set of equations for our model problem (hereafter often simply "our model") of a Klein-Gordon field within the conformally flat approximation to general relativity. Of course, however, this set of PDEs must be supplemented by boundary and initial conditions in order to complete the mathematical prescription of the model, and these will be discussed in the sections that follow.

Before moving on to that discussion though, it is worth mentioning that the derivation of equations of motion such as the above set is a non-trivial and error prone process. It is therefore very useful to use symbolic manipulation software to check calculations, and we have done so. Specifically, after having been derived by hand, the equations presented in this thesis were checked using Maple [119], including a tensor manipulation package due to Choptuik [120].

### 2.4.5 Boundary Conditions

In this section we discuss the spatial boundary conditions that are applied to the equations of motion. As already mentioned, a key issue is the fact that the spatial domain is infinite while, of course, any specific numerical computation based on a discretization technique such as the finite differencing used in this thesis, employs a finite number of discrete unknowns.

Three different basic strategies for boundary conditions were considered during the course of this project:

1. Compactification of the spatial domain
2. Asymptotic conditions
3. Truncation (fixed Dirichlet conditions imposed at a finite distance)

We discuss each in term, highlighting specific challenges that arise, and indicating possible future directions for improvement. We note that parts of the discussion rely on computational concepts and techniques that are discussed in detail in Chap. 3. The reader may thus wish to postpone a detailed study of this part of the thesis until after a perusal of that chapter.

#### Space Compactification

The space compactification was the first strategy used to deal with the practical issue of handling an infinite domain of integration in a computer. This technique consists in mapping the whole spatial domain into a finite region of the space, i.e. in choosing a map $f$ such that:

$$f : \mathbb{R}^3 \to U \subset \mathbb{R}^3 \quad | \quad U = [-1,1] \times [-1,1] \times [-1,1]. \tag{2.168}$$

Note that as the interval $[-1,1]$ can be remapped into any other finite interval belonging to $\mathbb{R}$, there is no loss of generality in choosing the interval $[-1,1]$ for each of the coordinate directions. All that it is left is to choose a particular function whose domain is $\mathbb{R}$ and the image range $[-1,1]$. Besides, it is desirable that this map be smooth or at least $C^2$. One particular choice is to use the *hyperbolic tangent* function. So each spatial coordinate can be compactified as:

$$\xi(x) = \tanh x, \qquad \eta(y) = \tanh y, \qquad \zeta(z) = \tanh z. \tag{2.169}$$

Naturally the equations of motion are affected by the way the differential operators are expressed under this new compactified coordinate system. Thus, the first order derivative operator transforms under this specific map to:

$$\frac{d}{dx} = \frac{d\xi}{dx}\frac{d}{d\xi} = (1 - \tanh^2 x)\frac{d}{d\xi} = (1 - \xi^2)\frac{d}{d\xi}, \tag{2.170}$$

while the second order derivative operator becomes:

$$\frac{d^2}{dx^2} = (1 - \xi^2)\frac{d}{d\xi}\left[(1 - \xi^2)\frac{d}{d\xi}\right] = (1 - \xi^2)^2\frac{d^2}{d\xi^2} - 2\xi(1 - \xi^2)\frac{d}{d\xi}. \tag{2.171}$$

Inserting (2.169)-(2.171) into the equations of motion, we obtain therefore its compactified form. This kind of compactification has been applied successfully before for hyperbolic equations such as those arising from the black-string problem [121] or those coming from the modelling of binary-black-hole spacetimes [38, 122]. However, to the best of our knowledge, it has not been applied yet to equations of type elliptic. We report thus the challenges involved in the implementation of this kind of compactification for elliptic equations and point out possible solutions for this problem. In order to understand the source of problem though it is more convenient to restrict our attention to a model problem. The poisson equation is then the best candidate that still captures the essential features of the elliptic equations of motion derived on the previous sections:

$$\nabla^2 u = \rho \qquad \Longleftrightarrow \qquad \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} = \rho(x, y, z), \tag{2.172}$$

where $u = u(x, y, z)$ is the unknown function while $\rho(x, y, z)$ is the source. The Poisson equation assumes the following compactified form when the second order operator are replaced by its compactified version:

$$(1 - \xi^2)^2\frac{\partial^2 \bar{u}}{\partial \xi^2} + (1 - \eta^2)^2\frac{\partial^2 \bar{u}}{\partial \eta^2} + (1 - \zeta^2)^2\frac{\partial^2 \bar{u}}{\partial \zeta^2} - 2\xi(1 - \xi^2)\frac{\partial \bar{u}}{\partial \xi} - 2\eta(1 - \eta^2)\frac{\partial \bar{u}}{\partial \eta} - 2\zeta(1 - \zeta^2)\frac{\partial \bar{u}}{\partial \zeta} = \rho(\xi, \eta, \zeta), \tag{2.173}$$

where $\bar{u} = \bar{u}(\xi, \eta, \zeta) = u(x, y, z)$. Now, since $-1 \leq \xi, \eta, \zeta \leq 1$ the issue of assigning a boundary condition to the equation at infinity disappeared. One can just set Dirichlet boundary conditions, i.e. $\bar{u} =$ constant, at the faces, edges and corners of the cube formed by the boundary of the region $[-1, 1]^3$. As mentioned before, the same sort of compactification applied to Poisson equation can also be employed to the elliptic equations for the metric variables. Despite the increase of the number of terms and factors in the equations, the compactification is quite advantageous and desirable in order to set the physical boundary conditions *exactly*. As said before, the lapse and conformal factor at the boundary of the cubic domain are set to their flat values, equals to 1, while

the shift vector vanishes, also a restraint physically in accordance to the values the shift vector would have in the Minkowski spacetime with cartesian coordinates.

Despite the appealing simplicity of this strategy to implement the boundary conditions, a serious disadvantage arises. Suppose, for example, that the Poisson equation is written with coefficients $A$, $B$ and $C$ accompanying its derivative terms:

$$A\frac{\partial^2 u}{\partial x^2} + B\frac{\partial^2 u}{\partial y^2} + C\frac{\partial^2 u}{\partial z^2} = \rho(x, y, z). \tag{2.174}$$

If the values of $A$, $B$ and $C$ differ considerably from each other, then the equation is said to be *anisotropic*. Note also that the anisotropic character of the equation can change through the domain of integration. Therefore, only some regions at particular instant of time (if the coefficients are time dependent as well) may be considered anisotropic. The equation (2.173), for example, describing the compactified form of Poisson equation shares this property.

The trouble with anisotropic elliptic equations comes from a technical issue on the numerical method to solve these sort of equations, the multigrid method[6]. As it is going to be mentioned again in Chap. 3, the multigrid method is the most efficient numerical algorithm to solve PDEs of the type elliptic. The amount of computational work usually required to obtain the solution scale linealy with the number of unknowns, i.e. it scale as $O(N)$ where $N$ is the number of unknowns. As the name already suggests, the multigrid method makes use of several different grids with different spacings between the grid points in order to solve the equations.

At the heart of a good multigrid solver lies the relaxation method. Its main function is to smooth the solution found on a particular grid (and because of this feature is also called smoother) before transfering that solution to a coarser or finer grid. Usually the smoother used acts on the equation at each point of the grid per sweep, i.e. the equations are locally solved on each point of the mesh whenever that point is visited through an iterative process. For most problems involving elliptic equations, this pointwise smoothing process along with the other usual multigrid components are good enough to obtain the solution as the expected $O(N)$ amount of numerical work.

For the case of anisotropic elliptic equations though, this smoothing process is not efficient to smooth the solution error along directions of weak coupling of the unknowns. For example, suppose that $B \simeq C >> A$ in the equation (2.174). In this case the equation is said to be anisotropic along the weak coupling direction, the $x$ direction. If the usual smoother is applied to solve the equation in this case, it would fail to make the whole solver to converge with the order expected. On the other hand, if a more sophisticated smoother like a line smoother were applied along the $x$ direction, then the efficient convergence properties of the multigrid solver would be recovered. The line smoother solves the equations *simultaneously* for a line of grid points where the unknowns are weakly coupled.

While the line smoothers or more generally the block smoothers (for a block of unknowns weakly coupled) are methods of choice for anisotropic problems, they introduce also new challenges. For example, when the number of unknowns becomes sufficiently large, it turns out to be impossible to solve the equations by making use only of the computational resources, memory and CPU time, of one computer. It is required then that the computational domain be decomposed into

[6]The reader unfamiliar with the multigrid jargon should refer to Chap. 3 before reading the end of this section.

several pieces to be handled by different processors and therefore to share the computational cost of obtaining the equations solutions; it is required then that the problem be parallelized. It turns out that the parallelization of a block smoother may be quite far from trivial. For special cases when the direction or region of the weak coupling is known a priori, it may be possible to assign a domain decomposition such that those regions are not decomposed. However, as it is the case of the compactified Poisson equation above, equation (2.173), or the compactified form of the equations of motion of the last section, that region or direction is not particularly clear and worse it may change with the resolution of the grid used. That requires an even more sophisticated smoother to capture this feature. Usually a smoother that alternates the smoothing directions of weak coupling is robust enough to be applied to several different problems and does a good job in smoothing the solution error. However this kind of smoothers are extremely hard to effectively parallelize.

The difficulty in parallelizing the alternating line smoothers, for example, forces one to look for another strategy to supplement the equations of motion with boundary conditions. Nonetheless, as this strategy still seems quite appealing for its simplicity and correctness, it is still an on-going project to look for alternatives to solve the compactified elliptic equations. One that seems extremely promising is to keep the pointwise smoother and alter the other multigrid components such as the coarsening direction. The idea is to coarsen the grid along one direction first and by doing so transforming the supposed direction of weak coupling into a strong one, where a pointwise smoother perfects at the job. Alternating the directions of coarsening could lead to a robust method to deal with a priori unknown weak regions of coupling. Furthermore, this change on the coarsening procedure of the multigrid algorithm, called alternate semi-coarsening, promises to be easily parallelizable since it is a local operation, i.e. it applies to each grid point at the time.

### Sommerfeld-like Boundary Conditions

While the first strategy for assigning boundary conditions are still under investigation, a second strategy was implemented. This consists in truncating the domain of integration to a finite region and evaluating the behaviour of the several unknown variables as they approach infinity in order to carefully assign a boundary condition that captures this feature. As mentioned earlier the spacetime for isolated bodies is asymptotically flat. Therefore both the metric components and the matter fields are expected to approach infinity such that they differ from the spatial flat metric $\mathrm{diag}(1,1,1)$ by terms $O(1/r)$ (as $r \to \infty$)[103], where $r$ is the usual variable from the spherical polar coordinate system and can be written in terms of cartesian coordinates by the usual formula: $r = \sqrt{x^2+y^2+z^2}$. The main disadvantage of this approach when compared to the compactification of the spatial domain resides on the lack of a priori knowledge of the region where the asymptotically flatness condition is valid. To overcome this problem though it is possible to push the coordinate of the boundaries of the domain as far as possible from the main region of interest where the stars coalesce. The domain of integration in this procedure becomes then $0 \leq t \leq t_{max}$, $x_{min} \leq x \leq x_{max}$, $y_{min} \leq y \leq y_{max}$ and $z_{min} \leq z \leq z_{max}$. For the metric components carrying the static or Newtonian-like behaviour of the gravitational field the asymptotic behaviour can be translated as:

$$u = u_\infty + \frac{k}{r}, \tag{2.175}$$

where $k$ is a unknown constant depending of the particular component being considered while $u = u(r)$ represents the metric component itself and $u_\infty$ is its expected value at spatial infinity.

The equation (2.175) can be expressed in a more convenient form where the explicit knowledge of the constant $k$ is not necessary. Taking its derivative with respect to the coordinate $r$ and substituting back the value of $k$ in terms of $u$ and $u_\infty$, the condition becomes:

$$r u_{,r} + u - u_\infty = 0, \tag{2.176}$$

where $u_{,r} \equiv \frac{\partial u}{\partial r}$. Performing the coordinate transformation from the spherical polar coordinates back into cartesian ones and applying the chain rule for the derivative with respect to $r$, i.e. inserting the following expression:

$$u_{,r} = u_{,x}\frac{\partial x}{\partial r} + u_{,y}\frac{\partial y}{\partial r} + u_{,z}\frac{\partial z}{\partial r} = u_{,x}\frac{x}{r} + u_{,y}\frac{y}{r} + u_{,z}\frac{z}{r} \tag{2.177}$$

into equation (2.176), the boundary condition finally becomes:

$$x u_{,x} + y u_{,y} + z u_{,z} + u - u_\infty = 0. \tag{2.178}$$

This sort of boundary condition is called Robin boundary condition since it specifies both the function and its derivatives at the boundary. It is also known as mixed boundary conditions.

The same idea can be applied to the radiative components of the metric (that it is absent in the case of conformally flat condition) and to the scalar field components. However, the wave-like behaviour has to be taken into account in this case. So a function of the coordinate combination $r - t$ captures this feature. This condition is also known as *Sommerfeld* boundary condition. Let the fall-off be expressed then by the following function:

$$v = \frac{h(r-t)}{r}, \tag{2.179}$$

where $h(r-t)$ is a unknown outgoing wave profile and $v = v(t, r)$ represents one of the scalar field variables. The same trick can be applied in this case as well. Differentiate with respect to time and coordinate $r$ separately and use the results to eliminate $h(r-t)$ from the condition:

$$\dot{v} = -\frac{h'(r-t)}{r} \qquad \text{and} \qquad v_{,r} = \frac{h'(r-t)}{r} - \frac{h(r-t)}{r^2}, \tag{2.180}$$

where the prime differentiation refers to the variable $p \equiv r - t$. Eliminate $h'$ by equating both expressions and insert the value of $h(r-t)$ in terms of $v$ and $r$, and the boundary condition for radiative variables becomes:

$$r\dot{v} + r v_{,r} + v = 0, \tag{2.181}$$

that can also be transformed back into cartesian coordinates with the final result given by:

$$\sqrt{x^2+y^2+z^2}\,\dot{v} + xv_{,x} + yv_{,y} + zv_{,z} + v = 0. \tag{2.182}$$

This condition, equation (2.182) was applied successfully for the complex scalar field components. The discrete version of this equation [7] is centered between the last two grid points along each of the 3 possible directions, i.e. the physical boundary condition expressed through equation (2.182) does not coincide with grid boundaries, it is centered between the last 2 grid points, one belonging to the face boundary and the other to the interior domain. The grid points at the boundaries are then expressed in terms of the interior grid points through the discrete version of equation (2.182). It is worth mentioning the edges and corners of the cubic boundary requires to be handle with a little more carefully. There the physical boundary condition is actually centered at the square formed by the 4 grid points at the edge boundary and centered at the the cube formed by the 8 grid points at the corner boundary.

On the other hand, the implementation of the equation (2.178) as boundary condition for the elliptic equations brought up some issues that are still under investigation. Despite the convergence of the relaxation method employed and the smoothness of its solution attained, the multigrid algorithm fails to converge. The reason comes from the disalignment of the points where the physical boundary conditions are imposed on the several grids of the multigrid hierarchy. As the grids get coarser and coarser the physical boundary resides more and more inside the domain of integration spoiling the convergence properties of the whole algorithm. It is worth mentioning that this problem did not show up for the scalar field components case since the numerical technique employed there to solve those evolution equations did not require a hierarchy of grids with different spacings between the grid points.

One possible solution for this problem is the adoption of ghost points around the boundary of the domain instead of insisting in applying the physical boundaries in between the last two grid points. This way both the boundary condition equations and the PDEs are supposed to be solved at the boundary. The idea is that the boundary condition equations be used to modify the equations of motion at the boundary points whenever a derivative operator references a ghost point. Despite the implementation of this modification be quite straightforward in 1D and 2D domains, it is not trivial for 3D domains. Due to the geometry that a cubic domain introduces (that would not be present on a square 2D domain, for example, where it is easily tractable) along with the presence of mixed derivatives in the equations of motion, the problem became hard to solve and it is still an open problem under investigation. Besides the pure geometric difficulty, another problem can make the implementation even more difficult. As the boundaries have PDEs to be solved, they require an special treatment regarding the smoothing process as well as the various transfers between grids of different resolutions. The bottom line is that a successful implementation of Sommerfeld-like boundary condition for the elliptic equations requires a careful study of the geometry close to the edges and corners of the cubic domain besides the implementation of a more sophisticated multigrid solver. It is still an on-going project.

[7] See Chap. 3 for a detailed discussion on discretization of PDEs.

**Truncated Boundary Conditions**

Finally the last strategy on implementing the boundary condition consists in applying Dirichlet conditions, i.e. all the variables are set to a constant at the boundaries of the domain. As in the case of Sommerfeld-like boundary conditions, the domain of integration is also truncated to a finite region, $0 \leq t \leq t_{max}$, $x_{min} \leq x \leq x_{max}$, $y_{min} \leq y \leq y_{max}$ and $z_{min} \leq z \leq z_{max}$. Since in this case the behaviour of the several functions as $r \to \infty$ is neglected by simply setting their known values at infinity at a finite value of $r$, essentially it is a cruder approximation than the Sommerfeld-like boundary condition. However, due to the simplicity in implementing this boundary condition, it is a strategy worth pursuing. One hopes that by pushing the boundaries of the truncated domain further away from the region of strong gravitational field where the two boson stars coalesce, the error associated with the boundary approximation be neglegible to the main physical properties of the system. Fortunately it is possible to do so numerically by the use of the technique called *Adaptive Mesh Refinement*. In essence this technique decreases or increases the number of grid points in the whole domain according with the necessity of resolving features of the solution. As the main features of the strong gravitation field are expected to happen in the middle of the domain, the grid are more densely populated there. Naturally this frees computer resources allowing then the use of larger domains of integration.

In short, at the boundaries of the cubic domain the lapse function $\alpha$, the conformal factor $\psi$ and the 3 shift vector components are set equal to their values at infinity. The same is valid for the 2 complex scalar field components $\phi_A$ and their conjugate momenta $\Pi_A$:

$$\alpha = 1, \qquad \psi = 1, \qquad \beta^i = 0, \qquad \phi_A = 0, \qquad \Pi_A = 0. \tag{2.183}$$

### 2.4.6 ADM/York Mass

In the hamiltonian formulation of general relativity the total spacetime energy, also called ADM mass, is associated with the numerical value of the Hamiltonian for the solutions taken on a surface $\partial V$ at spatial infinity $i^0$ enclosing the whole volume $V$ of the slice $\Sigma_t$. For an asymptotically flat spacetime $(\mathcal{M}, g_{ab})$, the total energy associated with the spacelike hypersurface $\Sigma_t$ is just a surface integral of metric component derivatives at $i^0$. In terms of the $3+1$ components and cartesian coordinates, this expression reads as:

$$M_{ADM} \equiv E \equiv \frac{H_\infty}{16\pi} = \frac{1}{16\pi} \lim_{r\to\infty} \oint_{\partial V} \left( \frac{\partial \gamma_{ij}}{\partial x^i} - \frac{\partial \gamma_{ii}}{\partial x^j} \right) N^j dS, \tag{2.184}$$

where $r = \sqrt{x^2+y^2+z^2}$, $H_\infty$ is the hamiltonian numerical value at $i^0$, $N^j$ is a unit vector pointing outwards the surface $\partial V$ and $dS$ is the area element on this surface, i.e. $dS = \sqrt{q}\, d^2y$ with $q$ being the determinant of the induced metric at that surface $\partial V$ and $d^2y$ the differentials corresponding to the coordinates at the surface. For example, in case of $\partial V$ being a spherical symmetric surface at $i^0$, the integral measure simply becomes $dS = \sqrt{q} d^2y = r^2 \sin\theta \, d\theta d\phi$ and $N^i = (1,0,0)$ in this coordinate system.

When implemented numerically, most of the times the equation (2.184) is actually approximated first by integrating along surfaces in which one of the coordinates is kept constant (particularly

true for cartesian coordinates) and second by evaluating the integral at finite $r$. The results are certainly spoiled by the proximity of the boundaries and the boundary conditions implemented there. Of coarse the best alternative for improvement is really pushing the boundaries and the integration surface away from the compact support region, what can be quite restrictive on the computation resources for 3D simulations.

Fortunately there is an alternative expression for the total mass present at the slice $\Sigma_t$ due to O'Murchadha and York [123]. They investigated the relationship between the ADM mass of two spatial metrics conformally related, i.e. for $\gamma_{ij} = \psi^4 \tilde{\gamma}_{ij}$. Starting from the expression 2.184 for the energy, they found out that the difference in energy is given by the following volume integral over the spacelike hypersurface:

$$16\pi \left( M_{ADM} - \tilde{M}_{ADM} \right) = -8 \int_V \sqrt{\tilde{\gamma}} \left( \tilde{\gamma}^{ij} \partial_i \partial_j \psi \right) \, d^3x. \tag{2.185}$$

This expression is particularly convenient when the conformal metric is flat, since the ADM mass vanishes by definition[8]:

$$16\pi M_{ADM} = -8 \int_V \sqrt{\hat{\gamma}} \left( \hat{\gamma}^{ij} \partial_i \partial_j \psi \right) \, d^3x. \tag{2.186}$$

That in cartesian coordinates becomes:

$$M_{ADM} = -\frac{1}{2\pi} \int_V \left( \psi_{,xx} + \psi_{,yy} + \psi_{,zz} \right) \, d^3x. \tag{2.187}$$

The last expression above is then used in this thesis as a diagnostic tool in order to identify regimes in which the total mass/energy has been conserved during the construction of the numerical solutions. It is thought to provide more accurate values than the equation (2.184) since the integral is taken over the whole volume of the slice. Comparisons between the calculations performed by the equation (2.187) on the evolution of one star and by the ADM mass on a spherically symmetric boson star (in a 1D domain) showed good agreement. However, it is worth noticing that no other theoretical investigation of conserved quantities under CFC has been pursued to date. Therefore these empirical evidences that the ADM mass is conserved and still well-defined under CFC should be taken with cautious.

[8] This result was generalized by the authors as a definition of the total energy valid only for asymptotically *conformally* flat spacelike slices.

## 2.5 Overview of the Equations of Motion

Previous sections discussed in detail the derivation of the equations of motion used in this thesis. The goal of this section is summarize those equations, given by (2.154)-(2.164), and discuss briefly some of their properties and how they were solved.

The equations of motion consist of a set of 5 quasi-linear, elliptic equations for the geometric variables lapse function $\alpha$, conformal factor $\psi$ and shift vector $\beta^i$ coupled to each other and to the set of 4 quasi-linear, hyperbolic equations to be solved for the components of the complex scalar field matter, $\phi_1$ and $\phi_2$, and its conjugate momenta, $\Pi_1$ and $\Pi_2$.

The elliptic equations were derived as follows. The maximal slicing condition applied to the trace of the evolution equation for the extrinsic curvature, $K^i_j$, resulted into an elliptic equation for the lapse function $\alpha$, equation (2.162), (after the hamiltonian constraint being used as well to simplify the equations) that can be written schematically as follows:

$$\alpha_{,xx} + \alpha_{,yy} + \alpha_{,zz} = N_\alpha \left(\alpha, \alpha_{,j}, \psi, \psi_{,j}, \beta^i{}_{,j}, \phi_A, \Pi_A\right), \tag{2.188}$$

where $N_\alpha$ is a non-linear function of its arguments. Note that this includes first order derivatives in the lapse function, conformal factor and shift vector.

The elliptic equation for the conformal factor $\psi$, equation (2.163) came exclusively from the hamiltonian constraint equation and it is written in a compact notation as:

$$\psi_{,xx} + \psi_{,yy} + \psi_{,zz} = N_\psi \left(\alpha, \psi, \beta^i{}_{,j}, \phi_A, \phi_{A,j}, \Pi_A\right), \tag{2.189}$$

where $N_\psi$ is also a non-linear function of its arguments but this time the non-linearities originated from first order derivatives are only for the shift vector and scalar field components.

At last, the momentum constraint equations were used to obtain the elliptic equations, equations (2.165)-(2.167), for the components of the shift vector $(\beta^x, \beta^y, \beta^z)$:

$$\frac{4}{3}\beta^x{}_{,xx} + \beta^x{}_{,yy} + \beta^x{}_{,zz} = N_{\beta^x}\left(\alpha, \alpha_{,j}, \psi, \psi_{,j}, \beta^x{}_{,j}, \beta^y{}_{,y}, \beta^y{}_{,x}, \beta^y{}_{,xy}, \beta^z{}_{,z}, \beta^z{}_{,x}, \beta^z{}_{,xz}, \phi_{A,x}, \Pi_A\right), \tag{2.190}$$

$$\beta^y{}_{,xx} + \frac{4}{3}\beta^y{}_{,yy} + \beta^y{}_{,zz} = N_{\beta^y}\left(\alpha, \alpha_{,j}, \psi, \psi_{,j}, \beta^y{}_{,j}, \beta^z{}_{,z}, \beta^z{}_{,y}, \beta^z{}_{,yz}, \beta^x{}_{,x}, \beta^x{}_{,y}, \beta^x{}_{,yx}, \phi_{A,y}, \Pi_A\right), \tag{2.191}$$

$$\beta^z{}_{,xx} + \beta^z{}_{,yy} + \frac{4}{3}\beta^z{}_{,zz} = N_{\beta^z}\left(\alpha, \alpha_{,j}, \psi, \psi_{,j}, \beta^z{}_{,j}, \beta^x{}_{,x}, \beta^x{}_{,z}, \beta^x{}_{,zx}, \beta^y{}_{,y}, \beta^y{}_{,z}, \beta^y{}_{,zy}, \phi_{A,z}, \Pi_A\right), \tag{2.192}$$

where the right-hand-side functions $N_{\beta^x}$, $N_{\beta^y}$ and $N_{\beta^z}$ are highly non-linear functions of its arguments, especially of derivatives of the shift vector components, inclusive with the presence of mixed derivatives.

These five elliptic PDEs have to be suplemented by the boundary conditions. The results of

the simulations shown on Chap. 4 used the following Dirichlet boundary conditions as representing the asymptotic flat spacetime condition:

$$\alpha = 1, \qquad \psi = 1, \qquad \beta^x = 0, \qquad \beta^y = 0, \qquad \beta^z = 0. \tag{2.193}$$

In order to solve this set of elliptic equations, the equations were discretized by finite difference techniques and represented on a 3D discrete domain of grid points. Chap. 3 discusses in details how to do the discretization (Sec. 3.1). The resulting set of discrete equations are then solved by Multigrid techniques (see Sec. 3.3 for details on these techniques).

The four hyperbolic equations consist of the Klein-Gordon equations for a complex scalar field cast into first order in time. They become two equations for the scalar field real and imaginary components, $\phi_A$ for $A = 1, 2$:

$$\dot{\phi}_A = N_{\phi_A}\left(\alpha, \psi, \beta^i, \phi_{A,j}, \Pi_A\right) \tag{2.194}$$

and two equations for its conjugate momenta, $\Pi_A$ for $A = 1, 2$:

$$\dot{\Pi}_A = N_{\Pi_A}\left(\alpha, \alpha_{,j}, \psi, \psi_{,j}, \beta^i, \beta^i{}_{,i}, \phi_A, \phi_{A,j}, \phi_{A,jj}, \Pi_A, \Pi_{A,j}\right), \tag{2.195}$$

where $N_{\phi_A}$ and $N_{\Pi_A}$ are non-linear function of their arguments. Both are suplemented by the following Dirichlet boundary conditions:

$$\phi_A = 0 \qquad \text{and} \qquad \Pi_A = 0. \tag{2.196}$$

The hyperbolic equations were also discretized by finite difference techniques, more specifically, the Crank-Nicholson finite difference scheme (see Sec. 3.1.2 for details). The set of algebraic equations originated from this approximation is then solved by point-wise Newton-Gauss-Seidel iterative technique (see App. D for details). Chap. 3 explains in detail all the numerical techniques necessary to solve these equations. The pseudo code explaining how both set of equations are solved together and in what order is discussed in detail on Sec. 4.1.

The last aspect to be dealt with in the formulation of the initial-boundary value problem concerns the initial data specification. In principle any configuration of the scalar field would settle the issue and in fact such a smooth configuration is used in Chap. 4 to investigate the correctness of the numerical code. However, the interest in this thesis lies on the modelling of the astrophysical scenario of two stars inspiriling towards each other. In order to specify this initial data, each of the stars needs to be modelled individually first. Next section discusses the modelling of boson stars in spherical symmetry and how the two stars are set together as initial data in order to fulfill the astrophysical scenario of interest.

## 2.6 Initial Data

The main goal of this section is to describe how to build the initial data configuration for the boson star binary. The approach taken here consists of building first the spacetime for a spherically symmetric Boson Star. More specifically, static solutions that represent Boson Stars in their ground state. The star solution is then transfered to the 3D spacelike hypersurface $\Sigma_t$. The same process is duplicated for the other star. The stars profiles are then located distant from each other in a way that only a tiny tail of scalar field belonging to the each star is overlapped. It is expected that gluing the two profiles far enough from each other the non-linearities naturally inherent in GR would not play a significant role in approximating the spacetime with two stars as matter source. In addition, the metric components describing the spherically symmetric spacetimes are also patched together into the 3D domain. However they are only used as a first approximation to the binary spacetime. The equations for the geometry are actually resolved using this approximation only as a initial guess.

Boson stars in spherical symmetry has already been studied by a great number of authors. Particularly though, this section follows closely the discussion on Boson Stars presented in Chap. 4 of Kevin Lai's thesis [107] and the numerical relativity lecture notes by Matthew Choptuik [124]. Their main results as well as their assumptions and derivations are only summarized here. The reader interested in further details on the derivations of the equations presented here should refer to one of these references. On the other hand, the end of this section discusses in more detail the derivation of the coordinate transformation law between the rest frame of the star and the lab frame, where the star is boosted. Thus expressions for the boosted metric components and scalar fields are found. The interpolation from the initial data calculated in spherical symmetry (1D) into the 3D spatial domain is also discussed.

### 2.6.1 Spherically Symmetric Space-time

Birkhoff's theorem ascertains that the only static spherically symmetric solution in vacuum, i.e. when $R_{ab} = 0$, is the Schwarzschild solution. As the spacetime solution being sought here is in presence of matter, dynamics may arise and the spacetime wouldn't be static anymore. This subsection starts then with the most general spherically symmetric solution and specializes later to the static case. Spherical symmetry is characterized by a group of symmetry, $SO(3)$. The action of this isometry into a collection of space points results into a 2-sphere. As $SO(3)$ can be physically associated with rotations, the metric components for a spherically symmetric spacetime remain invariant under rotations. Adopting the usual spherical coordinates $(r, \theta, \phi)$, the metric induced on the 2-sphere is going to be a multiple of the unit 2-sphere line element:

$$d\Omega^2 = d\theta^2 + \sin^2\theta d\phi^2 \tag{2.197}$$

and results into the following general spatial line element that it is spherically symmetric:

$$^{(3)}ds^2 = a^2(t, r)dr^2 + r^2b^2(t, r)d\Omega^2. \tag{2.198}$$

Note that the line element above is just a generalization of the flat line element in spherical coordinates. The generalization just assumes that the otherwise constant coefficients now become functions of the coordinates $(t, r)$ in order to allow for curvature due to the presence of the star [110]. The same argument can be applied to the construction of the 4-dimensional $3 + 1$ line element. However in this case one extra assumption must be taken. The shift vector must have only the radial component, otherwise the spherical symmetry would be destroyed, i.e. if the shift has only radial component, the 2-sphere would go to another 2-sphere from one slicing $\Sigma_t$ to the next one $\Sigma_{t+dt}$, while the presence of $\beta^\theta$ or $\beta^\phi$ prevents that to happen. So:

$$\beta^i = (\beta^r(t, r), 0, 0) \equiv (\beta, 0, 0). \tag{2.199}$$

Finally then the most general $4d$ line element for a spherically symmetric spacetime in the $3 + 1$ coordinate basis can be written as:

$$ds^2 = (-\alpha^2 + a^2\beta^2)dt^2 + 2a^2\beta\, dtdr + a^2dr^2 + r^2b^2d\Omega^2. \tag{2.200}$$

It is not hard to prove from the equation (2.40) that the extrinsic curvature corresponding to the spatial metric $\gamma_{ij}$ has only two independent components:

$$K^i_{\ j} = \mathrm{diag}(K^r_{\ r}(t, r), K^\theta_{\ \theta}(t, r), K^\theta_{\ \theta}(t, r)). \tag{2.201}$$

Before writing down the Einstein's equation in the $3+1$ form for this spacetime, it is convenient to introduce the following auxilliary fields to describe the massive complex scalar fields:

$$\Phi(t, r) \equiv \phi'(t, r) \equiv \partial_r\phi(t, r) \qquad \text{and} \qquad \Pi(t, r) \equiv \frac{a}{\alpha}(\dot{\phi} - \beta\phi'). \tag{2.202}$$

Note that besides its resemblance $\Pi$ is not the conjugate momentum of the field $\phi$. A factor of $\sqrt{\gamma}/a$ needs to be multiplied in order to make the expression the conjugate momentum. In terms of these auxiliary variables, and according to the equations (2.36)-(2.39), the non-vanishing components of the matter source stress-energy tensor becomes:

$$\rho = \frac{|\Phi|^2 + |\Pi|^2}{2a^2} + \frac{U(|\phi|^2)}{2}, \qquad j_r = -\frac{\Pi^*\Phi + \Pi\Phi^*}{2a} = a^2 j^r,$$

$$S^r_{\ r} = \rho - U(|\phi|^2), \qquad S^\theta_{\ \theta} = \frac{|\Pi|^2 - |\Phi|^2}{2a^2} - \frac{U(|\phi|^2)}{2},$$

$$S^\phi_{\ \phi} = S^\theta_{\ \theta}, \qquad S = \frac{3|\Pi|^2 - |\Phi|^2}{2a^2} - \frac{3}{2}U(|\phi|^2).$$

While the evolution equations for the complex scalar field and its auxiliary fields with a general self-interaction potential $U(|\phi|^2)$ are written as:

$$\dot{\phi} = \frac{\alpha}{a}\Pi + \beta\Phi, \tag{2.203}$$

$$\dot{\Phi} = \left(\frac{\alpha}{a}\Pi + \beta\Phi\right)', \tag{2.204}$$

$$\dot{\Pi} = \frac{1}{(rb)^2}\left[(rb)^2\left(\beta\Pi + \frac{\alpha}{a}\Phi\right)\right]' + 2\left[\alpha K^{\theta}_{\ \theta} - \beta\frac{(rb)'}{rb}\right]\Pi - \alpha a\frac{dU(|\phi|^2)}{d|\phi|^2}\phi. \tag{2.205}$$

On the other hand, according to equation (2.40) the evolution equations for the spatial metric $(\gamma_{ij})$ functions in spherical symmetry becomes:

$$\dot{a} = -\alpha a K^{r}_{\ r} + (a\beta)', \tag{2.206}$$

$$\dot{b} = -\alpha b K^{\theta}_{\ \theta} + \frac{\beta}{r}(rb)'. \tag{2.207}$$

While, according to equation (2.41), the evolution equations for the extrinsic curvature $K^{i}_{\ j}$ can be written as:

$$\dot{K}^{r}_{\ r} = \beta K^{r}_{\ r}{}' - \frac{1}{a}\left(\frac{\alpha'}{a}\right)' + \alpha\left\{-\frac{2}{arb}\left[\frac{(rb)'}{a}\right]' + KK^{r}_{\ r} - 4\pi\left[\frac{2|\Phi|^2}{a^2} + U(|\phi|^2)\right]\right\}, \tag{2.208}$$

$$\dot{K}^{\theta}_{\ \theta} = \beta K^{\theta}_{\ \theta}{}' + \frac{\alpha}{(rb)^2} - \frac{1}{a(rb)^2}\left[\frac{\alpha rb}{a}(rb)'\right]' + \alpha K K^{\theta}_{\ \theta} - 4\pi\alpha U(|\phi|^2). \tag{2.209}$$

At last, the hamiltonian constraint equation, equation (2.42), before and after the substitution of the expressions for the Ricci scalar in terms of the metric functions and the energy density in terms of the scalar fields:

$$\mathcal{R} + 4K^{r}_{\ r}K^{\theta}_{\ \theta} + 2K^{\theta\ 2}_{\ \theta} = 16\pi\rho, \tag{2.210}$$

$$-\frac{2}{arb}\left\{\left[\frac{(rb)'}{a}\right]' + \frac{1}{rb}\left[\left(\frac{rb}{a}(rb)'\right)' - a\right]\right\} + 4K^{r}_{\ r}K^{\theta}_{\ \theta} + 2K^{\theta\ 2}_{\ \theta} = 8\pi\left[\frac{|\Phi|^2 + |\Pi|^2}{a^2} + U(|\phi|^2)\right], \tag{2.211}$$

and equation (2.43), the momentum constraint:

$$K^{\theta}_{\ \theta}{}' + \frac{(rb)'}{rb}\left(K^{\theta}_{\ \theta} - K^{r}_{\ r}\right) = 2\pi\frac{\Pi^*\Phi + \Pi\Phi^*}{a}. \tag{2.212}$$

These set of equations are the $3+1$ equations for a time dependent spherically symmetric spacetime whose matter source is a complex scalar field. They are still in a quite general form. To cast them in a form appropriate for numerical calculation, the first step is to fix the coordinate system to be used. In the $3+1$ formalism that means to specify the values of the lapse, $\alpha$, and the shift vector component, $\beta$. There are different possibilities of doing so. Two of them though are quite suitable for the task of generating astrophysically relevant initial data. Next subsection specializes on maximal-isotropic coordinates, a coordinate system similar to the one adopted in

the 3D evolution and therefore appropriate to specify its initial data. The subsection afterwards discusses the other coordinate system, the polar-areal coordinates. The calculation of the initial data is performed more easily in this system and for this reason this is the coordinate system of choice to generate them.

### 2.6.2 Maximal-Isotropic Coordinates

In this coordinate system the slicing condition is chosen such that its mean extrinsic curvature always vanishes, hence the slicing is *maximal* at any time $t$ and position $r$ on $\Sigma_t$:

$$K \equiv K^i{}_i = 0 \qquad \text{and} \qquad \dot{K}(t,r) = 0. \tag{2.213}$$

Note that as a consequence of this slicing choice one of the extrinsic curvature components can be eliminated from the equations. Maximal slicing condition then implies:

$$K = 0 \qquad \Rightarrow \qquad K^\theta{}_\theta = -\frac{1}{2} K^r{}_r. \tag{2.214}$$

Once the slicing is imposed, only one coordinate freedom is left to be fixed. The choice of spatial coordinate is achieved by the so called *isotropic* condition which demands:

$$a = b \equiv \psi^2(t,r) \tag{2.215}$$

such that the spherically symmetric spatial metric is manifestly conformally flat (or isotropic):

$${}^{(3)}ds^2 = \psi^4 \left( dr^2 + r^2 d\Omega^2 \right). \tag{2.216}$$

The implementation of the isotropic condition requires that the initial data obeys the condition $a(0,r) = b(0,r)$ and that this condition holds for all times, i.e:

$$\dot{a}(t,r) = \dot{b}(t,r) \tag{2.217}$$

should be satisfied for all $t$ and $r$. According to equations (2.206) and (2.207), the isotropic condition results then in an ordinary differential equation (ODE) for the shift component $\beta$:

$$r \left( \frac{\beta}{r} \right)' = \alpha (K^r{}_r - K^\theta{}_\theta) \tag{2.218}$$

or, using the maximal slicing condition:

$$r \left( \frac{\beta}{r} \right)' = \frac{3}{2} \alpha K^r{}_r. \tag{2.219}$$

The maximal slicing and isotropic condition fix the coordinates through the specification of the lapse function, $\alpha$, and the shift vector component, $\beta$. The maximal slicing condition results in an ODE for the lapse function. That comes from the tracing of the evolution equation for the extrinsic curvature $K^i{}_j$, equation (2.41):

$$\dot{K} = \beta K' - \frac{1}{a(rb)^2}\left[\frac{(rb)^2}{a}\alpha'\right]' + \alpha\left[\mathcal{R} + K^2 + 4\pi\left(S - 3\rho\right)\right]. \tag{2.220}$$

While adopting $K = 0$ and substituting the values for $R$, $S$ and $\rho$ in terms of the metric and scalar field components, the equation above becomes a second order ODE for the lapse function $\alpha$:

$$\alpha'' + \frac{a}{(rb)^2}\left[\frac{(rb)^2}{a}\right]'\alpha' + \left[4\pi a^2 U(|\phi|^2) - 8\pi|\Pi|^2 - \frac{3}{2}a^2 {K^r_r}^2\right]\alpha = 0 \tag{2.221}$$

that turns into the following equation after the isotropic condition is imposed:

$$\alpha'' + \frac{2}{r\psi^2}\frac{\partial}{\partial r^2}\left(r^2\psi^2\right)\alpha' + \left[4\pi\psi^4 U(|\phi|^2) - 8\pi|\Pi|^2 - \frac{3}{2}\psi^4 {K^r_r}^2\right]\alpha = 0, \tag{2.222}$$

where the operator $\partial/\partial r^2$ is the derivative with respect to $r^2$. The reason for this change of operators from $\partial/\partial r$ to $\partial/\partial r^2$ is related to the regularity behaviour of these terms close to the origin. On the continuum both operators are identical, however their second order discrete approximations are distinct and only the discrete version of $\partial/\partial r^2$ preserves its continuum regularity behaviour. Sec. 2.6.8 will be devoted to the discussion of these kind of operators.

The other two geometric variables to be determined, the conformal factor $\psi$ and the extrinsic curvature component $K^r_r$, can be evaluated either by the hamiltonian and momentum constraints or by their respective evolution equations. Here they are updated through the constraints, a scheme also known as a *fully* constrained scheme. The hamiltonian and momentum constraints, equations (2.210) and (2.210), are then written as:

$$\frac{3}{\psi^5}\frac{d}{dr^3}\left(r^2\frac{d\psi}{dr}\right) + \frac{3}{16}{K^r_r}^2 = -\pi\left(\frac{|\Phi|^2 + |\Pi|^2}{\psi^4} + U(|\phi|^2)\right), \tag{2.223}$$

$${K^r_r}' + 3\frac{(r\psi^2)'}{r\psi^2}K^r_r = -\frac{4\pi}{\psi^2}\left(\Pi^*\Phi + \Pi\Phi^*\right). \tag{2.224}$$

On the other hand, the klein-gordon equation in the maximal-isotropic coordinate system becomes:

$$\dot{\phi} = \frac{\alpha}{\psi^2}\Pi + \beta\Phi, \tag{2.225}$$

$$\dot{\Phi} = \left(\frac{\alpha}{\psi^2}\Pi + \beta\Phi\right)', \tag{2.226}$$

$$\dot{\Pi} = \frac{3}{\psi^4}\frac{d}{dr^3}\left[r^2\psi^4\left(\beta\Pi + \frac{\alpha}{\psi^2}\Phi\right)\right] - 2\left[\frac{1}{2}\alpha K^r_r + \beta\frac{(r\psi^2)'}{r\psi^2}\right]\Pi - \alpha\psi^2\frac{dU(|\phi|^2)}{d|\phi|^2}\phi. \tag{2.227}$$

The set of equations (2.219), (2.222), (2.223), (2.224), (2.225), (2.226) and (2.227) fully determine the spherically symmetric Einstein-Klein-Gordon system in maximal-isotropic coordinates. Note however that those equations must be supplemented by boundary conditions in order to provide a unique solution. Despite these equations were implemented here as an exercise it is not

the focus of this dissertation. Therefore for further information on the implementation details, the reader should refer to appendix B of Lai's thesis [107]. The focus of this section is on the initial data generation. After introducing the polar-areal coordinates in the next subsection, the set of equations in both coordinates will be then especialized to initial data calculation.

### 2.6.3 Polar-Areal Coordinates

This coordinate system imposes the slicing condition such that the mean extrinsic curvature reduces to the following component of the extrinsic curvature:

$$K \equiv K^i_{\ i} = K^r_{\ r}. \tag{2.228}$$

This slicing condition is called polar slicing [125]. The immediate consequence from this slicing choice is the reduction of the number of components of the spherically symmetric extrinsic curvature:

$$K \equiv K^r_{\ r} + 2K^\theta_{\ \theta} = K^r_{\ r} \qquad \Rightarrow \qquad K^\theta_{\ \theta} = 0, \tag{2.229}$$

where the condition can be kept for all $t$ and $r$ by imposing:

$$K^\theta_{\ \theta}(t,r) = \dot{K}^\theta_{\ \theta}(t,r) = 0. \tag{2.230}$$

The spatial coordinate is fixed by demanding that the coordinate $r$ be associated with the proper area. A constant $r$ angular surface element $d\theta\, d\varphi$ has an area associated to it that by definition is $r^2 \sin\theta\, d\theta\, d\varphi$. The coordinate $r$ is endowed then of geometric meaning. This condition requires then that the spherical symmetric component $b(t,r)$ be constant for all $t$ and $r$ and equal to 1. Applying the polar-areal condition to the evolution equation for $b(t,r)$, equation (2.207), results in a identically vanishing shift component $\beta$. The $3+1$ line element in polar-areal coordinates reduces then to:

$$ds^2 = -\alpha^2 dt^2 + a^2 dr^2 + r^2 d\Omega^2. \tag{2.231}$$

One of the greatest advantages of the polar-areal coordinates is the considerable simplification of the equations of motion. Again, adopting a fully constrained scheme, the only non-vanishing component of the extrinsic curvature can be determined from the momentum constraint resulting in the following algebraic expression:

$$K^r_{\ r} = -\frac{2\pi r}{a}\left(\Pi^*\Phi + \Pi\Phi^*\right), \tag{2.232}$$

while the hamiltonian constraint can be used to determine the metric component function $a(t,r)$:

$$a' = \frac{1}{2}\left\{\frac{a}{r}\left(1-a^2\right) + 4\pi r a\left[|\Phi|^2 + |\Pi|^2 + a^2 U(|\phi|^2)\right]\right\}. \tag{2.233}$$

The equation for the lapse function $\alpha(t,r)$ originates from inserting the hamiltonian constraint into the evolution equation for $K^\theta_{\ \theta}$:

$$\alpha' = \frac{\alpha}{2}\left\{\frac{a^2-1}{r} + 4\pi r\left[|\Phi|^2 + |\Pi|^2 - a^2 U(|\phi|^2)\right]\right\}. \tag{2.234}$$

Finally the equations for the complex scalar field and its auxiliary variable turn out to be:

$$\dot{\phi} = \frac{\alpha}{a}\Pi, \tag{2.235}$$

$$\dot{\Phi} = \left(\frac{\alpha}{a}\Pi\right)', \tag{2.236}$$

$$\dot{\Pi} = 3\frac{d}{dr^3}\left[r^2\frac{\alpha}{a}\Phi\right] - \alpha a\frac{dU(|\phi|^2)}{d|\phi|^2}\phi. \tag{2.237}$$

Equations (2.232), (2.233), (2.234), (2.235), (2.236) and (2.237) correspond to the fully constrained set of equations in polar-areal coordinates used to solve the EKG system in spherical coordinate. Next subsection these equations are simplified even more by the *static* spacetime requirement and used to determine the boson star initial data.

### 2.6.4 Static Solution Ansatz

A spherically symmetric, localized, time independent matter model configuration captures the simplest notion of a star. Ideally this configuration also generates a spherically symmetric, time independent gravitational field that is regular everywhere. Unfortunately, it is not possible to have a time independent solution in case of the complex scalar field. Friedberg, Lee and Pang [126] showed that in order to have the lowest energy solution, i.e. a boson star in its ground state, the complex scalar field needs to have a time harmonic dependence. Therefore the following *ansatz* is adopted for the complex scalar field modelling the boson star:

$$\phi(t, r) = \phi_0(r)e^{-i\omega t}. \tag{2.238}$$

Note that with this time dependence, the complex scalar field stress-energy-momentum tensor becomes time independent. Consequently the spacetime sourced by this matter model is also expected to be time independent.

An extra condition on the coordinates needs to be imposed in order to guarantee staticity. A stationary spacetime is defined by the existence of a timelike killing vector field. A spacetime is said to be static when it is a stationary spacetime with time reversal symmetry, i.e. the metric is invariant under the transformation $t \to -t$. This extra condition on a stationary spacetime is achieved by demanding that the timelike killing vector field be orthogonal to the spacelike hypersurfaces $\Sigma_t$. In practice, the implementation of this condition is done via the coordinate requirement that the shift component vanishes for all times $t$, where $t$ is the orbit of the timelike killing vector field:

$$\beta(t, r) = 0. \tag{2.239}$$

Note that this condition is already satisfied in the polar-areal coordinate system. On the other

hand, it does needs to be imposed explicitly on the maximal-isotropic coordinates.

By the definition of the timelike killing vector field, all the time derivative of the metric components vanish ($\dot{a} = \dot{b} = 0$). Thus the static ansatz implies that the extrinsic curvature components also vanish:

$$K^r_{\ r} = K^\theta_{\ \theta} = 0, \tag{2.240}$$

as can be seen from the evolution equations (2.206) and (2.207) for the metric component functions $a$ and $b$. As a consequence the polar slicing is also maximal for the static solution.

Note that the static ansatz implies the following simplifications on the complex scalar fields:

$$\dot{\phi}(t,r) = -i\omega\phi_0(r)e^{-i\omega t}, \tag{2.241}$$

$$\phi'(t,r) = \phi_0'(r)e^{-i\omega t}, \tag{2.242}$$

which produce:

$$\Pi(t,r) = -i\omega\frac{a}{\alpha}\phi_0(r)e^{-i\omega t} \equiv \Pi_0(r)e^{-i\omega t}, \tag{2.243}$$

$$\Phi(t,r) = \phi_0'(r)e^{-i\omega t} \equiv \Phi_0(r)e^{-i\omega t}, \tag{2.244}$$

and

$$\Pi^*\Phi + \Pi\Phi^* = 0, \tag{2.245}$$

which indicates that the momentum constraint, equation (2.212), is satisfied identically.

Inserting the static ansatz solution into the set of equations describing the spherically symmetric EKG system in polar-areal coordinates simplifies considerably the equations of motion. After some manipulation, the hamiltonian constraint (2.233), the slicing condition, (2.234), as well as the scalar field equation (2.237) become the following system of coupled ODEs:

$$a' = \frac{1}{2}\left\{\frac{a}{r}\left(1-a^2\right) + 4\pi r a\left[\Phi_0^2 + \omega^2\frac{a^2}{\alpha^2}\phi_0^2 + a^2U(\phi_0^2)\right]\right\}, \tag{2.246}$$

$$\alpha' = \frac{\alpha}{2}\left\{\frac{a^2-1}{r} + 4\pi r\left[\Phi_0^2 + \omega^2\frac{a^2}{\alpha^2}\phi_0^2 - a^2U(\phi_0^2)\right]\right\}, \tag{2.247}$$

$$\phi_0' = \Phi_0, \tag{2.248}$$

$$\Phi_0' = -\left(1 + a^2 - 4\pi r^2a^2U(\phi_0^2)\right)\frac{\Phi_0}{r} + a^2\left[\frac{dU(\phi_0^2)}{d\phi_0^2} - \frac{\omega^2}{\alpha^2}\right]\phi_0. \tag{2.249}$$

This system of ODEs will be referred to from now on as the polar-areal ODE system.

