A new numerical method to construct binary neutron star initial data

Wolfgang Tichy

Department of Physics, Florida Atlantic University, Boca Raton, FL 33431, USA

Received 6 February 2009, in final form 30 June 2009
Published 21 August 2009
Online at stacks.iop.org/CQG/26/175018

Abstract
We present a new numerical method for the generation of binary neutron star initial data using a method along the lines of the Wilson–Mathews or the closely related conformal thin sandwich approach. Our method uses six different computational domains, which include spatial infinity. Each domain has its own coordinates which are chosen such that the star surfaces always coincide with domain boundaries. These properties facilitate the imposition of boundary conditions. Since all our fields are smooth inside each domain, we are able to use an efficient pseudospectral method to solve the elliptic equations associated with the conformal thin sandwich approach. Currently, we have implemented corotating configurations with arbitrary mass ratios, but an extension to arbitrary spins is possible. The main purpose of this paper is to introduce our new method and to test our code for several different configurations.

PACS numbers: 04.25.dk, 04.30.Db, 97.60.Jd, 97.80.Fk

(Some figures in this article are in colour only in the electronic version)

1. Introduction

Currently, several gravitational wave detectors such as LIGO [1, 2], Virgo [3, 4] or GEO [5] are already operating, while several others are in the planning or construction phase [6]. One of the most promising sources for these detectors is the inspirals and mergers of binary neutron stars (NS). In order to make predictions about the final phase of such inspirals and mergers, fully nonlinear numerical simulations of the Einstein equations are required. To start such simulations initial data are needed. The emission of gravitational waves tends to circularize the orbits [7, 8]. Thus, during the inspiral, we expect the two NSs to be in quasicircular orbits around each other with a radius which shrinks on a timescale much larger than the orbital timescale. This means that the initial data should have an approximate helical Killing vector \( \xi^\mu \). In addition, one would like to have the initial data in coordinates such that this approximate symmetry is manifest, i.e. the time evolution vector should lie along \( \xi^\mu \), so that...
the time derivatives of the evolved quantities are minimized. In order to achieve these goals, we use the Wilson–Mathews approach [9, 10], which is closely related to the conformal thin sandwich formalism [11]. The Wilson–Mathews approach has already been successfully used by several groups. Among them are results for corotating [12–15] and irrotational [16–20] NS binaries with equal masses. One group has also produced results for unequal mass systems [21, 22].

In this paper, we present a new numerical method to construct initial data for binary NSs in corotating configurations for arbitrary mass ratios. The main focus is on the numerical method rather than on new physics. We describe an efficient implementation of this method with the SGRID code [23], discuss code tests and compare with previous results.

Throughout we will use units where $G = c = 1$. Later when we present numerical results we will use fully dimensionless units by setting $\kappa$ in the polytropic equation of state to $\kappa = G = c = 1$. Latin indices such as $i$ run from 1 to 3, while Greek indices such as $\mu$ run from 0 to 3. The paper is organized as follows. In section 2, we describe the general relativistic equations that govern binary NSs described by perfect fluids. Section 3 describes our particular numerical implementation of these equations, followed by results for some particular configurations in section 4. We conclude with a discussion of our method in section 5.

2. Binary neutron stars in general relativity

In this section, we describe the equations governing binary NSs in quasicircular orbits.

2.1. Arnowitt–Deser–Misner (ADM) decomposition of Einstein’s equations

We use the Arnowitt–Deser–Misner (ADM) decomposition of Einstein’s equations (see, e.g., [24]) and write the line element,

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

in terms of the lapse $\alpha$, shift $\beta^i$ and the 3-metric $\gamma_{ij}$. The extrinsic curvature is defined by

$$ K_{