Note that this system of ODEs is singular at $r = 0$. In order to make it regular, certain conditions needs to be imposed on the functions $a(r)$ and $\Phi_0(r)$. Rigorously a function (or tensor) is defined to be regular if its cartesian components can be expanded into a convergent Taylor series

in a neighbourhood of $r = 0$ or equivalently $x = 0$, $y = 0$ and $z = 0$. It is possible to show [125, 127, 128] that the resulting expansion is in even powers of $x$, $y$ and $z$. Adopt here a simpler analysis though. Suppose there is a convergent Taylor expansion of the metric component $a(r)$ around $r = 0$ written as follows:

$$a(r) = a_0 + ra_1 + r^2 a_2 + \dots \tag{2.250}$$

Insert this expansion into the singular term of equation (2.247) and require that the result be at least $O(r)$ in order to make that term regular:

$$a^2 - 1 \equiv \left(a_0^2 - 1\right) + 2a_0 a_1 r + 2a_0 a_2 r^2 + \dots = O(r), \tag{2.251}$$

which is satisfied when $a_0^2 - 1 = 0$ or $a_0 = \pm 1$ (the choice of sign is arbitrary; here $a_0 = 1$). Therefore the first regularity condition is:

$$a(0) = 1. \tag{2.252}$$

The same analysis goes to make the singular term at equation (2.249) regular at $r = 0$ and the second regularity condition becomes:

$$\Phi_0(0) = 0. \tag{2.253}$$

The polar-areal ODE system need 4 conditions to fix all its integration constants. Two of them were already given by the regularity conditions (2.252) and (2.253). The requirement that the spacetime be asymptotically flat implies that the scalar field should vanish at infinity. This results then into the first outer boundary condition:

$$\lim_{r\to\infty} \phi(r) = 0. \tag{2.254}$$

The last boundary condition is a condition on the lapse function. It is obtained by the comparison of the Schwarzschild and polar-areal line element in the region where they coincide. The Schwarzschild spacetime metric components can be read from:

$$ds^2 = -\left(1 - \frac{2M}{r}\right) dt^2 + \left(1 - \frac{2M}{r}\right)^{-1} dr^2 + r^2 d\Omega^2 \tag{2.255}$$

and becomes the limiting case of the polar-areal metric components:

$$ds^2 = -\alpha^2 dt^2 + a^2 dr^2 + r^2 d\Omega^2, \tag{2.256}$$

when $r \to \infty$. Therefore the following limits are valid:

$$\lim_{r\to\infty} a(t,r)^2 = \left(1-\frac{2M}{r}\right)^{-1}, \tag{2.257}$$

$$\lim_{r\to\infty} \alpha(t,r)^2 = \left(1-\frac{2M}{r}\right) \tag{2.258}$$

and provide the lapse function outer boundary condition in terms of the metric component $a$:

$$\lim_{r\to\infty} \alpha(t,r) = \frac{1}{a(t,r)}. \tag{2.259}$$

The limit (2.257) also suggests the definition of a useful diagnostic function, the *mass aspect function*:

$$M(t,r) \equiv \frac{r}{2}\left(1-\frac{1}{a(t,r)^2}\right), \tag{2.260}$$

such that for $r \to \infty$ it recovers the Schwarzschild mass, also identified as the spacetime ADM mass:

$$\lim_{r\to\infty} M(t,r) = M \equiv M_{ADM}. \tag{2.261}$$

The polar-areal ODE system supplemented by the regularity conditions, equations (2.252) and (2.253), and the outer boundary conditions, equations (2.254) and (2.259), consists of an eigenvalue problem with eigenvalue $\omega$ parametrized by the central scalar field value $\phi_0 \equiv \phi_0(0)$, viz $\omega = \omega(\phi_0)$. This means that for any particular value of $\phi_0$ there will be a unique solution satisfying the polar-areal ODE system supplemented with the regularity and boundary conditions *only* for some specific values of $\omega$. For each $\phi_0$, it is expected then a discrete spectrum of values for $\omega$ , each associated with a solution profile with an increasingly number of nodes. Clearly only the solution with no nodes, the so called ground state of this system, is of interest to model a star-like solution.

Note that the implementation of the boundary condition (2.259) can be simplified by taking into account the homogeneity and linearity of the slicing condition equation (2.247). Since the lapse function can be arbitrarily rescaled without changing the equation, its central value is chosen as opposed to the boundary condition (2.259) and it is set to:

$$\alpha(0) = 1. \tag{2.262}$$

After the integration of the ODE system, then both the lapse function $\alpha$ and the eigenvalue $\omega$ needs to be rescaled in order to satisfy the outer boundary condition for $\alpha$:

$$\alpha(r) \to c\,\alpha(r), \tag{2.263}$$

$$\omega \to c\,\omega, \tag{2.264}$$

where $c$ is the rescaling constant calculated at the coordinate boundary of the numerical domain,

$r_{max}$ by:

$$c = \frac{1}{a(r_{max})\alpha(r_{max})}. \tag{2.265}$$

The set of conditions $a(0) = 1$, $\alpha(0) = 1$ and $\Phi(0) = 0$ is used then to integrate the polar-areal ODE system for a given a value for the central scalar field, $\phi_0$. The method for finding the eigenvalue $\omega$ consists of the *shooting* method. A bracket $[\omega_{hi}, \omega_{low}]$ around the possible values for $\omega$ needs to be found before applying the shooting method. $\omega_{low}$ is value of $\omega$ that results into a solution so that $\lim_{r\to\infty}\phi(r) = -\infty$. While the $\omega_{hi}$ value is such that $\lim_{r\to\infty}\phi(r) = \infty$. The shooting method adjusts the value of $\omega$ between these two values of the bracket such that the asymptotic behaviour for the scalar field is satisfied: $\lim_{r\to\infty}\phi(r) = 0$. This method was encoded in a FORTRAN subroutine called *bsidpa* that takes care of the integration of the polar areal ODE system, the shooting process per se and the functions rescaling. Appendix A describes in detail its user interface.

Once the polareal solution is computed, a coordinate transformation can be performed from the areal to isotropic coordinates. Remember that for the static solution the polar slicing is also maximal. The coordinate transformation involves the solution of the following ODE:

$$\begin{aligned} r|_{R=R_{max}} &= \left[\left(\frac{1+\sqrt{a}}{2}\right)^2 \frac{R}{a}\right]_{R=R_{max}}, \\ \frac{dr}{dR} &= a\frac{r}{R}, \end{aligned} \tag{2.266}$$

where $R$ is the areal radial coordinate and $r(R)$ is the isotropic radial coordinate. It is possible to show that the lapse is the same on both coordinates[9] and that the conformal factor can be obtained from the following expression:

$$\psi(r) = \sqrt{\frac{R}{r(R)}} \qquad \text{with} \qquad \psi(0) = \sqrt{\frac{1}{r'(0)}}, \tag{2.267}$$

where $'$ represents the derivative with respect to $R$.

### 2.6.5 Family of Spherically Symmetric Solutions - Properties

Figure 2.1 shows typical solutions of the polar-areal ODE system. They represent a static spherically symmetric boson star and are parametrized by the central scalar field value $\phi_0(0)$. Note that the star becomes more compact, meaning that its radius turns out to be smaller, as the central scalar field value increases. The star lapse function and the conformal factor, on the other hand, have their central value decreased and increased respectively as the star becomes more compact. Note that both approach unity at large $r$ as expected from the asymptotically flat condition. The mass aspect function of each configuration is also shown and it converges to a constant value, defined as the ADM mass of the spacetime, as $r \to \infty$. All star profiles shown on figure 2.1 are

[9] See appendix D of Lai's thesis for further details [107].

members of one-parameter family of solutions. More specifically they are spherically symmetric solutions stable against radial perturbations.

Figure 2.2 shows the ADM mass $M_{ADM}$ as a function of the central scalar field value $\phi_0(0)$ on the left panel and the ADM mass as a function of the star radius on the right panel. The plot of the ADM mass against $\phi_0(0)$ has an absolute maximum at $\phi_0(0) \approx 0.08$ of $M_{max} \approx 0.633 M_{pl}^2/m$ (where $M_{pl}$ is the planck mass and $m$ is the mass of the boson constituent particle). This result is analogous to the Chandrasekhar limit for white dwarfs and neutron stars in which above some mass limit there is no static configuration and the star is prompt to collapse into a black-hole. Note however that there is no degeneracy pressure in the case of boson stars. The supporting pressure against the gravity pressure can be thought as coming from the uncertainty principle if the system is studied semiclassically or from the wave dispersive nature of the Klein-Gordon field if the system is studied classically as it is the case in this thesis.

It is possible to show from perturbation analysis or full evolution of the equations of motion [129, 130, 131, 107] that the stars belonging to the branch $\phi_0(0) \lesssim 0.08$ (left panel of figure 2.2) are stable against radial perturbation. This branch is called then the *stable* branch. On the other hand, stars with $\phi_0(0) \gtrsim 0.08$ are unstable against radial perturbations and therefore belong to the *unstable* branch of the family of solutions. On every extremum of the plot an additional mode of the perturbation becomes unstable. All stars evolved as initial data in this thesis belong to the stable branch.

The solid triangles on both plots represent samples of stable stars belonging to the set $\phi_0(0) \in \{0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07\}$. As $\phi_0(0)$, belonging to the stable branch, increases the star ADM mass also increases as can be seen from left panel of figure 2.2. On the right panel, however, an increase of the ADM mass is associated with a decrease of the star radius for the same set of $\phi_0(0)$ values. This plot shows the dependency of the ADM mass to the radius of the star. The radius of the star is defined either as the radius containing 99% of the star ADM mass, $R_{99}$, or containing 95% of the star ADM mass, $R_{95}$. Both definitions are common in the literature and they are kept here for the sake of comparison.

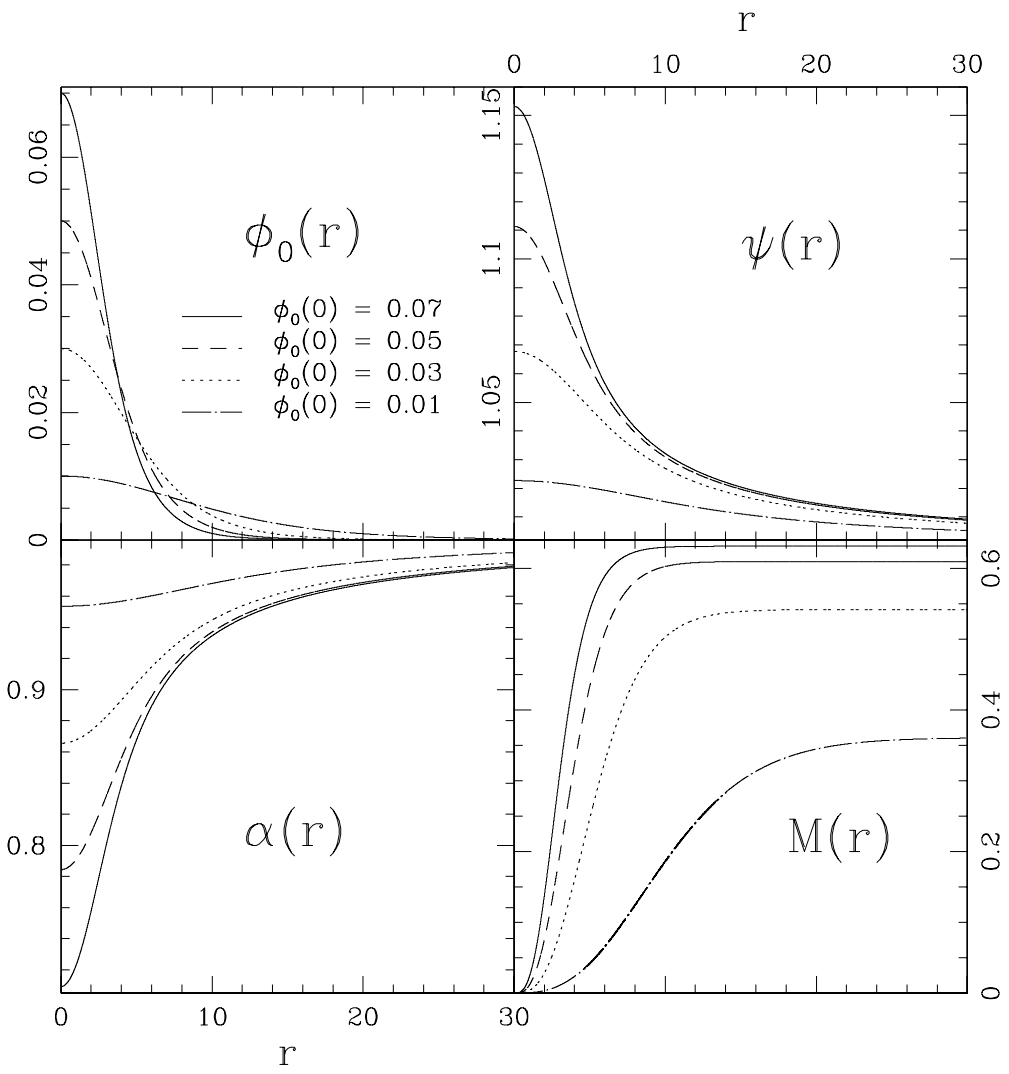


**Figure 2.1:** Static spherically symmetric solutions. These plots shows the values of $\phi_0(r), \psi(r), \alpha(r)$ and $M(r)$ for boson stars with the modulus of the central scalar field value ranging as $\phi_0(0) = 0.01$, 0.03, 0.05 and 0.07 in polar-areal coordinates. These configurations correspond to stable boson stars. They become more compact and increase their average density as the central value of the field is raised.

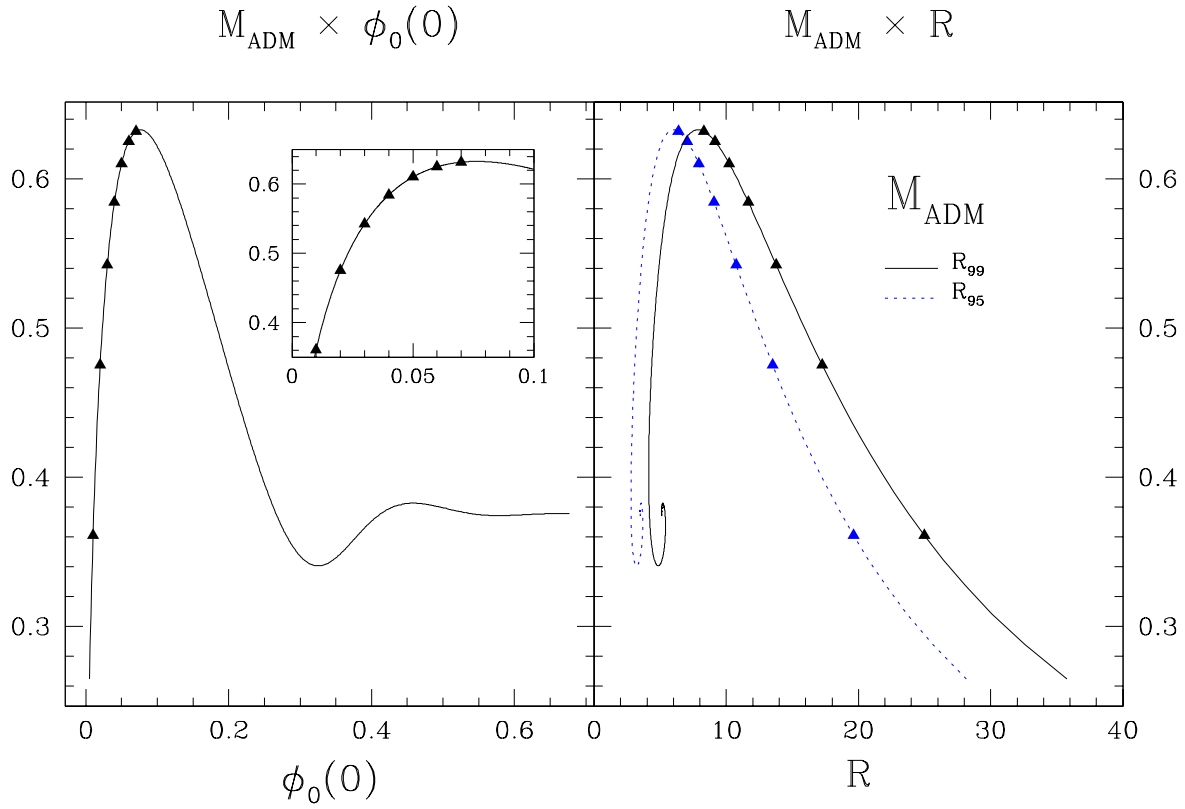


**Figure 2.2:** ADM mass $M_{ADM}$ as a function of central scalar field value $\phi_0(0)$ and the star radius $R$. The plot on the left panel shows the ADM mass $M_{ADM}$ as a function of the central scalar field value $\phi_0(0)$. There is a global maximum at $\phi_0(0) \approx 0.08$ of $M_{max} \approx 0.633 M_{pl}^2/m$ above which no static solution exists. The solid triangles represent members of the stable branch of the one-parameter family of solutions, $\phi_0(0) \in \{0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07\}$. The inside panel zooms in the stable branch range in order to show the details and help to visualize the difference in mass for the several $\phi_0(0)$. The plot on the right panel shows the ADM mass as a function of the star radius. The solid line plots in terms of $R_{99}$, the star radius contaning 99% of the star ADM mass, $M(R) = 99\% M(\infty) = 99\% M_{ADM}$, and the dashed line in terms of $R_{95}$, the star radius contaning 95% of the star mass. The solid triangles represent the same set of stable solution members introduced on the plot of the left panel.

### 2.6.6 1d to 3d Initial Data Interpolation

The complete set of one dimensional functions in polar-areal coordinates to be interpolated into the 3D maximal-isotropic numerical domain consists of $\{\alpha(R), \psi(R), \phi_0(R)\}$, where $R$ is the areal radius. The conformal factor $\psi(R)$ was obtained by the transformation (2.267). The transformation from 1D areal to 1D isotropic coordinates is trivially obtained[10] by simply considering $\alpha(r) \equiv \alpha(R)$, $\psi(r) \equiv \psi(R)$ and $\phi_0(r) \equiv \phi_0(R)$, where $r(R)$ is the isotropic radius obtained by the integration of equation (2.266). With these remarks in mind, the set of functions to be interpolated into the 3D isotropic domain becomes then $\{\alpha(r), \psi(r), \phi_0(r)\}$. The interpolation is performed by first mapping the 3D coordinate values into the 1D isotropic coordinates. This is achieved by use of the coordinate transformation law from cartesian to polar-spherical coordinates. Each point of the 3D domain in which the 3D functions are evaluated are then mapped to their equivalent radial positions $r_{3d}$ in the 1D isotropic coordinates by:

$$r_{3d} = \sqrt{(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2} \tag{2.268}$$

for a star centered at the $(x_0, y_0, z_0)$ position in the 3D domain. Then a quadratic lagrange polynomial interpolation is applied to the set $\{\alpha(r), \psi(r), \phi_0(r)\}$ in order to obtain the function values at $r_{3d}$, viz $\{\alpha(r_{3d}), \psi(r_{3d}), \phi_0(r_{3d})\}$. Once the lagrange interpolation is performed, it is just a matter of copying the values of the 1D functions back to the ones on the 3D domain:

$$\alpha(x, y, z) \equiv \alpha(r_{3d}), \qquad \psi(x, y, z) \equiv \psi(r_{3d}), \qquad \phi_0(x, y, z) \equiv \phi_0(r_{3d}) \tag{2.269}$$

and the interpolation procedure from 1D to 3D is complete.

If a second star is supposed to be interpolated in the 3D domain as well, then the procedure explained above is repeated and the solutions corresponding to each star are superposed as:

$$\alpha(x, y, z) = \alpha^{(1)}(x, y, z) + \alpha^{(2)}(x, y, z) - 1, \tag{2.270}$$

$$\psi(x, y, z) = \psi^{(1)}(x, y, z) + \psi^{(2)}(x, y, z) - 1, \tag{2.271}$$

$$\phi_0(x, y, z) = \phi_0^{(1)}(x, y, z) + \phi_0^{(2)}(x, y, z), \tag{2.272}$$

where the superscripts (1) and (2) refer to the interpolated solution of each star. Note that the superposition of the star solutions are such that the overlap of scalar matter is negligible, i.e. only the scalar tails of each star reach the other star. This minimizes possible non-linear effects originating from such initial data construction. It is worth noticing however that the initial data corresponding to the metric components $\alpha(x, y, z)$ and $\psi(x, y, z)$ are used only as initial guesses for the multigrid solver. The multigrid solver will then be responsible to calculate the appropriate metric component initial data for the star binary. See section 4.1 on how the initial data and the multigrid solvers are integrated on the whole solution scheme.

[10] According to appendix D of Lai's thesis [107]

### 2.6.7 Lorentz Boosted Initial Data

Last subsection described how to interpolate a static spherically symmetric solution for the EKG system into a three dimensional domain. The stars were not only interpolated but also translated into the 3D domain. This subsection supplements the binary initial data construction by providing the stars with speeds along the $x$ direction. In order to do so a lorentz transformation of the metric and scalar field related quantities from the star rest frame to the boosted or lab frame needs to be performed. This subsection discusses in detail how this transformation is done.

Let the primed coordinates $(t', x')$ to represent the star rest frame coordinates and the unprimed coordinates $(t, x)$ to be the lab frame coordinates. Consider a star boosted with velocity $v$ along the negative $x$ direction. So the lorentz coordinate transformation reads as[11]:

$$x = \gamma(x' + vt'), \tag{2.273}$$

$$t = \gamma(t' + vx'), \tag{2.274}$$

while the inverse lorentz transformation is:

$$x' = \gamma(x - vt), \tag{2.275}$$

$$t' = \gamma(t - vx). \tag{2.276}$$

Note that throughout this subsection the $y$ and $z$ coordinates of both systems will be supressed considering that they do not enter in the lorentz transformation law in a non-trivial form, i.e. $y' = y$ and $z' = z$, for the boost in the $x$ direction.

The real and imaginary components of the scalar field in the star rest frame follow from the static ansatz, equation (2.238):

$$\phi'_1(t', x') = \phi_0(x') \cos(\omega t'), \tag{2.277}$$

$$\phi'_2(t', x') = -\phi_0(x') \sin(\omega t'), \tag{2.278}$$

while their time derivatives are:

$$\bar{\Pi}'_1(t', x') = -\omega\phi_0(x') \sin(\omega t'), \tag{2.279}$$

$$\bar{\Pi}'_1(t', x') = -\omega\phi_0(x') \cos(\omega t'). \tag{2.280}$$

As the scalar field is invariant under coordinate transformations, the solution in the lab frame becomes:

[11] Remember that $c = 1$.

$$\phi_1(t,x) \equiv \phi_1'(t',x') = \phi_0(t,x)\cos\left[\omega\gamma(t-vx)\right], \tag{2.281}$$

$$\phi_2(t,x) \equiv \phi_2'(t',x') = -\phi_0(t,x)\sin\left[\omega\gamma(t-vx)\right], \tag{2.282}$$

where the inverse lorentz transformations, equations (2.275) and (2.276), have been inserted. The solution for the scalar field components in the lab frame at $t=0$ becomes:

$$\phi_1(0,x) = \phi_0(0,x)\cos\left(-\omega\gamma vx\right) = \phi_0(\gamma x)\cos\left(\omega\gamma vx\right), \tag{2.283}$$

$$\phi_2(0,x) = -\phi_0(0,x)\sin\left(-\omega\gamma vx\right) = \phi_0(\gamma x)\sin\left(\omega\gamma vx\right), \tag{2.284}$$

where

$$\phi_0(t,x) = \phi_0\left[\gamma(x-vt)\right] \qquad \Rightarrow \qquad \phi_0(0,x) = \phi_0(\gamma x) \tag{2.285}$$

has been used. As a remind, $\phi_0(\gamma x)$ is just the 1D solution interpolated in the 3D boosted domain, i.e. the following expression for the 3D isotropic radius is used instead of equation (2.268):

$$r_{3d} = \sqrt{(\gamma x - x_0)^2 + (y-y_0)^2 + (z-z_0)^2}. \tag{2.286}$$

Their time derivatives are also scalar fields and thus invariant under coordinate transformation as well. In order to write down an expression for them in the lab frame, one needs to go through the coordinate transformation in more details though. First note that the chain rule can be used to express the time derivatives at the star frame in terms of derivatives of the boosted (lab) frame:

$$\bar{\Pi}_1(t,x) \equiv \bar{\Pi}_1'(t',x') = \frac{\partial\phi_1'}{\partial t'}(t',x') = \frac{\partial\phi_1}{\partial t'}(t,x) = \frac{\partial\phi_1}{\partial x}(t,x)\frac{\partial x}{\partial t'} + \frac{\partial\phi_1}{\partial t}(t,x)\frac{\partial t}{\partial t'}, \tag{2.287}$$

$$\bar{\Pi}_2(t,x) \equiv \bar{\Pi}_2'(t',x') = \frac{\partial\phi_2'}{\partial t'}(t',x') = \frac{\partial\phi_2}{\partial t'}(t,x) = \frac{\partial\phi_2}{\partial x}(t,x)\frac{\partial x}{\partial t'} + \frac{\partial\phi_2}{\partial t}(t,x)\frac{\partial t}{\partial t'}. \tag{2.288}$$

The derivatives appearing on the expressions above can be calculated from the equations (2.281) and (2.282):

$$\frac{\partial\phi_1}{\partial x}(t,x) = \frac{\partial\phi_0}{\partial x}(t,x)\cos\left[\omega\gamma(t-vx)\right] + \omega\gamma v\phi_0(t,x)\sin\left[\omega\gamma(t-vx)\right], \tag{2.289}$$

$$\frac{\partial\phi_2}{\partial x}(t,x) = -\frac{\partial\phi_0}{\partial x}(t,x)\sin\left[\omega\gamma(t-vx)\right] + \omega\gamma v\phi_0(t,x)\cos\left[\omega\gamma(t-vx)\right] \tag{2.290}$$

and considering that

$$\frac{\partial\phi_0}{\partial t}(t,x) = \frac{\partial\phi_0}{\partial x'}(x')\frac{\partial x'}{\partial t} = -\gamma v\left.\frac{\partial\phi_0}{\partial x'}(x')\right|_{x'=\gamma(x-vt)}, \tag{2.291}$$

then

$$\begin{aligned}\frac{\partial\phi_1}{\partial t}(t,x) &= \frac{\partial\phi_0}{\partial t}(t,x)\cos\left[\omega\gamma(t-vx)\right]-\omega\gamma\phi_0(t,x)\sin\left[\omega\gamma(t-vx)\right]\\ &= -\gamma v\left.\frac{\partial\phi_0}{\partial x'}(x')\right|_{x'=\gamma(x-vt)}\cos\left[\omega\gamma(t-vx)\right]-\omega\gamma\phi_0(t,x)\sin\left[\omega\gamma(t-vx)\right] \quad (2.292)\end{aligned}$$

and

$$\begin{aligned}\frac{\partial\phi_2}{\partial t}(t,x) &= -\frac{\partial\phi_0}{\partial t}(t,x)\sin\left[\omega\gamma(t-vx)\right]-\omega\gamma\phi_0(t,x)\cos\left[\omega\gamma(t-vx)\right]\\ &= \gamma v\left.\frac{\partial\phi_0}{\partial x'}(x')\right|_{x'=\gamma(x-vt)}\sin\left[\omega\gamma(t-vx)\right]-\omega\gamma\phi_0(t,x)\cos\left[\omega\gamma(t-vx)\right]. \quad (2.293)\end{aligned}$$

Inserting equations (2.289), (2.290), (2.292) and (2.293) back into equations (2.287) and (2.288) results into:

$$\bar{\Pi}_1(t,x) = \gamma v\left[\frac{\partial\phi_0}{\partial x}(t,x)-\gamma\frac{\partial\phi_0}{\partial x'}(x')\right]\cos\left[\omega\gamma(t-vx)\right]-\omega\phi_0(t,x)\sin\left[\omega\gamma(t-vx)\right], \quad (2.294)$$

$$\bar{\Pi}_2(t,x) = \gamma v\left[\gamma\frac{\partial\phi_0}{\partial x'}(x')-\frac{\partial\phi_0}{\partial x}(t,x)\right]\sin\left[\omega\gamma(t-vx)\right]-\omega\phi_0(t,x)\cos\left[\omega\gamma(t-vx)\right], \quad (2.295)$$

where the derivative with respect to $x'$ is supposed to be evaluated at $x'=\gamma(x-vt)$. The solution in the lab frame at $t=0$ becomes then:

$$\bar{\Pi}_1(0,x) = \gamma v\left[\frac{\partial\phi_0}{\partial x}(\gamma x)-\gamma\left.\frac{\partial\phi_0}{\partial x'}(x')\right|_{x'=\gamma x}\right]\cos(\omega\gamma vx)+\omega\phi_0(\gamma x)\sin(\omega\gamma vx), \quad (2.296)$$

$$\bar{\Pi}_2(0,x) = \gamma v\left[\frac{\partial\phi_0}{\partial x}(\gamma x)-\gamma\left.\frac{\partial\phi_0}{\partial x'}(x')\right|_{x'=\gamma x}\right]\sin(\omega\gamma vx)-\omega\phi_0(\gamma x)\cos(\omega\gamma vx). \quad (2.297)$$

In order to completely determine the initial data, the metric components need to be boosted as well. The metric components for the static star is simply given by:

$$g'_{\mu\nu}(t',x') = \begin{pmatrix} -\alpha'^2(t',x') & 0 & 0 & 0\\ 0 & \psi'^4(t',x') & 0 & 0\\ 0 & 0 & \psi'^4(t',x') & 0\\ 0 & 0 & 0 & \psi'^4(t',x')\end{pmatrix}. \quad (2.298)$$

A coordinate transformation is supposed to relate the metric components from the static star frame to the boosted one:

$$g_{\lambda\delta}(t,x) = \frac{\partial x'^{\mu}}{\partial x^{\lambda}}\frac{\partial x'^{\nu}}{\partial x^{\delta}}g'_{\mu\nu}(t',x'), \quad (2.299)$$

where the lab frame metric components $g_{\lambda\delta}(t,x)$ is written as:

$$g_{\lambda\delta}(t,x) = \begin{pmatrix} -\alpha^2(t,x) + \gamma^{ij}\beta_i\beta_j & \beta_k(t,x) \\ \beta_l(t,x) & \gamma_{lk}(t,x) \end{pmatrix} \tag{2.300}$$

and the lorentz boost transformation as:

$$\Lambda^{\mu}_{\lambda} = \frac{\partial x'^{\mu}}{\partial x^{\lambda}} = \begin{pmatrix} \gamma & -\gamma v & 0 & 0 \\ -\gamma v & \gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{2.301}$$

For example, the $g_{0x}$ component of the boosted metric would be simply given by the following transformation:

$$g_{0x} \equiv \beta_x(t,x) = \Lambda^{\mu}_{\ 0}\Lambda^{\nu}_{\ x}g'_{\mu\nu} = \left(\Lambda^{0}_{\ 0}g'_{0\nu} + \Lambda^{x}_{\ 0}g'_{x\nu}\right)\Lambda^{\nu}_{\ x} \tag{2.302}$$

that results into the following expression:

$$\beta_x(t,x) = \gamma^2 v\left[\alpha'^2(t',x') - \psi'^4(t',x')\right]. \tag{2.303}$$

Analogously, it is possible to show that the boosted spatial metric is related to the static metric by:

$$\gamma_{ij} = \begin{pmatrix} \gamma^2\left[\psi'^4(t',x') - v^2\alpha'^2(t',x')\right] & 0 & 0 \\ 0 & \psi'^4(t',x') & 0 \\ 0 & 0 & \psi'^4(t',x') \end{pmatrix}, \tag{2.304}$$

what results then into the following form for the shift vector $x$ component:

$$\beta^x(t,x) = \frac{v\left[\alpha'^2(t',x') - \psi'^4(t',x')\right]}{\left[\psi'^4(t',x') - v^2\alpha'^2(t',x')\right]}. \tag{2.305}$$

Remember, however that the static metric components must be time independent. So the shift vector $x$ component becomes:

$$\beta^x(t,x) = \frac{v\left[\alpha'^2(x') - \psi'^4(x')\right]}{\left[\psi'^4(x') - v^2\alpha'^2(x')\right]} \tag{2.306}$$

and restricting to the initial time $t = 0$, the equation assumes its final form:

$$\beta^x(0,x) = \frac{v\left[\alpha'^2(\gamma x) - \psi'^4(\gamma x)\right]}{\left[\psi'^4(\gamma x) - v^2\alpha'^2(\gamma x)\right]}. \tag{2.307}$$

Applying a derivation analogous to the one used for $\beta^x(0,x)$, it is possible to show that the lapse function at $t = 0$ can be expressed as:

$$\alpha(0,x) = \frac{\alpha'(\gamma x)\psi'^2(\gamma x)}{\gamma\sqrt{\psi'^4(\gamma x) - v^2\alpha'^2(\gamma x)}}, \tag{2.308}$$

while the conformal factor is simply:

$$\psi(0,x) = \psi'(\gamma x). \tag{2.309}$$

Finally, with the results from equations (2.283), (2.284), (2.296), (2.297), (2.307), (2.308) and (2.309), the initial data for the scalar field conjugate momentum can be written as:

$$\Pi_1(0,x) = \frac{\psi^6(0,x)}{\alpha^2(0,x)}\left[\bar{\Pi}_1(0,x) - \beta^x(0,x)\frac{\partial\phi_1}{\partial x}(0,x)\right], \tag{2.310}$$

$$\Pi_2(0,x) = \frac{\psi^6(0,x)}{\alpha^2(0,x)}\left[\bar{\Pi}_2(0,x) - \beta^x(0,x)\frac{\partial\phi_2}{\partial x}(0,x)\right]. \tag{2.311}$$

To summarize, equations (2.283), (2.284), (2.310) and (2.311) initialize the scalar field components and their conjugate momenta in the boosted or lab reference frame. They are used as sources by the the multigrid solver. On the other hand, equations (2.307), (2.308) and (2.309) are used either as initial guesses for the shift vector $x$ component, lapse function and conformal factor, respectively, or they are used to construct the initial data for the scalar field conjugate momenta, equations (2.310) and (2.311). If a second star is added, then the same equations are used to set its initial data as well. The two sets of initial data corresponding to the two stars are then superposed as:

$$\phi_1(0,x) = \phi_1^{(1)}(0,x) + \phi_1^{(2)}(0,x), \tag{2.312}$$

$$\phi_2(0,x) = \phi_2^{(1)}(0,x) + \phi_2^{(2)}(0,x), \tag{2.313}$$

$$\Pi_1(0,x) = \Pi_1^{(1)}(0,x) + \Pi_1^{(2)}(0,x), \tag{2.314}$$

$$\Pi_2(0,x) = \Pi_2^{(1)}(0,x) + \Pi_2^{(2)}(0,x), \tag{2.315}$$

$$\beta^x(0,x) = \beta^{x\,(1)}(0,x) + \beta^{x\,(2)}(0,x), \tag{2.316}$$

$$\alpha(0,x) = \alpha^{(1)}(0,x) + \alpha^{(2)}(0,x) - 1, \tag{2.317}$$

$$\psi(0,x) = \psi^{(1)}(0,x) + \psi^{(2)}(0,x) - 1, \tag{2.318}$$

where the superscrips (1) and (2) refers to the solution of each star individually. Therefore the scalar field real component, $\phi_1$, imaginary component, $\phi_2$, and their conjugate momenta, $\Pi_1$ and $\Pi_2$, are specified for the binary star at the initial time step according to equations (2.312)-(2.315). They are used then as sources for the multigrid solver. The multigrid solver, on the other hand, accepts as initial guesses for the $x$ component of the shift vector, $\beta^x$, the lapse function, $\alpha$ and the conformal factor, $\psi$, as evaluated by equations (2.316)-(2.318). The multigrid solver then finds the solution for the complete set of geometric variables $\{\alpha, \psi, \beta^x, \beta^y, \beta^z\}$.

### 2.6.8 Note on $\partial/\partial r^p$ operators

Equation (2.222) had one of its differential operators replaced from $\partial/\partial r$ to $\partial/\partial r^2$. Other equations in this section had similar replacements as well. This subsection goal is to briefly discuss the reasons behind these operator modifications. First the equivalence between the two differential operators is shown and then an explicit example is worked out in order to show their similarities and differences.

The replacement of the differential operators is related to the necessity of keeping the metric functions regular at $r = 0$. A tensor quantity is said to be regular at $r = 0$ when its components relative to the cartesian coordinates (related to the spherical coordinates by the usual transformation law[12]) can all be expanded in non-negative integer powers of $x$, $y$ and $z$ [125]. In other words, when its cartesian components can be represented by a convergent Taylor series around $r = 0$ [128]. Loosely speaking, all that it is required is that the tensor components do not diverge as radial coordinate approaches to $r = 0$.

Let then $f(t, r)$ be a regular function with the following leading order behaviour as $r$ approaches to zero:

$$\lim_{r\to 0} f(t,r) = r^p f_p(t) + O(r^{p+2}), \tag{2.319}$$

where $p$ is a positive integer. Its first derivative with respect to $r$ is simply:

$$\lim_{r\to 0} \partial_r f(t,r) = pr^{p-1} f_p(t) + (p+2)O(r^{p+1}). \tag{2.320}$$

However the leading order dependence on $r$ can be eliminated immediately by differentiating with respect to $r^p$ instead:

$$\lim_{r\to 0} \partial_{r^p} f(t,r) = f_p(t) + \frac{(p+2)}{p} O(r^2) \tag{2.321}$$

and can be recovered by multiplying the equation above by $pr^{p-1}$:

$$\lim_{r\to 0} pr^{p-1}\partial_{r^p} f(t,r) = pr^{p-1} f_p(t) + (p+2)O(r^{p+1}). \tag{2.322}$$

Therefore, as $r$ goes to zero, $\partial_r f$ can be written as:

$$\partial_r f(t,r) = pr^{p-1}\partial_{r^p} f(t,r) + O(r^{p+1}). \tag{2.323}$$

This expression then ends the proof of the equivalence between this two differential operators. Let consider now an example that show the differences that arise between them when they are approximated by standard second-order centered finite difference operators. Suppose a laplacian term is part of a differential equation to be second-order discretized:

$$\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial \psi}{\partial r}\right) \tag{2.324}$$

[12] $x = r\sin\theta\cos\phi$, $y = r\sin\theta\sin\phi$ and $z = r\cos\theta$

and $\psi(t,r)$ has to satisfy the regularity condition by:

$$\lim_{r\to 0}\psi(t,r)=\psi_0(t)+\psi_2(t)r^2+O(r^4). \tag{2.325}$$

In this case, the laplacian becomes more natural if expressed in terms of derivatives of $r^3$ as can be noted by inserting the $\psi$ regularity condition on the laplacian term:

$$\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial\psi}{\partial r}\right)\simeq\frac{1}{r^2}\frac{\partial}{\partial r}\left[r^2\frac{\partial}{\partial r}\left(\psi_0(t)+\psi_2(t)r^2\right)\right]=\frac{1}{r^2}\frac{\partial}{\partial r}\left(2\psi_2(t)r^3\right) \tag{2.326}$$

and the last term of the line above can either be expressed as:

$$\frac{1}{r^2}\frac{\partial}{\partial r}\left(2\psi_2(t)r^3\right)=\frac{6\psi_2(t)r^2}{r^2}=6\psi_2(t)\qquad\text{or}\qquad\frac{1}{r^2}\frac{\partial}{\partial r}\left(2\psi_2(t)r^3\right)=3\frac{\partial}{\partial r^3}\left(2\psi_2(t)r^3\right)=6\psi_2(t) \tag{2.327}$$

that the results are identical. On the other hand, when both operators are discretized the results turn out to be different. Let the usual second-order discretization approximate[13] the laplacian term:

$$\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial\psi}{\partial r}\right)\simeq\frac{1}{r_j^2}\frac{\left[r_{j+\frac{1}{2}}^2\left(\psi_{j+1}^n-\psi_j^n\right)-r_{j-\frac{1}{2}}^2\left(\psi_j^n-\psi_{j-1}^n\right)\right]}{\left(r_j-r_{j-1}\right)^2}, \tag{2.328}$$

where $\psi_j^n$ represents $\psi(t,x)$ at time level $n$ and grid point $j$, $r_{j+\frac{1}{2}}=(r_{j+1}+r_j)/2$ and $r_{j-\frac{1}{2}}=(r_j+r_{j-1})/2$. Inserting the expansion for $\psi$, equation (2.325), into the finite difference approximation above results into:

$$6\psi_2(t)+\frac{1}{2}\psi_2(t)\frac{\left(r_{j+1}-r_j\right)^2}{r_j^2}, \tag{2.329}$$

where $\psi_2(t)$ is understood to be evaluated at time level $n$. Note that this result does differ from the continuum one, equation (2.327). The discretization of the laplacian with the $r$ derivative replaced by the derivative with respect to $r^3$ is achieved by:

$$3\frac{\partial}{\partial r^3}\left(r^2\frac{\partial\psi}{\partial r}\right)\simeq\frac{3}{\left(r_{j+\frac{1}{2}}^3-r_{j-\frac{1}{2}}^3\right)}\left[r_{j+\frac{1}{2}}^2\frac{\left(\psi_{j+1}^n-\psi_j^n\right)}{\left(r_j-r_{j-1}\right)}-r_{j-\frac{1}{2}}^2\frac{\left(\psi_j^n-\psi_{j-1}^n\right)}{\left(r_j-r_{j-1}\right)}\right]. \tag{2.330}$$

Inserting the equation (2.325) on the expression above results exactly into:

$$6\psi_2(t), \tag{2.331}$$

the same result as obtained by the continuum version, equation (2.327).

As a conclusion both differential operators are identical on the continuum domain for regular functions with behaviour given by equation (2.325). On the other hand, their second order discrete approximations are distinct and only the discrete version of $\partial/\partial r^3$ preserves the function continuum

[13] See Chap. 3 for full details on how differential operators are approximated by finite difference operators

regularity behaviour. For this reason, whenever the function behaviour as $r \to 0$ were identified on the equations appearing in this section, the operators were replaced by their counterparts that preserve the regularity conditions on the discrete domain.

# CHAPTER 3

# NUMERICAL ANALYSIS AND TECHNIQUES

This chapter discusses the numerical methods and analysis employed to find a numerical solution – besides testing it for its correctness – for the partial differential equations (PDE) comprising the Einstein-Klein-Gordon system under Conformal Flat Condition. Some topics here are discussed in more details than others but in essence all necessary concepts are mentioned. The contents of this chapter are heavily based on the Choptuik's numerical analysis for numerical relativists lecture notes [132] and the reader is referred to these notes for examples and a more detailed discussion of the concepts. Also the numerical analysis book by Burden and Faires [133] provides an excellent introduction to the field.

As mentioned in the last chapter, the Einstein-Klein-Gordon system under CFA comprises of a system of 4 first order hyperbolic and 5 second order elliptic equations coupled to each other. Since the equations are of a different kind, it is natural to expect that different numerical schemes are applied in order to obtain their numerical solution. In brief, the hyperbolic equations are solved in this thesis by an iterative Crank-Nicholson scheme while the solution of the elliptic equations are obtained through a multigrid technique. Both of them will be described in this chapter with a greater emphasis to the multigrid method.

Another key ingredient to find the solutions of PDEs concerns the discretization approach. The one adopted here is the finite difference approximation. It consists of substituting the differential operators present in the PDEs by appropriate finite difference operators. While the differential operators are applied on functions defined on a particular subdomain $\Omega$ of the continuum $\mathbb{R}^3$, the discrete operators are only applied to functions defined on a grid $\Omega^h$. One section is devoted on the derivation of the finite difference operators and the App. C describes the use of a set of Maple procedures created to automatically do the discretization of any PDE.

This chapter is organized as follows. Sec. 3.1 introduces the main concepts related to the finite difference discretization of partial differential equations. It also derives the discrete operators that approximate the differential operators appearing at the PDEs describing the EKG system under CFA. Sec. 3.2 introduces the analysis tools used to show the correctness of the numerical solution. Finally, Sec. 3.3 discusses in detail the multigrid technique. This is a numerical technique used to solve the system of algebraic equations originated from the discretization of the elliptic PDEs.

## 3.1 Discretization of Partial Differential Equations: Finite Difference Approximation

The fundamental assumptions of the finite difference approximation (FDA) of PDEs are:

1. Given boundary and/or initial conditions, there is a unique solution of the system of PDEs to be discretized.

2. This unique solution is smooth.

As argued in Chap. 2, the solution of the coupled elliptic-hyperbolic system of equations characterizing the boundary-initial value problem to be solved in this thesis is expected to be smooth once the prescribed initial data is smooth. Therefore the choice of finite difference approximation as the discretization approach becomes a natural one for the problem at hand.

The first step on the discretization of PDEs consists on the definition of a grid (or mesh) over the function domain of integration $\Omega \subset \mathbb{R}^n$. Loosely speaking the grid is defined as a set of points from this domain that obey some criteria. These criteria can be as diverse as the physical problems governed by PDEs. In general though the grid generation is governed by the geometry of the domain boundaries and the coordinate system in which the PDE is written. For boundaries with very complex geometries for example it is common to use unstructured grids that adapt themselves to the geometry. Grids used on simulations of car crash or airplane flight are usually of this kind. On the other hand, a physical problem with spherical symmetry for example has a coordinate system adapted to the symmetry. It is natural therefore to use a curvilinear grid that mimics the symmetry of the problem. In the case of the Einstein-Klein-Gordon system under CFA the coordinate system in which the equations are written is simply the cartesian coordinates. Thus, as anticipated, the grid adopted in their discretization should also be cartesian. Besides, the lack of complicated boundary geometries finally suggests the use of the simplest grid possible: a uniform cartesian grid. For a 3D spatial domain $\Omega \subset \mathbb{R}^3$, for example, this consists of an ordered set of points $(x_i, y_j, z_k) \equiv \Omega^h \subset \Omega$ such that their separation in the domain $\Omega$ is constant and equals to $h$:

$$x_{i+1} - x_i = h, \qquad y_{j+1} - y_j = h, \qquad z_{k+1} - z_k = h, \tag{3.1}$$

where $h$ is called mesh or grid spacing and acts as a control parameter of the discretization. In the remainder of this chapter, only a single discretization scale $h$ is considered.

Certainly the total number of grid points is finite considering the finite computational resources available. The uniform cartesian grid points are obtained from the coordinate range of the problem. In general for:

$$x_{min} \leq \quad x \quad \leq x_{max}, \tag{3.2}$$

$$y_{min} \leq \quad y \quad \leq y_{max}, \tag{3.3}$$

$$z_{min} \leq \quad z \quad \leq z_{max}, \tag{3.4}$$

the grid points are obtained by:

$$x_i = x_{min} + (i-1)\,h, \qquad i = 1 \ldots n_x, \tag{3.5}$$

$$y_j = y_{min} + (j-1)\,h, \qquad j = 1 \ldots n_y, \tag{3.6}$$

$$z_k = z_{min} + (k-1)\,h, \qquad k = 1 \ldots n_z, \tag{3.7}$$

such that:

$$x_1 = x_{min} \quad \text{and} \quad x_{n_x} = x_{max}, \tag{3.8}$$

$$y_1 = y_{min} \quad \text{and} \quad y_{n_y} = y_{max}, \tag{3.9}$$

$$z_1 = z_{min} \quad \text{and} \quad z_{n_z} = z_{max}. \tag{3.10}$$

The grid points are then used to define the grid functions $u^h : \Omega^h \to \mathbb{R}$. For the 3D grid used as an example in this section, the grid function $u^h$ is simply equivalent to $u_{ijk}$ defined on every grid point $(x_i, y_j, z_k)$. Such kind of grids are also called vertex centered grids since the unknowns are defined on the grid points instead of, for example, at the center of a cubic cell or at the faces of a cube formed by 8 neighbouring grid points. Fig. 3.2 shows a sample of a vertex centered 3D grid.

The derivative operators used in the PDEs are approximated by finite difference operators. Each finite difference operator is usually characterized by its difference stencil or difference star. This consists of the set of neighbouring grid points entering in the definition of the difference operator. For example, if the real function of one variable $u : \mathbb{R} \to \mathbb{R}$ is restricted to its values at the one dimensional grid $\Omega^h$, then the second order derivative of $u$ could be approximated by using the following combination of the grid functions $u^h : \Omega^h \to \mathbb{R}$:

$$\frac{d^2}{dx^2}u(x) \approx \frac{u_{i-1} - 2u_i + u_{i+1}}{h^2}. \tag{3.11}$$

The stencil of this derivative operator (see Fig. 3.1) has an overall multiplicative factor, $1/h^2$, and it is weighted by the ordered set of integers $(1, -2, 1)$ corresponding to the relative weight that neighbour grid functions have in the approximation of the differential operator. A later subsection will be devoted to the derivation and error analysis of these stencils. The focus now should shift on the meaning of this sort of approximation and on the basic concepts related to it.

App. C introduces a set of Maple procedures written by the author with the intention of automate the derivation of finite difference operators of any order and the subsequent discretization of partial differential equations of any order as well. Several examples illustrate how to use the procedures at the same time that output the most commonly used finite difference operators. A couple of PDEs are also discretized in order to demonstrate the flexibility of these routines. Please refer to App. C to more details.

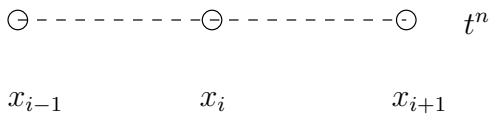


**Figure 3.1:** Second-order spatial derivative operator stencil.

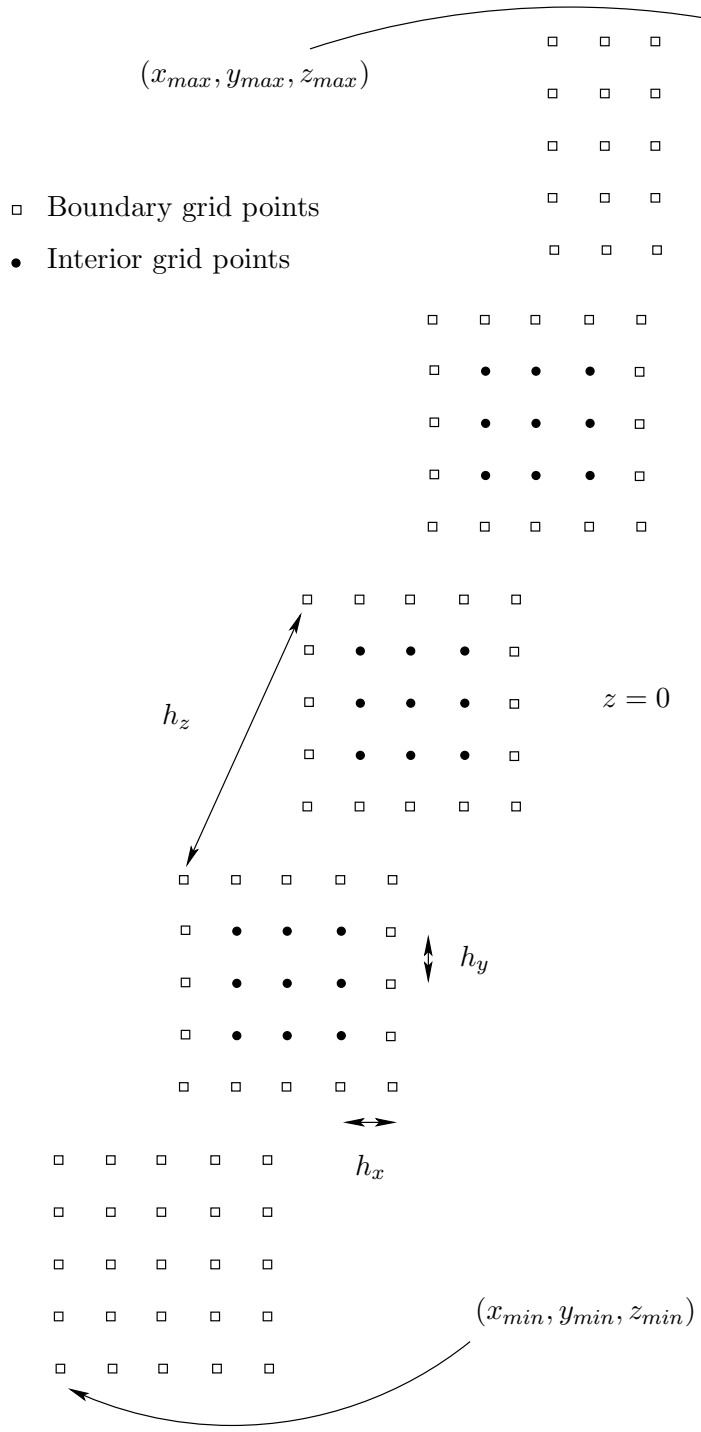


**Figure 3.2:** 3d grid sample. This figure shows a 3D grid drawn in perspective. This particular grid is composed by $5 \times 5 \times 5$ grid points covering the cartesian coordinates from $(x_{min}, y_{min}, z_{min})$ to $(x_{max}, y_{max}, z_{max})$. The solid circles represent interior grid points, while the hollow squares denote boundary grid points. The grid spacings were drawn out of scale for the sake of clarity. Here and throughout this manuscript: $h_x = h_y = h_z \equiv h$.

### 3.1.1 Basic Concepts Related to FDAs

Consider a system of differential equations described abstractly by:

$$Lu - f = 0, \tag{3.12}$$

where, in its all generality, $L$ is a set of differential operators acting on a vector $u$ of unknown dependent functions $u = (u_1, \ldots, u_m)$ of $n$ independent variables, $u_i = u_i(x_1, \ldots, x_n)$ for $i = 1 \ldots m$, and $f$ a prescribed $n$-dimensional function, also known as the source of the differential system.

The finite approximation of this generic differential system results into a system of algebraic equations. The number of algebraic equations and unknowns naturally depends on the number of grid points and therefore can be parametrized by the mesh spacing $h$. To reflect this fact, it is usual to write the FDA of equation (3.12) as:

$$L^h u^h - f^h = 0, \tag{3.13}$$

where the operator $L^h$ is the FDA of the differential operator $L$, $u^h$ the discrete solution of the algebraic equations and $f^h$ the values of the source function $f$ restricted to the grid points. The solution of the PDE $u : \Omega \to \mathbb{R}$ is approximated therefore by the solution of the algebraic system of equations, $u^h : \Omega^h \to \mathbb{R}$.

In order to capture and quantify how much the discrete solution $u^h$ deviates from the continuum solution $u$ in a given calculation for a fixed value of $h$, it is natural to define the *solution error* as:

$$e^h \equiv u^h - u. \tag{3.14}$$

It is assumed that the solution error does depend on the finite value of $h$. The analysis of approximation schemes try to answer questions associated to the relation between the solution error and the mesh spacing $h$ as this tends to zero: does the discrete solution approaches the continuum one? if yes, at what rate? Roughly speaking, numerical analysts judge the quality of the finite difference approximation by how fast the error goes to zero as a function of $h$. That motivates the introduction of the concept of *convergence* of the approximate solution. The approximation is said to converge iff:

$$\lim_{h \to 0} u^h = u \qquad \text{or} \qquad \lim_{h \to 0} e^h = 0. \tag{3.15}$$

The order of convergence measures the rate in which the error converges to zero. If

$$\lim_{h \to 0} e^h = O(h^p), \tag{3.16}$$

then the discrete solution is said to converge to the continuum one with order $p$. More specifically if $p = 1$ the solution is said to converge linearly and if $p = 2$ quadratically, for example.

The discrete solution $u^h$ of the algebraic system of equations, equation (3.13), can be calculated exactly (for a system that admits a non-singular solution). In the context of numerical computa-

tions, exact means as accurate as the floating-point number representation allows to be, i.e. down to machine precision. However, the calculation of the exact solution for systems where the number of unknowns is large may be computationally prohibitly expensive to obtain. This is usually the case for systems defined on 2D or 3D grids. There the usual ordering of the unknowns results into extremely large banded matrices whose bands increase with the total number of unknowns. As a result it is more advantageous to calculate an approximate solution through the so called iterative methods instead of obtaining the exact solution by direct methods as in the case of small systems.

The iterative methods, as the name suggests, iterates into a sequence of approximated solutions $\tilde{u}^h$ starting from an initial guess for what the solution should be. Each member of this sequence satisfies the equation (3.13) only approximately. The level of deviation from satisfaction of the FDA that the approximate solution produces is called *residual* $r^h$ and it is defined as:

$$r^h \equiv L^h \tilde{u}^h - f^h. \tag{3.17}$$

Consequently, to solve the equation (3.13) by an iterative process actually means to drive the residual to 0. The iterative process is said to converge if there is an arbitrarily small positive number $\epsilon > 0$ such that $r^h < \epsilon$ for all members of the approximate solution sequence coming after a finite number $N > 0$ of members.

So far the two new concepts introduced, solution error and residual, are related somehow to the solution itself. One last question is missing in the analysis: how good is the FDA $L^h$ with respect to its continuum counterpart $L$? In order to answer this question it is convenient to introduce a measure of this discrepancy. It is known in the literature as *truncation error* and it is defined as:

$$\tau^h \equiv L^h u - f^h, \tag{3.18}$$

where $u$ is the continuum solution of the differential system, equation (3.12). Note that if equation (3.12) is evaluated *at* the grid points then the subtraction of equation (3.18) gives exactly the truncation error, what is left[1] from a Taylor expansion of $L^h$ in terms of $L$:

$$(L^h - L)u = \tau^h. \tag{3.19}$$

The FDA is said to be *consistent* (with the PDE) if the truncation error goes to zero as $h$ tends to zero:

$$\lim_{h \to 0} \tau^h = 0 \tag{3.20}$$

and the FDA is declared to be $p$-th order accurate if:

$$\lim_{h \to 0} \tau^h = O(h^p). \tag{3.21}$$

The discretization of a PDE through finite differences addresses the question of what the truncation error $\tau^h$ functional form would be. Based on this information we should address what the functional form of the *solution* error would be.

[1] See App. B.

Richardson's Ansatz assumes that the solution for a centered, $O(h^p)$ (where $p$ is an even integer), finite difference approximation to the partial differential equation can be expanded in even powers of $h$. The argument comes from the following reasoning. The solution $u$ of the differential equation can be viewed or interpreted as the result of the integration of the PDE. The centered FDA approximating the PDE has an associated truncation error of the following form:

$$\tau^h = h^p \tau_p(t,x,y,z) + h^{p+2}\tau_{p+2}(t,x,y,z) + h^{p+4}\tau_{p+4}(t,x,y,z) + \dots \tag{3.22}$$

It is natural then to suppose that the error in the integration also preserves the same form as long as $\tau_p$, $\tau_{p+2}$, $\tau_{p+4}$, etc... do not depend on the mesh parameter $h$. The solution error should then be[2]:

$$e^h = h^p e_p(t,x,y,z) + h^{p+2}e_{p+2}(t,x,y,z) + h^{p+4}e_{p+4}(t,x,y,z) + \dots \tag{3.23}$$

This observation about the solution error is due to Richardson [134] and it is known in the literature as *Richardson's Ansatz*. We can argue then that *consistency* leads to *convergence*:

$$\lim_{h\to 0} \tau^h = O(h^p) \qquad \Rightarrow \qquad \lim_{h\to 0} e^h = O(h^p). \tag{3.24}$$

In general it is not possible to *prove* the existence of Richardson's ansatz. One can prove though for simple cases in which the PDE and its discretization are amenable to a closed form analysis. See page 111 of reference [135] for an example. In practice though the existence of the expansion is proven by construction. If the discrete solution obtained is consistent and convergent then the Richardson's ansatz exists.

### 3.1.2 Crank-Nicholson Discretization Scheme

To conclude this section it is worth working out an example of discretization of a time dependent PDE as well as some related concepts. The problem considered here is the advection equation. However the technique can be generalized to virtually any system of first order time dependent PDE. The end of this subsection shows how.

The one dimensional advection equation is an example of first order hyperbolic PDE. Without loss of generality this equation and its initial-boundary condition can be written as:

$$\frac{\partial u(t,x)}{\partial t} = \frac{\partial u(t,x)}{\partial x} \qquad -x_{min} < x < x_{max} \quad , \quad t \geq 0, \tag{3.25}$$

$$u(0,x) = u_0(x), \tag{3.26}$$

$$u(t,x_{min}) = 0, \tag{3.27}$$

$$u(t,x_{max}) = 0. \tag{3.28}$$

The best strategy for discretizing PDEs is to try to keep the stencils as centered as possible.

[2]This argument can be extended as well for off-centered FDAs as long as odd terms of the expansion in terms of $h$ are also included

Besides reducing the number of points required in the calculation, the symmetry of centered schemes translates into symmetric systems of linear equations. The properties of symmetric matrices are well known in the literature and usually used to prove existence and uniqueness of solutions for systems of this kind. The stability theorems of iterative techniques to solve linear systems of algebraic equations also makes use of this symmetry to state their results. Without elongating this discussion any further, consider then the following discrete domain $(t^n, x_i)$:

$$t^n \equiv n\Delta t, \qquad n = 0, 1, 2, \ldots \tag{3.29}$$

$$x_i \equiv x_{min} + (i-1)h, \qquad i = 1, 2, \ldots n_x, \tag{3.30}$$

$$u_i^n \equiv u^h(t^n, x_i), \tag{3.31}$$

$$h = \frac{(x_{max} - x_{min})}{(n_x - 1)}, \tag{3.32}$$

$$\Delta t = \lambda h, \tag{3.33}$$

where $\lambda$ is known as Courant factor.

The Crank-Nicholson discretization scheme is a two-level, second order approximation scheme centered both in space and time around the fictitious grid point $(t^{n+1/2}, x_i)$, as shown on Fig. 3.3. One obvious advantage of this method regards the use of only two time levels. Another not so obvious advantage – see [132]for a proof – is that Crank-Nicholson minimizes the truncation error for a certain mesh space $h$. In general, the Crank-Nicholson discretization also tends to minimize possible instabilities.

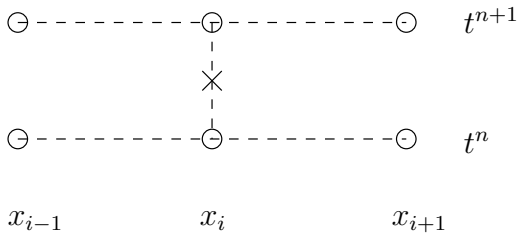


**Figure 3.3:** Crank-Nicholson stencil.

The advection equation discretized by the Crank-Nicholson scheme is written as follows:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \mu_t \left( \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x} \right), \tag{3.34}$$

where the operator $\mu_t$ is a $O(\Delta t^2)$ time average operator. Its action on a grid function consists in averaging its values between the time levels $t^n$ and $t^{n+1}$:

$$\mu_t u_i^n = \frac{1}{2} \left( u_i^{n+1} + u_i^n \right). \tag{3.35}$$

Inserting the definition of the average operator into equation (3.34) results into the following

expression:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{1}{2}\left[\frac{u_{i+1}^{n+1} - u_{i-1}^{n+1}}{2\Delta x} + \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x}\right], \tag{3.36}$$

that can be rewritten as follows:

$$-\frac{\lambda}{8}u_{i+1}^{n+1} + u_i^{n+1} + \frac{\lambda}{8}u_{i-1}^{n+1} = u_i^n + \frac{\lambda}{8}\left(u_{i+1}^n - u_{i-1}^n\right). \tag{3.37}$$

This form along with the boundary conditions is clearly a linear system for the unknowns $u_i^{n+1}$, $i = 1, 2, \ldots, n_x$ of the following form:

$$Au^{n+1} = b. \tag{3.38}$$

Note however that this system does couple the unknowns at advanced time level and for this reason it is called an *implicit* scheme. The matrix $A$ in this case is not diagonal. From equation (3.37) it is clear that $A$ is actually a tridiagonal matrix, i.e. only the elements of its diagonal, the immediate upper and lower diagonals are non-vanishing. Note that this kind of system can be solved very efficiently, $O(n_x)$, using specialized algorithms. LAPACK [136] is a linear algebra software package commonly used that contains plenty of such algorithms efficiently encoded.

The evolution equations used in this dissertation was cast in a system of first order, four dimensional PDEs, also known as canonical form, each of them taking the following general form:

$$\frac{\partial u(t,x,y,z)}{\partial t} = F(x,y,z,u,u_x,u_{xx}). \tag{3.39}$$

Their discretization was performed by a second-order accurate Crank-Nicholson scheme. As explained earlier for the advection equation, this consists in approximate the time derivative by a $O(h^2)$ finite difference operator centered at $t = t^n + \Delta t/2$:

$$\frac{\partial u(t,x,y,z)}{\partial t} = \frac{u_{ijk}^{n+1} - u_{ijk}^n}{\Delta t} + O(\Delta t^2), \tag{3.40}$$

while the spatial part is averaged between the advanced and present time levels using the time average operator $\mu_t$. The net result of the operation on a general spatial right hand side is just as follows:

$$F(x,y,z,u,u_x,u_{xx}) = \frac{1}{2}\left(F_{ijk}^{n+1} + F_{ijk}^n\right) + O(\Delta t^2). \tag{3.41}$$

## 3.2 Numerical Analysis Tools

This section introduces two numerical analysis tools employed to evaluate the correctness of the numerical solution and the FDA itself. First the convergence factor $Q^h(t)$ is defined and its functional properties analysed. Second the notion of independent residual evaluation is presented along with its expected behaviour as the numerical solution approaches its continuum counterpart.

### 3.2.1 Convergence Factor

In order to test for the convergence of the FDA solution to the continuum PDE solution, we need first to calculate the solution at three different levels of resolution. The solutions $u^h$, $u^{2h}$ and $u^{4h}$ correspond to FDA solutions with grid spacings of $h$, $2h$ and $4h$, respectively. They are generated using the same initial data (but in different meshes). Also the grids containing the solutions are supposed to align with the coarsest grid, i.e. the coarsest grid with spacings $4h$ for example is a subset of a finer grid with $2h$ spacings and the later is a subset of the finest grid with the mesh distance of $h$. Second, we need to find a convenient way to compare these three solutions at different resolutions in a way to capture the behaviour of the discrete solution as $h \to 0$. The *convergence factor* is a function of the solution at different resolutions and it does provide a convenient way to test for convergence of the discrete solution. It is defined as a function of time as:

$$Q(t) \equiv \frac{||u^{4h} - u^{2h}||}{||u^{2h} - u^h||}, \tag{3.42}$$

where $||\cdot||$ is any appropriate $l_p$ norm, like $l_2$ or $l_\infty$ norm, or root-mean-square norm:

$$||u^h||_{rms} = \frac{1}{\sqrt{n}}||u^h||_2 = \frac{1}{\sqrt{n}}\left[\sum_{i=1}^{n}|u_i^h|^2\right]^{1/2}, \qquad ||u^h||_\infty = \max_{1\le i\le n}|u_i^h|. \tag{3.43}$$

According to the Richardson's Ansatz (as discussed above, the solution for a centered FDA can be expanded in even powers of $h$. Here the ansatz is generalized to a $O(h^p)$ discretization scheme where $p$ is a even integer. The solutions with the different discretizations can then be expanded as a function of the three grid mesh spacings:

$$u^h = u + h^p e_p + h^{p+2} e_{p+2} + \dots \tag{3.44}$$

$$u^{2h} = u + (2h)^p e_p + (2h)^{p+2} e_{p+2} + \dots \tag{3.45}$$

$$u^{4h} = u + (4h)^p e_p + (4h)^{p+2} e_{p+2} + \dots \tag{3.46}$$

Subtracting the solutions at the *same* grid points, taking the appropriate norm and considering $E_p = ||e_p||$ the following expressions can be found:

$$||u^{4h} - u^{2h}|| = 2^p \left[ (2^p - 1)\, h^p E_p + (16 \cdot 2^p - 4)\, h^{p+2} E_{p+2} + \ldots \right], \tag{3.47}$$

$$||u^{2h} - u^{h}|| = (2^p - 1)\, h^p E_p \left[ 1 + \frac{(4 \cdot 2^p - 1)}{(2^p - 1)} h^2 \frac{E_{p+2}}{E_p} + \ldots \right]. \tag{3.48}$$

As long as $E_{p+2} < E_p$, as it tends to be the case if the continuum solution is analytic, then for $h \to 0$ the taylor expansion of one over the equation (3.48) becomes:

$$\frac{1}{||u^{2h} - u^{h}||} = \frac{1}{(2^p - 1)\, h^p E_p} \left[ 1 - \frac{(4 \cdot 2^p - 1)}{(2^p - 1)} h^2 \frac{E_{p+2}}{E_p} + \frac{(4 \cdot 2^p - 1)^2}{(2^p - 1)^2} h^4 \left( \frac{E_{p+2}}{E_p} \right)^2 + \ldots \right]. \tag{3.49}$$

After a bit of simplification the multiplication of equations (3.47) and (3.49) provide an useful formula for the convergence factor:

$$Q(t) = 2^p \left[ 1 + 3 \frac{(4 \cdot 2^p - 1)}{(2^p - 1)} h^2 \frac{E_{p+2}}{E_p} - 3 \frac{(4 \cdot 2^p - 1)^2}{(2^p - 1)^2} h^4 \left( \frac{E_{p+2}}{E_p} \right)^2 + \ldots \right]. \tag{3.50}$$

In particular for second order centered schemes, $p = 2$, $Q(t)$ becomes:

$$Q(t) = 4 + 60 h^2 \frac{E_4}{E_2} - 300 h^4 \left( \frac{E_4}{E_2} \right)^2 + \ldots \tag{3.51}$$

Therefore if the discrete solution $u^h$ of the FDA converges to the continuum solution $u$ of the PDE, the convergence factor must approach to $2^p$ (or 4 for second order schemes) as the mesh spacing $h$ tends to zero:

$$\lim_{h \to 0} Q(t) = 2^p. \tag{3.52}$$

Note that to perform a good convergence test it is desirable that the terms of order $h^2$ or higher be small in the expression for convergence factor, equation (3.50). Assuming that the terms containing the higher derivatives of $u$ becomes smaller in the Richardson's expansion (in analogy to Taylor expansion where those terms become increasingly smaller especially due to $1/n!$ factor term accompanying them), i.e. $E_{p+2} < E_p$, how close $Q(t)$ is to $2^p$ is controlled by how small $h$ is. Also, an ideal convergence test would calculate $Q(t)$ for at least three different base resolutions and compare the resulting function to see if it goes to $2^p$ quadratically as $h \to 0$, i.e. calculate $Q_h$ for $\{u^h, u^{2h}, u^{4h}\}$, $Q_{2h}$ for $\{u^{2h}, u^{4h}, u^{8h}\}$ and $Q_{4h}$ for $\{u^{4h}, u^{8h}, u^{16h}\}$ then $Q_{4h} - Q_{2h} \approx 4 \cdot (Q_{2h} - Q_h)$. This provides therefore an even stronger evidence that the solution is actually converging with the order of convergence expected from the FDA discretization scheme.

### 3.2.2 Independent Residual Evaluation

Suppose we evaluate the numerical solution from the FDA to some PDE and we find out, through the convergence factor for example, that the solution actually converges to the continuum one.

Before consider the problem as solved though, we should ask if the continuum solution, to which the numerical solution converges, is actually the solution to the PDE proposed to be solved initially. The reason that someone should take this issue seriously comes from the usual complexity involved in the algorithms to solve the FDA along with its encoding in a particular programming language. There are simply too many places in this solution process where things could go subtly and unexpectly awry. Fortunately there is an independent way of evaluating if the whole scheme is consistent with the PDE supposed to be approximated. Consider the PDE:

$$Lu - f = 0 \tag{3.53}$$

and its supposedly correct FDA:

$$L^h u^h - f^h = 0. \tag{3.54}$$

Suppose for example that the approximate discrete solution $\tilde{u}^h$ is found and showed to second order converge to a continuum solution $\bar{u}$. Therefore, by Richardson's Ansatz, the solution error comes from the expansion:

$$\tilde{u}^h = \bar{u} + h^2 e_2 + O(h^4), \tag{3.55}$$

where $\tilde{u}^h$ actually satisfies the residual equation to an arbitrary precision $\epsilon$:

$$L^h \tilde{u}^h - f^h = r^h < \epsilon \tag{3.56}$$

and it is the correct form as far as convergence is concerned in this example. What is missing to show is if $u \equiv \bar{u}$ is actually true. The missing test should aim therefore at evaluating if $L^h$ was actually implemented correctly; if it is in fact a valid approximation to $L$. In order to implement this test consider an independent, distinct discretization of the PDE:

$$\hat{L}^h \hat{u}^h - f^h = 0. \tag{3.57}$$

This new discrete operator $\hat{L}^h$ can be expanded in terms of the differential operator $L$ and the truncation error in the same fashion as $L^h$ could:

$$\hat{L}^h = L + h^2 \tau_2 + O(h^4) = L + \hat{\tau}^h. \tag{3.58}$$

The independent residual evaluation consists then in inserting the approximate solution of the original discrete system $\tilde{u}^h$ into this new discretization scheme. As a result the new residual $I^h$ can be easily read off from this operation as it is evident in the following expression:

$$I^h \equiv \hat{L}^h \tilde{u}^h - f^h = \left[L + h^2\tau_2 + O(h^4)\right]\left[\bar{u} + h^2 e_2 + O(h^4)\right] - f^h = L\bar{u} - f^h + O(h^2). \tag{3.59}$$

If the continuum solution is not a solution of the original PDE, i.e $\bar{u} = u + e_0$ then the independent residual converges to a number different from zero as $h \to 0$ and takes the following

form:

$$I^h \equiv \hat{L}^h \tilde{u}^h - f^h = Lu - f^h + Le_0 + O(h^2) = Le_0 + O(h^2) = O(1). \tag{3.60}$$

Unless $Le_0 = 0$, an extremely improbable event despite possible, the $O(1)$ convergence is an indication that something went wrong in the original discretization or through the process of obtaining the discrete solution. On the other hand, if $\bar{u} = u$ then the independent residual becomes:

$$I^h \equiv \hat{L}^h \tilde{u}^h - f^h = Lu - f^h + O(h^2) = O(h^2), \tag{3.61}$$

which means that $I^h$ goes to zero quadratically or with the same rate as the truncation error $\hat{\tau}^h$ of the new approximation scheme $\hat{L}^h$. This is then a strong indication that the approximate discrete solution $\tilde{u}^h$ is actually converging to the true continuum solution $u$ of the PDE.

The choice of the operator $\hat{L}^h$ for the independent residual evaluation is arbitrary. It is not necessary that the scheme be stable since it is not used to obtain the discrete solution but only as a posteriori check. The order of the operator doesn't need to match the original one either. For example the independent residual discretization operator for the time derivatives used in the evolution equations (Klein-Gordon equations) were chosen to be first order forward in time while the Crank-Nicholson was the discretization scheme used to get the numerical solution. So for the time derivatives the discretization was:

$$\hat{L}^h u^h \equiv \frac{u^{n+1}_{ijk} - u^n_{ijk}}{\lambda h} = \frac{\partial u(t,x,y,z)}{\partial t} + O(h). \tag{3.62}$$

On the other hand all spatial differential operators entering both the evolution and constraint equations were replaced by second order forward discrete operators. So for example a first order derivative along the $x$ coordinate would have the following independent discretization:

$$\hat{L}^h u^h \equiv \frac{-3u^n_{ijk} + 4u^n_{i+1\,jk} - u^n_{i+2\,jk}}{2h} = \frac{\partial u(t,x,y,z)}{\partial x} + O(h^2), \tag{3.63}$$

in contrast with a centered scheme used to solve the original discrete set of equations. The exception to this rule comes from a region of the grid domain where the discrete forward operators refers to grid points outside the domain of integration. In this case, the appropriate backward difference operator replaces the forward one. Chap. 4 presents the results of several different simulations for distinct sets of initial data and analyses the correctness of the results through the independent residual evaluation and convergence factor.

## 3.3 Multigrid Techniques

### 3.3.1 Introduction

Iterative techniques[3] such as the point-wise Newton-Gauss-Seidel (NGS) one are quite efficient to solve systems of non-linear algebraic equations resulting from the discretization of hyperbolic or parabolic PDEs. However, when applied to the FDAs originated from the discretization of elliptic-like PDEs their efficiency is only marginal. It is possible to show [132] that NGS convergence deteriorate as the mesh spacing tends to zero. For example, the Poisson equation in one spatial dimension:

$$u_{xx} = f(x) \tag{3.64}$$

is finite differenced by a centered $O(h^2)$ discretization scheme:

$$\frac{u_{i-1} - 2u_i + u_{i+1}}{h^2} - f_i = 0 \tag{3.65}$$

and has the $i$-th component of its running residual defined according to equation (D.13) by:

$$r_{ii}^{(k)} = \frac{\tilde{u}_{i-1}^{(k)} - 2\tilde{u}_i^{(k-1)} + \tilde{u}_{i+1}^{(k-1)}}{h^2} - f_i. \tag{3.66}$$

Each iteration is supposed to bring the approximate solution $\tilde{u}_i^{(k)}$ closer to the exact discrete solution $u_i$. The convergence of this process can also be viewed in terms of the reduction of the running residual. The effect of the iteration on the running residual vector is usually described by the residual amplification matrix $\mathbf{A}$, transforming the residual at iteration $k$ to the one at iteration $k+1$:

$$\mathbf{r}^{(k+1)} = \mathbf{A}\mathbf{r}^{(k)}. \tag{3.67}$$

An ideal iterative method would have the spectral radius of the amplification matrix close to zero, i.e. $\rho(\mathbf{A}) \ll 1$. Unfortunately the spectral radius for NGS amplification matrix $A_{NGS}$ when applied to the Poisson equation above turns out to be dependent on the number of unknowns of the system or equivalently to the mesh spacing $h$. Its dependence on $h$ have the following general form:

$$\rho(\mathbf{A}_{NGS}) = 1 - O(h^2). \tag{3.68}$$

This means that as the resolution increases, or, in other words, as the mesh spacing goes to zero, the amplification matrix spectral radius approaches unity, slowing considerably the rate by which the residuals are reduced to zero. The same sort of poor convergence appears virtually for all elliptic operators in spatial dimensions other than one, i.e. $d = 1, 2, 3$. This general behaviour of the NGS iterative method applied to FDAs originated from elliptic equations practically rule out

[3] See App. D for a short discussion.

its applicability as an equation solver.

On the other hand, expanding the residual vector in terms of the eigenvectors of the amplification matrix is possible to show that the asymptotic convergence is dominated by the eigenvectors with long wavelength associated to them. In other words, as the amplification matrix is applied to the residual vector its short wavelength components (or high frequency) are rapidly suppressed while its long wavelengths (low frequency) slowly converge to zero. This *smoothing* behaviour is typical of the NGS iteration when applied to elliptic-type FDAs. In this context the NGS iteration belongs to a class of methods known as *relaxation methods* where their prime feature is the ability to smooth both the solution and residual errors.

Fortunately there is an extremely efficient algorithm for solving the FDAs introduced by elliptic equations. The *Multigrid Methods* [137] are known to converge to the solution with $O(N)$ of computational work and storage (where $N$ is the total number of unknowns) not only for particular cases but for a whole class of problems. Furthermore its convergence rate is independent of the mesh spacing $h$.

The basic ideas behind the multigrid techniques reside on two simple principles. The first one consists in using relaxation methods to smooth both the solution and residual errors on a particular grid. The idea is not to try to solve the system of equations but only to smooth its errors. Therefore a few iterations should be enough for this task. Note that a smooth function on a finer grid can be well represented on a coarser grid. This fact gives rise to the second principle. This consists on transferring the problem of reducing the errors, or equivalently the solution convergence, from a finer grid to a coarser one. The slowly converging long wavelength components of the error on a fine grid have their wavelengths shortened when represented on a coarser grid. Applying the smoothing principle on the coarser grid then effectively and cheaply reduces these longer wavelength error components. The multigrid uses these two principles on a hierarchy of grids, where the coarser grids are used to quickly eliminate the smooth components of the fine grid errors, resulting therefore in a fast rate of convergence. Next subsection expose these ideas in more details for a particular multigrid technique specialized on the solution of non-linear problems, the so called *FAS* algorithm.

### 3.3.2 FAS Algorithm

Let a non-linear system of elliptic PDEs be written as:

$$N[u] = f, \tag{3.69}$$

where $N$ represents the set of non-linear differential operators acting on the solution vector $u$ and $f$ is a source function independent of $u$ or its derivatives. Their finite difference approximation is parametrized by the grid spacing $h$ and is then expressed as:

$$N^h[u^h] = f^h, \tag{3.70}$$

whose solution is expected to be built through some iterative process. Each step of the iteration defines the residual vector $r^h$ corresponding to the approximate solution $\tilde{u}^h$:

$$r^h \equiv N^h[\tilde{u}^h] - f^h. \tag{3.71}$$

The goal of the iterative process is to drive the residual to zero or equivalently to reduce to zero the error $v^h$ of the approximate solution relative to the exact discrete solution $u^h$, where $v^h$ is defined by the following expression:

$$u^h = \tilde{u}^h + v^h. \tag{3.72}$$

As mentioned in the last section, relaxation methods such as NGS iterative technique are excellent smoothers. The multigrid method takes advantage of this fact to smooth the correction $v^h$ on each grid of a hierarchy of grids. The hierarchy is usually composed such that $h_{l+1} = \frac{1}{2}h_l$, where $l$ labels the grids in the hierarchy ($l = 1$ corresponding to the coarsest and $l = l_{max}$ to the finest grid). After a few iterations of NGS relaxation, the problem of finding $u^h$ is then transfered to a coarser grid. Note that in order to pose a *coarse grid correction* the residual and error on the fine grid needs to be smooth and thus have an accurate representation on a grid with mesh spacing $2h$.

In order to define the coarse grid correction, first note that one could naively claim that applying the operator $N^h$ to equation (3.72) would result in:

$$N^h[u^h] = N^h[\tilde{u}^h + v^h] = N^h[\tilde{u}^h] + N^h[v^h] = r^h + f^h + N^h[v^h], \tag{3.73}$$

that could be simplified to:

$$N^h[v^h] = -r^h. \tag{3.74}$$

This result is true, and actually it is the basis of a multigrid version called *linear correction scheme*, when the operator $N^h$ is *linear*, since $N^h[\tilde{u}^h+v^h] = N^h[\tilde{u}^h]+N^h[v^h]$ takes into account its linearity. Nonetheless $N^h$ is by hypothesis a non-linear operator, what prevents this naive but convenient analysis to proceed in this case. On the other hand, the equation (3.74) suggests a similar treatment for the non-linear case. Subtracting the residual equation (3.71) from the FDA equation (3.70) results into:

$$N^h[u^h] - N^h[\tilde{u}^h] = N^h[\tilde{u}^h + v^h] - N^h[\tilde{u}^h] = -r^h. \tag{3.75}$$

Therefore instead of ending up with an equation for the correction $v^h$ only, equation (3.74), equation (3.75) poses the problem to be solved for a full approximation $u^h = \tilde{u}^h + v^h$, hence the denomination of the method as "Full Approximation Scheme" or FAS algorithm.

The idea of the coarse grid correction is to transfer all smoothed grid functions from the fine grid to the coarser grid, pose and solve the following coarse problem in analogy to the equation (3.75):

$$N^{2h}[u^{2h}] - N^{2h}[I_h^{2h}\tilde{u}^h] = -I_h^{2h}r^h, \tag{3.76}$$

where $I_h^{2h}$ is an appropriate transfer operator or in this case a restriction operator responsible to

set the values of the residual and approximate solution in the coarse grid. The equation (3.76) should then be solved for the unknown $u^{2h}$. After that the fine grid unknown is then updated by:

$$\tilde{u}^h = \tilde{u}^h + I_{2h}^h \left( u^{2h} - I_h^{2h} \tilde{u}^h \right), \tag{3.77}$$

where $I_{2h}^h$ is the prolongation operator responsible to interpolate the grid functions from the coarse grid into the finer one.

Equations (3.76) and (3.77) corresponds to the basic equations employed by the FAS algorithm. The point of view presented so far also corresponds to the canonical point of view of a multigrid solver as a solver that uses a hierarchy of coarser grids to accelerate the convergence of long wavelengths error components defined on a finer grid. Nevertheless the multigrid process can also be viewed as the fine grids providing high accurate corrections to the coarse grid equations in order to accelerate their convergence to the continuum PDEs. This *dual* point of view generates an algorithm equivalent to the one shown above however it relies on the concept of truncation error instead of residual. The idea is as follows:

The notion of truncation error, according to the definition by equation (3.18), arises from the replacement of the continuum operator $N$ acting on the continuum solution $u$ by the discrete operator $N^h$ acting on the continuum solution. The truncation error is simply the error associated with this approximation:

$$\tau^h \equiv N^h[u] - f^h, \tag{3.78}$$

that can be written in a more convenient form for this subsection by subtracting the FDA (3.70) from the equation (3.78):

$$\tau^h = N^h[u] - N^h[u^h]. \tag{3.79}$$

If it was possible to know the exact value of the truncation error $\tau^h$ in advance and that quantity was added to the right hand side of the FDA (3.70), i.e.:

$$N^h[u^h] = f^h + \tau^h, \tag{3.80}$$

then the discrete solution $u^h$ of the system above would be identical to the continuum solution $u$ of the PDE (3.69) when restricted to the mesh points. Unfortunately the exact knowledge of $\tau^h$ requires the knowledge of $u$ that it is actually the solution being sought. However the message here is that adding some sort of approximation $\tilde{\tau}^h$ of the truncation error to the right-hand side of equation (3.70) actually results in a better approximation of the discrete system:

$$N^h[u_\star^h] = f^h + \tilde{\tau}^h, \tag{3.81}$$

where the solution error is improved: $u_\star^h - u < u^h - u$.

Fortunately the multigrid hierarchy provides an excellent framework for estimating the approximate truncation error. First note that analogously to the definition of truncation error, the *relative* truncation error $\tau_h^{2h}$ is the error associated by the replacement of $N^h$, the operator on grid functions

with mesh spacing $h$, by the ones acting on grids with mesh spacing $2h$, $N^{2h}$. Then in analogy to equation (3.79), the relative truncation error is defined as:

$$\tau_h^{2h} \equiv N^{2h}[I_h^{2h} u^h] - I_h^{2h}\left(N^h[u^h]\right), \tag{3.82}$$

that can be promptly rewritten as:

$$N^{2h}[u^{2h}] = f^{2h} + \tau_h^{2h}. \tag{3.83}$$

Therefore, in analogy with the role played by $\tau^h$, $\tau_h^{2h}$ is interpreted as a correction of the source term on the coarse grid in order to make the coarse solution to coincide with the fine solution. After the coarse grid correction equations (3.83) being smoothed by NGS iterations the solution correction is then updated on the fine grid by equation (3.77).

The FAS algorithm posits the coarse grid correction recursively in a alternating sequence of smoothing and transference to coarser grid operations. Eventually the coarsest grid is reached. At this point it is computationally cheap to obtain the exact discrete solution and that is usually achieved by NGS iterations with a more stringent tolerance. Once the coarsest grid solution is evaluated, then another alternating sequence of smoothing, interpolation and update operations is performed until the solution on the finest grid is updated. This cycle is known as $V$-cycle in analogy with going down the $V$ leg to evaluate the coarse grid corrections and coming back up the $V$ leg updating the solutions. Its pseudo-code is given by Fig. (3.4).

The elliptic equations appearing in this thesis were discretized using a second order centered scheme and their non-linear FDAs were solved by the FAS algorithm just described above. The sources of the elliptic equations were the Klein-Gordon complex scalar field and its conjugate momenta, while the elliptic equations were solved for the geometric variables. As a last technical comment, the ordering applied by the NGS iterations was the *red-black* ordering and the transfers operators used were *full-weighted* restriction operator and *tri-linear* interpolation operator as prolongation. The reader should refer to one of the multigrid references cited on the beginning of this section for further details.

```
subroutine MG_update()
  cycle = 1
  do while (residual > tolerance)
     call vcycle(lmax,cycle,p,q)
     cycle = cycle + 1
  end do
end subroutine

subroutine vcycle(lmax,cycle,p,q)
  do l = lmax to 1 ( cycle from fine to coarse levels )
    if cycle = 1 or l not equal to lmax then
      repeat p times:
        perform a NGS relaxation sweep on level(h[l]):
          u[l] = relax_rb_NGS(u[l],f[l],h[l])

      restrict grid functions to level(h[l-1]):
          u[l-1] = I[l-1][l] u[l]

      compute the relative truncation error of the solution
        on level(h[l-1]):
        tau[l-1][l] = N[l-1] I[l-1][l] u[l] - I[l-1][l] N[l] u[l]

      compute the new RHS vector for level(h[l-1]), by adding the
        restricted RHS to the truncation error:
        f[l-1] = tau[l-1][l] + I[l-1][l] f[l]
    end if
  end do

  solve the system of FDAs on level(h[1]) exactly:
      do while (residual > coarsest grid tolerance)
          u[1] = relax_rb_NGS(u[1],f[1],h[1])
      end do

  do l = 2 to lmax ( cycle from coarse to fine levels )
     compute the CGC from level(h[l-1]) to level(h[l])
       and apply the CGC to unknown variables at level(h[l]):
       u[l] = u[l] + I[l][l-1] ( u[l-1] - I[l-1][l] u[l] )

     repeat q times:
        perform a relaxation sweep on level(h[l]):
          u[l] = relax_rb_NGS(u[l],f[l],h[l])
  end do
end subroutine
```

**Figure 3.4:** A pseudo-code representation of the FAS, $V$-cycle multigrid algorithm.

# CHAPTER 4

# CODE VALIDATION AND RESULTS

This chapter presents some results of the integration of the system of PDEs derived on Chap. 2 and overviewed on Sec. 2.5. These results are for the mini-boson stars, i.e. stars with the self-interaction potential $U(|\phi|^2) = m^2|\phi|^2$ where, without loss of generality, the mass $m$ of the constituent particle associated to the scalar field is kept constant: $m = 1$. The first part of the chapter is devoted to the validation of the numerical code. All functions involved were convergence tested and had their respective independent residual equations evaluated. The independent residual evaluation and convergence factors calculation are considered the strongest tests of self-consistency and correctness of the solution. They are tests independent of any previous assumption on the solution, aside from its smoothness, as it was detailed on Chap. 3. Besides these two tests, conservation of ADM mass and Noether charge tests were added to strength the pool of tests. After a brief description on how the unigrid (as opposed to AMR and parallel) numerical code works on Sec. 4.1, Sections 4.2-4.4 are dedicated to show the correctness of the results for 3 different types of initial data set. Despite only one set of smooth initial data profile be enough to judge about the rightness of the solution, the other sets were important to evaluate the correctness of the initial data itself. Thus Sec. 4.3 presents the results for a spherically symmetric initial data and discusses some issues related to its smoothness while Sec. 4.4 brings the results for the evolution of a boosted spherically symmetric initial data.

The second part of the chapter focus on the dynamics of two boson stars. Sec. 4.5 presents the simulation of a head-on collision between two boson stars. Sec. 4.6 discusses the orbital motion for three different initial configurations. It shows first the result of a long time evolution in which the stars orbit each other. The second result consists of an interrupted orbital motion with a final object being a rotating boson star. The section ends by presenting also an interrupted orbital motion but this time with a final object conjectured to be a black-hole. All simulations had an independent residual evaluated. Some of them were included and discussed in their respective sections. Finally, Sec. 4.7 summarizes the main results and proposes directions for further develpments of this project.

All the simulations presented here were run on single nodes of either **vnfe5** or **vnfe4** clusters located at UBC. Each node of the **vnfe5** cluster consists of four 2.4 GHz Dual-Core AMD Opteron(tm) processors with a total of 4 GB of memory. The **vnfe4** nodes, on the other hand, constitute of two 2.4 GHz Intel(R) Xeon(TM) CPUs and a total of 2 GB of memory. The most expensive calculations were run on the **vnfe5** cluster and last approximately 260 hours with use of approximately 700 MB of memory. Larger simulations, up to 1 GB, were possible but unpractical to provide reasonable results in a timescale of at least a few days and for that reason was avoided.

In order to facilitate the presentation of the numerical results in this chapter, consider the following definitions. The number of grid points of a particular simulation can be stored in a vector called *shape* and it is defined as:

$$shape = [N_x, N_y, N_z], \tag{4.1}$$

where $N_x$, $N_y$ and $N_z$ are the number of grid points per edge. Therefore this grid has $N_x$ grid points along the $x$ direction, $N_y$ grid points along the $y$ direction and $N_z$ along $z$ direction with a total of $N_x \times N_y \times N_z$ grid points. Also associated with the number of grid points is the concept of level. Let $N = \min\{N_x, N_y, N_z\}$, therefore the level $l$ associated with the grid is such that:

$$N = 2^l + 1. \tag{4.2}$$

Another convenient definition regards the range of coordinates of a particular grid. The *bounding box vector* bbox is defined such that:

$$bbox = [x_{min}, x_{max}, y_{min}, y_{max}, z_{min}, z_{max}], \tag{4.3}$$

where $\{x_{min}, x_{max}\}$ corresponds to the range of the coordinates along $x$ direction, $\{y_{min}, y_{max}\}$ along $y$ direction and $\{z_{min}, z_{max}\}$ along $z$.

As a last remark, it is worth noticing that the grid lines on the surface plots presented in this chapter do not represent the actual grid structure where the simulations were performed. They are all for visual aid only. On the same plots, unless otherwise stated, the height of the surface plots correspond to the function values.

## 4.1 Unigrid code

The set of PDEs comprising the boundary-initial-value problem described on Chap. 2 was discretized with the techniques presented on Chap. 3. To be more specific the second order accurate Crank-Nicholson discretization scheme was applied to the hyperbolic equations; those describing the time evolution of the real and imaginary components of the complex scalar field $\{\phi_1, \phi_2\}$ and their conjugate momenta $\{\Pi_1, \Pi_2\}$. Additionally, a second order accurate centred FDA was employed to discretize the set of elliptic PDEs; those originated from the slicing condition, hamiltonian and momentum constraints solving for the geometric quantities lapse function, conformal factor and shift vector respectively, $\{\alpha, \psi, \beta^x, \beta^y, \beta^z\}$. Dirichlet boundary conditions were enforced on all variables for the simulations presented in this chapter.

A numerical code was written to solve the FDAs originated from the discretization of the boundary-initial-value problem. Since part of the equations are elliptic and part are hyperbolic in nature, different numerical strategies were required. The hyperbolic FDA equations were encoded through RNPL (Rapid Numerical Prototyping Language) [138, 139] where a pointwise Newton-Gauss-Seidel iterative technique was applied. To solve the elliptic FDA equations, a numerical code was written in FORTRAN to implement the FAS multigrid algorithm as described in Chap. 3.

The flow of the numerical code is presented in Fig. 4.1 as a pseudo-code. The program starts its execution by calling the initial value solver routine. This consists of a set of subroutines used to generate a static solution of spherically symmetric boson stars. Chap. 2 describes in detail how this 1D solution is obtained and appendix A describes the user interface of its main subroutine, *bsidpa*,

written with the purpose of finding the spherically symmetric solution not only for mini-boson stars but also for boson stars with *any* self-interaction potential.

Once the one dimensional solution is obtained, the following set of spherically symmetric functions in polar-areal coordinates $\{\alpha, a, \phi, \Phi\}$ is interpolated in the maximal-isotropic 3D numerical domain according to Sections 2.6.6 and 2.6.7. If two stars are to be generated, then the initial data process is duplicated and the solution superimposed. The stars are considered far enough from each other such that the scalar field superposition happens in a region of weak field where linear superposition of solutions still holds. The interpolated—and possibly boosted— scalar field variables $\{\phi_1, \phi_2, \Pi_1, \Pi_2\}$ are then used as sources for the multigrid solver. On the other hand, the geometric variables interpolated, $\{\alpha, \psi, \beta^x\}$ (recall that the boost is only implemented in the $x$ direction), are used as initial guesses for the multigrid $V$-cycle. After convergence of the multigrid iteration, the control is given back to the main routine where it proceeds with the Crank-Nicholson iteration to update the scalar field variables. The role of sources and unknown variables are reversed and the pointwise NGS step applied to the evolution equations. This cycle repeats itself until the norm of the scalar field residuals is driven below an arbitrarily small tolerance ($10^{-7}$ in the simulations presented here). At this point the time step is then updated and the solution analysed (independent residual evaluated, mass and noether charges calculated) and written to disk. The whole time step is repeated until a maximum value for time $t_{max}$ is reached. Fig. 4.1 synthesizes the flow of the code explained in this section.

```
f(t) = scalar field variables at time t
g(t) = geometry variables at time t


t = 0
call  IVP_solver[f(t),g(t)]

do while (t < tmax)

   call  initial_guess[f(t) -> f(t+dt), g(t) -> g(t+dt)]

   do while (residual[f(t+dt)] > tolerance)

      call  MG_update[source = f(t+dt); solve for g(t+dt)]

      call  CN_update[source = g(t),g(t+dt),f(t); solve for f(t+dt)]

      call  evaluate_residual[f(t+dt)]

   end do

   call  swap_levels[f(t) <--> f(t+dt); g(t) <--> g(t+dt)]

   t = t + dt

   call  analysis[g(t),g(t+dt),f(t),f(t+dt)]

end do
```

**Figure 4.1:** A pseudo-code representation of the unigrid code.

## 4.2 Generic Initial Data

The easiest way to test for convergence and consistency of the numerical solution is to assign a smooth, generic and closed form initial data and evaluate the results of its time evolution. The initial configuration presented in this section was modelled by a gaussian lump of scalar field given by the following formula:

$$\phi(x,y,z) = \phi_0 \exp\left[-\left(\frac{x-x_0}{\delta_x}\right)^2 - \left(\frac{y-y_0}{\delta_y}\right)^2 - \left(\frac{z-z_0}{\delta_z}\right)^2\right]. \tag{4.4}$$

The freely specifiable variables $\{\phi_1, \phi_2, \Pi_1, \Pi_2\}$ were initialized as follows. $\phi_1$ was assigned the formula (4.4) with $(x_0, y_0, z_0) = (0, 0, 0)$ (centered at the origin), a central amplitude $\phi_0 = 0.05$ and a slightly prolate gaussian width of $(\delta_x, \delta_y, \delta_z) = (4.5, 4, 4)$ (in order induce some non-trivial dynamics). In order to try to mimic a spherically symmetric star (at least a perturbed version) as much as possible with this initial data $\Pi_2$ was assigned such that $\Pi_2 = -3\phi_1$ while $\phi_2 = \Pi_1 = 0$.

The bounding box of this simulation was $bbox = [-15, 15, -15, 15, -15, 15]$. Three different grids with resolutions on the proportion 2 : 1 were used for the convergence tests and independent residual evaluations. The grid with the finest resolution, level 7, had a shape of $shape = [129, 129, 129]$, while for the coarsest $shape = [33, 33, 33]$. The courant factor adopted here and for all simulations showed in this chapter was $\lambda = 0.4$.

Fig. 4.2 shows a short period of the evolution of this initial data configuration. Despite the trial to mimic static stars, the resemblance is minimum. The solution oscillation probably reflects the fact that a gaussian profile can also be represented by a superposition of the excited boson star solutions, each with a particular mode of oscillation. It is not the goal here to determine this series though. What it is important to note here is the fact that the solution remains considerably smooth during this evolution. As a result the convergence factor $Q$ for various quantities remain close to its theoretical value, 4, as it is expected for a second order accurate discretization scheme according to equation (3.51). Left panels of Figures 4.3 to 4.5 show the convergence factor as a function of time for all variables. Their right panels, on the other hand, show the $l2$-norm of the independent residual equations, $I$, for all variables as a function of time. The geometry variables $\{\alpha, \psi, \beta^x, \beta^y, \beta^z\}$ had their respective PDE equations discretized with second order accurate forward difference operators (note that this is a different set of FDAs from the ones used to obtain the solution) and for this reason it is expected to have $I$ converging to zero as $O(h^2)$. An eye ball inspection of the plots shows evidence of this convergence. Furthermore the shape of the various functions $I(t)$ does not change much as the resolution is increased. This observation support the statement that the independent residual goes as:

$$I^h = I_2(t)h^2 \qquad \text{and not as} \qquad I^h = I_2(t,h)h^2. \tag{4.5}$$

The same is true for the scalar field variables $\{\phi_1, \phi_2, \Pi_1, \Pi_2\}$, except that the convergence in this case is linear, $O(h)$, since the independent discretization of the time derivatives was a first order forward stencil.

Fig. 4.6 shows the ADM mass $M_{ADM}$ and the noether charge $Q_N$ as a function of time for

levels 5, 6 and 7. As the resolution increases, the functions tends to constant functions of time resulting therefore in the conservation of each quantity as a function of time. It is estimated from the plot that the rate of convergence is also $O(h^2)$.

In the approximation of the differential operator to the finite difference, the error associated is composed of two main parts. First, there is a power of $h$, the discretization parameter or mesh spacing, characterizing the order of convergence of the approximation. For second order schemes, for example, the error is a function of $h^2$. Second, multiplying the factor $h$, there is a continuum function of higher derivative of the solution being sought. For example, a second order approximation of the first order derivative of the function $u$ can be written as:

$$\frac{u_{i+1} - u_{i-1}}{2h} \equiv \frac{u(x+h) - u(x-h)}{2h} = u'(x) + \frac{h^2}{12}u'''(\xi), \tag{4.6}$$

where $\xi \in [x-h, x+h]$ and $h^2/12u'''(\xi)$ represents the error of the approximation. Note however that this error term is applicable *only* if $u'''$ exists and it is continuum in the approximation interval. This remark emphasizes the care that the reader has to take while interpreting the results from the independent residual evaluations as well as from the convergence factors. Most of the times such results will be only an approximation to its theoretical prediction. That is understandable in terms of the smoothness of the solution function calculated numerically $u^h$. Such a solution may be approximating a continuum solution that do not have a continuum third derivative for example and the error formula is not applicable anymore. The natural way to overcome this problem is to increase the number of grid points of the domain of computation where the function is represented as well as to push the boundaries (specially when truncated boundary conditions are applied [1].) away from the region where the solution changes considerably when compared to the mesh spacing $h$. That may be an unsurmountable task for 3D domains where limitations on the computer physical memory constrains to a reasonably low number of grid points. This caveat should be taken into account when interpreting the code validation results, especially for unigrid code tests. Next section an example will be worked out to show how small non-smoothness of the solution can destroy the results of these tests.

[1] See Sec. 2.4.5

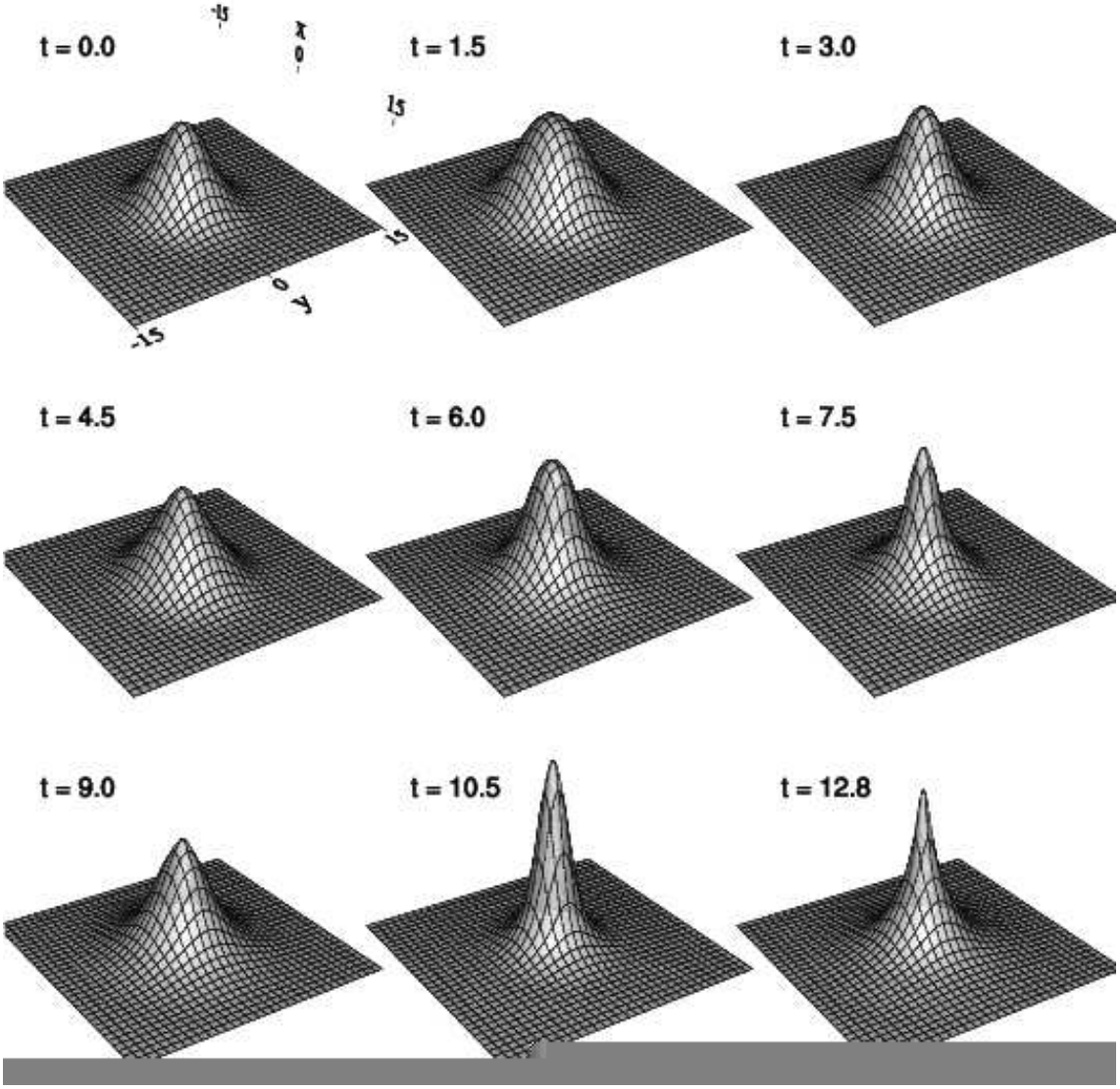


**Figure 4.2:** Evolution of a Gaussian Lump as Initial Data. These plots are snaphots of the modulus of the scalar field $|\phi(t, x, y, z)|$ for a short time evolution of the coupled system of elliptic-hyperbolic equations describing the Einstein-Klein-Gordon system under Conformally Flat Condition. The real component of the scalar field initial data is modelled by a gaussian function with a central value field $\phi_0 = 0.05$. The evolution shows an oscillatory solution with an increase in the scalar field amplitude. Note, however, that the amplitude is bounded at this time interval: $0 \leq |\phi(t, x, y, z)| \leq 0.117$.

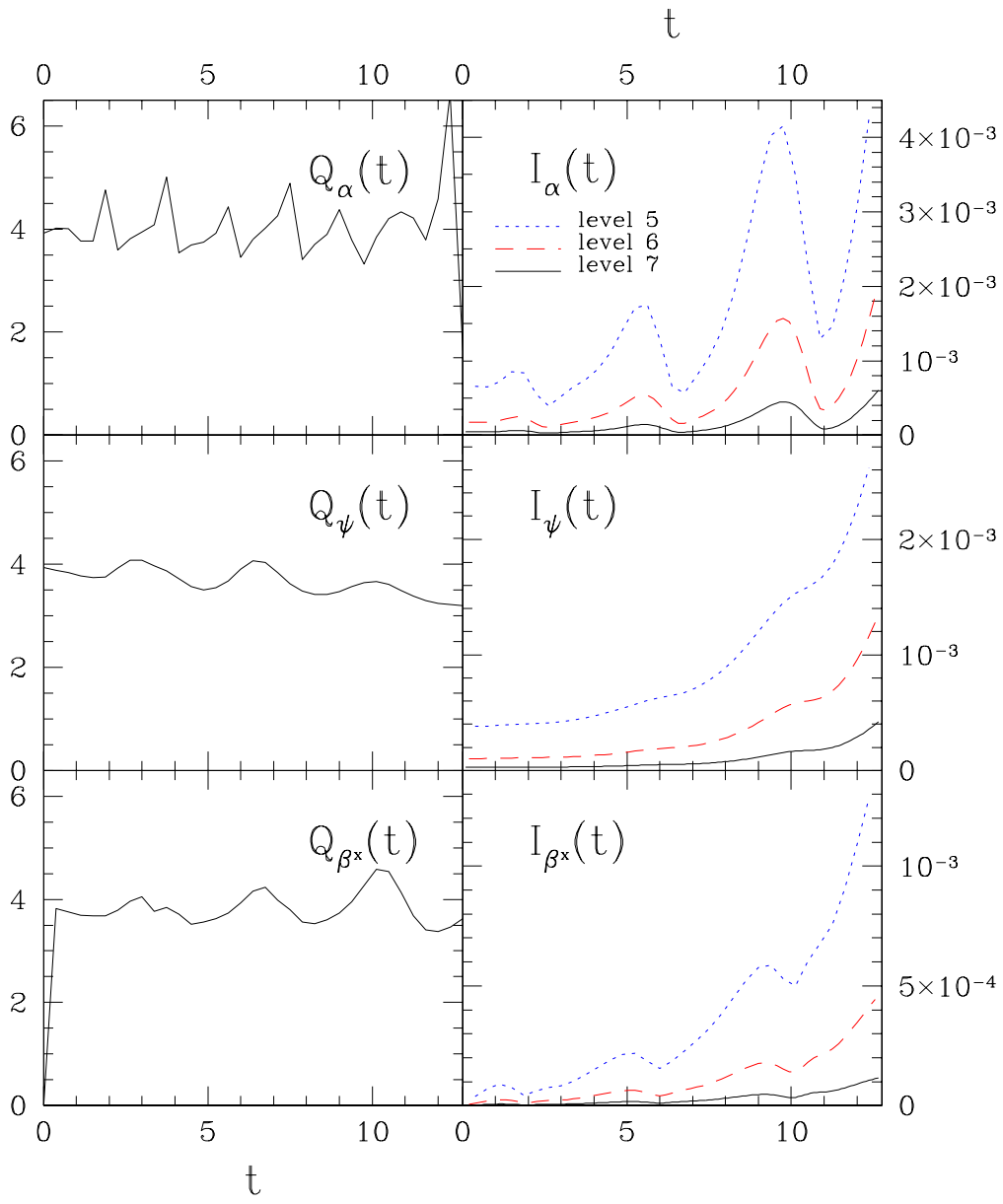


**Figure 4.3:** Q and I for $\alpha$, $\psi$ and $\beta^x$. The left panels show the plots of the convergence factor Q as a function of time t for the lapse function $\alpha$, the conformal factor $\psi$ and the $x$ component of the shift vector $\beta^x$. Note that for $\beta^x$ the first data point is set to zero since $\beta^x(0, x, y, z) = 0$. The right panels show the $L_2$-norm I of the independent residual equations for the same set of variables as a function of time.

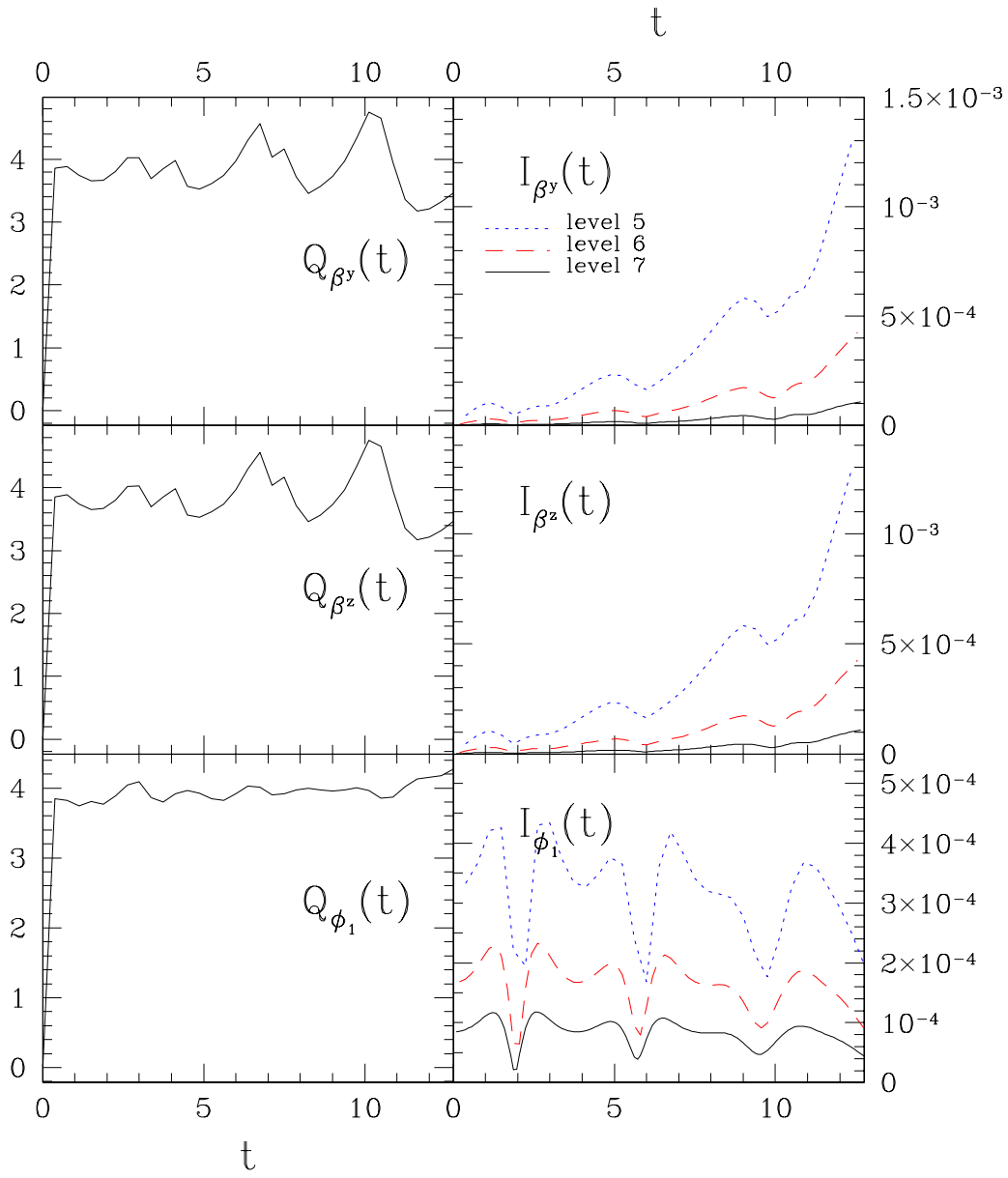


**Figure 4.4:** Q and I for $\beta^y$, $\beta^z$ and $\phi_1$. The left panels show the plots of the convergence factor Q as a function of time t for the $y$ and $z$ shift components, $\beta^y$ and $\beta^z$, and the real component of the complex scalar field, $\phi_1$. The right panels show the $L_2$-norm I of the independent residual equations for the same set of variables as a function of time.

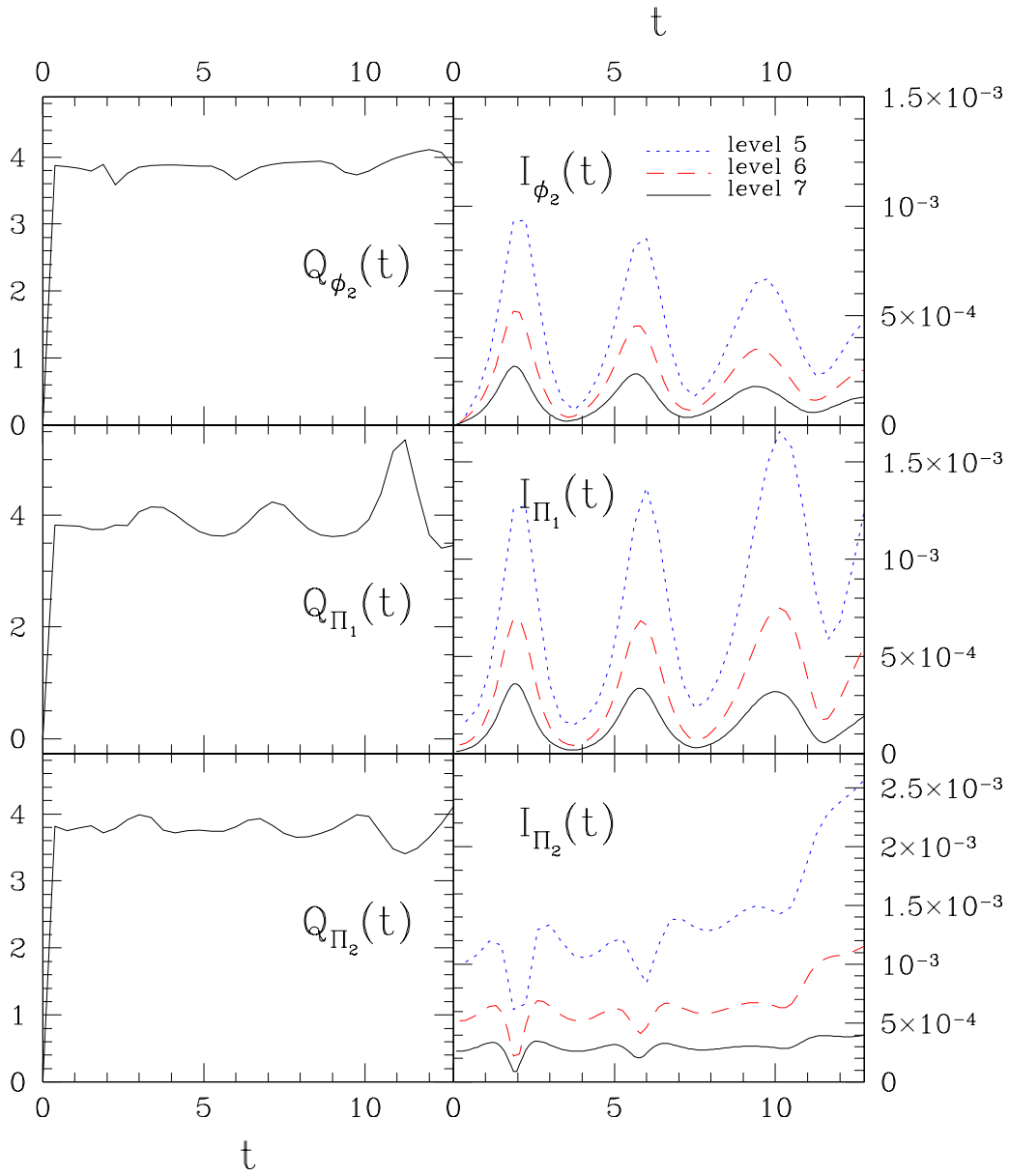


**Figure 4.5:** Q and I for $\phi_2$, $\Pi_1$ and $\Pi_2$. The left panels show the plots of the convergence factor Q as a function of time t for imaginary component of the complex scalar field $\phi_2$ and the real and imaginary components of its conjugate field, $\Pi_1$ and $\Pi_2$. The right panels show the $L_2$-norm I of the independent residual equations for the same set of variables as a function of time.

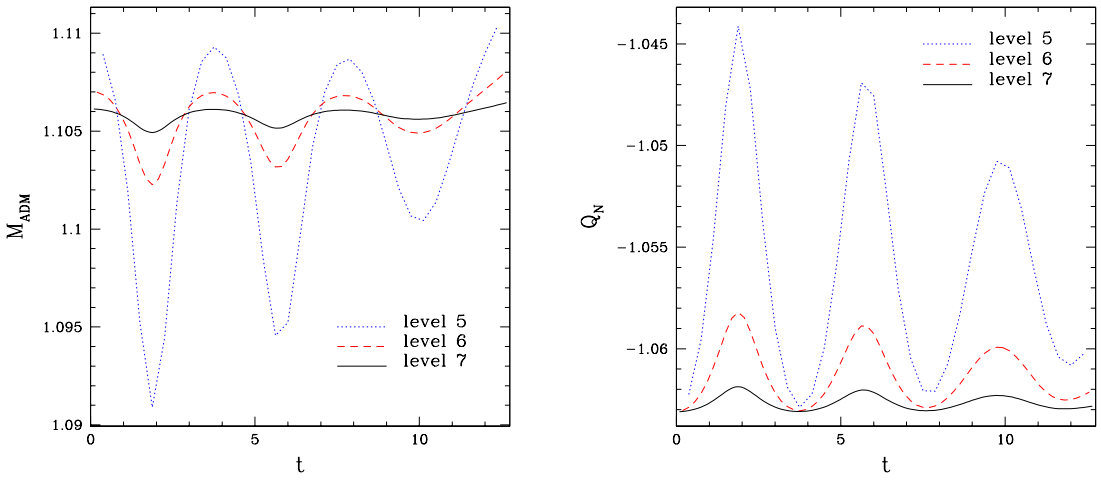


**Figure 4.6:** ADM mass $M_{ADM}$ and Noether charge $Q_N$ for Gaussian initial data as a function of time t. Both plots indicate a trend to conservation of the quantity with a convergence rate of $O(h^2)$.

## 4.3 Static Spherically Symmetric Initial Data - 1 star

Last section used a generic type of initial data to test for the correctness of the numerical code. The convergence tests and independent residual evaluations there were quite satisfatory in assuring that the equations of motion were implemented correctly and that the numerical solution obtained converged to the continuum solution. Those are already strong enough tests of the validity of the numerical code. This section continues the battery of tests though with a slightly different goal. The equations of motion is already known to be correctly implemented. Any possible deviation found here or elsewhere should be related then to the initial data itself or the resolution in which the code was run for that particular initial data set. The testing idea behind this section is therefore twofold. First, the static spherically symmetric initial data, calculated according to the ansatz for the complex scalar field (see Subsection 2.6.4):

$$\phi(t, r) = \phi_0(r)e^{-i\omega t}, \tag{4.7}$$

has not been tested yet in the evolution context. It is expected then that such initial data sets do not show any dynamics as the time evolution occurs. The second goal of these experiments is to compare the results between two static stars with different total masses. The idea is to clarify if a strong gravitational field initial data would lead to poorer results as opposed to weak field ones when all the other grid parameters are kept the same.

In order to pursue the goals briefly described on last paragraph, two members of the family of static spherically symmetric boson stars are used as initial data in this section. Both belong to the stable branch of the family (see Fig. 2.2) and have as the central value for the scalar field $\phi_0 \equiv \phi_0(0) = 0.03$ and $\phi_0 \equiv \phi_0(0) = 0.06$ respectively. The scalar field configuration with $\phi_0 = 0.03$

is less massive and thus less compact than the one parametrized by $\phi_0 = 0.06$, as can be inferred from Fig. 2.1. The grid parameters used in the evolution of both fields are the same as the ones used for the generic initial data, in Sec. 4.2. So, $bbox = [-15, 15, -15, 15, -15, 15]$ and the finest $shape = [129, 129, 129]$.

Figures 4.7 and 4.8 show some snapshots of the time evolution of the initial data parametrized by $\phi_0 = 0.03$ and $\phi_0 = 0.06$ respectively. A naked eye inspection of both figures confirm that both evolutions are time independent. As a conclusion, the set of initial data seems to be static as expected and the first goal of this experiment seems to be fulfilled.

On the other hand, as Figures 4.9, 4.10 and 4.11 can attest, the convergence factor $Q$ for each of the variables of the set $\{\alpha, \psi, \beta^x, \beta^y, \beta^z, \phi_1, \phi_2, \Pi_1, \Pi_2\}$ fails to approach its predicted theoretical value of 4 when the $\phi_0 = 0.03$ initial data set is used in the evolution. That is a strong evidence that there is something wrong with this initial data set (since the possibility of a wrong implementation of the equations of motion has already been discarded with the use of a generic initial data). The star configuration with $\phi_0 = 0.06$ shows a reasonably clean convergence for the geometric variables and a not so immaculate one, but with a strong trend to, in the case of the scalar fields (Note that there is a large glitch for $Q_{\Pi_1}$ for $t \leq 2$ in Fig. 4.11).

Naturally a non-convergent solution does not satisfies its independent residual equation. That can be testified on Figures 4.12, 4.13 and 4.13 where plots of the $l2$-norm of the independent residual equations, $I$, for all variables as a function of time are displayed for three different levels of resolution. Note that on the left panels are the plots for $\phi_0 = 0.06$ configuration while on the right for $\phi_0 = 0.03$ one. The shift vector equations and the scalar field conjugate momenta equations for $\phi_0 = 0.03$ case are affected the most and show the poorest independent residual results. In the case of $\phi_0 = 0.06$ basically all independent residual $l2$-norms are driven to zero as the resolution increases. It is worth noting though that $I_{\Pi_1}$ is on the verge of failing this test.

Regarding the mass and noether charge conservation displayed on Fig. 4.15, both cases seems to have them conserved. However it is clear from the top left panel that the convergence for the constant mass is not even close to $O(h^2)$ rate.

It would be natural to suppose that when one of the spherically symmetric star solutions fails to converge or to be consistent to the continuum solution then this failure would be associated with the most relativistic, compact configuration. Surprisingly, the results discussed so far reveal exactly the opposite. In both configuration cases then there is enough points to represent the stars without a great loss of accuracy, discarding thus the hypothesis that huge gradients on the most relativistic solution function would be spoiling its smoothness on the grids shown here. What is left of the analysis so far is the boundary conditions. As discussed on Subsection 2.4.5 the boundary conditions applied on these simulations were of Dirichlet type. That can be a pretty good approximation of the physical boundary conditions at the spatial infinity as long as they are kept far enough from the compact object being modelled. Clearly that it is not the case for the configuration $\phi_0 = 0.03$.

A closer look on the frame $t = 3.0$ of Fig. 4.7, as showed in Fig. 4.16, reveals a discontinous wave of scalar matter originated at the boundaries of the numerical domain and propagating inwards. The jump in the solution is estimated to be of the order of 1% of its central value $\phi_0$. This apparently

small spurious boundary effect has drastic effects on the solution convergence. Both the convergence factor and the independent residual assume that the continuum solution of a system of PDEs is smooth. That is the uttermost hypothesis behind the finite difference approximation of partial differential equations. The star profile with $\phi_0 = 0.03$ (see Figures 2.1 and 2.2) has a long tail that eventually had to be truncated to fit on the numerical domains of the present simulations. That discontinuity between the last two grid points of the domain spoils the smoothness of the initial data. As the solution evolves, this discontinuity propagates over the whole domain of integration ruining the smoothness of the whole solution and causing the convergence tests to fail.

A look on Fig. 4.17, on the other hand, shows that this boundary discontinuity is much smaller than any length scale of the problem. That means that whatever the discontinuity is its effects on the system of FDAs are smaller than or of the order of the truncation error associated or in simpler words it goes unnoticeable as far as the solution is concerned. $\phi_0 = 0.06$ is an example then of a smooth initial data set where the approximation of the boundary conditions to Dirichlet type is considered satisfactory for the current grid and coordinate bounding box set up.

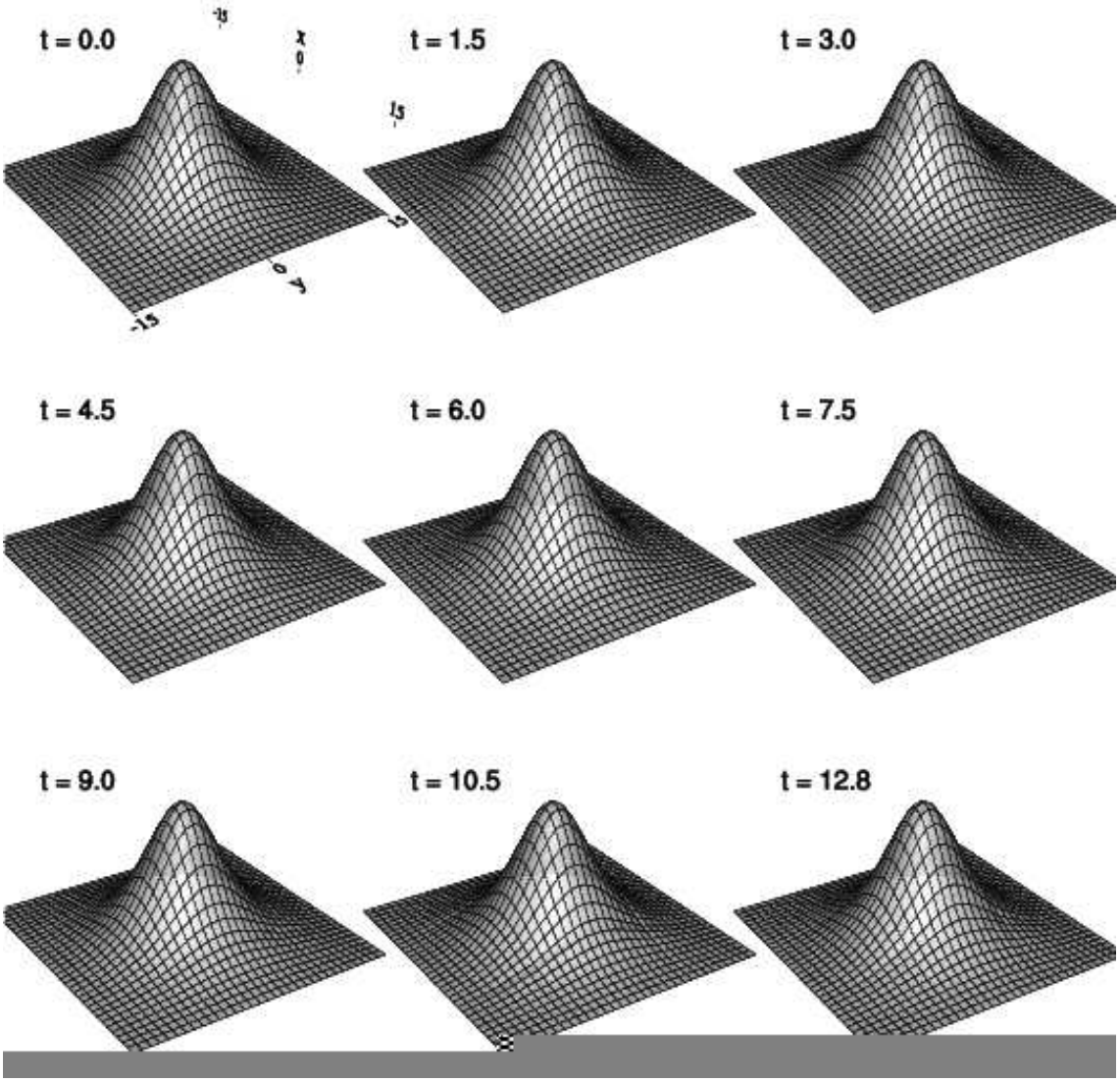


**Figure 4.7:** Evolution of a $\phi_0 = 0.03$ Spherically Symmetric Star as Initial Data. This plot shows some snapshots of the time evolution of a spherically symmetric star parametrized by a central value field of $\phi_0 = 0.03$. The evolution does not reveal any apparent dynamics as it is expected of a static initial data set.

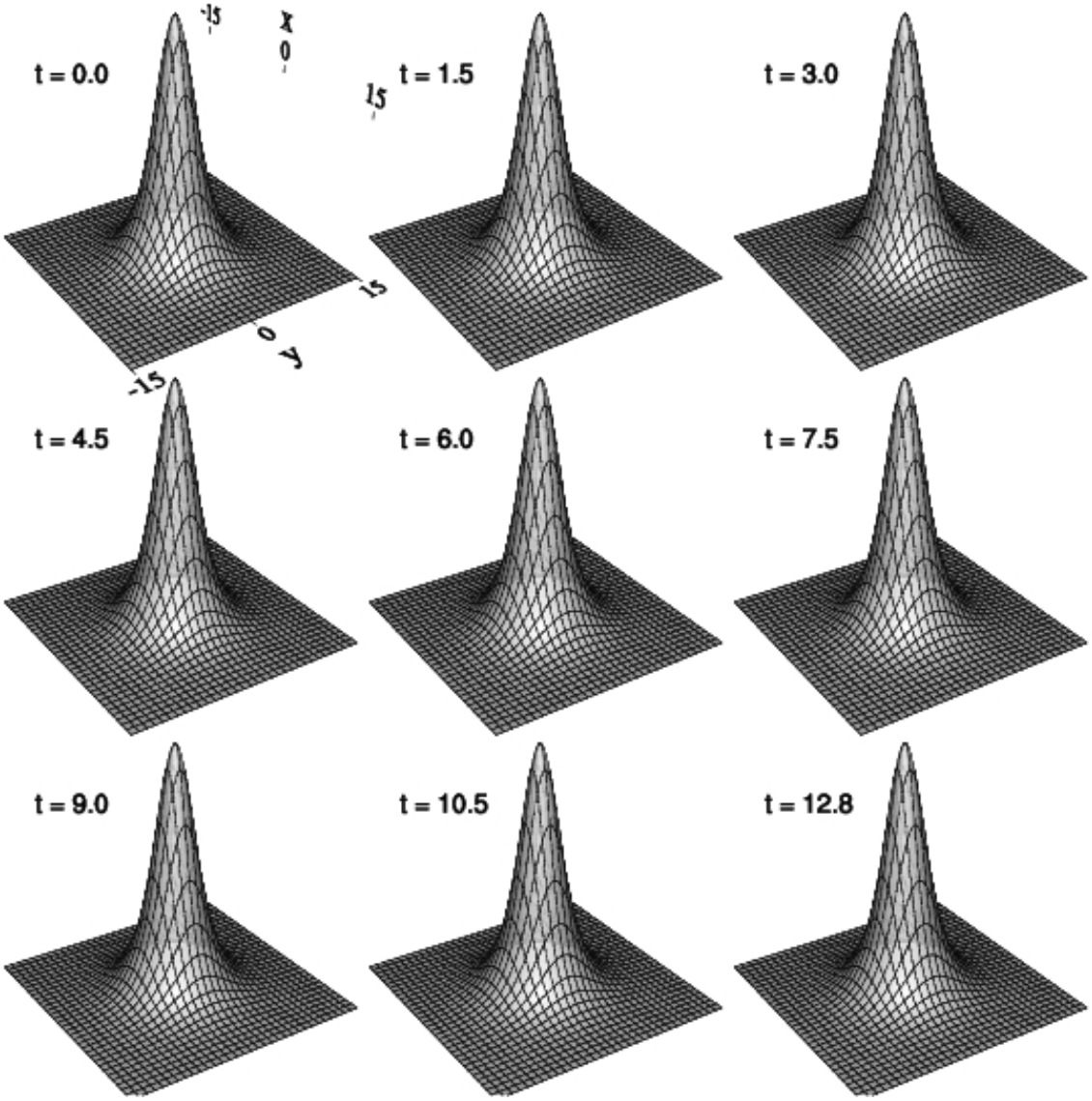


**Figure 4.8:** Evolution of a $\phi_0 = 0.06$ Spherically Symmetric Star as Initial Data. This plot shows some snapshots of the time evolution of a spherically symmetric star parametrized by a central value field of $\phi_0 = 0.06$. The evolution does not reveal any apparent dynamics as it is expected of a static initial data set.

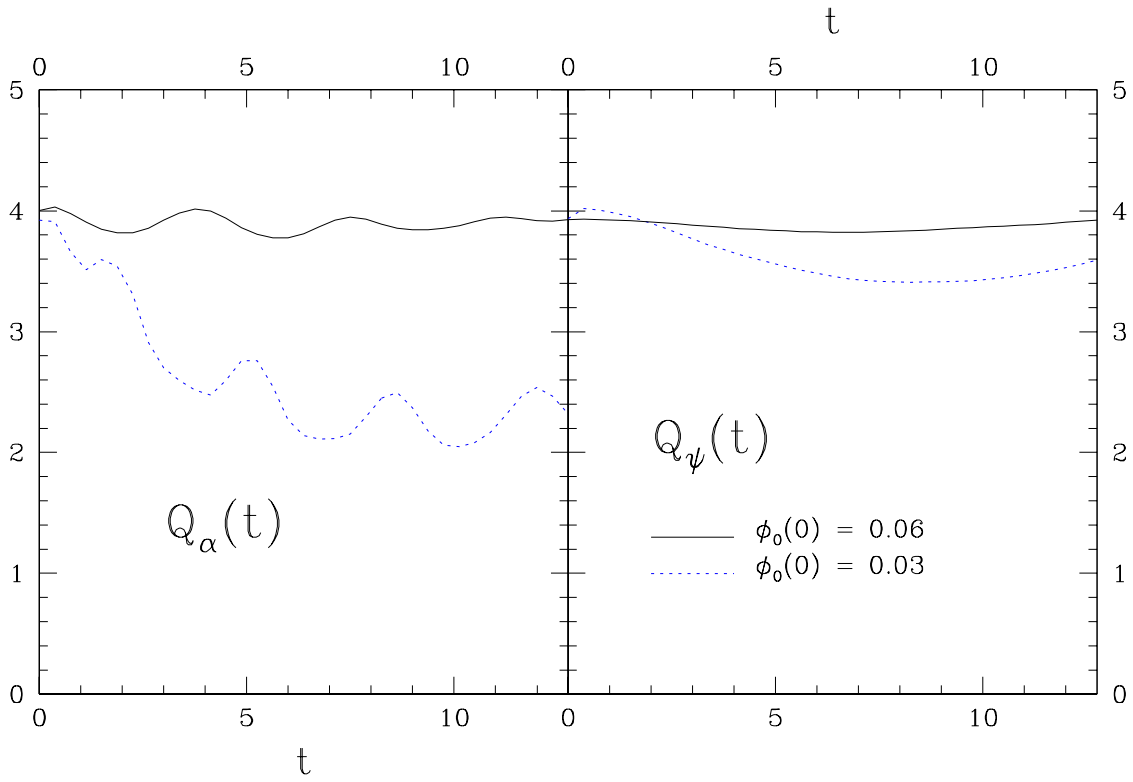


**Figure 4.9:** Q for $\alpha$ and $\psi$. Plots of the convergence factor for the lapse function, $\alpha$ and the conformal factor $\psi$ as a function of time. The functions are plotted for two different values of the central scalar field: $\phi_0 = 0.03$ and $\phi_0 = 0.06$. The improvement is noticeable as the star turns into a more compact one and the computational boundary becomes further apart from the star surface.

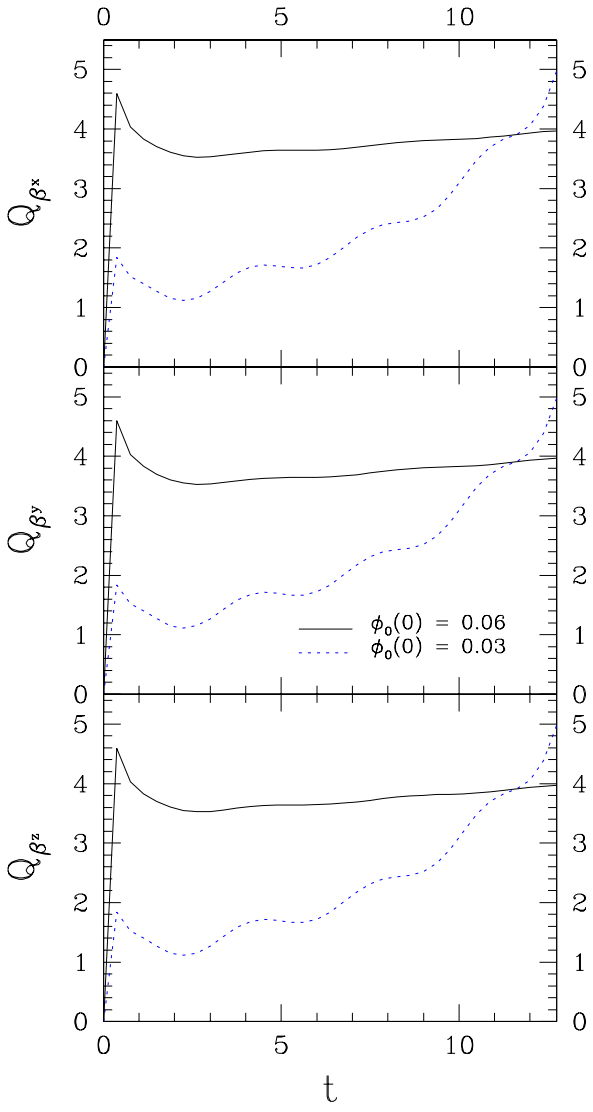


**Figure 4.10:** Q for $\beta^i$. Plots showing the convergence factor Q for all shift vector components $\beta^i$. The functions are plotted for two different values of the central scalar field: $\phi_0 = 0.03$ and $\phi_0 = 0.06$. Note again that as the star turns into a more compact one and the computational boundary becomes further apart from the star surface the convergence factor approaches its expected value of 4. Also, due to the symmetry of the shift vector equations with respect to each of its components in addition to the identical vanishing initial data for all three components, the results presented in these plots seem practically identical.

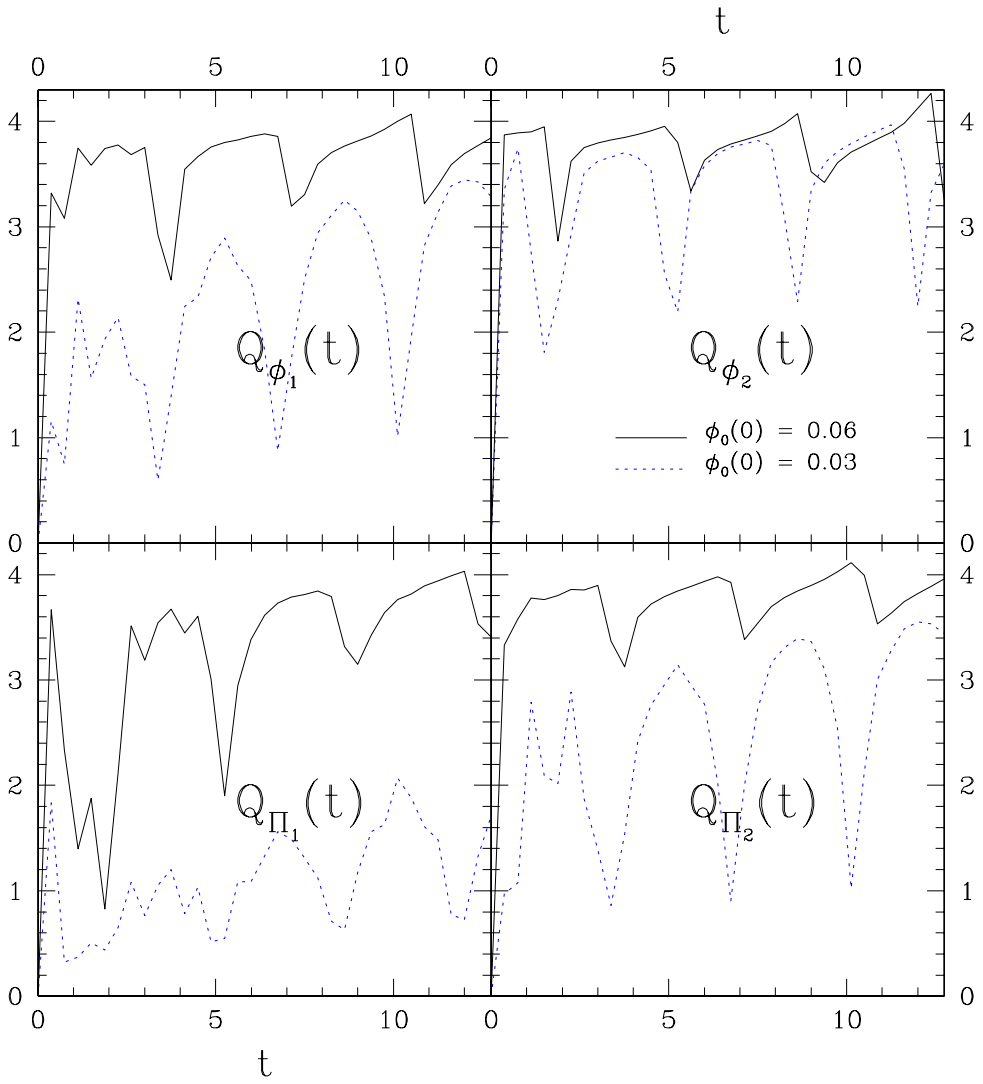


**Figure 4.11:** Q for $\phi_A$ and $\Pi_A$. These set of plots display the convergence factor for the real and imaginary components of the complex scalar field and its conjugate momenta field. The functions are plotted for two different values of the central scalar field, $\phi_0 = 0.03$ and $\phi_0 = 0.06$, in order to highlight the boundary effects on the convergence of the solutions.

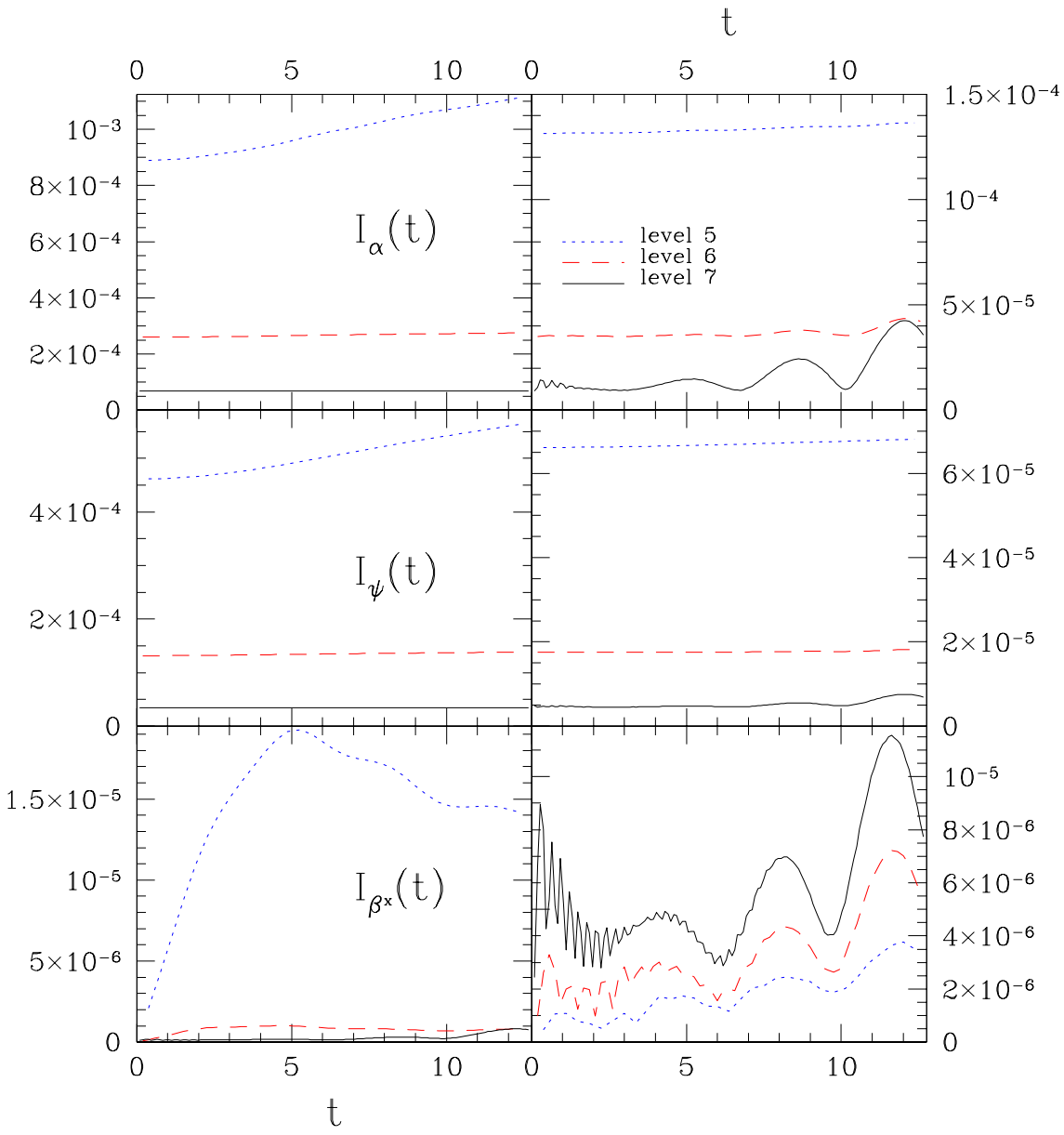


**Figure 4.12:** I for $\alpha$, $\psi$ and $\beta^x$. The plots on the left column correspond to the $L_2$-norm of the independent residuals as a function of time for the lapse function $\alpha$, the conformal factor $\psi$ and the $x$ component of the shift vector $\beta^x$. The left column results are based on a calculation for a star whose its initial central value field is $\phi_0(0) = 0.06$. The right column plots the same quantities as the left but for a star with initial central value field of $\phi_0(0) = 0.03$. On the left, the independent residuals drop to zero as the level of resolution is increased. The rate of convergence is roughly $O(h^2)$. The same seems to happen on the right column with the exception of the bottom panel for $I_{\beta^x}$. It is evident then that the discrete equations are not consistent with their continuum counterpart for this initial data.

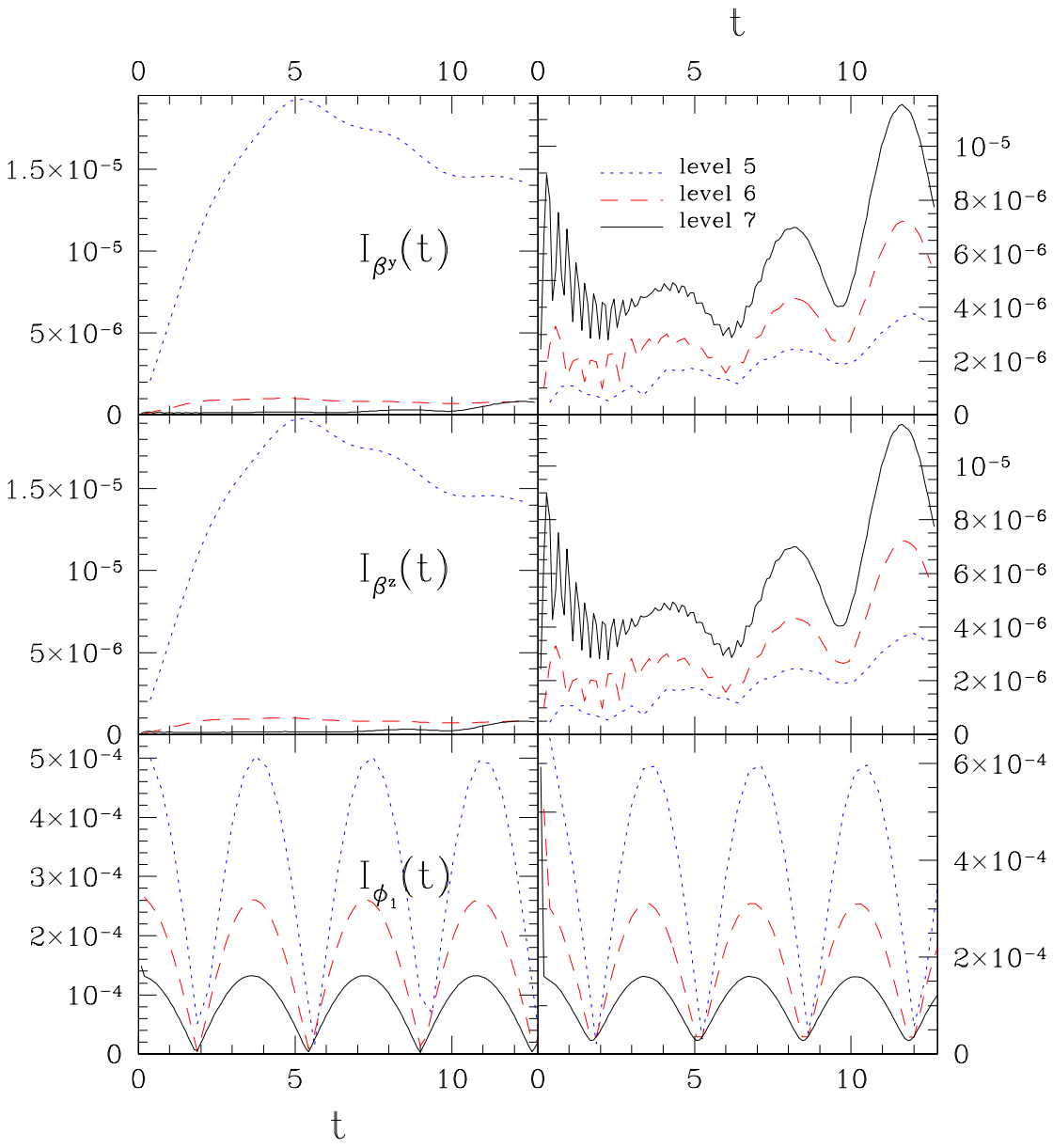


**Figure 4.13:** I for $\beta^y$, $\beta^z$ and $\phi_1$. The plots on the left column result from a calculation for a star whose its initial central value field is $\phi_0(0) = 0.06$, while plots on the right column are for $\phi_0(0) = 0.03$ stars. These plots correspond to the $L_2$-norm of the independent residuals as a function of time for the $y$ and $x$ components of the shift vector, $\beta^y$ and $\beta^z$. On the left, the independent residuals drop to zero as the level of resolution is increased. The rate of convergence is roughly $O(h^2)$ for the shift components and $O(h)$ for the real component of the scalar field. The same seems to happen on the right column, except for the shift components, where it is evident that the discrete equations are not consistent with their continuum counterpart for that initial data.

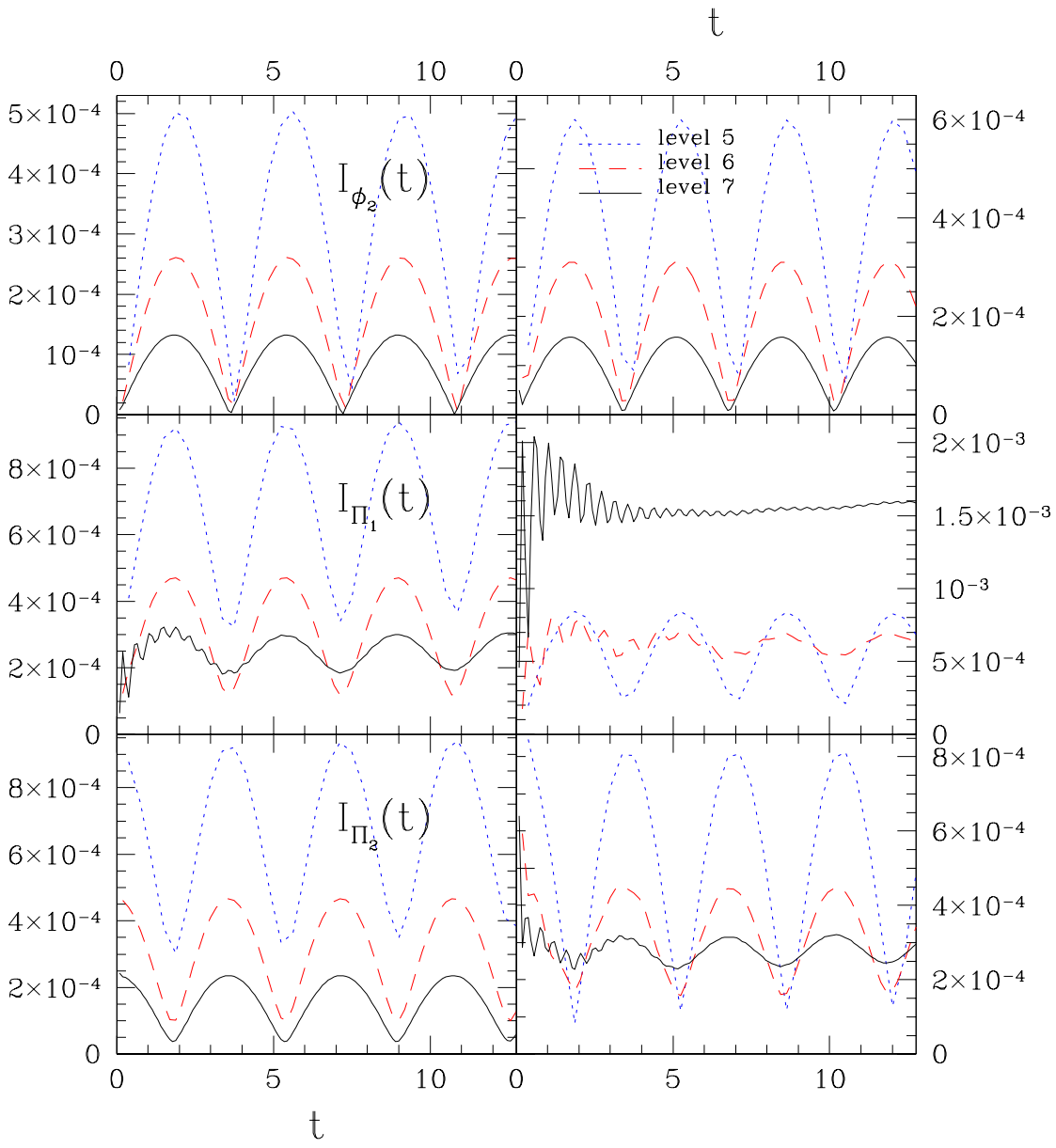


**Figure 4.14:** I for $\phi_2$, $\Pi_1$ and $\Pi_2$. Again plots on the left column refer to stars with $\phi_0 = 0.06$, while plots on the right one to stars with $\phi_0 = 0.03$. The $L_2$-norm of the independent residuals as a function of time for imaginary component of the scalar field, $\phi_2$ and both real and imaginary parts of the conjugate field, $\Pi_1$ and $\Pi_2$, are plotted. All quantities plotted here use first order in time discretization entering the independent residual equations. It is expected then that the residuals I drop to zero with $O(h)$. Clearly that fails for $I_{\Pi_1}$ and it is not convincing for $I_{\Pi_2}$ on the right. For the more compact star, on the left column, all of them seem to approach zero with the rate expected. Note that the result for $I_{\Pi_1}$ is not totally convincing though.

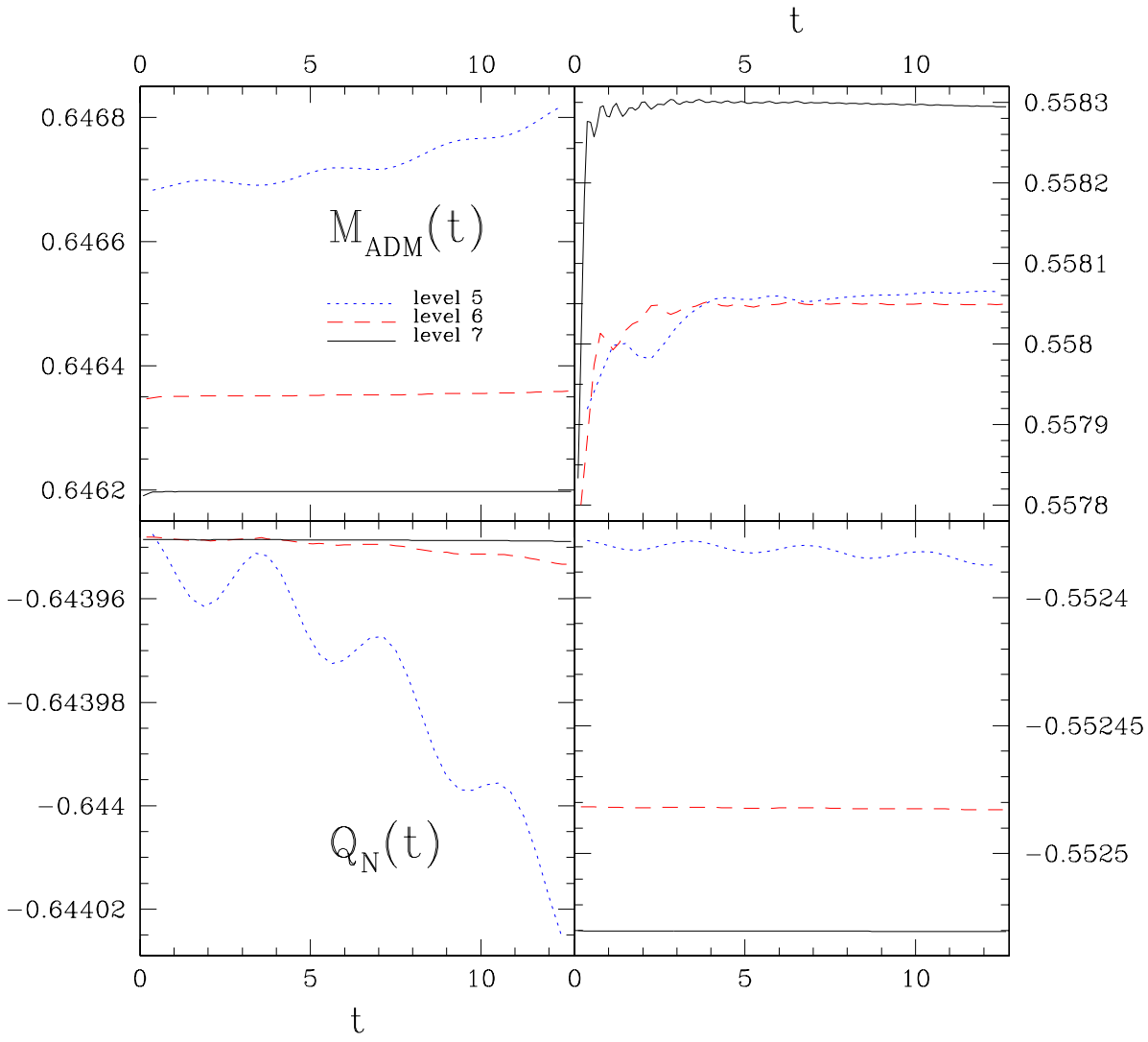


**Figure 4.15:** ADM mass $M_{ADM}$ and Noether charge $Q_N$ for spherical symmetric initial data. Plots showing the conservation of ADM mass $M_{ADM}$ and Noether charge $Q_N$ as a function of time for different levels of resolution. Again, results on the left correspond to stars with $\phi_0 = 0.06$ and those on the right to stars with $\phi_0 = 0.03$.

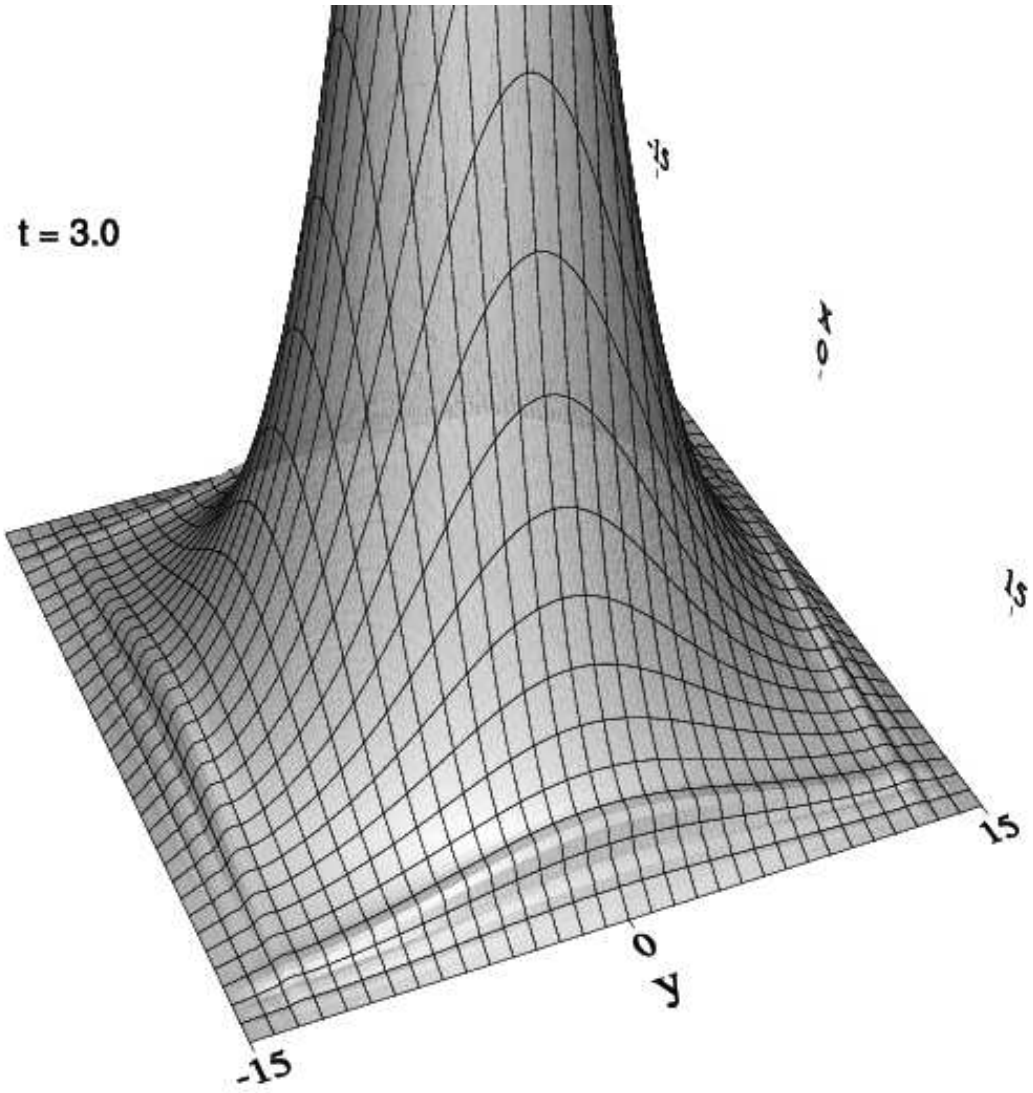


**Figure 4.16:** Evolution Frame Zoomed at $t = 3.0$ for $\phi_0 = 0.03$ Spherically Symmetric Star as Initial Data. The jump in the solution is estimated to be of the order of 1% of $\phi_0$.

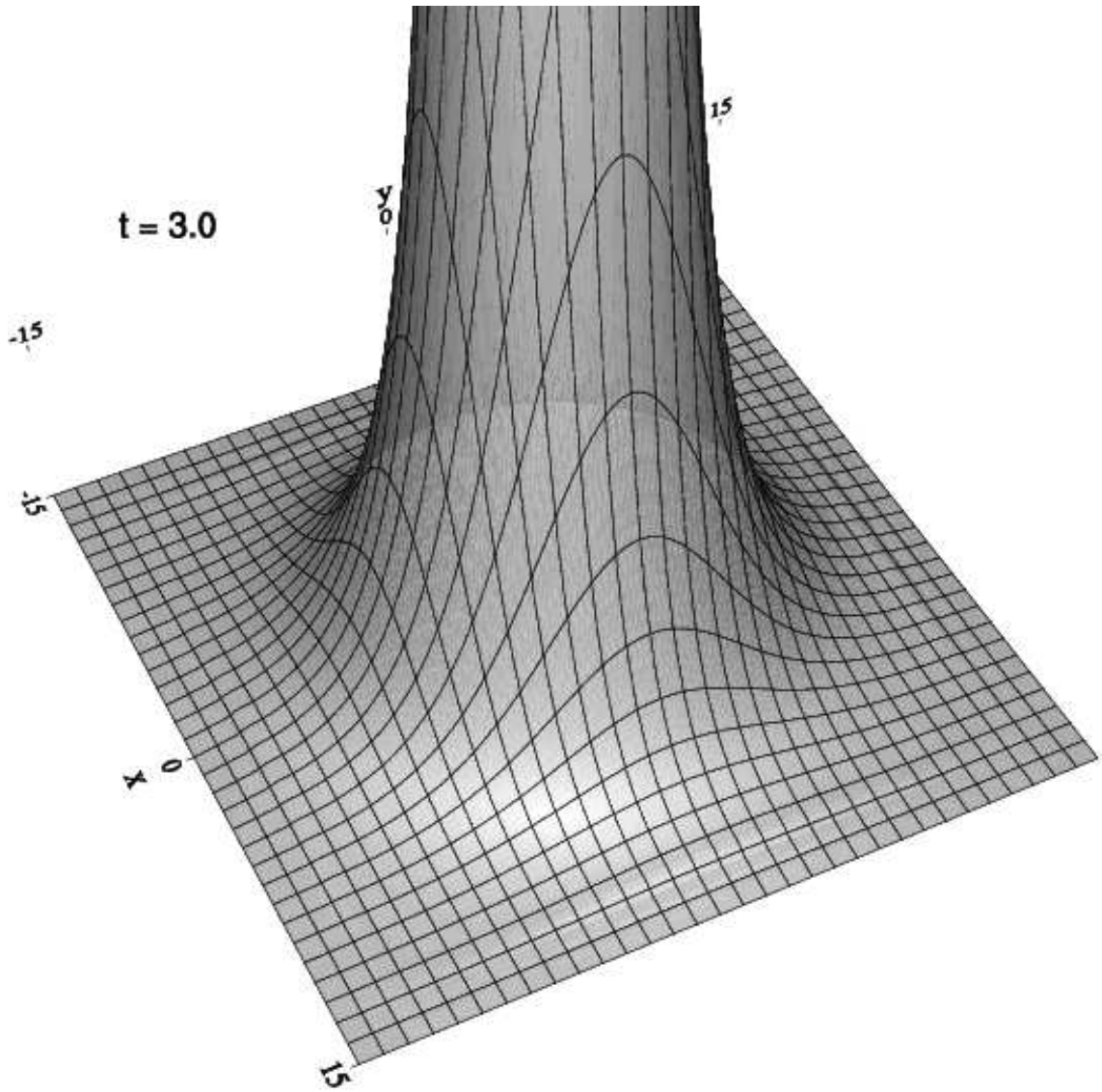


**Figure 4.17:** Evolution Frame Zoomed at $t = 3.0$ for $\phi_0 = 0.06$ Spherically Symmetric Star as Initial Data. Virtually impossible to "eye ball" estimate at the spurious jump in the solution coming from the boundary.

## 4.4 Lorentz Boosted Spherically Symmetric Initial Data - 1 Star

Last section tested the static spherically symmetric solution as initial data. The results were considered satisfactory what testify in favor of the correctness of the implementation of the initial value problem solvers. One last piece of these solvers is left untested though. Once the spherically symmetric solution is generated and interpolated in the 3D numerical domain, according to Subsection 2.6.6, the star profile needs to be boosted, according to Subsection 2.6.7, in order to initiate the orbital motion for example. This section applies then the same battery of tests as the previous Sec. 4.3 in order to test this last non-trivial step to obtain this special initial data set, the boosted spherically symmetric solution.

Fig. 4.18 displays some snapshots of the evolution of a boosted solution. The spherically symmetric solution used is parametrized by $\phi_0 = 0.06$, being therefore a member of the stable branch. The boost velocity adopted in this simulation was $v_x = -0.1$. The snapshots show that the star is slowly moving along the $x$ direction towards $x = -15$. A rough numerical estimate gave the value $\Delta x/\Delta t \approx -0.1$ according to the expected. The grid parameters used here was the same as in the previous sections: $bbox = [-15, 15, -15, 15, -15, 15]$ and the finest $shape = [129, 129, 129]$.

The left panels of Figures 4.19, 4.20 and 4.21 attest that the boosted solution is converging. The convergence factor approaches to 4 for all variables. The right panels of the same set of figures indicates that besides being convergent the solution is also consistent. The independent residuals of variables do converge to zero with the rate expected: $O(h)$ for the scalar field variables and $O(h^2)$ for the geometric variables. At last, Fig. 4.22 shows that both ADM mass and noether charge are being conserved.

After this extensive battery of tests on all different kinds of initial data used in this dissertation the reader should be convinced about the correctness of the implementation of the equations of motion and the smoothness of all initial data sets used. Next section starts then second part of this chapter, the investigation of the collision between two stars.

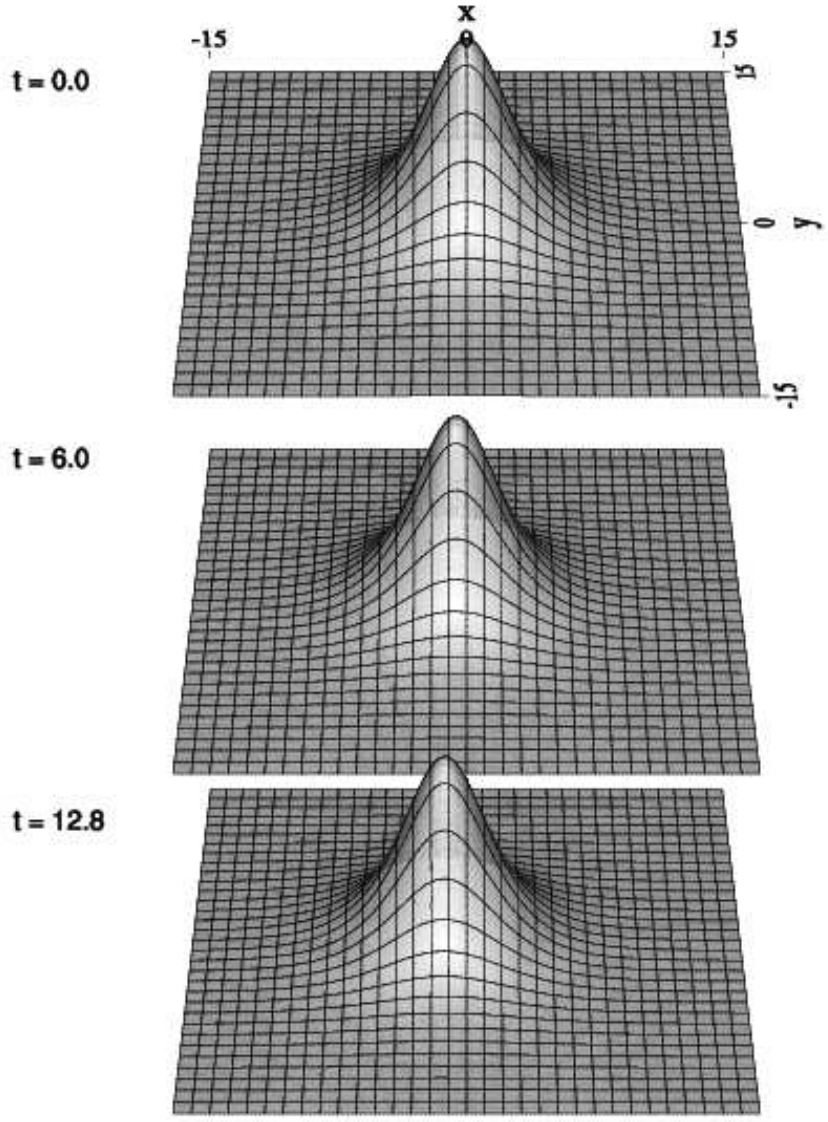


**Figure 4.18:** Snapshots of the Evolution of a Boosted $\phi_0 = 0.06$ Spherically Symmetric Star as Initial Data. A static spherically symmetric star with central value field of $\phi_0 = 0.06$ is boosted along the $x$ direction with boost speed of $v_x = -0.1$. Note that a rough numerical estimation of this velocity was possible and provided a consistent result, i.e. the numerically measured quantity $\Delta x/\Delta t \approx -0.1$.

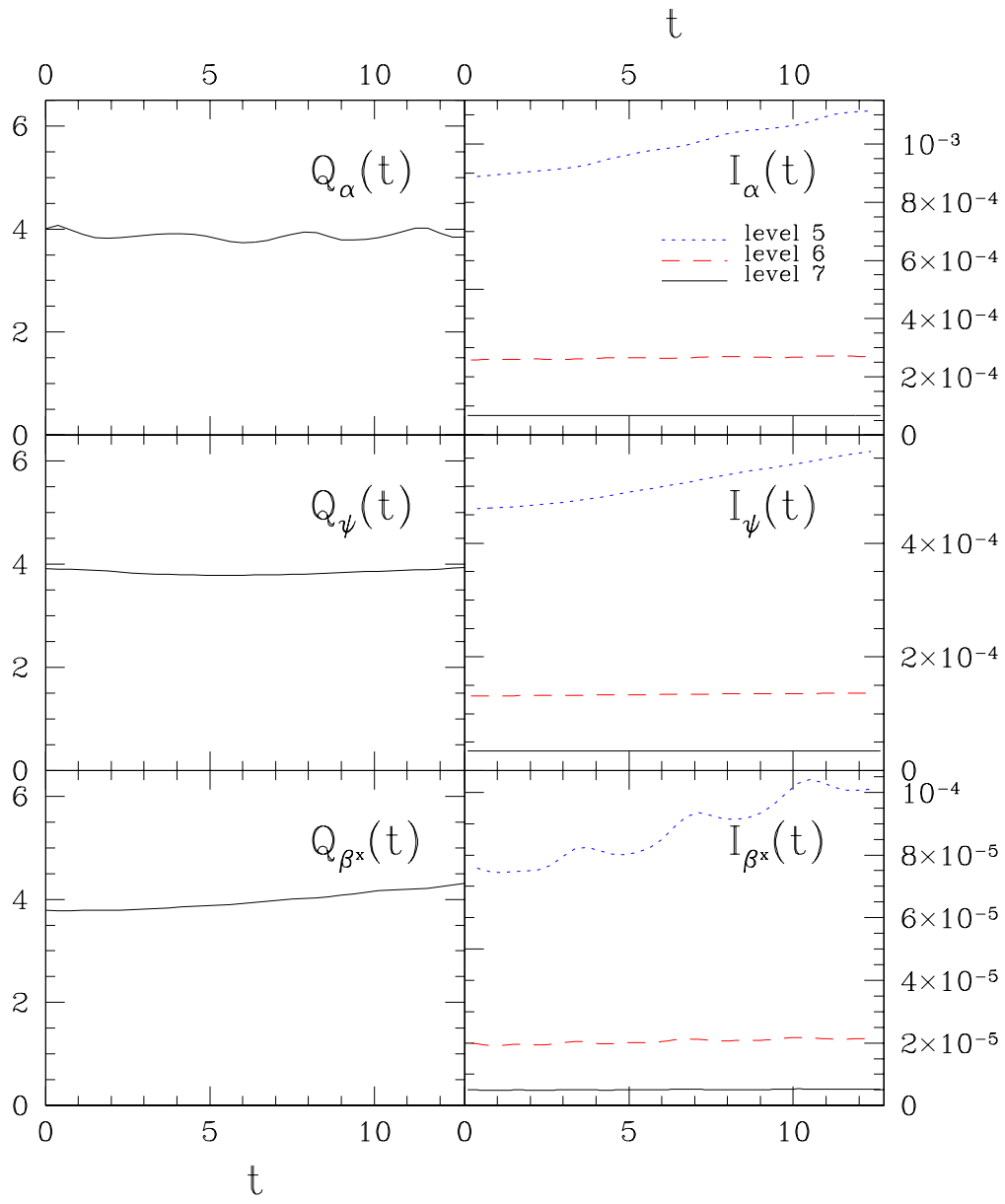


**Figure 4.19:** Q and I for $\alpha$, $\psi$ and $\beta^x$ for a boosted star. The left panels show the plots of the convergence factor Q as a function of time t for the lapse function $\alpha$, the conformal factor $\psi$ and the $x$ component of the shift vector $\beta^x$. The right panels show the $L_2$-norm I of the independent residual equations for the same set of variables as a function of time. The results reported here are for a boosted star along the $x$ direction with speed $v = -0.1$. The central value of the scalar field parametrizing this star is $\phi_0 = 0.06$.

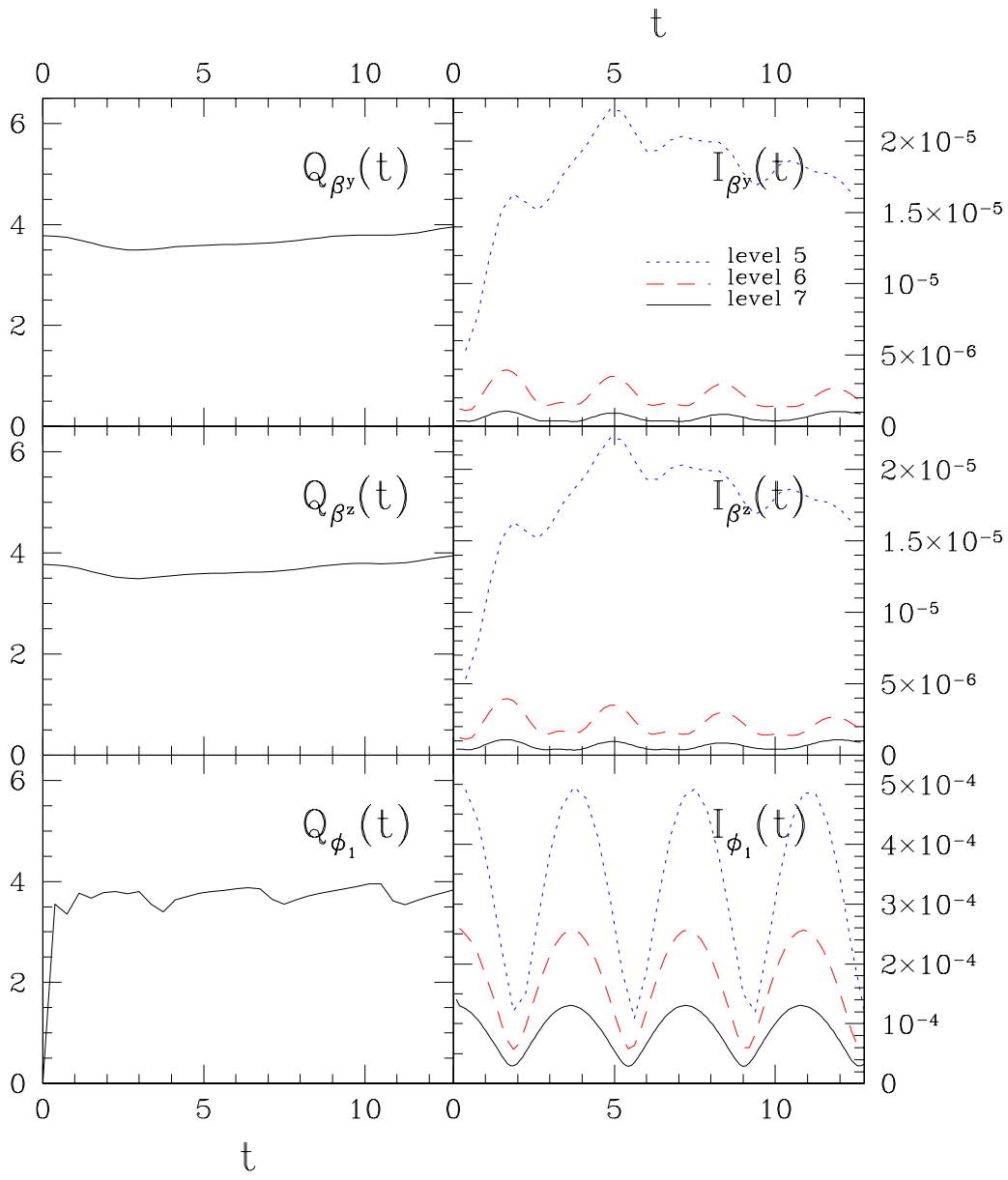


**Figure 4.20:** Q and I for $\beta^y$, $\beta^z$ and $\phi_1$ for a boosted star. The left panels show the plots of the convergence factor Q as a function of time t for the $y$ and $z$ shift components, $\beta^y$ and $\beta^z$, and the real component of the complex scalar field, $\phi_1$. The right panels show the $L_2$-norm I of the independent residual equations for the same set of variables as a function of time. The results reported here are for a boosted star along the $x$ direction with speed $v = -0.1$. The central value of the scalar field parametrizing this star is $\phi_0 = 0.06$.

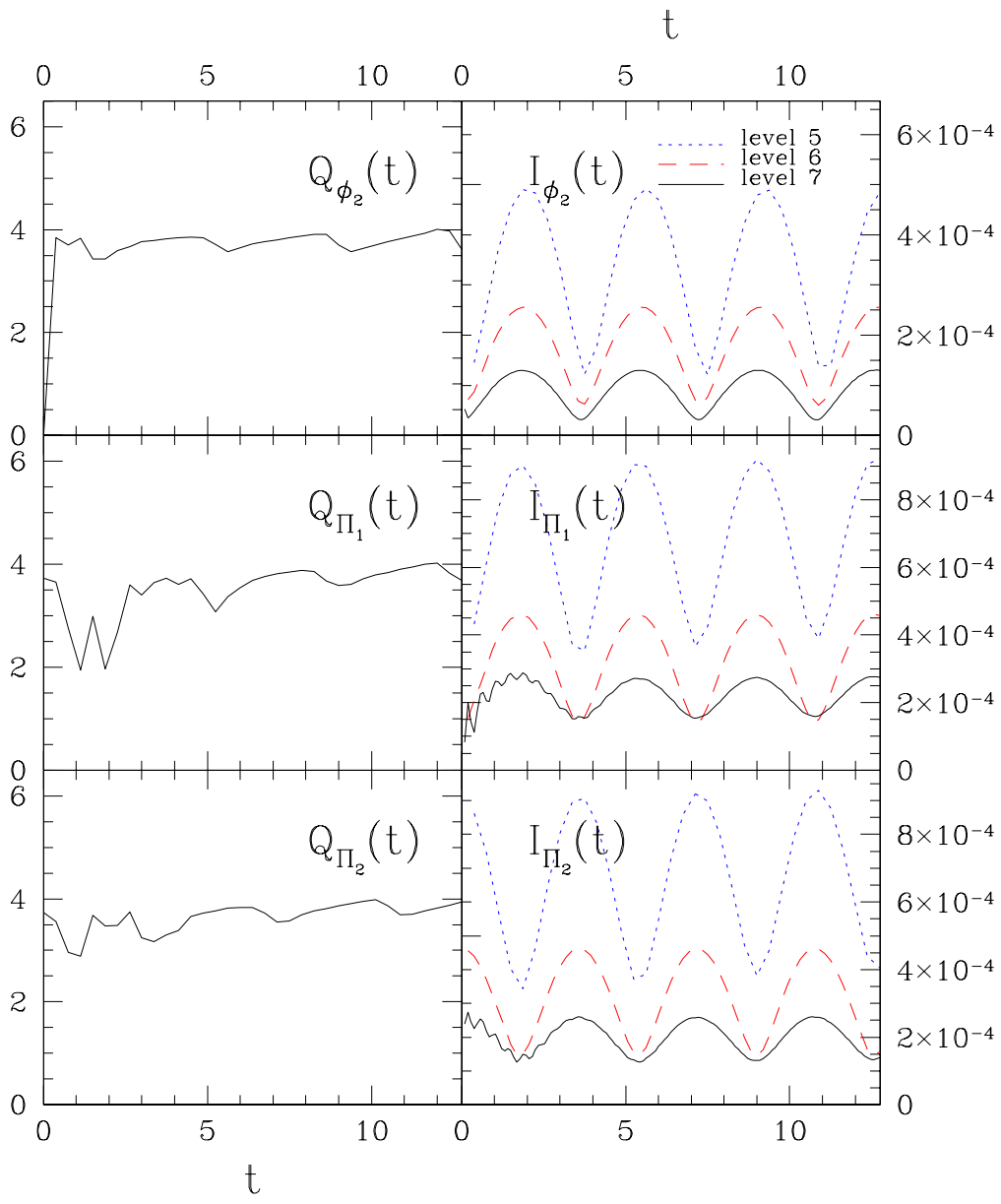


**Figure 4.21:** Q and I for $\phi_2$, $\Pi_1$ and $\Pi_2$ for a boosted star. The left panels show the plots of the convergence factor Q as a function of time t for imaginary component of the complex scalar field $\phi_2$ and the real and imaginary components of its conjugate field, $\Pi_1$ and $\Pi_2$. The right panels show the $L_2$-norm I of the independent residual equations for the same set of variables as a function of time. The star with a central value field of $\phi_0 = 0.06$ is boosted along the $x$ coordinate with boost speed of $v = -0.1$.

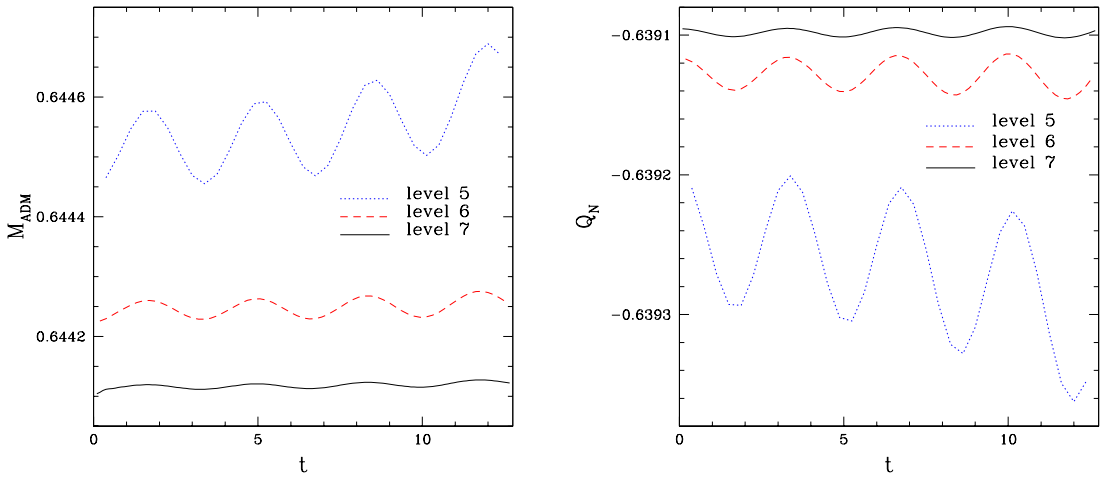


**Figure 4.22:** ADM mass $M_{ADM}$ and Noether charge $Q_N$ for boosted spherical symmetric initial data as a function of time t. Both plots indicate a trend to conservation of the quantity with a convergence rate of $O(h^2)$.

## 4.5 Head-on Collision of Two Boson Stars

Head-on collision of boson stars have been investigated both on newtonian regime [140] as well as in the full relativistic regime [107, 6]. A solitonic behaviour of the solution was first shown to exist in the newtonian regime by Choi [140]. This behaviour was also confirmed in the full general relativist case in axisymmetry by Lai [107]. This section has the purpose of reproducing this solitonic behaviour for the Einstein-Klein-Gordon system of equations (boson star) under the conformally flat condition.

The experiment was prepared as follows. Two spherically symmetric boson stars were generated each with the central value field of $\phi_0 = 0.02$. The stars were interpolated into the 3D numerical domain, $bbox = [-50, 50, -25, 25, -25, 25]$, and centred at $(25, 0, 0)$ (star 1) and $(-25, 0, 0)$ (star 2). An initial boost with velocity $v_x = 0.4$ was given to both stars such that $-v_x^{(2)} = v_x^{(1)} = v_x$. Fig. 4.23 shows the evolution of these two stars on a grid with $shape = [129, 65, 65]$ for $0 \leq t \leq 140$. The stars approach each other and start overlapping around $t \approx 30$. It becomes hard then to disguish between the two stars around $t = 55$. They are in a central region of the domain with a strong interference between them. About $t = 100$ it is possible to distingsh again between the two stars and they follow their paths in opposite directions towards the ends of the domain. During the time interval they interact the modulus of the scalar field reaches a maximum of $|\phi(t, x, y, z)| \simeq 0.0472$ returning afterwards to a value close to its initial configuration, $|\phi(t, x, y, z)| \simeq 0.0182$.

This result is important to evaluate how good the conformally flat condition applied to the Einstein-Klein-Gordon system is. This is the first experiment that allows to conclude something about this approximation. Qualitatively, the solitonic behaviour was observed as in the axisymmetric fully general relativistic case suggesting the viability of the CFC as a satisfactory approximation

to general relativity in the context of head-on collision of stars.

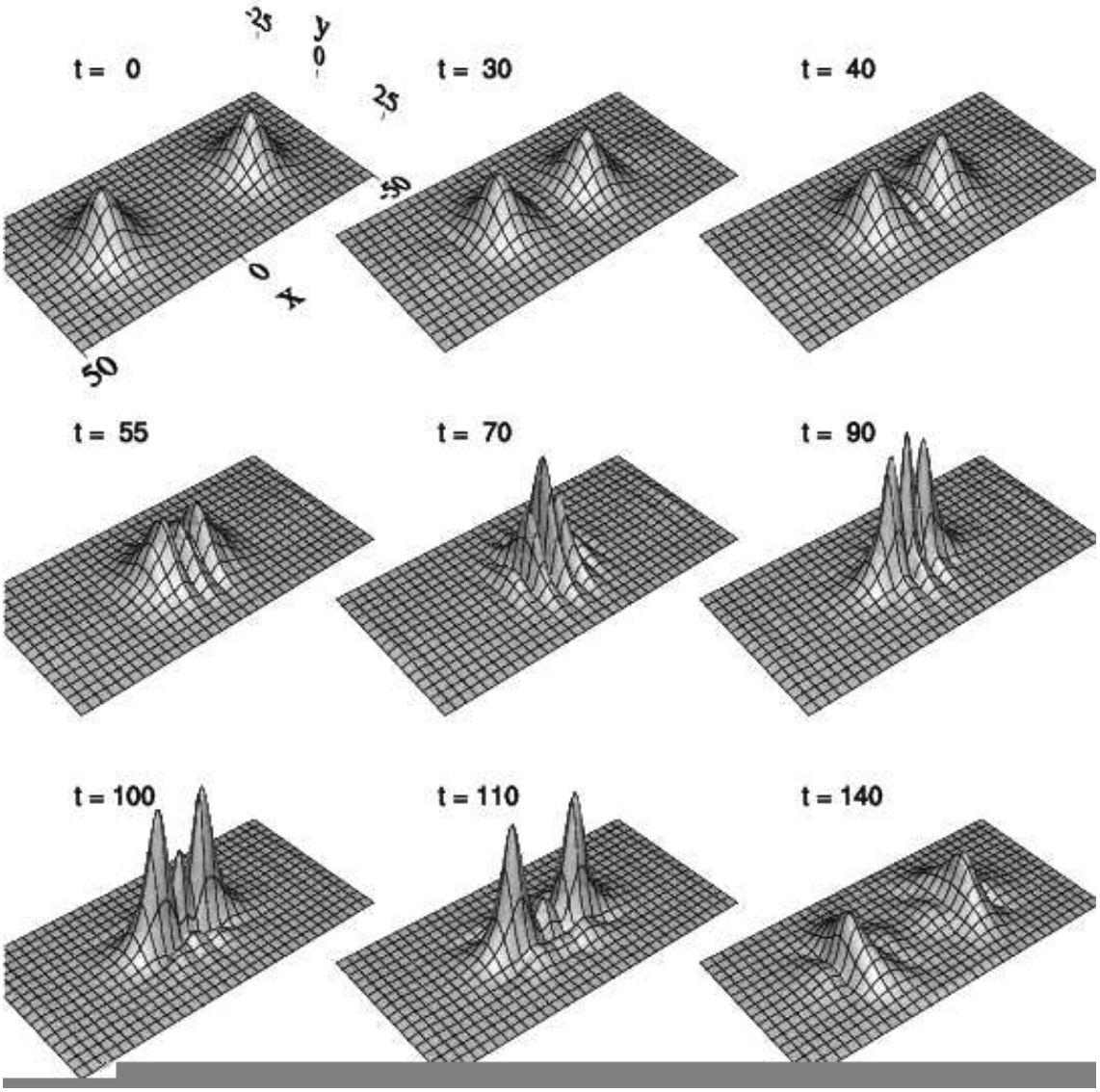


**Figure 4.23:** Boson star head-on collision: solitonic behaviour. Both stars are members of the stable branch of the family of spherically symmetric solution with central value field of $\phi_0 = 0.02$. The stars are initialized via interpolation, translation and boosting of a spherically symmetric solution. The stars are initially centered at $(25, 0, 0)$ and $(-25, 0, 0)$, with initial velocity $-v_x^{(2)} = v_x^{(1)} = v_x = 0.4$. The two stars start to overlap at $t \approx 30$, interfere with each other, and then separate at $t \approx 110$. Throughout the evolution $0.0 \leq |\phi(t, x, y, z)| \leq 0.0472$. Other simulation parameters are: $bbox = [-50, 50, -25, 25, -25, 25]$ and $shape = [129, 65, 65]$. Note that the time snapshots showed in this figure are not uniformely distributed in time. There are more snapshots during the time interval the stars overlap than when they are apart.

## 4.6 Orbital Dynamics of Two Boson Stars

This section presents the results for three different cases of the orbital dynamics classified according to the nature of the final configuration. Several different simulations were performed (but not all shown in this chapter) corresponding to configurations with initial velocities along the $x$ direction of $v_x \in \{0, 0.03, 0.05, 0.07, 0.08, 0.09, 0.1, 0.11\}$. The final result of the mergers could then be classified as orbital motion for $0.09 \leq v_x \leq 0.11$; boson star formation when $0.07 \leq v_x < 0.09$ and (possibly) black-hole formation for $v_x$ in $0 \leq v_x < 0.07$. Note however that the limits of the range of these intervals are not precisely determined yet, they are just a rough estimate based on the results of the few experiments performed. The first case showed consists of the two stars orbiting each other in a slightly eccentric orbit. On the second case, the stars trajectories approach each other close enough to initiate a grazing collision followed by the full merger of the two stars. The final object is still a boson star as far as the simulations were run. The third and last result presents a very eccentric orbit interrupted by the collision of the two compact bodies possibly resulting into a black-hole.

All the simulations presented in this section were performed with four different grid resolutions. The grids are labelled such that $G_1, G_2, G_3, G_4$ represent the sequence of grids from the coarser to the finer. Their several shapes are:

$$G_1 : \quad shape_1 = [65, 65, 65], \tag{4.8}$$

$$G_2 : \quad shape_2 = [81, 81, 81], \tag{4.9}$$

$$G_3 : \quad shape_3 = [97, 97, 97], \tag{4.10}$$

$$G_4 : \quad shape_4 = [113, 113, 113]. \tag{4.11}$$

The coordinate range for all grids used in this section are identical and stored into the bounding box vector as $bbox = [-60, 60, -60, 60, -60, 60]$. All the surface and trajectory plots refer to the $z = 0$ slice of the numerical domain. The basic experiment set up consists of generating two individual 1D spherically symmetric boson stars parametrized[2] by $\phi_0 = 0.02$. Each of them are then interpolated into the 3D domain and centred at $(0, 20, 0)$ and $(0, -20, 0)$ for all experiments presented in this section. The stars interpolation and translation processes happen at the same time the stars are boosted according to Subsection 2.6.7. The binary of boosted stars form therefore the initial data for these experiments. The experiments differ among them only by the initial velocities assigned to the stars. All the other parameters are kept the same.

The first experiment is shown on Fig. 4.24. Each star is assigned a velocity of $v_x = 0.09$ along the $x$ axis and in opposite directions. The plots shows the evolution of the binary for approximately two orbits and a quarter corresponding to the time between $0 \leq t \leq 4211$. The orbits are slightly elliptic as a visual inspection of Fig. 4.25 displaying the orbital trajectory of both stars can attest. The eccentricity of the orbit has not been measured numerically though. An even closer examination of the same figure shows that the orbits seem to slowly precess. This qualitative result is quite promising towards the evaluation of the CFC as an accurate approximation of general relativity

[2] See Fig. 2.1 for properties of the individual stars.

since this effect is a purely relativistic one. Further quantitative examination is still required though.

The evolution showed on Fig. 4.24 is monitored by the $l2$-norm of the independent residual for all geometric and scalar field quantities[3]. The $l2$-norm independent residual for the real component of the scalar field, $I_{\phi_1}(t)$, is shown on Fig. 4.26. The green dots appearing on that plot[4] correspond to $I_{\phi_1}(t)$ calculated on the $G_1$ grid, the blue dots to the one evaluated at the $G_2$ grid, the red ones to the $G_3$ and finally the black dots to the $G_4$. The rate of convergence was roughly measured and found to be close to $O(h)$, as expected. Fig. 4.27 shows the same results but for a short time interval of $0 \leq t \leq 20$. It becomes clearer from there that the functional shape of the independent residual as a function of time does not change with resolution. The $l2$-norm of the independent residual for the lapse function as a function of time, $I_\alpha(t)$, is displayed on Fig. 4.28. In this case the convergence is expected to be quadratic. Again, a rough estimate from the data on the plot agree with $O(h^2)$ expected.

The second set of experiment assigns the initial velocity $v_x = 0.07$ to the stars. The evolution is shown on Fig. 4.29 and its correspondent trajectory on Fig. 4.30. They orbit each other for almost half an orbit when the grazing collision takes place. The stars then plunge towards each other resulting into a spinning boson star. The final star has a central value field of approximately $\phi_0 \simeq 0.05$. The Fig. 4.34 shows the $l2$-norm for the lapse function independent residual as a function of time. The independent residual does goes to zero as a function of the grid resolution. However, as the plot shows, the convergence deteriorates as a function of time specially after the merger of the stars. The coarsest grids $G_1$ and $G_2$ can be safely discarded after $t \simeq 600$ since they are too coarse to represent the solution at those times. Lots of scalar matter are sent out after the merger of the two stars. Unfortunately they are reflected back onto the domain by the boundaries. These spurious reflections also contribute to pollute the solution domain.

The last experiment shows a short orbital motion of two stars with initial velocities given by $v_x = 0.05$. The stars trajectories quickly plunge towards each other, as the Figures 4.32 and 4.33 show, and a black-hole is believed to form. At the moment no apparent horizon finder has been implemented to make sure a black-hole has formed. The conclusion that a black-hole actually formed comes from the inspection of the metric components functions such as the lapse function. The lapse function for example reaches extremely small values in the region the scalar matter spikes, freezing therefore the evolution of the slicing in that region. That is a very well known feature of the maximum slicing and it was taken as an indication that a black-hole is forming in that region. Another caveat about this last set of results comes from the independent residual as shown on Fig. 4.34. It does converges approximately as expected, $O(h^2)$, until around $t \simeq 380$. About that time the stars are merged and about to collapse. In the collapse process, the resolution of *all* grids used for this simulation fails to resolve the solution. Unfortunately, it is impossible to increase the resolution at the moment since the unigrid code is already using all the computational resources available. So this result still needs to be handle with care. Only a substantial increase in

[3]Recall that the *independent* finite difference operators for the elliptic set of equations are second oder accurate while for the hyperbolic equations they are only first order accurate.

[4]The map between the point colours of the independent residual functions and the grids at which they are evaluated are the same for all plots of this kind discussed in this chapter.

resolution will provide a definitive answer on the final object resulting from this collision.

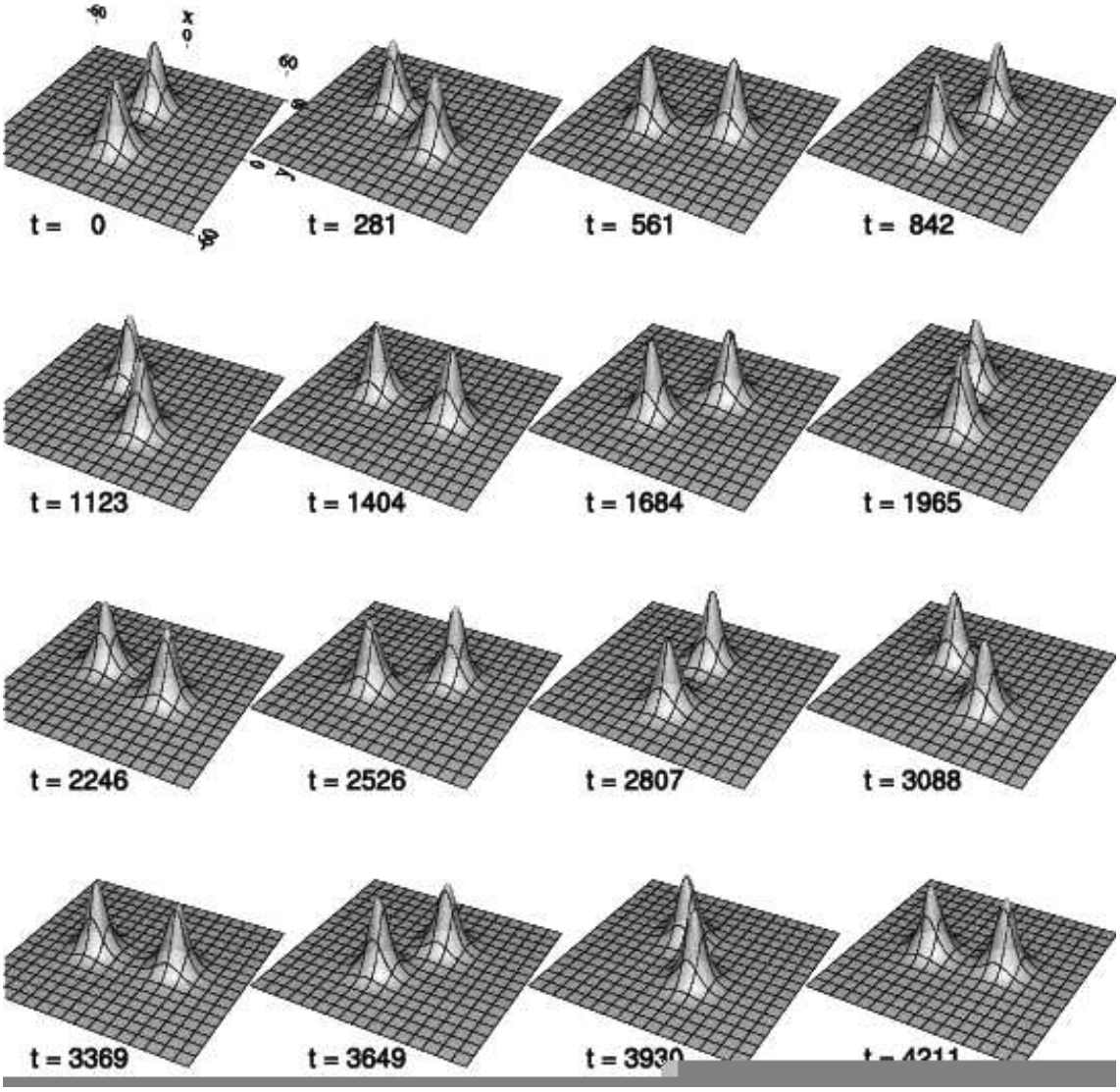


**Figure 4.24:** Orbital Evolution of two $\phi_0 = 0.02$ boson stars with initial speed $v_x = 0.09$. This plot shows a long term time evolution of two spherically symmetric stars used as initial data. The stars are both parametrized by a central value field of $\phi_0 = 0.02$. They are centred in the cubic numerical domain at $(0, 20, 0)$ and $(0, -20, 0)$. The initial star velocities are such that $v_x = -v_x^{(1)} = v_x^{(2)} = 0.09$. Note that the stars orbit around each other for almost two and a half orbits. Throughout the evolution the complex scalar field modulo is bounded approximately by its initial central value, $0.0 \leq |\phi(t, x, y, z)| \leq \phi_0$.

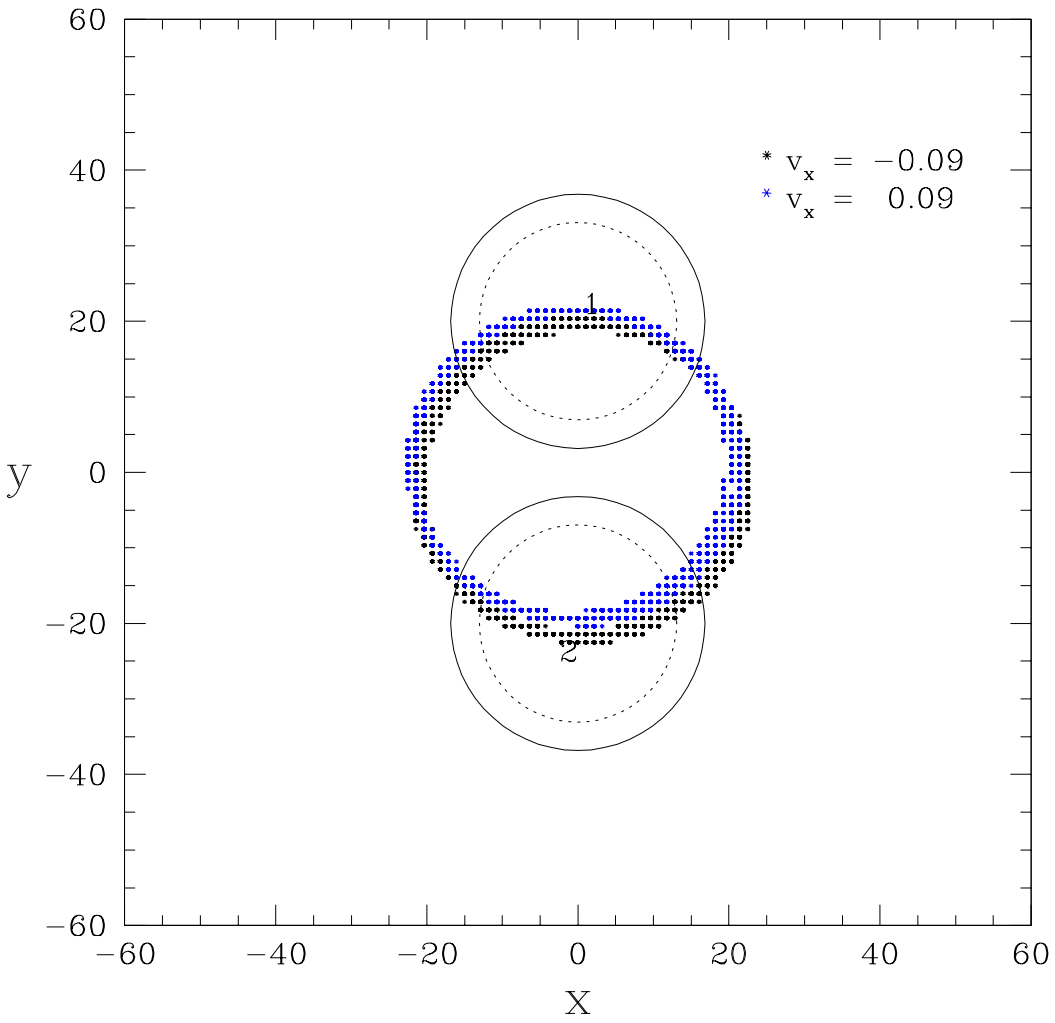


**Figure 4.25:** Two-star trajectory for initial speed $v_x = 0.09$ and coordinate separation $d = 40$. This plot shows the trajectories of stars 1 and 2 initially centred at $(0, 20, 0)$ and $(0, -20, 0)$ for $0 \leq t \leq 4500$. It corresponds to the same orbital motion shown on Fig. 4.24. At $t = 0$, the solid lines represent the circumferences of the stars at $R_{99}$, the radius containg 99% of its ADM mass $M_{ADM}$, while the dotted lines draw the star surface at $R_{95}$, the radius containg 95% of its ADM mass $M_{ADM}$. Both stars are members of the stable branch family of solutions and are parametrized by $\phi_0(0) = 0.02$. The initial star velocities are such that $v_x = -v_x^{(1)} = v_x^{(2)}$. Note that the evolution corresponds approximately to two and a half complete orbits around each other. Furthermore, the orbit seems to be elliptic in shape and precess slightly as the simulation evolves.

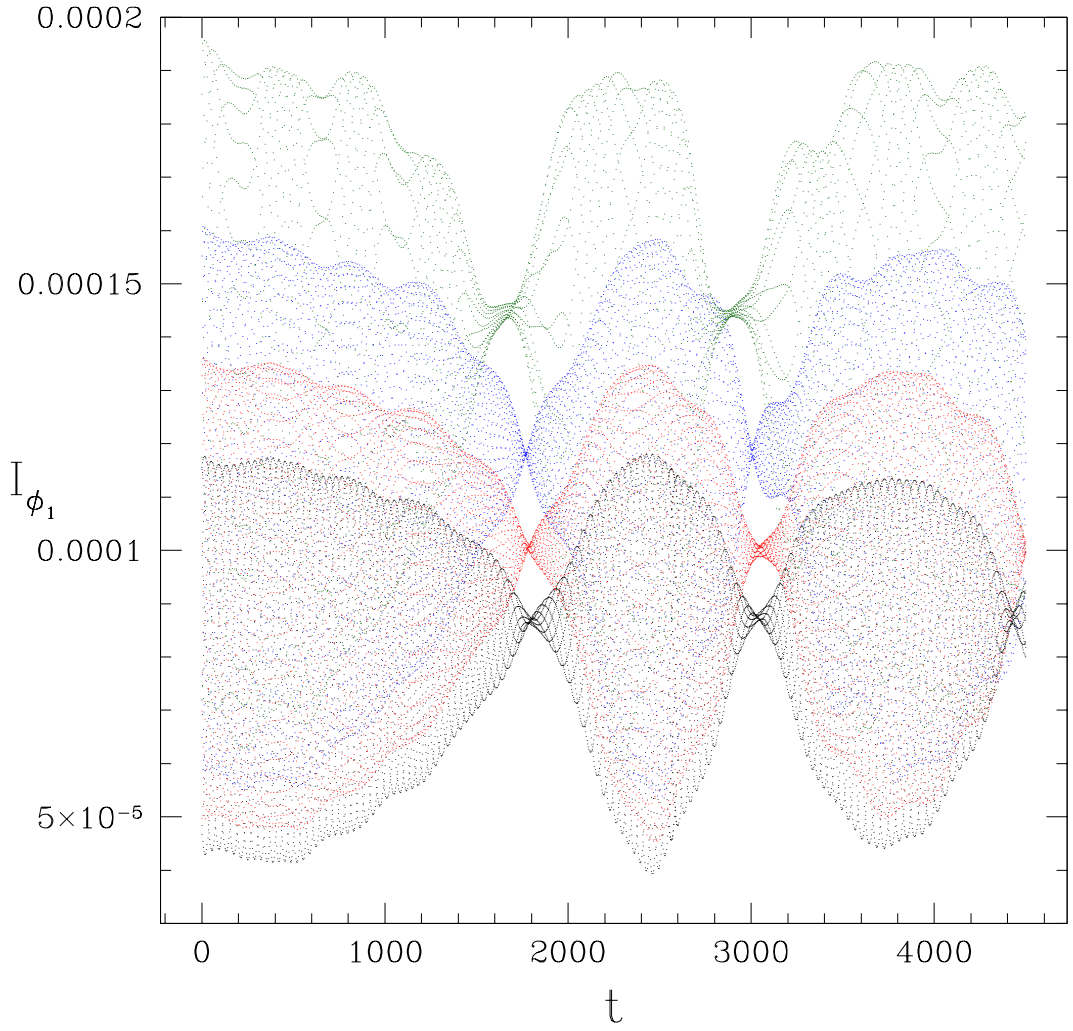


**Figure 4.26:** Long term $I_{\phi_1}(t)$ during the orbital dynamics with initial speed $v_x = 0.09$. This figure shows the independent residual $l2$-norm $I_{\phi_1}(t)$ for the real scalar field component $\phi_1$ during the orbital motion presented in the Fig. 4.24 for $0 \leq t \leq 4500$. The different coulored dots represent $I_{\phi_1}(t)$ function calculated with different grid resolutions; The number of grid points per edge of the cubic domain is $n_1 = 65$ (green dots), $n_2 = 81$ (blue), $n_3 = 97$ (red) and $n_4 = 113$ (black) with corresponding mesh spacings of $h_1 = 1.875$, $h_2 = 1.5$, $h_3 \simeq 1.237$ and $h_4 \simeq 1.062$. The convergence rate expected for this evaluation is $O(h)$. A rough estimate indicate that the average value goes to zero as $(1.45 \rightarrow 1.18 \rightarrow 1 \rightarrow 0.87) \times 10^{-4}$. When multiplied by the linear conversion factors, $h_{12} = h_1/h_2 = 1.25$, $h_{13} = h_1/h_3 \simeq 1.516$ and $h_{14} = h_1/h_4 \simeq 1.766$ the same sequence of average $I_{\phi_1}$ becomes almost coincident: $(1.45 \simeq 1.475 \simeq 1.516 \simeq 1.536) \times 10^{-4}$. That is a good evidence that the linear convergence is attained throughout the evolution.

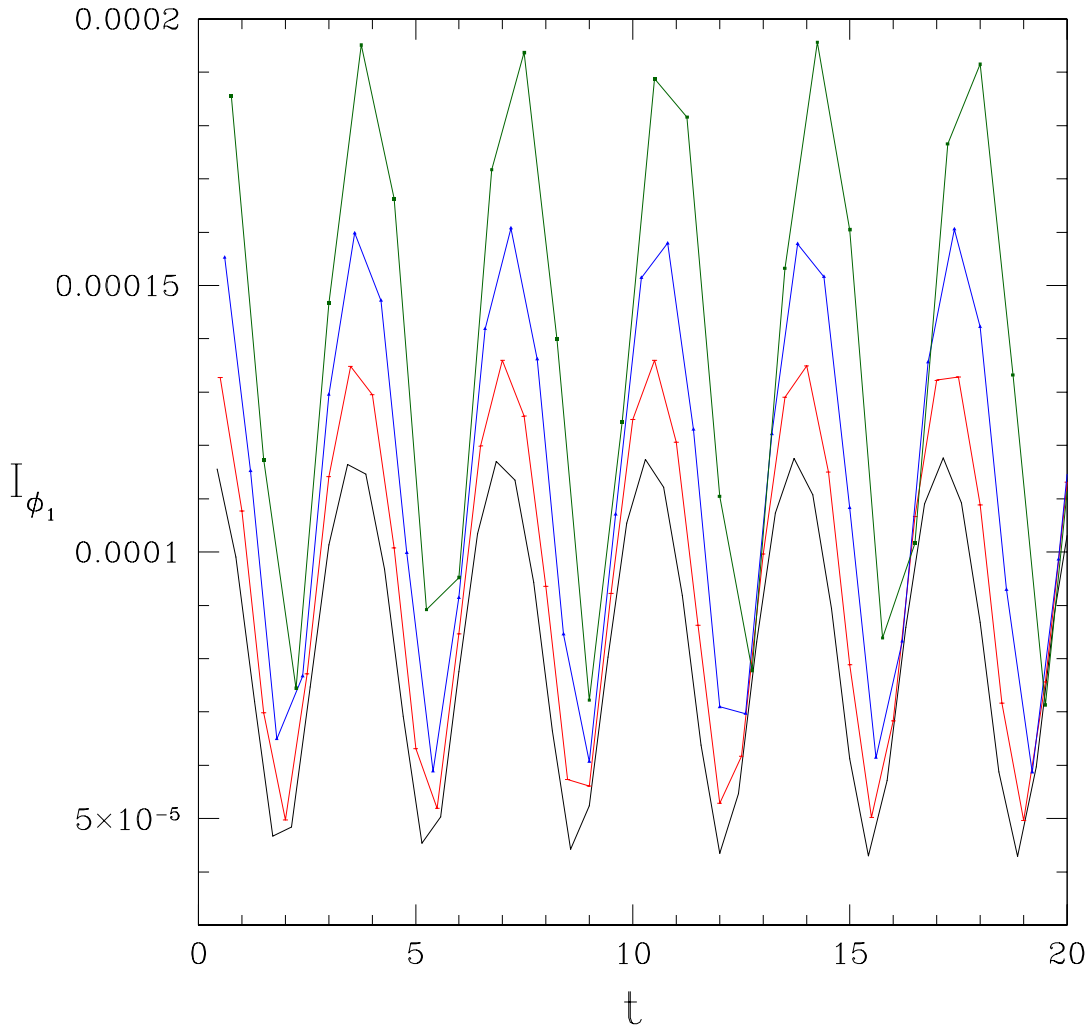


**Figure 4.27:** Short term $I_{\phi_1}(t)$ for the orbital dynamics with initial speed $v_x = 0.09$. This figure shows the independent residual evaluation for the orbital motion presented in the Fig. 4.24 for $0 \leq t \leq 20$. It shows the same results as Fig. 4.26 but for a shorter time interval. Note that the shape of the $I_{\phi_1}(t)$ is kept approximately the same for all four grid resolutions showed in this figure.

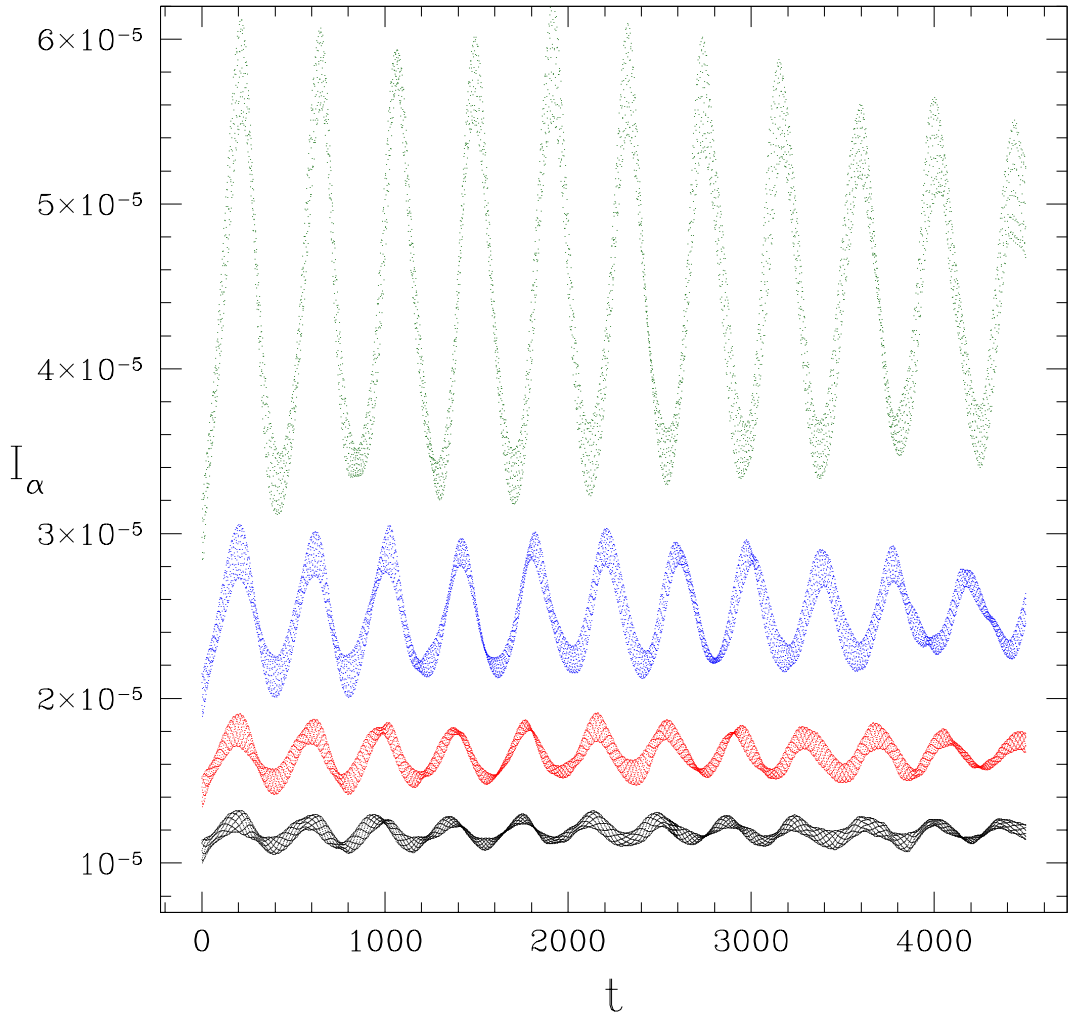


**Figure 4.28:** Long term $I_\alpha(t)$ during the orbital dynamics with initial speed $v_x = 0.09$. This figure shows the independent residual $l2$-norm $I_\alpha(t)$ for the lapse function $\alpha$ during the orbital motion presented in the Fig. 4.24 for $0 \leq t \leq 4500$. The different coulored dots represent $I_\alpha(t)$ function calculated with different grid resolutions; The number of grid points per edge of the cubic domain is $n_1 = 65$ (green dots), $n_2 = 81$ (blue), $n_3 = 97$ (red) and $n_4 = 113$ (black) with corresponding mesh spacings of $h_1 = 1.875$, $h_2 = 1.5$, $h_3 \simeq 1.237$ and $h_4 \simeq 1.062$. The convergence rate expected for this evaluation is $O(h^2)$. A rough estimate indicate that the average value goes to zero as $(4.5 \rightarrow 2.5 \rightarrow 1.64 \rightarrow 1.18) \times 10^{-5}$. When multiplied by the quadratic conversion factors, $h_{12}^2 = (h_1/h_2)^2 = (1.25)^2$, $h_{13}^2 = (h_1/h_3)^2 \simeq (1.516)^2$ and $h_{14}^2 = (h_1/h_4)^2 \simeq (1.766)^2$ the same sequence of average $I_\alpha$ becomes almost coincident: $(4.5 \simeq 3.9 \simeq 3.8 \simeq 3.7) \times 10^{-5}$. This is a good indication that the quadratic convergence is attained throughout this evolution.

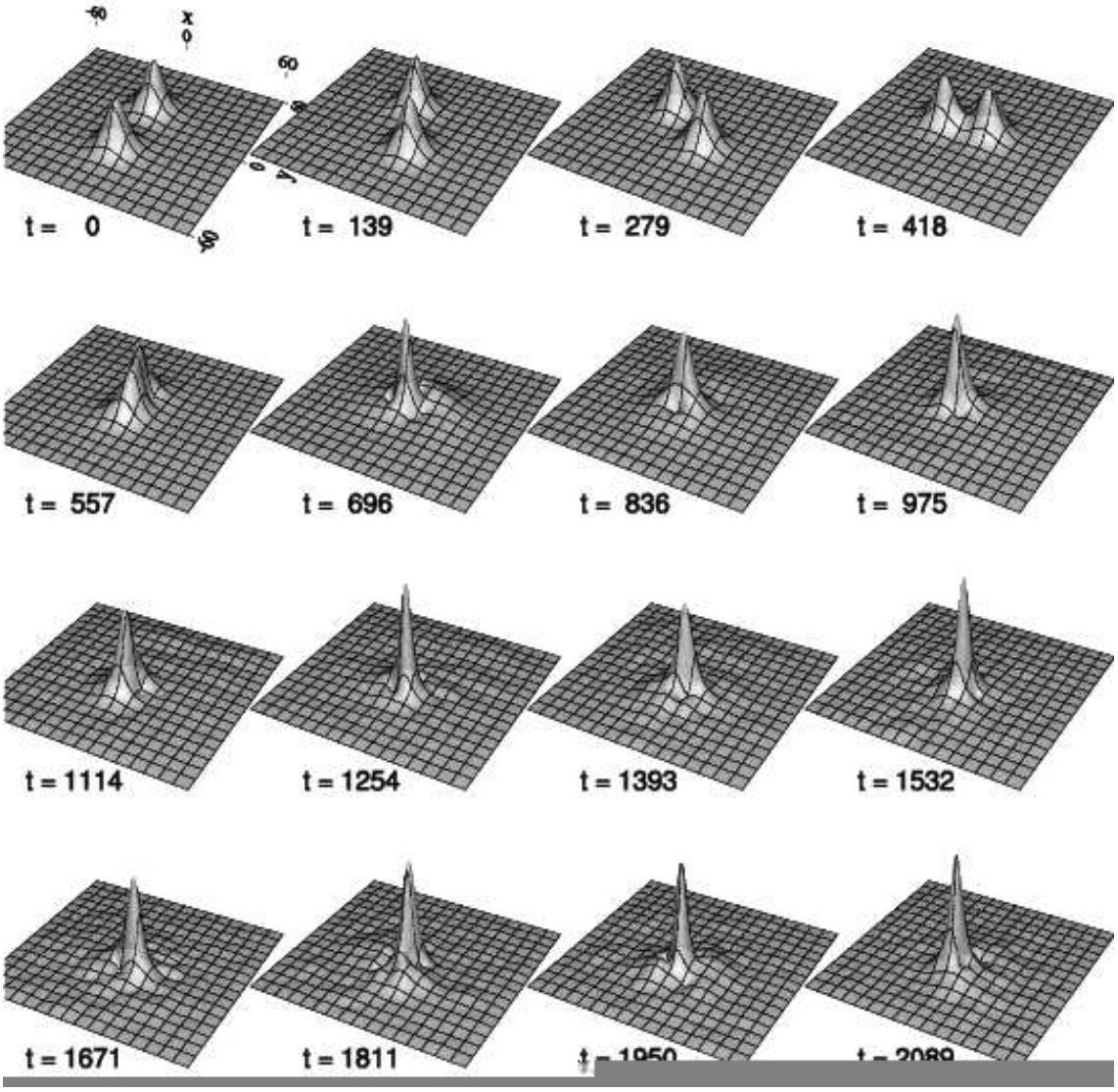


**Figure 4.29:** Orbital Evolution of two $\phi_0 = 0.02$ boson stars with initial speed $v_x = 0.07$. This plot shows a long term time evolution of two spherically symmetric stars used as initial data. The stars are both parametrized by a central value field of $\phi_0 = 0.02$. They are centred in the cubic numerical domain at $(0, 20, 0)$ and $(0, -20, 0)$. The initial star velocities are such that $v_x = -v_x^{(1)} = v_x^{(2)} = 0.07$. The star orbits approach each other around $t = 418$, initiating a grazing collision. The stars quickly plunge and merge into a another rotating boson star. Throughout the evolution the complex scalar field modulo is bounded approximately to $0.0 \leq |\phi(t, x, y, z)| \lesssim 0.05$.

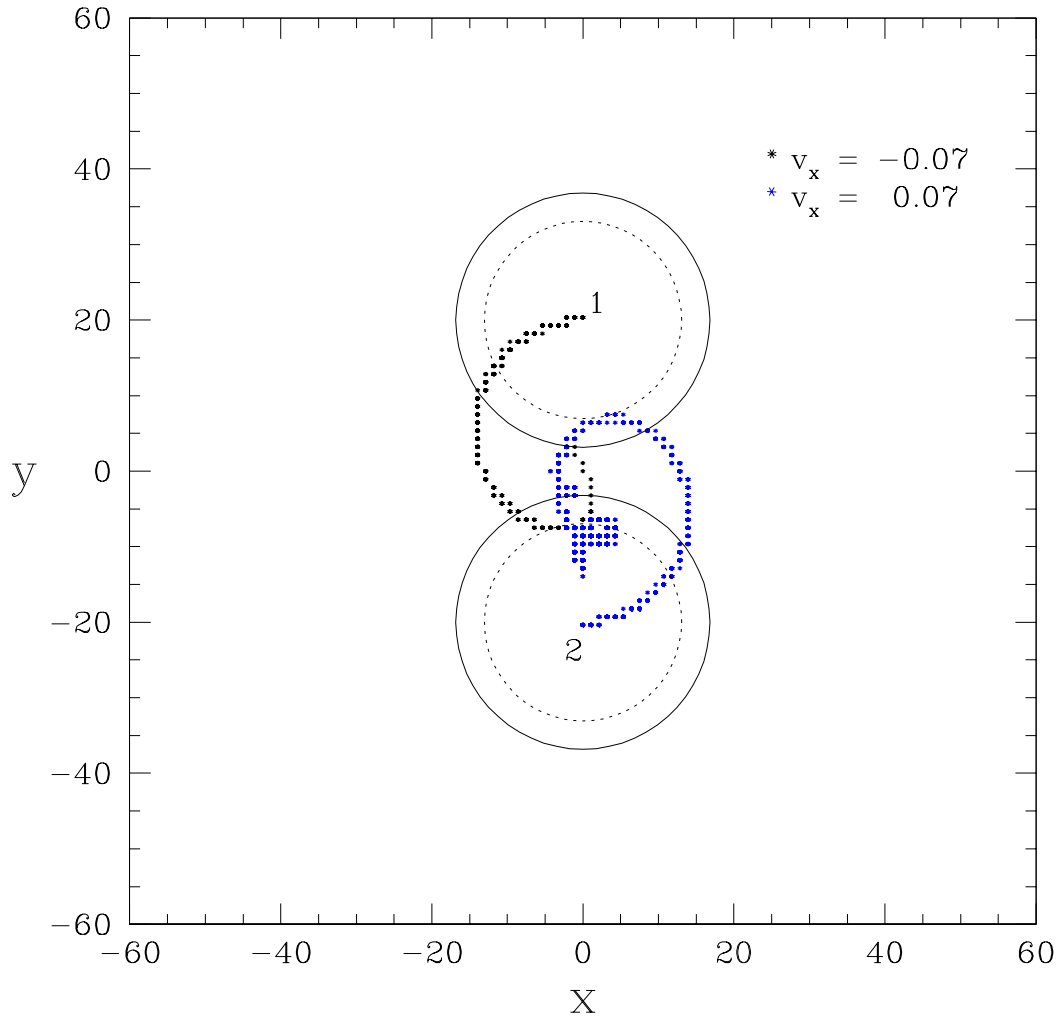


**Figure 4.30:** Two-star trajectory for initial speed $v_x = 0.07$ and coordinate separation $d = 40$. This plot shows the trajectories of stars 1 and 2 centred at $(0, 20, 0)$ and $(0, -20, 0)$ for $0 \leq t \leq 2250$. It corresponds to the same orbital motion shown on Fig. 4.29. At $t = 0$, the solid lines represent the circumference of the stars at $R_{99}$, the radius containg 99% of its ADM mass $M_{ADM}$, while the dotted lines draw the star surface at $R_{95}$, the radius containg 95% of its ADM mass $M_{ADM}$. Both stars are members of the stable branch family of solutions and are parametrized by $\phi_0(0) = 0.02$. The initial stars velocities are such that $v_x = -v_x^{(1)} = v_x^{(2)}$. Note that after a short period of orbital dynamics the stars trajectories approach each other close enough for the stellar matter interact and initiate a collision phase. The end state of the collision is a spinning boson star.

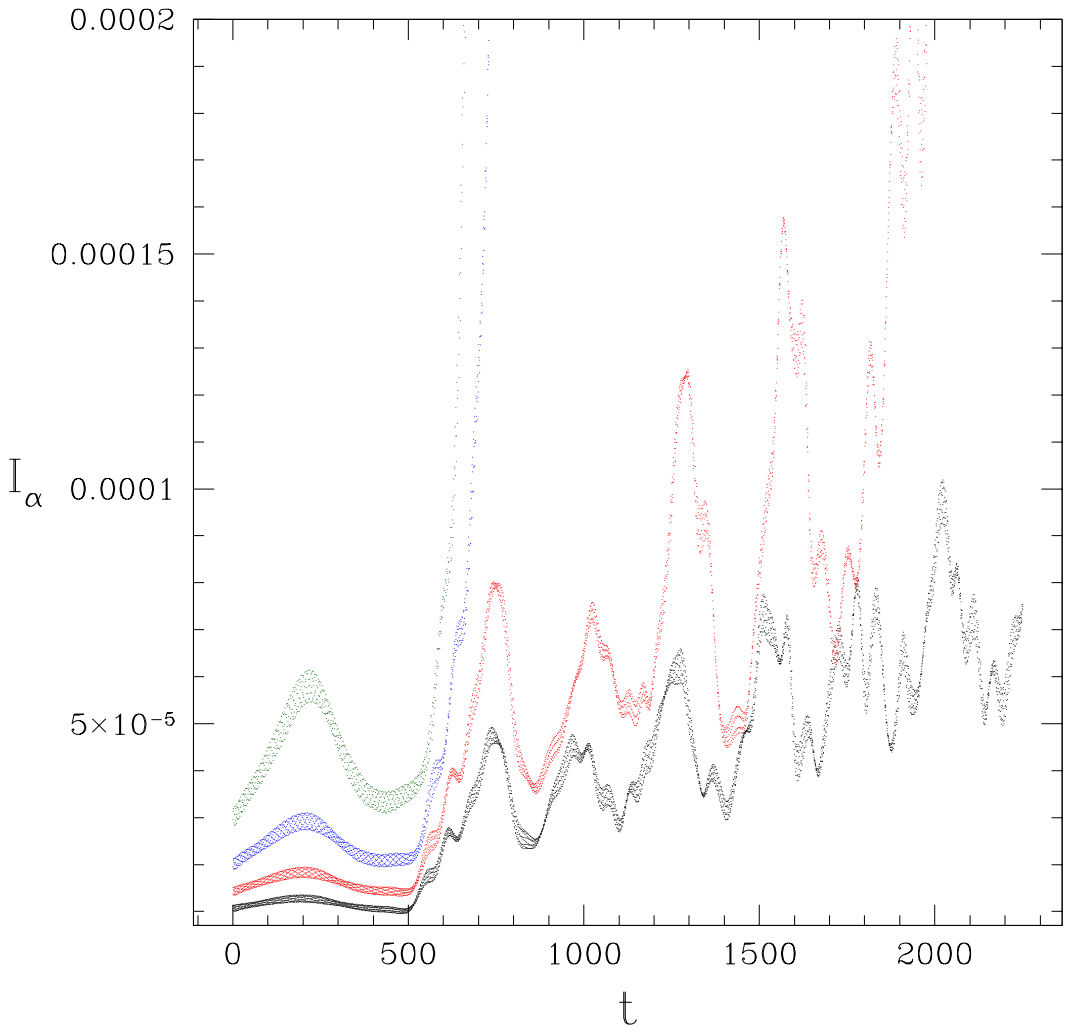


**Figure 4.31:** Long term $I_\alpha(t)$ during the orbital dynamics with initial speed $v_x = 0.07$. This figure shows the independent residual $l2$-norm $I_\alpha(t)$ for the lapse function $\alpha$ during the orbital motion presented in the Fig. 4.29 for $0 \leq t \leq 2250$. The different coulored dots represent $I_\alpha(t)$ function calculated with different grid resolutions; Grids $G_1$, $G_2$, $G_3$ and $G_4$ have corresponding mesh spacings of $h_1 = 1.875$, $h_2 = 1.5$, $h_3 \simeq 1.237$ and $h_4 \simeq 1.062$. The convergence rate expected for this evaluation is $O(h^2)$. Note that the convergence deteriorate in the merging phase. The $G_1$ and $G_2$ grids are too coarse to resolve the merger of the two stars. They leave the converging regime for $t \gtrsim 600$. The same happens with $G_3$ grid at $t \gtrsim 1500$ as the numerical domain becomes polluted by scalar matter and spurious reflections from the boundaries.

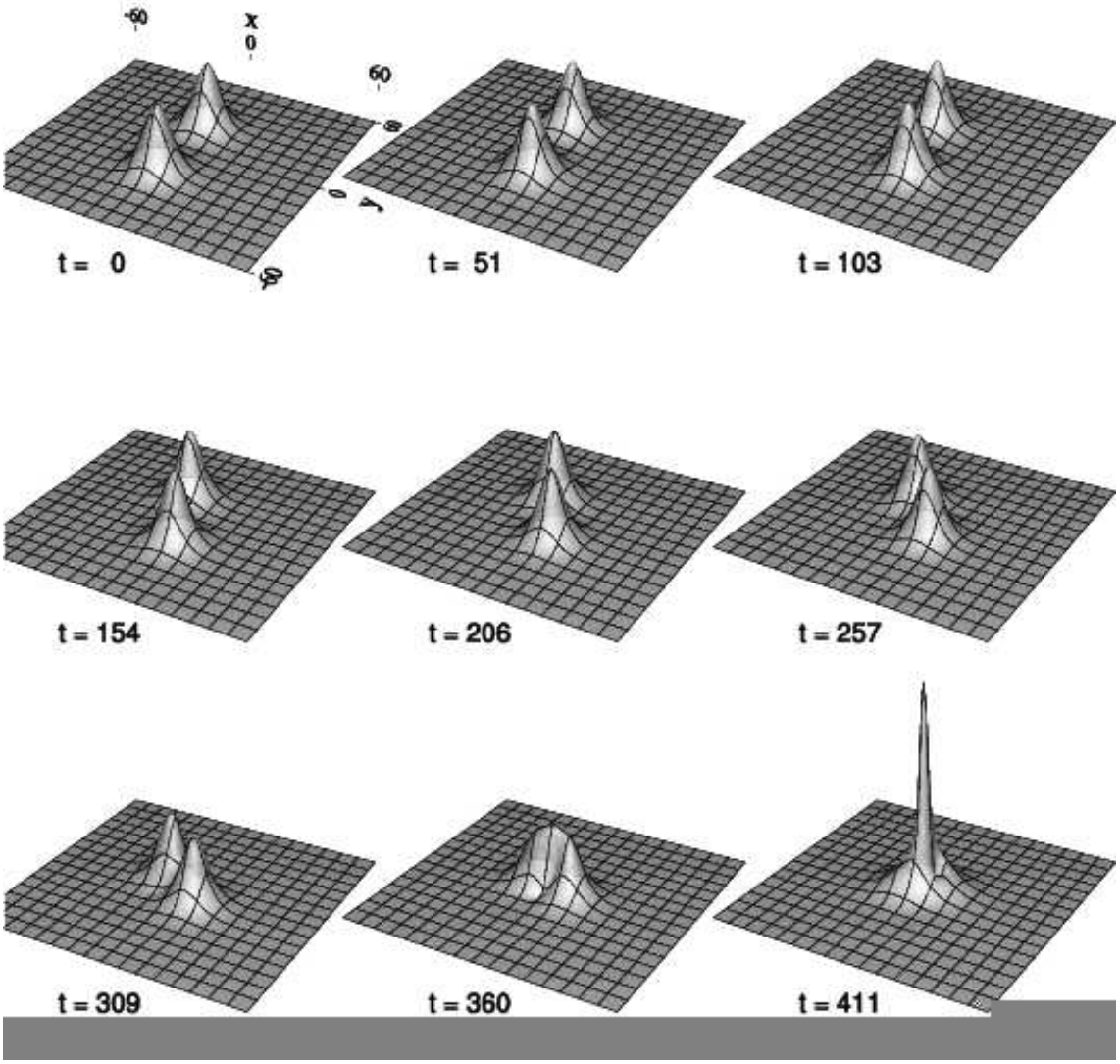


**Figure 4.32:** Orbital Evolution of two $\phi_0 = 0.02$ boson stars with initial speed $v_x = 0.05$. This plot shows a long term time evolution of two spherically symmetric star parametrized by a central value field of $\phi_0 = 0.02$. They are centred in the cubic numerical domain at $(0, 20, 0)$ and $(0, -20, 0)$. The initial star velocities are such that $v_x = -v_x^{(1)} = v_x^{(2)} = 0.05$. The stars barely orbit each other before they plunge and merge into a compact object believed to be a black-hole. The complex scalar field modulo reaches a maximum during the evolution of $\max|\phi(t, x, y, z)| \simeq 0.14$. For comparison, $\phi_0 \simeq 0.08$ represents a stable boson star with the maximum possible mass before being driven to collapse into a black-hole.

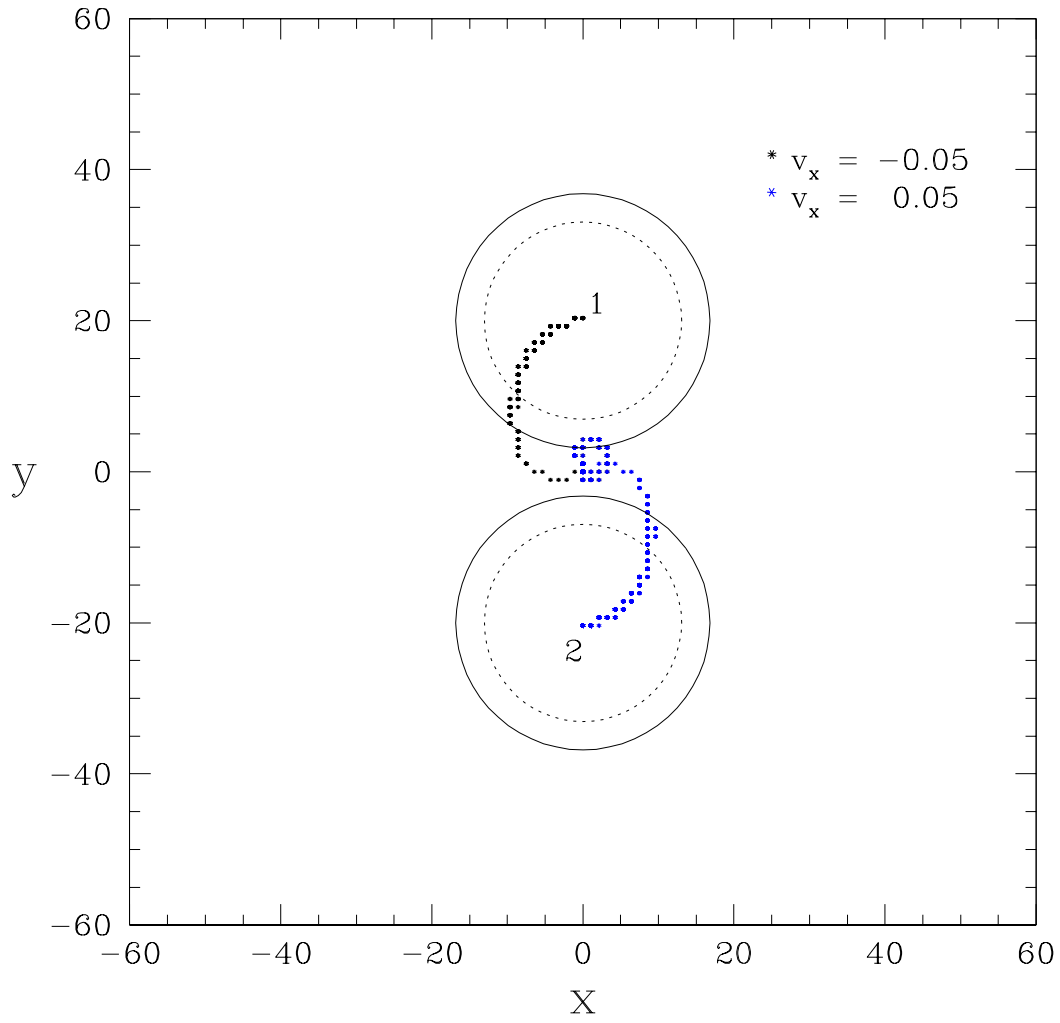


**Figure 4.33:** Two-star trajectory for initial speed $v_x = 0.05$ and coordinate separation $d = 40$. This plot shows the trajectories of stars 1 and 2 centred at $(0, 20, 0)$ and $(0, -20, 0)$ for $0 \leq t \leq 1500$. This trajectories corresponds to the same orbital motion displayed on Fig. 4.32. At $t = 0$, the solid lines represent the circumference of the stars at $R_{99}$, the radius containg 99% of its ADM mass $M_{ADM}$, while the dotted lines draw the star surface at $R_{95}$, the radius containg 95% of its ADM mass $M_{ADM}$. Both stars are members of the stable branch family of solutions and are parametrized by $\phi_0(0) = 0.02$. The initial stars velocities are such that $v_x = -v_x^{(1)} = v_x^{(2)}$. Note that after a short period of orbital dynamics the stars trajectories approach each other close enough for the stellar matter interact and initiate a collision phase. The final merger happens and it is believed to result into a black-hole.

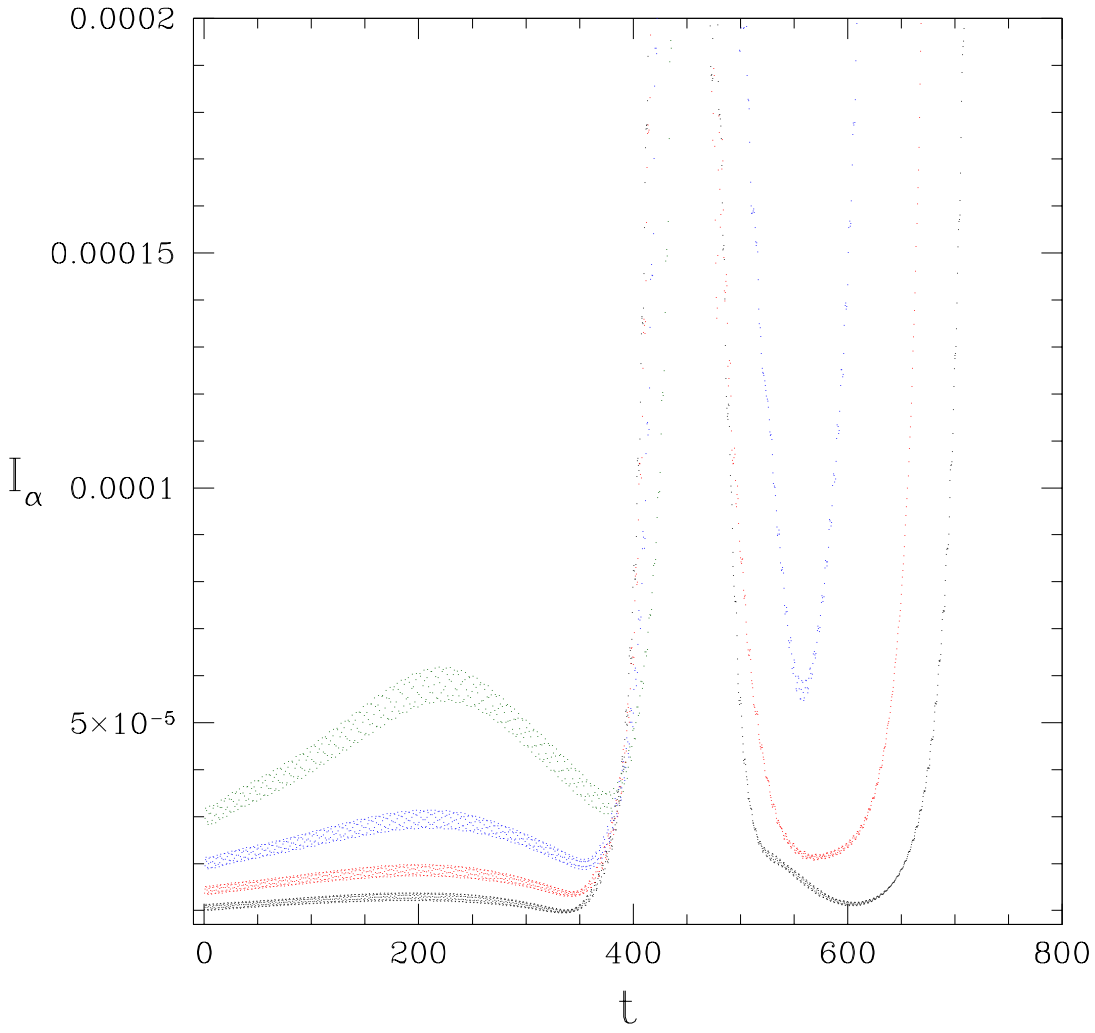


**Figure 4.34:** Long term $I_\alpha(t)$ during the orbital dynamics with initial speed $v_x = 0.05$. This figure shows the independent residual $l2$-norm $I_\alpha(t)$ for the lapse function $\alpha$ during the orbital motion presented in the Fig. 4.32 for $0 \leq t \leq 411$. The different coulored dots represent $I_\alpha(t)$ function calculated with different grid resolutions; The number of grid points per edge of the cubic domain is $n_1 = 65$ (green dots), $n_2 = 81$ (blue), $n_3 = 97$ (red) and $n_4 = 113$ (black) with corresponding mesh spacings of $h_1 = 1.875$, $h_2 = 1.5$, $h_3 \simeq 1.237$ and $h_4 \simeq 1.062$. The convergence rate expected for this evaluation is $O(h^2)$. Note that this convergence rate is roughly satisfied until $t \simeq 380$ when the final merged stars collapse into a black-hole.

## 4.7 Discussions and Further Developments

It is worth noticing that the second and third experiments shown last section are qualitatively consistent with the results obtained by Palenzuela et. al. [7]. They have simulated a fully relativistic implementation of the Einstein-Klein-Gordon system having as initial data two boosted spherically symmetric boson stars. The stars used in their simulations though had an ADM mass of $M_{ADM} = 0.5$ while the ones used here, parametrized by $\phi_0 = 0.02$, had masses of $M_{ADM} \simeq 0.475$. Also they used a closer separation for the stars initial configuration, $d = 32$, while here the stars were further away apart,$d = 40$, in order to avoid that their "surfaces" touch each other considering that each star has $R_{99} = 17.25$. Despite these differences some qualitative comparison was possible. What they claim to be a spinning bar with dispersal of scalar matter agree with the result of the second experiment shown in this section where the final object was a rotating boson star. The range of initial velocities that they claim would result into black-hole formation is approximately the range of possible black-hole formation, for boost velocity of $0 \leq v_x < 0.07$, in which the third experiment was shown as an example. Those authors also reported an interesting case of the dynamics in which the stars merge, keep themselves together as one lump of deformed or perturbed scalar matter for a while and eventually decouple from each other and follow trajectories in opposite directions. As far as the experiments performed for this thesis goes, this case was not observed. The novelty of the work done here though comes from the first experiment. It is the first time that more than two orbits of the boson star binary is reported. Furthermore, the trajectories observed clearly seem to precess, a good indication that the system preserved the main features of the strong gravitational field.

These experiments just started to investigate the parameter space of orbital dynamics. A full parameter space survey would have to take into account all possible ratios between *dimensional* parameters of the system in such a way to fully determine and characterize the system by a set of *adimensional* parameters. For the orbital motion of two non-spinning stars, the dimensional[5] parameters involved are the following: The masses of the two stars $M_1$ and $M_2$; the coordinate separation between their centers, $d$; the velocity $v$ of each star. Note however that actually only the velocity of one of the stars is necessary since the other can be arbitrarily fixed by the coordinate choice. Therefore the total number of dimensional parameters is 6 while the total number of adimensional parameters that can be formed from this set is given by the Newton binomial coefficient $C_2^6 = 15$. Unfortunately, this is a huge parameter space to be explored. Howeverm, if the study of the orbital motion is restricted to a plane then the number of dimensional parameters is reduced to 4 and the number of adimensional ones to $C_2^4 = 6$. Despite this parameter space being still big, it looks more promising to be explored. The fully relativistic 3D numerical simulation of the binaries are computationally too expensive. The CFC approximation provides a set of simpler PDEs that shows promising results in capturing the same phenomena reported in the fully relativistic case. It is expected then that once this code is callibrated with the fully relativistic one it will provide results faster turning this code into a perfect tool to survey the huge parameter space of the orbital motion.

Concerning the comparison of the Wilson-Mathews compression effect reported on the neutron

[5] recall that the system of units used here is the planck units in which $\hbar = G = c = 1$

star binary case, the results shown here are inconclusive with respect to the existence or not of this effect on boson stars. In one of the experiments ($v_x = 0.08$), the strong gravitational field did appear to compress one of the stars. However, due to the low resolution in which this particular simulation was run, this result was not trustworthy and therefore not explored in details in this thesis. Besides a resolution increase, it may be necessary to adopt a different self-interaction potential $U(|\phi|^2)$ in order to observe this phenomena in the same way that for only certain neutron star equations of state this phenomena was evident. Further studies are still required.

To wrap up this section, a few comments on the future developments of this numerical code are welcome. First and foremost, resolution is the key that is missing to study the most interesting parts of the star dynamics. More sophisticated numerical techniques needs to be implemented in order to use the available computational resources more effectively. Crutial to this purpose, the parallelization of the code is mandatory. This consists in decomposing the numerical domain and sharing the work load for obtaining the solution with several different processing units with their own local memory and CPU. Besides parallelization, applying a technique called adaptive mesh refinement (AMR) provides resolution where it is needed. The philosophy behind AMR consists of increasing the number of grid points only where and when the detailed solution features needs to be resolved. Both techniques together provide the means for a substantial increase in resolution. At the moment the unigrid code used to obtain the results presented in this chapter has been adapted to both techniques. It is still in its testing and debugging phase though.

A second proposal for code improvement involves the implementation of boundary conditions that approximate better the physical boundary condition, viz the aysmptotically flat spacetime. As pointed out on Sec. 2.4.5 an exact implementation of the physical boundary conditions could be implemented via compactification of the spatial domain. This option, despite promising, would involve a complete revamp of the equations of motion besides the development of a more sophisticated multigrid solver to deal with the anisotropic aspect of the equations. A more suited possibility though consists of implementing boundary conditions that approximate the fall-off to infinity of the various quantities, the so called Sommerfeld boundary conditions. This implementation is the next priority in the development of this code.

# CHAPTER 5

# CONCLUSION

The main result presented in this thesis consists of an implementation of a $3d$ numerical code to solve a set of coupled elliptic-hyperbolic partial differential equations describing the Einstein-Klei-Gordon system under conformally flat condition. This code was thoroughly tested with satisfactory results regarding the correctness of its implementation. Moreover, this code was applied to the study of orbital dynamics of mini-boson stars. Solitonic behaviour of the solution for head-on collisions has been confirmed according to predictions from previous studies of the system under newtonian approximation [140] and in axisymmetric relativity [107]. Three main results should also be mentioned. They consist of the three final possibilities discovered for the boson star orbital dynamics: an eternal, precessing elliptic orbit; the merger of the two stars into one final, more compact boson star; and the merger into a black-hole. These results are consistent with the ones obtained by Palenzuela et. al. [7] with a fully relativistic set of equations. This suggests that the approximation used in this thesis may be accurate enough to capture the main features of the strong gravitational field.

One main drawback on the simulations presented in this thesis concerns their grid resolution. Most of the runs had a reasonable resolution to allow conclusions about the physics of the system to be stated. However, in some cases the gravitational field was really strong to the point to cause the stars to collapse into black-holes. In those cases further conclusions were limited due to the poor resolution of the solution. In order to resolve the features of the solution, more grid points should be added where it is needed. This philosophy is the base of a technique called *adaptive mesh refinement* (AMR) [141, 142, 143]. It creates a hierarchy of grids with finer and finer resolutions covering regions of the domain where detailed features of the solution needs to be resolved. Besides AMR, the numerical domain decomposition and distribution of the work load to several different CPUs, also known as code parallelization, allows a substantial increase in resolution of the simulations as well. At the moment the unigrid code used in this thesis has been adapted to use these two techniques but it is still in its test and debugging phase.

Calibration under what conditions CFC approximation is a satisfactory approximation of GR and when is not is the next priority. This opens an exciting direction of research that can shed some light on how to identify the true degrees of freedom of GR, or in other words on the identification of the gravitational radiation. Besides, it can clarify what is really left out in applying the conformally flat condition when compared to the fully general relativistic solution.

# BIBLIOGRAPHY

[1] Keita Kawabe. Status of LIGO. *J. Phys. Conf. Ser.*, 120:032003, 2008.

[2] F. Acernese et al. The status of Virgo. *J. Phys. Conf. Ser.*, 110:062025, 2008.

[3] R. Takahashi et al. Operational status of TAMA300 with the Seismic Attenuation System (SAS). *Class. Quant. Grav.*, 25:114036, 2008.

[4] Benno Willke. GEO600: Status and plans. *Class. Quant. Grav.*, 24:S389–S397, 2007.

[5] Curt Cutler and Eanna E. Flanagan. Gravitational waves from merging compact binaries: How accurately can one extract the binary's parameters from the inspiral wave form? *Phys. Rev.*, D49:2658–2697, 1994.

[6] C. Palenzuela, I. Olabarrieta, L. Lehner, and S. Liebling. Head-on collisions of boson stars. *Phys. Rev.*, D75:064005, 2007.

[7] C. Palenzuela, L. Lehner, and S. L. Liebling. Orbital Dynamics of Binary Boson Star Systems. *Phys. Rev.*, D77:044036, 2008.

[8] Gregory B. Cook, Stuart L. Shapiro, and Saul A. Teukolsky. Testing a simplified version of einstein's equations for numerical relativity. *Phys. Rev. D*, 53(10):5533–5540, May 1996.

[9] Willy Kley and Gerhard Schaefer. Relativistic dust disks and the Wilson-Mathews approach. *Phys. Rev.*, D60:027501, 1999.

[10] Harald Dimmelmeier, Jose A. Font, and Ewald Muller. Relativistic simulations of rotational core collapse. I. Methods, initial models, and code tests. *Astron. Astrophys.*, 388:917–935, 2002.

[11] Harald Dimmelmeier, Jose A. Font, and Ewald Muller. Gravitational waves from relativistic rotational core collapse. *Astrophys. J.*, 560:L163–L166, 2001.

[12] James A. Isenberg. Waveless approximation theories of gravity. *Int. J. Mod. Phys.*, D17:265–273, 2008.

[13] J. R. Wilson and G. J. Mathews. Instabilities in close neutron star binaries. *Phys. Rev. Lett.*, 75:4161–4164, 1995.

[14] J. R. Wilson, G. J. Mathews, and P. Marronetti. Relativistic numerical model for close neutron star binaries. *Phys. Rev.*, D54:1317–1331, 1996.

[15] G. J. Mathews and J. R. Wilson. Binary induced neutron-star compression, heating, and collapse. *Astrophys. J.*, 482:929–941, 1997.

[16] G. J. Mathews, P. Marronetti, and J. R. Wilson. Relativistic studies of close neutron star binaries. 1997.

[17] G. J. Mathews, P. Marronetti, and J. R. Wilson. Relativistic hydrodynamics in close binary systems: Analysis of neutron-star collapse. *Phys. Rev. D*, 58(4):043003, Jul 1998.

[18] Pedro Marronetti, Grant J. Mathews, and James R. Wilson. Binary neutron stars systems: Irrotational quasi- equilibrium sequences. 1998.

[19] P. Marronetti, G. J. Mathews, and J. R. Wilson. Binary neutron-star systems: From the newtonian regime to the last stable orbit. *Phys. Rev. D*, 58(10):107503, Oct 1998.

[20] P. Marronetti, G. J. Mathews, and J. R. Wilson. Irrotational binary neutron stars in quasiequilibrium. *Phys. Rev. D*, 60(8):087301, Sep 1999.

[21] Éanna É. Flanagan. Possible explanation for star-crushing effect in binary neutron star simulations. *Phys. Rev. Lett.*, 82(7):1354–1357, Feb 1999.

[22] G. J. Mathews and J. R. Wilson. Revised relativistic hydrodynamical model for neutron-star binaries. *Phys. Rev.*, D61:127304, 2000.

[23] James R. Wilson. Precompression of colliding neutron stars. *Phys. Rev.*, D66:084015, 2002.

[24] Mark A. Miller. General relativistic decompression of binary neutron stars during inspiral. *Phys. Rev.*, D75:024001, 2007.

[25] Marc Favata. Are neutron stars crushed? Gravitomagnetic tidal fields as a mechanism for binary-induced collapse. *Phys. Rev.*, D73:104005, 2006.

[26] Masaru Shibata, Koji Uryu, and John L. Friedman. Deriving formulations for numerical computation of binary neutron stars in quasicircular orbits. *Phys. Rev.*, D70:044044, 2004.

[27] Gerhard Schaefer and Achamveedu Gopakumar. A minimal no-radiation approximation to Einstein's field equations. *Phys. Rev.*, D69:021501, 2004.

[28] Koji Uryu, Francois Limousin, John L. Friedman, Eric Gourgoulhon, and Masaru Shibata. Binary neutron stars in a waveless approximation. *Phys. Rev. Lett.*, 97:171101, 2006.

[29] R. Arnowitt, S. Deser, and C. W. Misner. The dynamics of general relativity. In L. Witten, editor, *Gravitation: An Introduction to Current Research*. New York, Wiley, (1962).

[30] A. Lichnerowicz. L'intégration des équations de la gravitation relativiste et le problème des n corps. *J. Math. Pures et Appl.*, **23**:37, (1944).

[31] Frederic A. Rasio and Stuart L. Shapiro. Coalescing binary neutron stars. *Class. Quant. Grav.*, 16:R1–R29, 1999.

[32] T.W. Baumgarte and S.L. Shapiro. Numerical relativity and compact binaries. *Phys. Rep.*, **376**:41–131, (2003).

[33] Frans Pretorius. Binary Black Hole Coalescence. 2007.

[34] L. Smarr, A. Cadez, Bryce S. DeWitt, and K. Eppley. Collision of Two Black Holes: Theoretical Framework. *Phys. Rev.*, D14:2443–2452, 1976.

[35] Peter Anninos, David Hobill, Edward Seidel, Larry Smarr, and Wai-Mo Suen. The Collision of two black holes. *Phys. Rev. Lett.*, 71:2851–2854, 1993.

[36] G. B. Cook et al. Boosted three-dimensional black-hole evolutions with singularity excision. *Phys. Rev. lett.*, 80:2512–2516, 1998.

[37] Bernd Bruegmann, Wolfgang Tichy, and Nina Jansen. Numerical simulation of orbiting black holes. *Phys. Rev. Lett.*, 92:211101, 2004.

[38] Frans Pretorius. Evolution of Binary Black Hole Spacetimes. *Phys. Rev. Lett.*, 95:121101, 2005.

[39] Frans Pretorius. Simulation of binary black hole spacetimes with a harmonic evolution scheme. *Class. Quant. Grav.*, 23:S529–S552, 2006.

[40] Manuela Campanelli, C. O. Lousto, P. Marronetti, and Y. Zlochower. Accurate evolutions of orbiting black-hole binaries without excision. *Phys. Rev. Lett.*, 96:111101, 2006.

[41] Manuela Campanelli, C. O. Lousto, and Y. Zlochower. The last orbit of binary black holes. *Phys. Rev.*, D73:061501, 2006.

[42] Y. Zlochower, J. G. Baker, Manuela Campanelli, and C. O. Lousto. Accurate black hole evolutions by fourth-order numerical relativity. *Phys. Rev.*, D72:024021, 2005.

[43] John G. Baker, Joan Centrella, Dae-Il Choi, Michael Koppitz, and James van Meter. Gravitational wave extraction from an inspiraling configuration of merging black holes. *Phys. Rev. Lett.*, 96:111102, 2006.

[44] John G. Baker, Joan Centrella, Dae-Il Choi, Michael Koppitz, and James van Meter. Binary black hole merger dynamics and waveforms. *Phys. Rev.*, D73:104002, 2006.

[45] Bela Szilagyi, Denis Pollney, Luciano Rezzolla, Jonathan Thornburg, and Jeffrey Winicour. An explicit harmonic code for black-hole evolution using excision. *Class. Quant. Grav.*, 24:S275–S293, 2007.

[46] Peter Diener et al. Accurate evolution of orbiting binary black holes. *Phys. Rev. Lett.*, 96:121101, 2006.

[47] Bernd Bruegmann et al. Calibration of Moving Puncture Simulations. *Phys. Rev.*, D77:024027, 2008.

[48] John G. Baker, Manuela Campanelli, Frans Pretorius, and Yosef Zlochower. Comparisons of binary black hole merger waveforms. *Class. Quant. Grav.*, 24:S25–S31, 2007.

[49] Carlos F. Sopuerta, Nicolas Yunes, and Pablo Laguna. Gravitational recoil velocities from eccentric binary black hole mergers. *Astrophys. J.*, 656:L9–L12, 2007.

[50] John G. Baker et al. Getting a kick out of numerical relativity. *Astrophys. J.*, 653:L93–L96, 2006.

[51] Jose A. Gonzalez, Ulrich Sperhake, Bernd Bruegmann, Mark Hannam, and Sascha Husa. Total recoil: the maximum kick from nonspinning black-hole binary inspiral. *Phys. Rev. Lett.*, 98:091101, 2007.

[52] Frank Herrmann, Ian Hinder, Deirdre Shoemaker, and Pablo Laguna. Unequal mass binary black hole plunges and gravitational recoil. *Class. Quant. Grav.*, 24:S33–S42, 2007.

[53] Dae-Il Choi et al. Recoiling from a kick in the head-on collision of spinning black holes. *Phys. Rev.*, D76:104026, 2007.

[54] Frank Herrmann, Ian Hinder, Deirdre M. Shoemaker, Pablo Laguna, and Richard A. Matzner. Binary Black Holes: Spin Dynamics and Gravitational Recoil. *Phys. Rev.*, D76:084032, 2007.

[55] Pedro Marronetti, Wolfgang Tichy, Bernd Bruegmann, Jose Gonzalez, and Ulrich Sperhake. High-spin binary black hole mergers. *Phys. Rev.*, D77:064010, 2008.

[56] John G. Baker et al. Modeling kicks from the merger of non-precessing black- hole binaries. *Astrophys. J.*, 668:1140–1144, 2007.

[57] J. A. Gonzalez, M. D. Hannam, U. Sperhake, Bernd Bruegmann, and S. Husa. Supermassive kicks for spinning black holes. *Phys. Rev. Lett.*, 98:231101, 2007.

[58] Bernd Bruegmann, Jose A. Gonzalez, Mark Hannam, Sascha Husa, and Ulrich Sperhake. Exploring black hole superkicks. *Phys. Rev.*, D77:124047, 2008.

[59] Manuela Campanelli, C. O. Lousto, and Y. Zlochower. Gravitational radiation from spinning-black-hole binaries: The orbital hang up. *Phys. Rev.*, D74:041501, 2006.

[60] Manuela Campanelli, C. O. Lousto, and Yosef Zlochower. Spin-orbit interactions in black-hole binaries. *Phys. Rev.*, D74:084023, 2006.

[61] Manuela Campanelli, Carlos O. Lousto, Yosef Zlochower, Badri Krishnan, and David Merritt. Spin Flips and Precession in Black-Hole-Binary Mergers. *Phys. Rev.*, D75:064030, 2007.

[62] Manuela Campanelli, Carlos O. Lousto, Yosef Zlochower, and David Merritt. Maximum gravitational recoil. *Phys. Rev. Lett.*, 98:231102, 2007.

[63] Ian Hinder, Birjoo Vaishnav, Frank Herrmann, Deirdre Shoemaker, and Pablo Laguna. Universality and Final Spin in Eccentric Binary Black Hole Inspirals. *Phys. Rev.*, D77:081502, 2008.

[64] Ulrich Sperhake et al. Eccentric binary black-hole mergers: The transition from inspiral to plunge in general relativity. *Phys. Rev.*, D78:064069, 2008.

[65] Frans Pretorius and Deepak Khurana. Black hole mergers and unstable circular orbits. *Class. Quant. Grav.*, 24:S83–S108, 2007.

[66] Matthew C. Washik et al. Binary Black Hole Encounters, Gravitational Bursts and Maximum Final Spin. *Phys. Rev. Lett.*, 101:061102, 2008.

[67] John G. Baker et al. Binary black hole late inspiral: Simulations for gravitational wave observations. *Phys. Rev.*, D75:124024, 2007.

[68] Alessandra Buonanno, Gregory B. Cook, and Frans Pretorius. Inspiral, merger and ring-down of equal-mass black-hole binaries. *Phys. Rev.*, D75:124018, 2007.

[69] Emanuele Berti et al. Inspiral, merger and ringdown of unequal mass black hole binaries: A multipolar analysis. *Phys. Rev.*, D76:064034, 2007.

[70] Mark Hannam, Sascha Husa, Ulrich Sperhake, Bernd Bruegmann, and Jose A. Gonzalez. Where post-Newtonian and numerical-relativity waveforms meet. *Phys. Rev.*, D77:044020, 2008.

[71] John G. Baker et al. Mergers of non-spinning black-hole binaries: Gravitational radiation characteristics. *Phys. Rev.*, D78:044046, 2008.

[72] Mark A. Scheel et al. Solving Einstein's equations with dual coordinate frames. *Phys. Rev.*, D74:104006, 2006.

[73] Harald P. Pfeiffer et al. Reducing orbital eccentricity in binary black hole simulations. *Class. Quant. Grav.*, 24:S59–S82, 2007.

[74] Michael Boyle et al. High-accuracy comparison of numerical relativity simulations with post-Newtonian expansions. *Phys. Rev.*, D76:124038, 2007.

[75] Ulrich Sperhake, Vitor Cardoso, Frans Pretorius, Emanuele Berti, and Jose A. Gonzalez. The high-energy collision of two black holes. *Phys. Rev. Lett.*, 101:161101, 2008.

[76] Masaru Shibata, Hirotada Okawa, and Tetsuro Yamamoto. High-velocity collision of two black holes. *Phys. Rev.*, D78:101501, 2008.

[77] Masaru Shibata and Koji Uryu. Simulation of merging binary neutron stars in full general relativity: $\Gamma = 2$ case. *Phys. Rev.*, D61:064001, 2000.

[78] Masaru Shibata and Koji Uryu. Gravitational waves from the merger of binary neutron stars in a fully general relativistic simulation. *Prog. Theor. Phys.*, 107:265, 2002.

[79] Masaru Shibata, Keisuke Taniguchi, and Koji Uryu. Merger of binary neutron stars of unequal mass in full general relativity. *Phys. Rev.*, D68:084020, 2003.

[80] Masaru Shibata, Keisuke Taniguchi, and Koji Uryu. Merger of binary neutron stars with realistic equations of state in full general relativity. *Phys. Rev.*, D71:084021, 2005.

[81] Masaru Shibata and Keisuke Taniguchi. Merger of binary neutron stars to a black hole: Disk mass, short gamma-ray bursts, and quasinormal mode ringing. *Phys. Rev.*, D73:064027, 2006.

[82] Matthew D. Duez, Pedro Marronetti, Stuart L. Shapiro, and Thomas W. Baumgarte. Hydrodynamic simulations in 3+1 general relativity. *Phys. Rev.*, D67:024004, 2003.

[83] Mark A. Miller, Philip Gressman, and Wai-Mo Suen. Towards a Realistic Neutron Star Binary Inspiral: Initial Data and Multiple Orbit Evolution in Full General Relativity. *Phys. Rev.*, D69:064026, 2004.

[84] Matthew Anderson et al. Simulating binary neutron stars: dynamics and gravitational waves. *Phys. Rev.*, D77:024006, 2008.

[85] Luca Baiotti, Bruno Giacomazzo, and Luciano Rezzolla. Accurate evolutions of inspiralling neutron-star binaries: prompt and delayed collapse to black hole. *Phys. Rev.*, D78:084033, 2008.

[86] Yuk Tung Liu, Stuart L. Shapiro, Zachariah B. Etienne, and Keisuke Taniguchi. General relativistic simulations of magnetized binary neutron star mergers. *Phys. Rev.*, D78:024012, 2008.

[87] Matthew Anderson et al. Magnetized Neutron Star Mergers and Gravitational Wave Signals. *Phys. Rev. Lett.*, 100:191101, 2008.

[88] Masaru Shibata and Koji Uryu. Merger of black hole - neutron star binaries: Nonspinning black hole case. *Phys. Rev.*, D74:121503, 2006.

[89] Masaru Shibata and Koji Uryu. Merger of black hole-neutron star binaries in full general relativity. *Class. Quant. Grav.*, 24:S125–S138, 2007.

[90] Frank Loffler, Luciano Rezzolla, and Marcus Ansorg. Numerical evolutions of a black hole-neutron star system in full general relativity: Head-on collision. *Phys. Rev.*, D74:104018, 2006.

[91] Zachariah B. Etienne et al. Fully General Relativistic Simulations of Black Hole- Neutron Star Mergers. *Phys. Rev.*, D77:084002, 2008.

[92] Tetsuro Yamamoto, Masaru Shibata, and Keisuke Taniguchi. Simulating coalescing compact binaries by a new code SACRA. 2008.

[93] J. Balakrishna. *A Numerical Study of Boson Stars: Einstein Equations with a Matter Source*. PhD thesis, Washington University, (1999).

[94] F. Siddhartha Guzman. Evolving spherical boson stars on a 3d cartesian grid. *Phys. Rev.*, D70:044033, 2004.

[95] Jayashree Balakrishna, Ruxandra Bondarescu, Gregory Daues, F. Siddhartha Guzman, and Edward Seidel. Evolution of 3d boson stars with waveform extraction. *Class. Quant. Grav.*, 23:2631–2652, 2006.

[96] J. A. Wheeler. Geons. *Phys. Rev.*, **97**:511, (1955).

[97] David J. Kaup. Klein-gordon geon. *Phys. Rev.*, 172(5):1331–1342, Aug 1968.

[98] Philippe Jetzer. Boson stars. *Phys. Rept.*, 220:163–227, 1992.

[99] R. Ruffini and S. Bonazzola. Systems of self gravitating particles in general relativity and the concept of an equation of state. *Phys. Rev.*, **187**:1767, (1969).

[100] Alan H. Guth. The Inflationary Universe: A Possible Solution to the Horizon and Flatness Problems. *Phys. Rev.*, D23:347–356, 1981.

[101] F. E. Schunck and E. W. Mielke. General relativistic boson stars. *Class. Quantum Grav.*, **20**:R301–R356, (2003).

[102] M. Colpi, S. L. Shapiro, and I. Wasserman. Boson stars: Gravitational equilibria of self-interacting scalar fields. *Phys. Rev. Lett.*, **57**:2485–2488, (1986).

[103] R. M. Wald. *General Relativity*. University of Chicago Press, Chicago IL, (1984).

[104] Tsvi Piran. Numerical codes for cylindrical general relativistic systems. *Journal of Computational Physics*, **35**:254–283, (1980).

[105] E. Seidel and W. M. Suen. Oscillating soliton stars. *Phys. Rev. Lett.*, 66:1659–1662, 1991.

[106] M. W. Choptuik. Universality and scaling in gravitational collapse of a massless scalar field. *Phys. Rev. Lett.*, **70**:9, (1993).

[107] C. W. Lai. *A Numerical Study of Boson Stars*. PhD thesis, The University of British Columbia, (2004).

[108] B. Rousseau. Axisymmetric boson stars in the conformally flat approximation. Master's thesis, The University of British Columbia, (2003).

[109] Frank Estabrook, Hugo Wahlquist, Steven Christensen, Bryce DeWitt, Larry Smarr, and Elaine Tsiang. Maximally slicing a black hole. *Phys. Rev. D*, 7(10):2814–2817, May 1973.

[110] C. W. Misner, K. S. Thorne, and J. A. Wheeler. *Gravitation*. W.H. Freeman and Company, New York, (1973).

[111] Jr. York, James W. Gravitational degrees of freedom and the initial-value problem. *Phys. Rev. Lett.*, 26:1656–1658, 1971.

[112] Jr. York, James W. Role of conformal three geometry in the dynamics of gravitation. *Phys. Rev. Lett.*, 28:1082–1085, 1972.

[113] S. Deser. Covariant decomposition of symmetric tensors and the gravitational cauchy problem. *Annales de l'institut Henri Poincaré (A) Physique théorique*, **7**(2):149–188, (1967).

[114] M. Berger and D. Ebin. Some decompositions of the space of symmetric tensors on a riemannian manifold. *J. Differential Geometry*, **3**:379–392, (1969).

[115] Jr. York, James W. Conformatlly invariant orthogonal decomposition of symmetric tensors on Riemannian manifolds and the initial value problem of general relativity. *J. Math. Phys.*, 14:456–464, 1973.

[116] E. Newman and R. Penrose. An approach to gravitational radiation by a method of spin coefficients. *J. Math. Phys.*, **3**:566, (1962).

[117] Kip S. Thorne. Multipole expansions of gravitational radiation. *Rev. Mod. Phys.*, 52(2):299–339, Apr 1980.

[118] Eric Gourgoulhon. 3+1 Formalism and Bases of Numerical Relativity. 2007.

[119] http://www.maplesoft.com/.

[120] M. Choptuik. ftp://laplace.physics.ubc.ca/pub/TensorV6/.

[121] Matthew W. Choptuik et al. Towards the final fate of an unstable black string. *Phys. Rev.*, D68:044001, 2003.

[122] Frans Pretorius. Numerical Relativity Using a Generalized Harmonic Decomposition. *Class. Quant. Grav.*, 22:425–452, 2005.

[123] Niall O'Murchadha and James W. York. Gravitational energy. *Phys. Rev.*, D10:2345–2357, 1974.

[124] M. Choptuik. Unpublished Numerical Relativity Lecture Notes (1998).

[125] Tsvi Piran James M. Bardeen. General relativistic axisymmetric rotating systems: Coordinates and equations. *Physics Reports*, **96**:205–250, (1983).

[126] R. Friedberg, T.D. Lee, and Y. Pang. Mini-soliton stars. *Phys. Rev.*, **D35**:3640–57, (1987).

[127] M. W. Choptuik. *A Study of Numerical Techniques for Radiative Problems in General Relativity*. PhD thesis, The University of British Columbia, (1986).

[128] O. Rinne. *Axisymmetric Numerical Relativity*. PhD thesis, Trinity College, Cambridge, (2005).

[129] M. Gleiser and R. Watkins. Gravitational stability of scalar matter. *Nucl. Phys.*, **B319**:733, (1989).

[130] Scott H. Hawley and Mathew W. Choptuik. Boson stars driven to the brink of black hole formation. *Phys. Rev. D*, 62(10):104024, Oct 2000.

[131] S. H. Hawley. *Scalar Analogues of Compact Astrophysical Systems*. PhD thesis, The University of Texas at Austin, (2000).

[132] M. W. Choptuik. *Lectures for VII Mexican School on Gravitation and Mathematical Physics, Numerical Analysis for Numerical Relativists*, (2006).

[133] R. Burden and J. Faires. *Numerical Analysis*. Thomson Brooks-Cole Publishing, 8th edition, (2005).

[134] L. F. Richardson. The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam. *Philosophical Transactions of the Royal Society of London. Series A*, **210**:307–357, (1911).

[135] B. Gustafsson. *High Order Difference Methods for Time Dependent PDE*. Springer, (2008).

[136] http://www.netlib.org/lapack.

[137] U. Trottenberg, C. W. Oosterlee, and A. Schueller. *Multigrid*. Academic Press, (2001).

[138] R. L. Marsa. *Radiative Problems in Black-Hole Spacetimes*. PhD thesis, The University of Texas at Austin, (1995).

[139] M. Choptuik and R. L. Marsa. http://laplace.physics.ubc.ca/People/matt/Rnpl/index.html.

[140] D. Choi. *Numerical Studies of Nonlinear Schrödinger and Klein-Gordon Systems: Techniques and Applications*. PhD thesis, The University of Texas at Austin, (1998).

[141] Joseph Oliger Marsha J Berger. Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of Computational Physics*, 53:484–512, March 1984.

[142] Frans Pretorius and Matthew W. Choptuik. Adaptive mesh refinement for coupled elliptic-hyperbolic systems. *J. Comput. Phys.*, 218:246–274, 2006.

[143] F. Pretorius. *Numerical Simulations of Gravitational Collapse*. PhD thesis, The University of British Columbia, (2002).

[144] B. C. Mundim. See BSIDPA distribution at ftp://laplace.physics.ubc.ca/pub/.

[145] http://www.netlib.org/odepack.

[146] B. C. Mundim. See PDEFDAOFF distribution at ftp://laplace.physics.ubc.ca/pub/.

# APPENDIX A

# BSIDPA - BOSON STAR INITIAL DATA FUNCTION IN POLAR AREAL COORDINATES

The problem of numerically solving the Einstein equations formulated as an initial value problem (as in the $3+1$ formalism case) splits into two parts: the initial data specification and the evolution of that initial data. While the evolution problem is still pervaded with open questions, the initial data calculation is approaching more and more to a routine type of problem. Particularly for spherically symmetric boson stars as matter model, it can virtually be considered as a solved problem. It becomes interesting then to provide this solution for a larger community of users mainly interesting on problems originated from the evolution of these boson stars. This way the user have the benefit of an initial data as accurate as desired, without the necessity of a detailed knowledge on how the initial data was calculated numerically or the need to implement herself the initial data solver. With this intention in mind, a general purpose initial data solver for boson stars was written.

BSIDPA[1] is a FORTRAN subroutine written with the purpose of generating static spherically symmetric boson stars as initial data. As the name suggests, BSIDPA stands for **B**oson **S**tar **I**nitial **D**ata in **P**olar-**A**real coordinates. It produces a boson star by integrating numerically the set of ODEs in polar-areal coordinates, equations (2.246)-(2.249), as described in detail in Section 2.6, for a particular value of the ansatz eigenfrequency $\omega$. Furthermore, it not only integrates the polar areal ODE system but also it finds the bracket of values $[\omega_{hi}, \omega_{low}]$ to which the boson star eigenfrequency $\omega$ belongs and decreases the bracket width by root bisection in order to provide $\omega$ with the desired accuracy. It is worth noticing that BSIDPA provides the boson star solution for a general polynomial self-interaction potential:

$$U(\phi_0) = \sum_{i=1}^{pn} p_i \phi_0^i, \tag{A.1}$$

where $pn$ and $p_i$ are supplied by the user and the $p_i$s are the adimensional coupling constants of the model. Note that $i = 2$ corresponds to the massive Klein-Gordon scalar field or to the so called mini-boson stars. The numerical integration of the ODE system is performed by a routine called LSODA of the ODEPACK [145] package.

The BSIDPA subroutine is called by its driver routine as follows:

```
...

integer   sht, n, pn, pt, dft, tail_type
real*8    phi0, rmax, lsoda_tol, w_tol, dr, whi, wlo, wshoot, wresca
```

[1] BSIDPA and auxiliary routines are kept at Choptuik's ftp repository site [144]

```
real*8    a(n), alpha(n), phi(n), pp(n), m(n), zr(n), r(n)
real*8    p(pn)

...

call bsidpa(a,alpha,phi,pp,m,zr,r,p,phi0,rmax,lsoda_tol,w_tol,dr,
&            whi,wlo,wshoot,wresca,sht,n,pn,pt,dft,tail_type)

...
```

The user has to provide the following parameters:

**phi0:** the central value scalar field $\phi_0(0)$ that parametrize the family of static spherically symmetric boson star solutions.

**rmax:** the integration starts at the radial coordinate $r = 0$ and ends at $r = rmax$.

**pt:** chooses the potential type. At the moment only the massless potential, $pt = 0$, and the polynomial potential, $pt = 1$, are implemented.

**p(pn):** real*8 vector of length $pn$ to storage the user supplied values for the coupling constants $p_i$ appearing in the self-interacting potential (A.1).

**n:** a number of grid points in the interval $0 \leq r \leq rmax$ to output the result of the ODE integration.

**sht:** Switch to determine whether or not BSIDPA is to find the bracket of $\omega$ values and shoot to determine $\omega$ to the accuracy prescribed by the user. For $sht = 1$, BSIDPA finds the interval corresponding to the ground state of the eigenvalue system and shoots to find $\omega$ and the solution of the ODE system. For $sht = 0$, it doesn't shoot and the user is responsible to provide $wshoot$, the correct value for the ground state eigenfrequency $\omega$.

**wshoot:** eigenfrequency associated with a particular value of $\phi_0(0)$ that results into a solution for $\phi(r)$ with no nodes, i.e. the ground state solution. $\phi(r)$ is therefore a smooth strictly positive function of the areal coordinate $r$ that approaches to zero as $r$ tends to infinity. It is expected to be provided by the user when $sht = 0$. It is calculated by shooting (or root bisection) otherwise.

**dft:** flag to switch on and off the set of default parameters to be used. $dft = 0$ assigns lsoda_tol, w_tol, dr, whi and wlo their the default values. $dft = 1$ assigns only dr, whi and wlo to their default values. Finally, $dft = 2$ assigns no default value, i.e. the user must specify all the relevant parameter values.

On the other hand, the following parameters may or may not be specified by the user according to the value assigned to $dft$:

**lsoda_tol:** LSODA tolerance parameter. The local solution error is kept below this tolerance by the ODE solver. Its default value is hard-coded as $lsoda\_tol = 10^{-10}$.

**w_tol:** Tolerance for estimating the eigenvalue $\omega$ coming from the static ansatz $\phi(t,r) = \phi_0(r)\exp(-i\omega t)$. Therefore, for any $\phi_0$, there is a single $\omega \in [\omega_0 - w\_tol, \omega_0 + w\_tol]$ which results into a star-like solution (i.e. no nodes) of the ODE system satisfing the fields and metric components boundary and regularity conditions. Its default value is $w\_tol = 10^{-9}$.

**dr:** solution output interval used by BSIDPA when the algorithm is looking for the bracket that constains $\omega$ values. It is useful to have a different outpupt control parameter until the bracket is located since it may happen that for certain values of $\omega$, away from the correct eigenvalue, the system do not integrate even on the first step. Decreasing the output step interval helps to overcome this problem. Its default value is $dr = 10^{-4}$.

**[whi,wlo]:** these two guesses for $\omega$ value are the eigenfrequencies in which the solution $\phi_0(r)$ blows up or down as $r$ gets large. It is the user responsibility to assure that the values of whi and wlo actually bracket the desired value of $\omega$. BSIDPA *does* have an algorithm encoded to automatically look for an appropriate bracket, however it should be handle with care. Bracketing a particular value of $\omega$ may be tricky specially if the ground state eigenfrequencies and the excited states ones have their values too close to each other. That does happen for some extreme values for the parameter vector $p_i$ entering the self-interacting potential. When exploring the parameter space for this problem is recommended that the new bracket actually be formed based on the previous set of values. A continuum survey of the parameter space starting from known values of $\omega$ helps then to avoid problems related to appropriate bracketing. Its default values are based on the solution of mini-boson stars with associated particle mass set to $m = 1$ and $\phi_0(0) = 0.01$. Therefore $whi = 1$ and $wlo = 1.01$ by default.

**tail_type:** determines the type of tail to be matched as the boson star "atmosphere". When $\phi_0(r)$ reaches a cut-off point or when the concavity of the boson star tail changes its sign from positive to negative (remember: the shooting process to find the boson star solution ends up blowing up or down. This code makes sure the last trial always blows down) a tail is matched at the cut-off point until the end of the prescribed domain. Three different tails $T(r)$ are encoded:

$$T_1(r) = \frac{A}{r} + \frac{B}{r^2} + \frac{C}{r^3}, \tag{A.2}$$

$$T_2(r) = A\exp(-Br) + \frac{C}{r}, \tag{A.3}$$

$$T_3(r) = A\exp(-Br). \tag{A.4}$$

The default tail is $T_1(r)$. Since it has three parameters to fit, it matches then the function, its first and second derivatives at the cut-off point. Besides being a smoother match, $T_1(r)$ also assigns to the field a fall-off similar to the metric components (asymptotically flat condition).

The only caveat though is that not always it is possible to fit $T_1(r)$ without assigning a negative number to one of the constants $A$, $B$ or $C$. As a result the tail may cross the $\phi_0 = 0$ axis as it approaches to zero. That is not the solution behaviour sought though. $T_2(r)$ and $T_3(r)$ are then tried in sequence. $T_3(r)$ fits the function and its first derivative only but can be considered already a reasonably good tail (remember that it usually assigns values for $\phi_0(r) \leq 10^{-5}\phi_0(0)$ after the cut-off point). A warning message is output if failure to fit occurs. If one insists in fitting a particular tail, then it may be necessary to increase the frequency in which the solution is output in order to make the numerical solution approximation smoother.

At last, the code returns the following quantities:

**r(n):** vector of length $n$ storaging the areal coordinate $r$.

**a(n):** vector of length $n$ storaging the solution for the metric function $a(r)$.

**alpha(n):** vector of length $n$ storaging the solution for the metric function $\alpha(r)$. Note that $\alpha(r)$ has already been rescaled.

**phi(n):** vector of length $n$ storaging the solution for the scalar field modulo $\phi_0(r)$.

**pp(n):** vector of length $n$ storaging the solution for the spatial derivative of the scalar field modulo $\Phi_0(r)$.

**m(n):** vector of length $n$ storaging a derived quantity, the mass aspect function $m(r)$.

**zr(n):** vector of length $n$ storaging a derived quantity, the compactness function $z(r) = 2m(r)/r$. As brief purpose description, this function has been used in the past to determine upper bounds on how compact a star can be before it collapses into a black-hole. If $\max(z(r)) = 1$ then the star is a black-hole and its radius is the Schwarzschild radius $r = r_{sch}$.

**wresca:** rescaled value of $\omega$.

BSIDPA code flows as follows. After the user supplies all the required parameters, the code will look for a pair of values for $\omega$ such that *whi* corresponds to a $\phi_0(r)$ solution that blows up at large values of the areal coordinate $r$ (due to accumulated truncation and round-off errors) and *wlo* corresponds to a solution that blows down. Note that it may happen that the algorithm brackets an "excited" solution, i.e. a solution with nodes (usually crossing the $\phi_0(0)$ axis). The code identifies this type of solution and discards it from the bracket search (and adjust the bracket accordingly). Once the bracket is found, a simple root bisection algorithm is applied to constrain the value of $\omega$ to a very narrow interval of width $2\,w\_tol$. The solution in which $\phi_0(r)$ blows down is picked as the eigenfunction associated with the eigenfrequency (or eigenvalue) $\omega$. This solution has a property that allows to easily identify the $r$ coordinate in which the solution is not trustworthy anymore. As $r$ increases, the second derivative of the numerical solution for $\phi_0(r)$ changes signs from negative to positive and then reaches a point where it undesirably changes the sign again to negative until the solution blows down at large $r$. BSIDPA identifies when that happens and fits an exponential tail starting from a few grid points before. Note that this identification happens

on top of another condition: if the value of the scalar field is $\phi_0(r) \leq 10^{-5}\phi_0(0)$ then a tail is fit even if the concavity sign has not changed yet. Unless the user specifies otherwise, it tries first to fit the smoothest possible tail to $\phi_0(r)$ and its spatial derivative $\Phi_0(r)$. In case of failure the other two cases are tried in sequence. The last tail option is simply an exponential and has never fail to fit so far. $a(r)$ and $\alpha(r)$ are approximated after the point the tail is fit by use of the formula for mass aspect function, equation (2.260), and their limiting values to the Schwarzschild metric components, equations (2.257) and (2.258). Once the tail is incorporated to the solution, the values of $\omega$ and $\alpha(r)$ are rescaled according to formulas (2.263) and (2.265). The code returns then the solution of the system: $a(r)$, $\alpha(r)$, $\phi_0(r)$ and $\Phi_0(r)$, along with the derived quantities mass aspect function $m(r)$ and compactness function $z(r) = 2m(r)/r$. The rescaled value of $\omega$, wresca, is also returned.

Table A.1 show some values of the eigenfrequency calculated with BSIDPA for a family of spherically symmetric solutions parametrized by the central field value $\phi_0(0)$ in the case of mini-boson stars. Table A.2 collects some properties refering to the mini-boson star solutions. More specifically it shows where the cut-off radius is and how the mass aspect function changes in that region. Also the ADM mass and the star radius are tabulated for several different values of the central field value $\phi_0(0)$. Note that the contribution of the tail for the total star mass is negligible. Table A.3, on the other hand, tabulates the maxima and minima of some metric components and, more interestingly, the compactness function $z(r)$. One way of interpret those results is to take as reference the same value for a black-hole. In this case $\max z(r) = 1$ and $r_{\max z} = R_{sch}$. Therefore as the star becomes more and more compact $\max z(r)$ approaches its maximum possible value at the coordinate radius tending to the Schwarzschild radius.

| $\phi_0$ | wshoot | wresca |
|---|---|---|
| 0.005 | 1.01173913657665 | 0.987921656870952 |
| 0.006 | 1.01414229214191 | 0.985551416322374 |
| 0.007 | 1.01656427949667 | 0.983196152260810 |
| 0.008 | 1.01900526873767 | 0.980855788252950 |
| 0.009 | 1.02146543309092 | 0.978530270851457 |
| 0.01 | 1.02394494622946 | 0.976219519074950 |
| 0.02 | 1.04984409445897 | 0.953910295230613 |
| 0.03 | 1.07788555219769 | 0.933010708580807 |
| 0.04 | 1.10828046126291 | 0.913467522154004 |
| 0.05 | 1.14126635619905 | 0.895235350281201 |
| 0.06 | 1.17711107464507 | 0.878276194485022 |
| 0.07 | 1.21611731750891 | 0.862559036385476 |
| 0.08 | 1.25862797105685 | 0.848059515066944 |
| 0.09 | 1.30503232180141 | 0.834759601661149 |

**Table A.1:** Mini-boson star central scalar field values and its eigenfrequencies $\omega$. This table shows the values of $\omega$, either its shooting value wshoot or its rescaled value wresca, for several values for the central scalar field values. The parameters used to obtain these values were all the defaults except for: $n = 2^{16} + 1$, $rmax = 100$ and $dr = 10^{-4}$. Also the self-interaction coupling constants for mini-boson stars are just $p_2 = 1.0$ and $p_i = 0$ for all $i \neq 2$.

| $\phi_0$ | $r_{cut-off}$ | $dm/dr\|_{r_{cut-off}}$ | $M_{ADM}$ | $R_{99}$ |
|---|---|---|---|---|
| 0.005 | 91.12 | $1.34\times10^{-9}$ | 0.2646 | 35.78 |
| 0.006 | 81.42 | $1.13\times10^{-9}$ | 0.2878 | 32.58 |
| 0.007 | 73.58 | $1.56\times10^{-9}$ | 0.3086 | 30.09 |
| 0.008 | 71.63 | $0.84\times10^{-9}$ | 0.3275 | 28.08 |
| 0.009 | 65.72 | $0.92\times10^{-9}$ | 0.3449 | 26.40 |
| 0.01 | 63.85 | $1.22\times10^{-9}$ | 0.3609 | 24.99 |
| 0.02 | 45.50 | $2.06\times10^{-9}$ | 0.4751 | 17.26 |
| 0.03 | 36.62 | $2.62\times10^{-9}$ | 0.5424 | 13.77 |
| 0.04 | 31.99 | $3.25\times10^{-9}$ | 0.5844 | 11.67 |
| 0.05 | 28.39 | $3.94\times10^{-9}$ | 0.6103 | 10.22 |
| 0.06 | 25.44 | $4.83\times10^{-9}$ | 0.6251 | 9.148 |
| 0.07 | 23.74 | $5.38\times10^{-9}$ | 0.6319 | 8.311 |
| 0.08 | 22.16 | $5.99\times10^{-9}$ | 0.6327 | 7.637 |
| 0.09 | 20.84 | $7.21\times10^{-9}$ | 0.6290 | 7.081 |

**Table A.2:** Mini-boson star properties: tail, mass and radius. This table shows where the scalar field tail was fit, $r_{cut-off}$, the fraction of the total mass change around that region, $dm/dr|_{r_{cut-off}}$, the ADM mass of the star, $M_{ADM}$ and the radius containg $99\%M_{ADM}$. The parameters used to obtain these values were the same as table A.1 Note that the change in mass around $r_{cut-off}$ is very small. Therefore the increment in the ADM mass due to the tail is negligible.

| $\phi_0$ | $\min\alpha(r)$ | $\max\psi(r)$ | $\max z(r)$ | $r_{\max z}$ | $R_{sch}$ |
|---|---|---|---|---|---|
| 0.005 | 0.9764 | 1.0111 | 0.0207 | 19.03 | 0.529 |
| 0.006 | 0.9718 | 1.0134 | 0.0248 | 17.32 | 0.575 |
| 0.007 | 0.9672 | 1.0157 | 0.0288 | 15.98 | 0.617 |
| 0.008 | 0.9626 | 1.0180 | 0.0328 | 14.90 | 0.655 |
| 0.009 | 0.9580 | 1.0203 | 0.0367 | 14.00 | 0.690 |
| 0.01 | 0.9534 | 1.0226 | 0.0406 | 13.24 | 0.722 |
| 0.02 | 0.9086 | 1.0454 | 0.0780 | 9.05 | 0.950 |
| 0.03 | 0.8656 | 1.0678 | 0.1124 | 7.14 | 1.085 |
| 0.04 | 0.8242 | 1.0898 | 0.1441 | 5.98 | 1.169 |
| 0.05 | 0.7844 | 1.1114 | 0.1734 | 5.17 | 1.220 |
| 0.06 | 0.7461 | 1.1325 | 0.2004 | 4.56 | 1.250 |
| 0.07 | 0.7093 | 1.1532 | 0.2253 | 4.08 | 1.264 |
| 0.08 | 0.6738 | 1.1735 | 0.2483 | 3.69 | 1.265 |
| 0.09 | 0.6397 | 1.1933 | 0.2694 | 3.36 | 1.258 |

**Table A.3:** Mini-boson star properties: metric components maxima and minima. This table shows the minimum value for the lapse function $\min\alpha(r)$, the maximum value for the conformal factor, $\max\psi(r)$ and the maximum of the compactness function $\max z(r)$ and where it occurs, $r_{\max z}$. For comparison, the Schwarzschild radius of the star, $R_{sch}$, is also tabulated. The parameters used to obtain these values were the same as table A.1. Note that as the central field value increases the radius in which $z(r)$ reaches its maximum decreases and approaches more and more the star Schwarzschild radius. A black-hole is formed once they coincide.

# APPENDIX B

# STENCIL DERIVATION

The derivation of the finite difference operators or stencils, their pictorial equivalent, is usually achieved by Taylor series expansion of the PDE continuum solution $u$. In order to simplify the discussion, this section restricts its attention to one dimensional operators only[1]. The continuum solution evaluated at the grid points $x_i$ is denoted by $u(x_i) \equiv u_i$. On the other hand, the discrete solution $u^h$ at the grid location $i$ is labelled as $u_i^h$.

To begin with, consider the general form of a finite difference operator $L^h$ approximating an arbitrary constant-coefficient differential operator $L$. Let $L^h$ act at a grid function $u_i^h$. The expression for this FDA is simply defined to be a weighted sum of neighbour grid functions:

$$L^h u_i^h \equiv \sigma \sum_{j=j_{min}}^{j=j_{max}} c_j u_{i+j}^h, \tag{B.1}$$

where $\sigma$, $j_{min}$, $j_{max}$ and $c_{j_{min}} \dots c_{j_{max}}$ are attributes of the finite difference operator $L^h$ and can be defined as follows:

1. The integers $j_{min}$ and $j_{max}$ define the limits of the stencil, i.e. the extension in which neighbour grid functions are used to define the discrete operator. The total width of the stencil is then given by $j_{max} - j_{min} + 1$ and it is also called stencil diameter. Whenever $j_{min} = -j_{max}$, the FDA is said to be *centred*. It is called *off-centred* otherwise.

2. The relative weights of the unknowns $u_{i+j}^h$, the $c_{j_{min}} \dots c_{j_{max}}$, are integer coefficients.

3. $\sigma$ is an overall scale factor of the form $1/(\kappa h^m)$, where $m$ is the order of the differential operator being approximated and $\kappa$ an integer chosen in order to guarantee that all $c_j$ are integers.

As a means to determine which differential operator is approximated by $L^h$ for a given set of coefficients $c_j$, each $u_i^h$ in (B.1) should be replaced by its continuum counterpart evaluated at the different grid positions, $u_{i+j}$, and Taylor expanded around the grid point $x = x_i$:

$$L^h u_i^h \equiv \sigma \sum_{j=j_{min}}^{j=j_{max}} c_j u_{i+j} = \sigma \sum_{j=j_{min}}^{j=j_{max}} c_j \left[ \sum_{k=0}^{\infty} \frac{u_i^{(k)}}{k!} (x_j - x_i)^k \right] = \sigma \sum_{j=j_{min}}^{j=j_{max}} c_j \left[ \sum_{k=0}^{\infty} \frac{u_i^{(k)}}{k!} (jh)^k \right], \tag{B.2}$$

where $u_i^{(k)} \equiv d^k u(x)/dx^k|_{x=x_i}$ and $u_i^{(0)} \equiv u(x_i)$. Regrouping the terms in (B.2), the approximation becomes:

[1] Higher order operators can be obtained by applying the same techniques exposed here at each dimension at a time.

$$L^h u_i^h \equiv \sigma \sum_{j=j_{min}}^{j=j_{max}} c_j \left[ \sum_{k=0}^{\infty} \frac{u_i^{(k)}}{k!} (jh)^k \right] = \sigma \sum_{k=0}^{\infty} \frac{u_i^{(k)}}{k!} h^k \left[ \sum_{j=j_{min}}^{j=j_{max}} j^k c_j \right] = \sigma \sum_{k=0}^{\infty} \frac{u_i^{(k)}}{k!} h^k S_k. \tag{B.3}$$

Suppose one wishes to find a $O(h^p)$ finite difference approximation to a $m$-th order differential operator, i.e. to find $c_j$ and $\sigma$ in (B.1) such that $L^h u_i^h = d^m u/dx^m + O(h^p)$. In order for that to happen all the coefficients in the series (B.3) that are not multiplying $u_i^{(m)}$ or the terms containing $h^{m+p}$ or higher order in $h$ must vanish. As a result all the $m$ sums present in the sum over $k$ in (B.3) such that $k < m$ must vanish, i.e. $S_k = 0$ for $k < m$, the $m$th term has to be adjusted to $S_m = m!/(\sigma h^m)$, and finally all the other $p-1$ sums must also vanish, i.e. $S_k = 0$ for $m < k < m+p$. Thus there are $m+p$ equations for the sums of the $c_j$ coefficients, $S_k$. As a consequence, the number of required non-zero coefficients $c_j$ to obtain a determined system of linear equations should be also $m + p$ or in other words, $j_{max} - j_{min} + 1 = m + p$. The linear system is solved then for the coefficients $c_j$ and the FDA is built through the sum (B.1).

As an explicit example, suppose one is looking for a centred second order approximation to the second order differential operator or $L^h u_i^h = d^2u/dx^2 + O(h^2)$. According to the counting argument exposed in the last paragraph, $j_{max} - j_{min} + 1 = m + p = 4$. However since for centred stencils $j_{min} = -j_{max}$ then $j_{max} = 3/2$ in order to satisfy the counting argument. The integer fraction has actually to be truncated to $j_{max} = 1$. This example is good to illustrate what happens when one estimates a centred stencil diameter using the reasoning of the previous paragraph. The otherwise correct stencil diameter of $(m + p =)$ 4 is actually reduced to $(m + p - 1 =)$ 3 for centred stencils. The reason for that comes from the symmetry between the terms in $S_k$. It is possible to show in general that $c_j = \pm c_{-j}$ for centred approximations, with the sign depending on the derivative order $m$ being even or odd. This eliminates the need for one of the equations $S_k$ for $k$ odd, i.e. one equation for $k$ odd is equivalent to any of the other $k$-odd equations. Carrying on with the example, the possible $c_j$ are then $c_{-1}$, $c_0$ and $c_1$ and the system of linear equations becomes (according to the expected counting):

$$S_0 \equiv c_{-1} + c_0 + c_1 = 0, \tag{B.4}$$

$$S_1 \equiv (-1)c_{-1} + (0)c_0 + (1)c_1 = 0, \tag{B.5}$$

$$S_2 \equiv (-1)^2 c_{-1} + (0)^2 c_0 + (1)^2 c_1 = \frac{2!}{\sigma h^2}, \tag{B.6}$$

$$S_3 \equiv (-1)^3 c_{-1} + (0)^3 c_0 + (1)^3 c_1 = 0. \tag{B.7}$$

It is evident that $S_3 \equiv S_1$ and the system reduces to 3 equations for 3 unknowns. The solution of this linear system results in $c_{-1} = c_1 = 1/(\sigma h^2)$ and $c_0 = -2/(\sigma h^2)$ where $\sigma = 1/h^2$ can be read from this solution according to the definition of $\sigma$. Inserting these values on the sum $S_4$ one can find explicitly the leading order term of the approximation. As a result the FDA can be written to leading order in $h$ as:

$$L^h u_i^h \equiv \frac{u_{i-1}^h - 2u_i^h + u_{i+1}^h}{h^2} \approx \left.\frac{d^2u(x)}{dx^2}\right|_{x=x_i} + \frac{h^2}{12}\left.\frac{d^4u(x)}{dx^4}\right|_{x=x_i}. \tag{B.8}$$

Once the finite difference operators are derived to the order of accuracy desired, to discretize a PDE is simply a matter of substituting all differential operators on functions by their correspondent finite difference operators acting on the grid functions instead. Each function is also replaced by its grid function approximation. The order of accuracy of the whole scheme depends then on the lowest order of the finite difference operators employed on the discretization.

App. C introduces a set of Maple procedures written by the author with the intention of automate the derivation of finite difference operators of any order and the subsequent discretization of partial differential equations of any order as well. Several examples illustrate how to use the procedures at the same time that output the most commonly used finite difference operators. A couple of PDEs are also discretized in order to demonstrate the flexibility of these routines. Please refer to App. C for more details.

# APPENDIX C

# PDEFDAOFF - PDE DISCRETIZATION MAPLE PROCEDURE

This appendix describes the use of the package PDEFDAOFF (that stands for Partial Differential Equation to Finite Difference Aproximation using OFF-centered and centered approximation schemes). This package is a set of MAPLE procedures written by the author to assist on the translation of the PDEs describing the equations of motion into a discrete form convenient for numerical computation. This appendix consists of two main sections. Each of them describes through examples the use of the two main procedures contained in the package, '*dfdandoff*' and '*resndoff*'. The examples were extracted straight from the MAPLE demo worksheet included in the distribution[146].

## C.1 Finite difference approximation to n-dimensional differential operators

### C.1.1 Introduction

'*dfdandoff*' is a maple procedure to finite difference a differential operator of a function of an arbitrary number of coordinates. It returns a finite difference approximation to $O(h^p)$ for all coordinates, where $p$ records the order of accuracy of the difference scheme required and it is user specified. This procedure returns both centered and off-centered finite difference schemes[1]. The rationale behind the procedure name is the following: *d* stants for differential, *fda* for finite difference approximation, *nd* for *n* dimensional functions and *off* to remind that this procedure also calculates off-centered schemes.

To load the package just open a MAPLE session and type:

```
> restart;
> read PDEFDAOFF:
```

### C.1.2 Differential operators on 1D functions

The first argument of dfdandoff must be a function. The reason for that is that maple returns zero when it differentiates a symbol/name. It is usual to define an alias for the function and hide its coordinate dependency in order to make the equations look tidier:

```
> alias(u=u(x));
```

$$u \tag{C.1}$$

[1]Historical note: this procedure evolved from '*dfdand*' where only centered schemes were calculated.

The second argument is a list of the independent variables (coordinates) entered in lexicographical (alphabetical) order. The third argument is a name for the finite difference function. Note that must be a name (i.e. just a symbol). The forth argument is a list with the names of the indexes corresponding to each coordinate. The fifth one is a list naming the coordinate spacing between gridpoints for each coordinate. The sixth one is a list with the order of the differential operator along each coordinate direction. The seventh one specifies the order of the approximation scheme and the last one is a list, with each member associated with the respective coordinate, indicating how off-centered the scheme should be when compared to the centered one. The examples below should make them clearer.

The first examples are for first order differential operator. As it is evident below the sixth argument is a list of one element corresponding to only one coordinate and its value is 1 to indicate a first order differential operator. In the first three examples the seventh argument assumes the value 1 to represent first order schemes, i.e. $O(hx)$. Note that if one naively try to find a first order centered scheme, entering the value 0 in the last argument, the procedure returns:

```
> dfdandoff(u,[x],ud,[i],[hx],[1],1,[0]);
Error, (in dfdandoff) invalid input: rhs received 1, which is not valid for its 1st argument, expr
```

This is actually not a bug since there is no $O(h)$ centered scheme for first order differential operators. Therefore the simplest FDA to first order differential operators is given by a backward or forward scheme ($-1$ or 1 as the eighth argument):

```
> dfdandoff(u,[x],ud,[i],[hx],[1],1,[-1]);
```

$$\frac{\partial}{\partial x}u_i = -\frac{ud_{i-1}}{hx} + \frac{ud_i}{hx} \tag{C.2}$$

To verify the correctness of the finite difference operator, meaning the order of convergence of the FDA, an independent procedure called '*fdaeval*' written by Matt Choptuik (and included in the file 'PDEFDAOFF') needs be applied. To do so, the procedure Taylor expand around $ud_i$ the right-hand side of the expression above in terms of the mesh spacing ($hx$ in this case) and collect the terms of the same spacing power. The procedure requires as first argument a rank-1 Maple array with the coefficients of entering the approximation represented as integers. The second argument is the overall scale factor that allows all the coefficients to be integers. The last example can be tested then as:

```
> abwh:=array(-1..0,[-1,1]);
```

$$abwh := ARRAY\left(-1..0, [0 = 1, -1 = -1]\right) \tag{C.3}$$

```
> fdaeval(abwh,1/h);
```

$$\mathrm{D}\left(u\right)\left(x\right) - \frac{1}{2}\,\mathrm{D}^{(2)}\left(u\right)\left(x\right)h + \frac{1}{6}\,\mathrm{D}^{(3)}\left(u\right)\left(x\right)h^2 \tag{C.4}$$

As expected the backward approximation to the first order differential operator is a first order

finite difference approximation.

> dfdandoff(u,[x],ud,[i],[hx],[1],1,[1]);

$$\frac{\partial}{\partial x}u_i = -\frac{ud_i}{hx} + \frac{ud_{i+1}}{hx} \tag{C.5}$$

> afwh:=array(0..1,[-1,1]):
> fdaeval(afwh,1/h);

$$\mathrm{D}\left(u\right)\left(x\right) + \frac{1}{2}\,\mathrm{D}^{(2)}\left(u\right)\left(x\right)h + \frac{1}{6}\,\mathrm{D}^{(3)}\left(u\right)\left(x\right)h^2 \tag{C.6}$$

On the other hand, the centered, backward and forward second order approximation and respective checks are given by:

> dfdandoff(u,[x],ud,[i],[hx],[1],2,[0]);

$$\frac{\partial}{\partial x}u_i = -\frac{1}{2}\,\frac{ud_{i-1}}{hx} + \frac{1}{2}\,\frac{ud_{i+1}}{hx} \tag{C.7}$$

> ach2:=array(-1..1,[-1,0,1]):
> fdaeval(ach2,1/(2*h));

$$\mathrm{D}\left(u\right)\left(x\right) + \frac{1}{6}\,\mathrm{D}^{(3)}\left(u\right)\left(x\right)h^2 \tag{C.8}$$

> dfdandoff(u,[x],ud,[i],[hx],[1],2,[-1]);

$$\frac{\partial}{\partial x}u_i = \frac{1}{2}\,\frac{ud_{i-2}}{hx} - 2\,\frac{ud_{i-1}}{hx} + \frac{3}{2}\,\frac{ud_i}{hx} \tag{C.9}$$

> abwh2:=array(-2..0,[1,-4,3]):
> fdaeval(abwh2,1/(2*h));

$$\mathrm{D}\left(u\right)\left(x\right) - \frac{1}{3}\,\mathrm{D}^{(3)}\left(u\right)\left(x\right)h^2 + \frac{1}{4}\,\mathrm{D}^{(4)}\left(u\right)\left(x\right)h^3 \tag{C.10}$$

> dfdandoff(u,[x],ud,[i],[hx],[1],2,[1]);

$$\frac{\partial}{\partial x}u_i = -\frac{3}{2}\,\frac{ud_i}{hx} + 2\,\frac{ud_{i+1}}{hx} - \frac{1}{2}\,\frac{ud_{i+2}}{hx} \tag{C.11}$$

> afwh2:=array(0..2,[-3,4,-1]):
> fdaeval(afwh2,1/(2*h));

$$\mathrm{D}\left(u\right)\left(x\right) - \frac{1}{3}\,\mathrm{D}^{(3)}\left(u\right)\left(x\right)h^2 - \frac{1}{4}\,\mathrm{D}^{(4)}\left(u\right)\left(x\right)h^3 \tag{C.12}$$

The seven examples below are supposed to elucidate what the last function argument means. Let $i$ represent a generic point of the grid where the Taylor expansion is done and let $j$ represent

an index to determine how off-centered the scheme is. For $j = 0$ the scheme is centered. For $j = 1$ the sheme is centered at $i + 1$ and thus off-centered by 1 grid point. For the examples below the maximum allowed is $j = 2$ since the scheme is supposed to reference the grid point $i$.

```
> dfdandoff(u,[x],ud,[i],[hx],[1],4,[0]);
```

$$\frac{\partial}{\partial x}u_i = \frac{1}{12}\,\frac{ud_{i-2}}{hx} - \frac{2}{3}\,\frac{ud_{i-1}}{hx} + \frac{2}{3}\,\frac{ud_{i+1}}{hx} - \frac{1}{12}\,\frac{ud_{i+2}}{hx} \tag{C.13}$$

```
> ach4:=array(-2..2,[1,-8,0,8,-1]):
> fdaeval(ach4,1/(12*h));
```

$$\mathrm{D}\left(u\right)\left(x\right) - \frac{1}{30}\,\mathrm{D}^{(5)}\left(u\right)\left(x\right)h^4 \tag{C.14}$$

```
> dfdandoff(u,[x],ud,[i],[hx],[1],4,[-1]);
```

$$\frac{\partial}{\partial x}u_i = -\frac{1}{12}\,\frac{ud_{i-3}}{hx} + \frac{1}{2}\,\frac{ud_{i-2}}{hx} - \frac{3}{2}\,\frac{ud_{i-1}}{hx} + \frac{5}{6}\,\frac{ud_i}{hx} + \frac{1}{4}\,\frac{ud_{i+1}}{hx} \tag{C.15}$$

```
> abw1h4:=array(-3..1,[-1,6,-18,10,3]):
> fdaeval(abw1h4,1/(12*h));
```

$$\mathrm{D}\left(u\right)\left(x\right) + \frac{1}{20}\,\mathrm{D}^{(5)}\left(u\right)\left(x\right)h^4 - \frac{1}{24}\,\mathrm{D}^{(6)}\left(u\right)\left(x\right)h^5 \tag{C.16}$$

```
> dfdandoff(u,[x],ud,[i],[hx],[1],4,[1]);
```

$$\frac{\partial}{\partial x}u_i = -\frac{1}{4}\,\frac{ud_{i-1}}{hx} - \frac{5}{6}\,\frac{ud_i}{hx} + \frac{3}{2}\,\frac{ud_{i+1}}{hx} - \frac{1}{2}\,\frac{ud_{i+2}}{hx} + \frac{1}{12}\,\frac{ud_{i+3}}{hx} \tag{C.17}$$

```
> afw1h4:=array(-1..3,[-3,-10,18,-6,1]):
> fdaeval(afw1h4,1/(12*h));
```

$$\mathrm{D}\left(u\right)\left(x\right) + \frac{1}{20}\,\mathrm{D}^{(5)}\left(u\right)\left(x\right)h^4 + \frac{1}{24}\,\mathrm{D}^{(6)}\left(u\right)\left(x\right)h^5 \tag{C.18}$$

```
> dfdandoff(u,[x],ud,[i],[hx],[1],4,[-2]);
```

$$\frac{\partial}{\partial x}u_i = \frac{1}{4}\,\frac{ud_{i-4}}{hx} - \frac{4}{3}\,\frac{ud_{i-3}}{hx} + 3\,\frac{ud_{i-2}}{hx} - 4\,\frac{ud_{i-1}}{hx} + \frac{25}{12}\,\frac{ud_i}{hx} \tag{C.19}$$

```
> abw2h4:=array(-4..0,[3,-16,36,-48,25]):
> fdaeval(abw2h4,1/(12*h));
```

$$\mathrm{D}\left(u\right)\left(x\right) - \frac{1}{5}\,\mathrm{D}^{(5)}\left(u\right)\left(x\right)h^4 + \frac{1}{3}\,\mathrm{D}^{(6)}\left(u\right)\left(x\right)h^5 \tag{C.20}$$

```
> dfdandoff(u,[x],ud,[i],[hx],[1],4,[2]);
```

$$\frac{\partial}{\partial x}u_i = -\frac{25}{12}\frac{ud_i}{hx} + 4\frac{ud_{i+1}}{hx} - 3\frac{ud_{i+2}}{hx} + \frac{4}{3}\frac{ud_{i+3}}{hx} - \frac{1}{4}\frac{ud_{i+4}}{hx} \tag{C.21}$$

> afw2h4:=array(0..4,[-25,48,-36,16,-3]):
> fdaeval(afw2h4,1/(12*h));

$$\mathrm{D}\left(u\right)\left(x\right) - \frac{1}{5}\mathrm{D}^{(5)}\left(u\right)\left(x\right)h^4 - \frac{1}{3}\mathrm{D}^{(6)}\left(u\right)\left(x\right)h^5 \tag{C.22}$$

The last two examples for this differential operator illustrate the fact that if the user try to use a value of $j$ large enough to center the scheme outside the allowed range, i.e. such that it includes the grid $i$. In these two cases below it just returns the scheme centered at the maximum possible value, $i-2$ and $i+2$:

> dfdandoff(u,[x],ud,[i],[hx],[1],4,[-3]);

$$\frac{\partial}{\partial x}u_i = \frac{1}{4}\frac{ud_{i-4}}{hx} - \frac{4}{3}\frac{ud_{i-3}}{hx} + 3\frac{ud_{i-2}}{hx} - 4\frac{ud_{i-1}}{hx} + \frac{25}{12}\frac{ud_i}{hx} \tag{C.23}$$

> dfdandoff(u,[x],ud,[i],[hx],[1],4,[3]);

$$\frac{\partial}{\partial x}u_i = -\frac{25}{12}\frac{ud_i}{hx} + 4\frac{ud_{i+1}}{hx} - 3\frac{ud_{i+2}}{hx} + \frac{4}{3}\frac{ud_{i+3}}{hx} - \frac{1}{4}\frac{ud_{i+4}}{hx} \tag{C.24}$$

The next three examples refers to second order approximation to a second order differential operator. The examples are for centered, forward and backward schemes respectively:

> dfdandoff(u,[x],ud,[i],[hx],[2],2,[0]);

$$\frac{\partial^2}{\partial x^2}u_i = \frac{ud_{i-1}}{hx^2} - 2\frac{ud_i}{hx^2} + \frac{ud_{i+1}}{hx^2} \tag{C.25}$$

> a2ch2:=array(-1..1,[1,-2,1]):
> fdaeval(a2ch2,1/(h**2));

$$\mathrm{D}^{(2)}\left(u\right)\left(x\right) + \frac{1}{12}\mathrm{D}^{(4)}\left(u\right)\left(x\right)h^2 \tag{C.26}$$

> dfdandoff(u,[x],ud,[i],[hx],[2],2,[1]);

$$\frac{\partial^2}{\partial x^2}u_i = 2\frac{ud_i}{hx^2} - 5\frac{ud_{i+1}}{hx^2} + 4\frac{ud_{i+2}}{hx^2} - \frac{ud_{i+3}}{hx^2} \tag{C.27}$$

> a2fw1h2:=array(0..3,[2,-5,4,-1]):
> fdaeval(a2fw1h2,1/(h**2));

$$\mathrm{D}^{(2)}\left(u\right)\left(x\right) - \frac{11}{12}\mathrm{D}^{(4)}\left(u\right)\left(x\right)h^2 - \mathrm{D}^{(5)}\left(u\right)\left(x\right)h^3 \tag{C.28}$$

> dfdandoff(u,[x],ud,[i],[hx],[2],2,[-1]);

$$\frac{\partial^2}{\partial x^2}u_i = -\frac{ud_{i-3}}{hx^2} + 4\,\frac{ud_{i-2}}{hx^2} - 5\,\frac{ud_{i-1}}{hx^2} + 2\,\frac{ud_i}{hx^2} \tag{C.29}$$

```
> a2bw1h2:=array(-3..0,[-1,4,-5,2]):
> fdaeval(a2bw1h2,1/(h**2));
```

$$\mathrm{D}^{(2)}(u)(x) - \frac{11}{12}\,\mathrm{D}^{(4)}(u)(x)\,h^2 + \mathrm{D}^{(5)}(u)(x)\,h^3 \tag{C.30}$$

Turning the attention to the forth order approximation to a second order differential operator:

```
> dfdandoff(u,[x],ud,[i],[hx],[2],4,[0]);
```

$$\frac{\partial^2}{\partial x^2}u_i = -\frac{1}{12}\,\frac{ud_{i-2}}{hx^2} + \frac{4}{3}\,\frac{ud_{i-1}}{hx^2} - \frac{5}{2}\,\frac{ud_i}{hx^2} + \frac{4}{3}\,\frac{ud_{i+1}}{hx^2} - \frac{1}{12}\,\frac{ud_{i+2}}{hx^2} \tag{C.31}$$

```
> a2ch4:=array(-2..2,[-1,16,-30,16,-1]):
> fdaeval(a2ch4,1/(12*h**2));
```

$$\mathrm{D}^{(2)}(u)(x) - \frac{1}{90}\,\mathrm{D}^{(6)}(u)(x)\,h^4 \tag{C.32}$$

Note that on the following example the approximation is not totally backward. These kind of operators may be useful to set for regions of the domain very close to the boundaries.

```
> dfdandoff(u,[x],ud,[i],[hx],[2],4,[-1]);
```

$$\frac{\partial^2}{\partial x^2}u_i = \frac{1}{12}\,\frac{ud_{i-4}}{hx^2} - \frac{1}{2}\,\frac{ud_{i-3}}{hx^2} + \frac{7}{6}\,\frac{ud_{i-2}}{hx^2} - \frac{1}{3}\,\frac{ud_{i-1}}{hx^2} - \frac{5}{4}\,\frac{ud_i}{hx^2} + \frac{5}{6}\,\frac{ud_{i+1}}{hx^2} \tag{C.33}$$

```
> a2bw1h4:=array(-4..1,[1,-6,14,-4,-15,10]):
> fdaeval(a2bw1h4,1/(12*h**2));
```

$$\mathrm{D}^{(2)}(u)(x) + \frac{13}{180}\,\mathrm{D}^{(6)}(u)(x)\,h^4 - \frac{1}{12}\,\mathrm{D}^{(7)}(u)(x)\,h^5 \tag{C.34}$$

```
> dfdandoff(u,[x],ud,[i],[hx],[2],4,[1]);
```

$$\frac{\partial^2}{\partial x^2}u_i = \frac{5}{6}\,\frac{ud_{i-1}}{hx^2} - \frac{5}{4}\,\frac{ud_i}{hx^2} - \frac{1}{3}\,\frac{ud_{i+1}}{hx^2} + \frac{7}{6}\,\frac{ud_{i+2}}{hx^2} - \frac{1}{2}\,\frac{ud_{i+3}}{hx^2} + \frac{1}{12}\,\frac{ud_{i+4}}{hx^2} \tag{C.35}$$

```
> a2fw1h4:=array(-1..4,[10,-15,-4,14,-6,1]):
> fdaeval(a2fw1h4,1/(12*h**2));
```

$$\mathrm{D}^{(2)}(u)(x) + \frac{13}{180}\,\mathrm{D}^{(6)}(u)(x)\,h^4 + \frac{1}{12}\,\mathrm{D}^{(7)}(u)(x)\,h^5 \tag{C.36}$$

On the other hand a fully forward operator version would be like:

```
> dfdandoff(u,[x],ud,[i],[hx],[2],4,[2]);
```

$$\frac{\partial^2}{\partial x^2}u_i = \frac{15}{4}\frac{ud_i}{hx^2} - \frac{77}{6}\frac{ud_{i+1}}{hx^2} + \frac{107}{6}\frac{ud_{i+2}}{hx^2} - 13\frac{ud_{i+3}}{hx^2} + \frac{61}{12}\frac{ud_{i+4}}{hx^2} - \frac{5}{6}\frac{ud_{i+5}}{hx^2} \tag{C.37}$$

```
> a2fw2h4:=array(0..5,[45,-154,214,-156,61,-10]):
> fdaeval(a2fw2h4,1/(12*h**2));
```

$$\mathrm{D}^{(2)}(u)(x) - \frac{137}{180}\mathrm{D}^{(6)}(u)(x)h^4 - \frac{19}{12}\mathrm{D}^{(7)}(u)(x)h^5 \tag{C.38}$$

### C.1.3 Higher order differential operators on 1D functions $O(h^2)$

In order to illustrate the versatility of the present procedure, some higher order differential operators are shown below with their respective independent check for correctness. Despite the appearance of such operators are rare, they are still present on some important equations in mathematics and enginnering.

```
> dfdandoff(u,[x],ud,[i],[hx],[4],2,[0]);
```

$$\frac{\partial^4}{\partial x^4}u_i = \frac{ud_{i-2}}{hx^4} - 4\frac{ud_{i-1}}{hx^4} + 6\frac{ud_i}{hx^4} - 4\frac{ud_{i+1}}{hx^4} + \frac{ud_{i+2}}{hx^4} \tag{C.39}$$

```
> a4ch2:=array(-2..2,[1,-4,6,-4,1]):
> fdaeval(a4ch2,1/h**4);
```

$$\mathrm{D}^{(4)}(u)(x) + \frac{1}{6}\mathrm{D}^{(6)}(u)(x)h^2 \tag{C.40}$$

```
> dfdandoff(u,[x],ud,[i],[hx],[4],2,[-1]);
```

$$\frac{\partial^4}{\partial x^4}u_i = -\frac{ud_{i-4}}{hx^4} + 6\frac{ud_{i-3}}{hx^4} - 14\frac{ud_{i-2}}{hx^4} + 16\frac{ud_{i-1}}{hx^4} - 9\frac{ud_i}{hx^4} + 2\frac{ud_{i+1}}{hx^4} \tag{C.41}$$

```
> a4bw1h2:=array(-4..1,[-1,6,-14,16,-9,2]):
> fdaeval(a4bw1h2,1/h**4);
```

$$\mathrm{D}^{(4)}(u)(x) - \frac{5}{6}\mathrm{D}^{(6)}(u)(x)h^2 + \mathrm{D}^{(7)}(u)(x)h^3 \tag{C.42}$$

```
> dfdandoff(u,[x],ud,[i],[hx],[4],2,[-2]);
```

$$\frac{\partial^4}{\partial x^4}u_i = -2\frac{ud_{i-5}}{hx^4} + 11\frac{ud_{i-4}}{hx^4} - 24\frac{ud_{i-3}}{hx^4} + 26\frac{ud_{i-2}}{hx^4} - 14\frac{ud_{i-1}}{hx^4} + 3\frac{ud_i}{hx^4} \tag{C.43}$$

```
> a4bw2h2:=array(-5..0,[-2,11,-24,26,-14,3]):
> fdaeval(a4bw2h2,1/h**4);
```

$$\mathrm{D}^{(4)}(u)(x) - \frac{17}{6}\mathrm{D}^{(6)}(u)(x)h^2 + 5\mathrm{D}^{(7)}(u)(x)h^3 \tag{C.44}$$

> dfdandoff(u,[x],ud,[i],[hx],[3],2,[0]);

$$\frac{\partial^3}{\partial x^3}u_i = -\frac{1}{2}\,\frac{ud_{i-2}}{hx^3} + \frac{ud_{i-1}}{hx^3} - \frac{ud_{i+1}}{hx^3} + \frac{1}{2}\,\frac{ud_{i+2}}{hx^3} \tag{C.45}$$

> a3ch2:=array(-2..2,[-1,2,0,-2,1]):
> fdaeval(a3ch2,1/(2*h**3));

$$\mathrm{D}^{(3)}\left(u\right)\left(x\right) + \frac{1}{4}\,\mathrm{D}^{(5)}\left(u\right)\left(x\right)h^2 \tag{C.46}$$

> dfdandoff(u,[x],ud,[i],[hx],[3],2,[1]);

$$\frac{\partial^3}{\partial x^3}u_i = -\frac{3}{2}\,\frac{ud_{i-1}}{hx^3} + 5\,\frac{ud_i}{hx^3} - 6\,\frac{ud_{i+1}}{hx^3} + 3\,\frac{ud_{i+2}}{hx^3} - \frac{1}{2}\,\frac{ud_{i+3}}{hx^3} \tag{C.47}$$

> a3fw1h2:=array(-1..3,[-3,10,-12,6,-1]):
> fdaeval(a3fw1h2,1/(2*h**3));

$$\mathrm{D}^{(3)}\left(u\right)\left(x\right) - \frac{1}{4}\,\mathrm{D}^{(5)}\left(u\right)\left(x\right)h^2 - \frac{1}{4}\,\mathrm{D}^{(6)}\left(u\right)\left(x\right)h^3 \tag{C.48}$$

> dfdandoff(u,[x],ud,[i],[hx],[3],2,[2]);

$$\frac{\partial^3}{\partial x^3}u_i = -\frac{5}{2}\,\frac{ud_i}{hx^3} + 9\,\frac{ud_{i+1}}{hx^3} - 12\,\frac{ud_{i+2}}{hx^3} + 7\,\frac{ud_{i+3}}{hx^3} - \frac{3}{2}\,\frac{ud_{i+4}}{hx^3} \tag{C.49}$$

> a3fw2h2:=array(0..4,[-5,18,-24,14,-3]):
> fdaeval(a3fw2h2,1/(2*h**3));

$$\mathrm{D}^{(3)}\left(u\right)\left(x\right) - \frac{7}{4}\,\mathrm{D}^{(5)}\left(u\right)\left(x\right)h^2 - \frac{5}{2}\,\mathrm{D}^{(6)}\left(u\right)\left(x\right)h^3 \tag{C.50}$$

### C.1.4 Differential operators on 2D functions

This section illustrate the application of the discretization procedure on 2 dimensional functions.

> alias(v=v(x,y));

$$u,\ v \tag{C.51}$$

First remember that the coordinates should be entered in lexicographic order. If one tries otherwise it returns the following error:

> dfdandoff(v,[y,x],vd,[j,i],[hy,hx],[1,1],2,[0,0]);
Error, (in dfdandoff) Coordinate list, 2nd argument, must be entered in lexicographical order.

The command sort can help you to decide if you entered in lexicographic order or not:

> sort([y,x],'lexorder');

$$[x, y] \tag{C.52}$$

> dfdandoff(v,[x,y],vd,[i,j],[hx,hy],[1,1],2,[0,0]);

$$\frac{\partial^2}{\partial x \partial y} v_{i,j} = \frac{1}{4}\,\frac{vd_{i-1,j-1}}{hx\,hy} - \frac{1}{4}\,\frac{vd_{i-1,j+1}}{hx\,hy} - \frac{1}{4}\,\frac{vd_{i+1,j-1}}{hx\,hy} + \frac{1}{4}\,\frac{vd_{i+1,j+1}}{hx\,hy} \tag{C.53}$$

Note that for each dimension a centered scheme is evaluated in the corresponding coordinate direction. This totally centered scheme may not be the way to go for higher dimensions in general but at least can be used to verify if a numerical solution satisfies the differential operator to the order prescribed, i.e. it may be a building block for an independent residual evaluator.

Most of the time the boundary conditions and regions of the domain close to the boundaries require a modification of the approximation scheme. The following examples demonstrate some of the possibilities for these changes:

> dfdandoff(v,[x,y],vd,[i,j],[hx,hy],[1,0],2,[1,0]);

$$\frac{\partial}{\partial x} v_{i,j} = -\frac{3}{2}\,\frac{vd_{i,j}}{hx} + 2\,\frac{vd_{i+1,j}}{hx} - \frac{1}{2}\,\frac{vd_{i+2,j}}{hx} \tag{C.54}$$

> dfdandoff(v,[x,y],vd,[i,j],[hx,hy],[1,1],2,[1,0]);

$$\frac{\partial^2}{\partial x \partial y} v_{i,j} = \frac{3}{4}\,\frac{vd_{i,j-1}}{hx\,hy} - \frac{3}{4}\,\frac{vd_{i,j+1}}{hx\,hy} - \frac{vd_{i+1,j-1}}{hx\,hy} + \frac{vd_{i+1,j+1}}{hx\,hy} + \frac{1}{4}\,\frac{vd_{i+2,j-1}}{hx\,hy} - \frac{1}{4}\,\frac{vd_{i+2,j+1}}{hx\,hy} \tag{C.55}$$

> dfdandoff(v,[x,y],vd,[i,j],[hx,hy],[1,1],2,[1,-1]);

$$\begin{aligned}\frac{\partial^2}{\partial x \partial y} v_{i,j} = &-\frac{3}{4}\,\frac{vd_{i,j-2}}{hx\,hy} + 3\,\frac{vd_{i,j-1}}{hx\,hy} - \frac{9}{4}\,\frac{vd_{i,j}}{hx\,hy} + \frac{vd_{i+1,j-2}}{hx\,hy} \\ &- 4\,\frac{vd_{i+1,j-1}}{hx\,hy} + 3\,\frac{vd_{i+1,j}}{hx\,hy} - \frac{1}{4}\,\frac{vd_{i+2,j-2}}{hx\,hy} + \frac{vd_{i+2,j-1}}{hx\,hy} - \frac{3}{4}\,\frac{vd_{i+2,j}}{hx\,hy}\end{aligned} \tag{C.56}$$

> dfdandoff(v,[x,y],vd,[i,j],[hx,hy],[0,2],2,[0,1]);

$$\frac{\partial^2}{\partial y^2} v_{i,j} = 2\,\frac{vd_{i,j}}{hy^2} - 5\,\frac{vd_{i,j+1}}{hy^2} + 4\,\frac{vd_{i,j+2}}{hy^2} - \frac{vd_{i,j+3}}{hy^2} \tag{C.57}$$

The error below comes from $[0, 0]$ input. The focus of the procedure is on differential operators. Therefore at least one first order derivative is needed to avoid erroneous output.

> dfdandoff(v,[x,y],vd,[i,j],[hx,hy],[0,0],2,[1,-1]);
Error, (in dfdandoff) invalid input: diff expects 2 or more arguments, but received 1

As one last remark, the names of the various symbols are quite arbitrary:

> dfdandoff(v,[x,y],vd,[n,m],[h,h],[1,1],2,[0,0]);

$$\frac{\partial^2}{\partial x \partial y} v_{n,m} = \frac{1}{4}\,\frac{vd_{n-1,m-1}}{h^2} - \frac{1}{4}\,\frac{vd_{n-1,m+1}}{h^2} - \frac{1}{4}\,\frac{vd_{n+1,m-1}}{h^2} + \frac{1}{4}\,\frac{vd_{n+1,m+1}}{h^2} \tag{C.58}$$

### C.1.5 Differential operators on 3D and higher dimensional functions

A couple of examples to illustrate the use of the procedure on 3d functions:

> alias(w=w(x,y,z));

$$u,\ v,\ w \tag{C.59}$$

> dfdandoff(w,[x,y,z],wd,[i,j,k],[hx,hy,hz],[0,0,1],2,[1,0,1]);

$$\frac{\partial}{\partial z}w_{i,j,k} = -\frac{3}{2}\,\frac{wd_{i,j,k}}{hz} + 2\,\frac{wd_{i,j,k+1}}{hz} - \frac{1}{2}\,\frac{wd_{i,j,k+2}}{hz} \tag{C.60}$$

> dfdandoff(w,[x,y,z],wd,[i,j,k],[hx,hy,hz],[0,0,1],2,[0,0,1]);

$$\frac{\partial}{\partial z}w_{i,j,k} = -\frac{3}{2}\,\frac{wd_{i,j,k}}{hz} + 2\,\frac{wd_{i,j,k+1}}{hz} - \frac{1}{2}\,\frac{wd_{i,j,k+2}}{hz} \tag{C.61}$$

> dfdandoff(w,[x,y,z],wd,[i,j,k],[hx,hy,hz],[1,1,1],2,[1,0,1]);

$$\begin{aligned}\frac{\partial^3}{\partial y\partial x\partial z}w_{i,j,k} = &-\frac{3}{8}\,\frac{wd_{i+2,j-1,k}}{hx\ hy\ hz} - \frac{1}{8}\,\frac{wd_{i+2,j-1,k+2}}{hx\ hy\ hz} + \frac{1}{2}\,\frac{wd_{i+2,j-1,k+1}}{hx\ hy\ hz}\\ &+ \frac{3}{2}\,\frac{wd_{i+1,j-1,k}}{hx\ hy\ hz} - 2\,\frac{wd_{i+1,j-1,k+1}}{hx\ hy\ hz} + \frac{1}{2}\,\frac{wd_{i+1,j-1,k+2}}{hx\ hy\ hz} - \frac{3}{2}\,\frac{wd_{i+1,j+1,k}}{hx\ hy\ hz}\\ &+ 2\,\frac{wd_{i+1,j+1,k+1}}{hx\ hy\ hz} - \frac{1}{2}\,\frac{wd_{i+1,j+1,k+2}}{hx\ hy\ hz} + \frac{3}{8}\,\frac{wd_{i+2,j+1,k}}{hx\ hy\ hz} - \frac{1}{2}\,\frac{wd_{i+2,j+1,k+1}}{hx\ hy\ hz}\\ &+ \frac{1}{8}\,\frac{wd_{i+2,j+1,k+2}}{hx\ hy\ hz} + \frac{9}{8}\,\frac{wd_{i,j+1,k}}{hx\ hy\ hz} - \frac{3}{2}\,\frac{wd_{i,j+1,k+1}}{hx\ hy\ hz} + \frac{3}{8}\,\frac{wd_{i,j+1,k+2}}{hx\ hy\ hz}\\ &- \frac{9}{8}\,\frac{wd_{i,j-1,k}}{hx\ hy\ hz} + \frac{3}{2}\,\frac{wd_{i,j-1,k+1}}{hx\ hy\ hz} - \frac{3}{8}\,\frac{wd_{i,j-1,k+2}}{hx\ hy\ hz}\end{aligned} \tag{C.62}$$

and to higher dimensional functions; as high as the memory resources allow:

> alias(f=f(t,x,y,z));

$$u,\ v,\ w,\ f \tag{C.63}$$

> dfdandoff(f,[t,x,y,z],fd,[l,i,j,k],[h,h,h,h],[1,1,1,1],2,[0,0,0,0]);

$$\begin{aligned}\frac{\partial^4}{\partial y\partial x\partial t\partial z}f_{l,i,j,k} = &-\frac{1}{16}\,\frac{fd_{l-1,i+1,j+1,k+1}}{h^4} + \frac{1}{16}\,\frac{fd_{l-1,i+1,j+1,k-1}}{h^4} - \frac{1}{16}\,\frac{fd_{l+1,i+1,j+1,k-1}}{h^4}\\ &+ \frac{1}{16}\,\frac{fd_{l+1,i+1,j+1,k+1}}{h^4} + \frac{1}{16}\,\frac{fd_{l+1,i-1,j+1,k-1}}{h^4} - \frac{1}{16}\,\frac{fd_{l+1,i-1,j+1,k+1}}{h^4} + \frac{1}{16}\,\frac{fd_{l-1,i-1,j+1,k+1}}{h^4}\\ &- \frac{1}{16}\,\frac{fd_{l-1,i-1,j+1,k-1}}{h^4} - \frac{1}{16}\,\frac{fd_{l-1,i+1,j-1,k-1}}{h^4} + \frac{1}{16}\,\frac{fd_{l-1,i+1,j-1,k+1}}{h^4} - \frac{1}{16}\,\frac{fd_{l+1,i+1,j-1,k+1}}{h^4}\\ &- \frac{1}{16}\,\frac{fd_{l-1,i-1,j-1,k+1}}{h^4} + \frac{1}{16}\,\frac{fd_{l-1,i-1,j-1,k-1}}{h^4} + \frac{1}{16}\,\frac{fd_{l+1,i+1,j-1,k-1}}{h^4} + \frac{1}{16}\,\frac{fd_{l+1,i-1,j-1,k+1}}{h^4}\\ &- \frac{1}{16}\,\frac{fd_{l+1,i-1,j-1,k-1}}{h^4}\end{aligned} \tag{C.64}$$

## C.2 Residual evaluator for n-dimensional PDEs

### C.2.1 Introduction

'resndoff' is a Maple procedure deviced to evaluate residual equations from a multidimensional, multi-order partial differential differential equation. This routine was initially devised with elliptic PDEs in mind, i.e. devised to return its discrete equation version, the residual equation. Nonetheless it can in principle handle time dependent PDE as well. The finite difference equations returned may not be appropriated to solve the equations though. For example, when using a centered scheme in time it returns an equation referencing only point at $t - dt$ and $t + dt$, while for time dependent PDEs one of the most convenient method, the Crank-Nicholson method, is centered at the mid point in time, i.e. at $t+dt/2$. Still this procedure can return a finite difference equation appropriate to evaluate independently the residual of the discrete version of the differential equation.

The use of 'resndoff' is illustraded below for non-linear Poisson equations in different dimensions. The advection equation in 1D is taken as a simple case study of time-dependent PDE.

### C.2.2 Poisson equation in 1D

> alias(u=u(x),f=f(x));

$$u,\ v,\ w,\ f \tag{C.65}$$

> POI1D:=diff(u,x$2)+sigma*u**2=f;

$$POI1D := \frac{d^2}{dx^2}u + \sigma\, u^2 = f \tag{C.66}$$

> RES_POI1D:=resndoff(POI1D,[u,f],[x],[ud,fd],[i],[h],2,[0]);

$$RES\_POI1D := \frac{ud_{i-1}}{h^2} - 2\,\frac{ud_i}{h^2} + \frac{ud_{i+1}}{h^2} + \sigma\, {ud_i}^2 - fd_i = 0 \tag{C.67}$$

The first argument is the equation to be discretized. The second is a list of the functions appearing in that equation. The third is a list of the coordinates in alphabetical (lexicographical) order. The fourth is a list with the names of the the functions defined only on the mesh points (see 'dfdandoff' above). The fifth one is a list with the names of the finite difference indexes. The sixth one is a list of names for coordinate spacings for each of the coordinates. The seventh argument is the order of the finite difference scheme. At last, the list indicating whether or not the scheme is centered or off-centered along a particular coordinate. For example, the same equation but with a backward scheme:

> resndoff(POI1D,[u,f],[x],[ud,fd],[i],[hx],2,[-1]);

$$-\frac{ud_{i-3}}{hx^2} + 4\,\frac{ud_{i-2}}{hx^2} - 5\,\frac{ud_{i-1}}{hx^2} + 2\,\frac{ud_i}{hx^2} + \sigma\, {ud_i}^2 - fd_i = 0 \tag{C.68}$$

If you don't like the fact the continuum function and its discrete version have different names,

you can always, after the residual equation is written, unalias and rewrite the residual equation with the continuum function names:

> alias(u=u,f=f);

$$v,\ w \tag{C.69}$$

> RES_POI1D:=subs(ud=u,fd=f,RES_POI1D);

$$RES\_POI1D := \frac{u_{i-1}}{h^2} - 2\,\frac{u_i}{h^2} + \frac{u_{i+1}}{h^2} + \sigma\,{u_i}^2 - f_i = 0 \tag{C.70}$$

### C.2.3 Poisson equation in 2D

Poisson equation in 2D for centered schemes: > alias(u=u(x,y),f=f(x,y));

$$v,\ w,\ u,\ f \tag{C.71}$$

> POI2D:=diff(u,x$2)+diff(u,y$2)+sigma*u**2=f;

$$POI2D := \frac{\partial^2}{\partial x^2}u + \frac{\partial^2}{\partial y^2}u + \sigma\,u^2 = f \tag{C.72}$$

> resndoff(POI2D,[u,f],[x,y],[ud,fd],[i,j],[hx,hy],2,[0,0]);

$$\frac{ud_{i-1,j}}{{hx}^2} - 2\,\frac{ud_{i,j}}{{hx}^2} + \frac{ud_{i+1,j}}{{hx}^2} + \frac{ud_{i,j-1}}{{hy}^2} - 2\,\frac{ud_{i,j}}{{hy}^2} + \frac{ud_{i,j+1}}{{hy}^2} + \sigma\,{ud_{i,j}}^2 - fd_{i,j} = 0 \tag{C.73}$$

The same equation but in a form convenient to be applied at the right edge of the 2D domain:

> resndoff(POI2D,[u,f],[x,y],[ud,fd],[i,j],[hx,hy],2,[-1,0]);

$$-\frac{ud_{i-3,j}}{{hx}^2} + 4\,\frac{ud_{i-2,j}}{{hx}^2} - 5\,\frac{ud_{i-1,j}}{{hx}^2} + 2\,\frac{ud_{i,j}}{{hx}^2} + \frac{ud_{i,j-1}}{{hy}^2} - 2\,\frac{ud_{i,j}}{{hy}^2} + \frac{ud_{i,j+1}}{{hy}^2} + \sigma\,{ud_{i,j}}^2 - fd_{i,j} = 0 \tag{C.74}$$

To illustrate the versatility of the procedure consider the following example where mixed derivatives, non-constant and non-linear terms are present:

> alias(u=u(x,y),v=v(x,y));

$$v,\ w,\ u \tag{C.75}$$

> EQN:=diff(u,x,y)=(1-x)*(1-y)*u**4*diff(u,y)*diff(v,x$2)-v*diff(u,x)*x;

$$EQN := \frac{\partial^2}{\partial x \partial y}u = (1-x)\,(1-y)\,u^4 \left(\frac{\partial}{\partial y}u\right) \frac{\partial^2}{\partial x^2}v\,(t,r) - v\,(t,r) \left(\frac{d}{dx}u\right) x \tag{C.76}$$

> resndoff(EQN,[u,v],[x,y],[ud,vd],[i,j],[h,h],2,[0,0]);

$$\frac{1}{4}\,\frac{ud_{i-1,j-1}}{h^2}-\frac{1}{4}\,\frac{ud_{i-1,j+1}}{h^2}-\frac{1}{4}\,\frac{ud_{i+1,j-1}}{h^2}+\frac{1}{4}\,\frac{ud_{i+1,j+1}}{h^2}$$
$$-\left(1-x_i\right)\left(1-y_j\right)ud_{i,j}{}^4\left(-\frac{1}{2}\,\frac{ud_{i,j-1}}{h}+\frac{1}{2}\,\frac{ud_{i,j+1}}{h}\right)\left(\frac{vd_{i-1,j}}{h^2}-2\,\frac{vd_{i,j}}{h^2}+\frac{vd_{i+1,j}}{h^2}\right)$$
$$+vd_{i,j}\left(-\frac{1}{2}\,\frac{ud_{i-1,j}}{h}+\frac{1}{2}\,\frac{ud_{i+1,j}}{h}\right)x_i=0 \tag{C.77}$$

### C.2.4 Poisson equation in 3D

> alias(u=u(x,y,z),f=f(x,y,z));

$$w,\ u,\ f \tag{C.78}$$

> POI3D:=diff(u,x$2)+diff(u,y$2)+diff(u,z$2)+sigma*u**2=f;

$$POI3D:=\frac{\partial^2}{\partial x^2}u+\frac{\partial^2}{\partial y^2}u+\frac{\partial^2}{\partial z^2}u+\sigma\,u^2=f \tag{C.79}$$

> resndoff(POI3D,[u,f],[x,y,z],[ud,fd],[i,j,k],[h,h,h],2,[0,0,0]);

$$\frac{ud_{i-1,j,k}}{h^2}-6\,\frac{ud_{i,j,k}}{h^2}+\frac{ud_{i+1,j,k}}{h^2}+\frac{ud_{i,j-1,k}}{h^2}+\frac{ud_{i,j+1,k}}{h^2}+\frac{ud_{i,j,k-1}}{h^2}+\frac{ud_{i,j,k+1}}{h^2}+\sigma\,ud_{i,j,k}{}^2-fd_{i,j,k}=0 \tag{C.80}$$

### C.2.5 Poisson equation in 3D compactified cartesian coordinates

> alias(u=u(chi,eta,zeta),f=f(chi,eta,zeta));

$$w,\ u,\ f \tag{C.81}$$

> POI3DCP:=(1-chi**2)*diff((1-chi**2)*diff(u,chi),chi)+(1-eta**2)*diff((1-eta**2)*diff(u,eta),eta)
+(1-zeta**2)*diff((1-zeta**2)*diff(u,zeta),zeta)+sigma*u**2=f;

$$\left(1-\chi^2\right)\left(-2\,\chi\,\frac{d}{d\chi}u+\left(1-\chi^2\right)\frac{d^2}{d\chi^2}u\right)+\left(1-\eta^2\right)\left(-2\,\eta\,\frac{d}{d\eta}u+\left(1-\eta^2\right)\frac{d^2}{d\eta^2}u\right)$$
$$+\left(1-\zeta^2\right)\left(-2\,\zeta\,\frac{d}{d\zeta}u+\left(1-\zeta^2\right)\frac{d^2}{d\zeta^2}u\right)+\sigma\,u^2=f \tag{C.82}$$

> resndoff(POI3DCP,[u,f],[chi,eta,zeta],[ud,fd],[i,j,k],[hx,hy,hz],2,[0,0,0]);

$$\left(1-{\chi_i}^2\right)\left[-2\,\chi_i\left(-\frac{1}{2}\,\frac{ud_{i-1,j,k}}{hx}+\frac{1}{2}\,\frac{ud_{i+1,j,k}}{hx}\right)+\left(1-{\chi_i}^2\right)\left(\frac{ud_{i-1,j,k}}{{hx}^2}-2\,\frac{ud_{i,j,k}}{{hx}^2}+\frac{ud_{i+1,j,k}}{{hx}^2}\right)\right]$$
$$+\left(1-{\eta_j}^2\right)\left[-2\,\eta_j\left(-\frac{1}{2}\,\frac{ud_{i,j-1,k}}{hy}+\frac{1}{2}\,\frac{ud_{i,j+1,k}}{hy}\right)+\left(1-{\eta_j}^2\right)\left(\frac{ud_{i,j-1,k}}{{hy}^2}-2\,\frac{ud_{i,j,k}}{{hy}^2}+\frac{ud_{i,j+1,k}}{{hy}^2}\right)\right]$$
$$+\left(1-{\zeta_k}^2\right)\left[-2\,\zeta_k\left(-\frac{1}{2}\,\frac{ud_{i,j,k-1}}{hz}+\frac{1}{2}\,\frac{ud_{i,j,k+1}}{hz}\right)+\left(1-{\zeta_k}^2\right)\left(\frac{ud_{i,j,k-1}}{{hz}^2}-2\,\frac{ud_{i,j,k}}{{hz}^2}+\frac{ud_{i,j,k+1}}{{hz}^2}\right)\right]$$
$$+\sigma\,{ud_{i,j,k}}^2-fd_{i,j,k}=0 \quad \text{(C.83)}$$

### C.2.6 Advection equation in 1D

This section turns the attention to time dependent PDEs. The analysis is focused on the advection equation but the principles are easily generalized.

> alias(v=v(t,r));

$$w,\ v \quad \text{(C.84)}$$

> ADVEQ:=diff(v,t)-diff(v,r)=0;

$$ADVEQ := \frac{\partial}{\partial t}v\left(t,r\right)-\frac{\partial}{\partial r}v\left(t,r\right)=0 \quad \text{(C.85)}$$

> RES_ADVEQ:=resndoff(ADVEQ,[v],[r,t],[vd],[i,k],[hr,ht],2,[0,0]);

$$RES\_ADVEQ := -\frac{1}{2}\,\frac{vd_{i,k-1}}{ht}+\frac{1}{2}\,\frac{vd_{i,k+1}}{ht}+\frac{1}{2}\,\frac{vd_{i-1,k}}{hr}-\frac{1}{2}\,\frac{vd_{i+1,k}}{hr}=0 \quad \text{(C.86)}$$

In principle there is no problem with this residual equation. However in practice this form won't probably be very useful to either solve the equation or evaluate the independent residual. First, it is not common to store the data along the time direction. Usually only the present time step, the future and at most one at the past are stored. The common practice then is to rename the variable at different time steps. For example, $v(t,r)$ would go to $vnm1(r)$, $v(r)$ and $vp1(r)$ for the past, present and future time step and as the solution are obtained previous time steps are discarded. The way to proceed then is first discretize the spatial operators of the equation and then substitute by hand the discrete version of the time operator:

> RES_ADVEQ:=resndoff(ADVEQ,[v],[r],[vd],[i],[hr],2,[0]);

$$RES\_ADVEQ := \frac{\partial}{\partial t}vd_i+\frac{1}{2}\,\frac{vd_{i-1}}{hr}-\frac{1}{2}\,\frac{vd_{i+1}}{hr}=0 \quad \text{(C.87)}$$

> RES_ADVEQ:=subs(diff(vd[i],t)=(vdnp1[i]-vd[i])/ht,RES_ADVEQ);

$$RES\_ADVEQ := \frac{1}{2}\,\frac{vd_{i-1}}{hr}-\frac{1}{2}\,\frac{vd_{i+1}}{hr}=\frac{{vdnp1}_i-vd_i}{ht} \quad \text{(C.88)}$$

If you look very carefully at the result of the discrete equation above, you can notice that there was an overall sign flip when compared to the continuum equation ADVEQ. The reason for that comes from the fact that $vd_i$ is a symbol instead of a function and a derivative of a symbol in MAPLE vanishes. That subtle fact can be very hard to detect since the only effect on equations of the kind above is an overall sign flip. To avoid this problem, MAPLE provides a mechanism to delay evaluation of the expression: enclose the whole expression in quotes. The same example above then becomes:

> RES_ADVEQ:=‘resndoff(ADVEQ,[v],[r],[vd],[i],[hr],2,[0])’;

$$RES\_ADVEQ := resndoff\ (ADVEQ, [v], [r], [vd], [i], [hr], 2, [0]) \tag{C.89}$$

> RES_ADVEQ:=subs(‘diff(vd[i],t)’=(vdnp1[i]-vd[i])/ht,RES_ADVEQ);

$$RES\_ADVEQ := \frac{vdnp1_{\,i} - vd_i}{ht} + \frac{1}{2}\,\frac{vd_{i-1}}{hr} - \frac{1}{2}\,\frac{vd_{i+1}}{hr} = 0 \tag{C.90}$$

In this case the evaluation of the derivative of the symbol $vd_i$ is delayed and the substitution by its discrete version occurs first and gives the correct discrete form for the equation.

The second point to pay attention is that the equation was discretized with second order approximation to the spatial operators but only first order in time. This doesn't actually represent a problem if the goal is to use the discrete equation only to evaluate the independent residual of the advection equation. In this case the convergence would be first order but enough to evaluate the consistency and convergence of the numerical solution. On the other hand if one intends to use this discrete equation to actually find the solution via Crank-Nicholson iteration for example, then one last step would still be missing. This discrete form is centered between the present and future time level and in order to make it second order the part of the equation that doesn't have time derivative operators must be averaged between the present and future time, according to the Crank-Nicholson scheme. As at the current time the package PDEFDAOFF doesn't have a built-in time average operator this operation is done on a case by case basis.

# APPENDIX D

# ITERATIVE SOLUTIONS OF LINEAR AND NON-LINEAR SYSTEMS

Given the discretization of a system of PDE approximated by a particular FDA scheme, the result is a set of linear or non-linear system of algebraic equations. The next questions to be asked are: how to solve this system? are there efficient ways to obtain the solution? This appendix addresses these issues in two sections: one for the linear case and the other for the non-linear one. The main techniques are explained and basic properties stated.

## D.1 Jacobi and Gauss-Seidel Iteration Methods

A system of linear algebraic equations, for example one derived from the discretization of a PDE, can be written in a matrix form as:

$$Lu = f, \tag{D.1}$$

where $L$ is a $N \times N$ matrix, while $u$ and $f$ are $N$-component vectors. Note that in this section and in the following the superscript $h$ has been dropped from the FDA notation. It should not be confused then with the notation for PDE.

Usually such a system can be solved directly by LU decomposition (Gaussian elimination) for example. That it is the case for small ($N < 100$) linear systems. However in most practical cases originated from FDA of PDEs the size of the matrix can easily become fairly large ($N \geq 10000$). Furthermore these matrices are usually banded matrices (only a few of their diagonals are filled with numbers other than zero) with an associated computational cost of $O(w^2N)$, where $w$ is the band width. The discretization of 2D or 3D PDEs results in sparse matrices with very large band widths, increasing then the computational cost to approximately $O(N^2)$. Along with the usual large values of $N$, that practically eliminates the feasibility of direct solvers implementation on the solution of FDAs. The applicability of direct solvers is restricted then to the solution of FDA from 1D PDEs where the usually small bandwidth of the resulting matrix yields the method efficient.

Fortunately there is an alternative to directly solving a linear system. The idea is to rewrite the linear system of algebraic equations in such a way that the solution is obtained through an iterated series of solution guesses. In the infinite limit, the sequence of approximations converges to the solution of the linear system. Such methods are known as iterative or fixed point methods. So for example the system of equations (D.1) can be rewritten in an equivalent system as follows:

$$u = Tu + c, \tag{D.2}$$

where $T$ is an update matrix and $c$ a constant vector characteristic of the iterative method used.

The goal is then to find a solution $u$ such that when multiplied by $T$ and added to $c$ gets the result mapped back to itself. That can be achieved iteratively by generating a sequence of approximations $\{u^{(k)}\}_{k=0}^{\infty}$ starting from an initial guess $u^{(0)}$:

$$u^{(0)} \rightarrow u^{(1)} \rightarrow u^{(2)} \rightarrow \cdots \rightarrow u^{(n)} \rightarrow \cdots \rightarrow u \tag{D.3}$$

such that each approximation labelled by the superscript $(k)$ is obtained by:

$$u^{(k)} = Tu^{(k-1)} + c, \tag{D.4}$$

where $k$ runs from 1 to $n$. The convergence or not of the sequence is controlled by the eigenvalues of the update matrix $T$. More precisely, the necessary and sufficient condition for the sequence to converge is to have the so called *spectral radius* of $T$ smaller than 1: $\rho(T) < 1$. The spectral radius is defined as the maximum of the matrix eigenvalue norms:

$$\rho(T) = \max_{1 \leq i \leq N} |\lambda_i|, \tag{D.5}$$

where $\lambda_i$ are the eigenvalues of $T$. For strictly diagonally dominant matrices $T$, i.e.

$$|T_{ii}| > \sum_{\substack{j=1 \\ j \neq i}}^{N} |T_{ij}|, \tag{D.6}$$

two iterative schemes become particularly interesting since they are known to converge for any initial guess $u^{(0)}$. The idea behind them consists of solving the $i$-th equation in (D.1) in terms of the $i$-th unknown component $u_i$ of the solution vector. In terms of components, equation (D.1) can be written as:

$$\sum_{j=1}^{N} L_{ij} u_j = f_i, \qquad \text{for} \qquad i = 1, 2, \ldots, N, \tag{D.7}$$

where the sum can be split in order to isolate the component $u_i$:

$$u_i = \frac{1}{L_{ii}} \left[ \sum_{\substack{j=1 \\ j \neq i}}^{N} -L_{ij} u_j + f_i \right] \tag{D.8}$$

and the sequence of approximations starting from the initial guess $u_i^{(0)}$ can be generated component by component such as:

$$u_i^{(k)} = \frac{1}{L_{ii}} \left[ \sum_{\substack{j=1 \\ j \neq i}}^{N} -L_{ij} u_j^{(k-1)} + f_i \right], \qquad \text{for} \qquad i = 1, 2, \ldots, N. \tag{D.9}$$

Such an iterative method is known as *Jacobi iterative method*. Note that both the current calculated approximation $u_i^{(k)}$ and the previous one $u_i^{(k-1)}$ need to be stored in the Jacobi iteration. The other iterative method is simply an improvement of the Jacobi algorithm such that as the components

of solution vector are calculated they are used in the evaluation of the subsequent components. As it is supposed to be a better approximation, it should therefore provide a better estimation of the components left. Furthermore it saves storage since only one solution vector per iteration needs to be saved. For example suppose the $i$-th component is being evaluated. Then all $j < i$ components had already been updated and their new values are used to calculate the $i$-th component:

$$u_i^{(k)} = \frac{1}{L_{ii}} \left[ \sum_{j=1}^{i-1} -L_{ij} u_j^{(k)} \sum_{j=i+1}^{N} -L_{ij} u_j^{(k-1)} + f_i \right], \qquad \text{for} \qquad i = 1, 2, \ldots, N. \tag{D.10}$$

Such a method is known as *Gauss-Seidel iterative method*.

It is useful to analyse and rewrite the iteration sequence in terms of the *residual* vector associated with the approximate solution. Considering that the components of the residual vector are not updated simultaneously, the notion of residual needs to be extended to what is called *running* residual. Let the approximate solution vector $\mathbf{u}_i^{(k)}$ be defined as:

$$\mathbf{u}_i^{(k)} = (u_1^{(k)}, u_2^{(k)}, \ldots, u_{i-1}^{(k)}, u_i^{(k-1)}, \ldots, u_N^{(k-1)})^t, \tag{D.11}$$

where the subscript $i$ in the boldface $u$ denoting the approximate vector just adds the information that the $i$-th component is being updated at that moment. That explains as well why $u_j^{(k-1)}$ is used for $j > i$: since each component is visited and updated in a lexicographic order, those components haven't been updated yet. Therefore, the corresponding running residual to the approximate solution vector can be written as:

$$\mathbf{r}_i^{(k)} = (r_{1i}^{(k)}, r_{2i}^{(k)}, \ldots, r_{Ni}^{(k)})^t, \tag{D.12}$$

where its $m$-th component is defined to be:

$$r_{mi}^{(k)} = \sum_{j=1}^{i-1} L_{mj} u_j^{(k)} + \sum_{j=i+1}^{N} L_{mj} u_j^{(k-1)} + L_{mi} u_i^{(k-1)} - f_m. \tag{D.13}$$

In particular the residual component corresponding to the approximate solution vector component being updated is simply:

$$r_{ii}^{(k)} = \sum_{j=1}^{i-1} L_{ij} u_j^{(k)} + \sum_{j=i+1}^{N} L_{ij} u_j^{(k-1)} + L_{ii} u_i^{(k-1)} - f_i. \tag{D.14}$$

Inserting the expression above for the running residual into equation (D.10) results into a more compact expression for the Gauss-Seidel iteration procedure:

$$u_i^{(k)} = u_i^{(k-1)} - \frac{r_{ii}^{(k)}}{L_{ii}}. \tag{D.15}$$

This expression is useful for comparison with a similar expression for non-linear system of algebraic equations to be derived in the next section.

## D.2 Newton-Gauss-Seidel Iteration Method

In general the FDAs resulting from the discretization of the Einstein equations for a particular matter model are actually a non-linear system of algebraic equations. Based on the assumption that the solution for this non-linear system is smooth, there is a technique extremely efficient used to obtain its solution called *Newton's method*. This method consists in linearizing the non-linear equations around the known solution $u_i$. It works as follows:

Let the system of non-linear equations be expressed in the following canonical form:

$$N_i[u_j] = 0 \qquad \text{for} \qquad i,\, j = 1, 2, \ldots, N, \tag{D.16}$$

where $N_i$ are the $N$ non-linear discrete operators acting on the grid functions $u_j$ for $j = 1, 2, \ldots, N$. This system needs then to be rewritten as a fixed point iteration system in order to define an iterative solution process for $k = 1, 2, \ldots, n$:

$$u_i^{(k)} = G_i[u_j^{(k-1)}], \tag{D.17}$$

where $G_i$ are a set of non-linear iteration update operators. Let the $i$-th component of the solution approximation at iteration $k$ be denoted by $u_i^{(k)}$. The solution of the discrete equation (D.16) can be written in terms of its $k$-th approximation $u_i^{(k)}$ as:

$$u_i = u_i^{(k)} - \delta u_i^{(k)}, \tag{D.18}$$

where $\delta u_i^{(k)}$ is the $i$-th component of the approximate solution error relative to the exact discrete solution $u_i$. Taylor expanding or linearizing around $u_i^{(k)}$, equation (D.16) becomes:

$$N_i[u_j] \equiv N_i[u_j^{(k)}] - \sum_{j=1}^{N} \left.\frac{\partial N_i}{\partial u_j}\right|_{u_j = u_j^{(k)}} \delta u_j^{(k)} + O\left((\delta u_j^{(k)})^2\right) = 0, \tag{D.19}$$

from where the running residual vector can be defined in analogy with the linear case, equation (D.13), as:

$$r_{im}^{(k)} \equiv N_m[u_1^{(k)}, u_2^{(k)}, \ldots, u_i^{(k-1)}, \ldots, u_N^{(k-1)}], \tag{D.20}$$

where $i$ indicates that the $u_i$ approximate solution component is being updated, while $m$ tells the component of the running residual vector in consideration. Also the form of equation (D.19) motivates the definition of the Jacobian matrix:

$$J_{ij}[u_p^{(k)}, u_q^{(k-1)}] = \left.\frac{\partial N_i}{\partial u_j}\right|_{u_j = u_j^{(k)}}, \tag{D.21}$$

with $p < i$ and $q \geq i$. Therefore the equation (D.19) can be written to second order in $\delta u_j^{(k)}$ as:

$$\sum_{j=1}^{N} J_{ij}[u_p^{(k)}, u_q^{(k-1)}] \delta u_j^{(k)} = r_{ii}^{(k)}. \tag{D.22}$$

One Newton's method step consists then in solving the linear system (D.22) for $\delta u_i^{(k)}$ and updating the solution approximation through the expression:

$$u_i^{(k)} = u_i^{(k-1)} - \delta u_i^{(k)}. \tag{D.23}$$

As long as the initial guess denoted by $u_i^{(0)}$ is good enough or close enough to the solution $u_i$, Newton's method is known to generate a sequence of approximations that converge quadratically to the solution of the non-linear system of equations, i.e.

$$\lim_{k\to\infty} \frac{\|u^{(k)} - u\|_\infty}{\|u^{(k-1)} - u\|_\infty^2} = c, \tag{D.24}$$

where $c$ is a constant.

Besides the choice of a convenient initial guess – that may not be so obvious for a whole class of problems – there is another drawback affecting Newton's method robustness and efficiency. Note that on each Newton's step a full linear system, equation (D.22), needs to be solved. That may represent a problem for large non-linear systems. In those cases the linear system is also solved iteratively by use for example of Gauss-Seidel iterations. A direct combination of both methods would result on a solution scheme that looks like the following:

A sequence of approximations for the error component at Newton's iteration $k$ is generated by Gauss-Seidel iterations such as:

$$\delta u_i^{(k)(0)} \to \delta u_i^{(k)(1)} \to \cdots \to \delta u_i^{(k)(l)} \to \cdots \to \delta u_i^{(k)}, \tag{D.25}$$

where the superscript $l$ labels the Gauss-Seidel iteration step. Usually $\delta u_i^{(k)(0)} = 0$ is considered as a sensible choice for initialising the sequence. The sequence is then generated by use of the equation (D.15) that, when applied on the linear system (D.22), becomes:

$$\delta u_i^{(k)(l)} = \delta u_i^{(k)(l-1)} - \frac{R_{ii}^{(k)(l)}}{J_{ii}}, \tag{D.26}$$

where $R_{ii}^{(k)(l)}$ is the running residual as defined by equation (D.13) for equation (D.22), or more explicitly:

$$R_{ii}^{(k)(l)} = \sum_{j=1}^{i-1} J_{ij}\delta u_j^{(k)(l)} + \sum_{j=i+1}^{N} J_{ij}\delta u_j^{(k)(l-1)} + J_{ii}\delta u_i^{(k)(l-1)} - r_{ii}^{(k)}. \tag{D.27}$$

After convergence of the Gauss-Seidel iteration to the solution $\delta u_i^{(k)}$, then the Newton's step would take place through equation (D.23).

This method is also known as *global* Newton-Gauss-Seidel iteration method since it solves for the corrections of *all* unknowns per Newton's step. Despite being more efficient than using direct linear solvers, the method is rarely implemented in practice. A simplified or approximated version called *local* or *point-wise* Newton-Gauss-Seidel iteration method is the method of choice to solve most of the large non-linear systems of algebraic equations. The method consists in executing only *one* Gauss-Seidel step per Newton step. Furthermore, each equation in (D.22) is only solved for

one variable at the time. Therefore, for the $i$-th equation in (D.22) all the correction variables are considered constant but the $\delta u_i^{(k)}$. Given the initial guess $\delta u_i^{(k)(0)} = 0$, this is equivalent to set the corrections $\delta u_j^{(k)(l)}$ and $\delta u_j^{(k)(l-1)}$ all equal to zero for all $j \neq i$ in equation (D.27) after the only one Gauss-Seidel step be executed. This results in:

$$R_{ii}^{(k)(l)} = J_{ii}\delta u_i^{(k)(l-1)} - r_{ii}^{(k)} \tag{D.28}$$

and, for $l = 1$, the correction becomes:

$$\delta u_i^{(k)} \equiv \delta u_i^{(k)(l)} = \frac{r_{ii}^{(k)}}{J_{ii}}, \tag{D.29}$$

from where the Newton step proceeds as usual:

$$u_i^{(k)} = u_i^{(k-1)} - \delta u_i^{(k)} = u_i^{(k-1)} - \frac{r_{ii}^{(k)}}{J_{ii}}. \tag{D.30}$$

The point-wise Newton-Gauss-Seidel iterative method explained in this section was the method applied to obtain the solutions of the FDAs approximating the evolution equations in this dissertation. It is worth remembering that those FDAs were obtained by the Crank-Nicholson discretization scheme of the evolution PDEs.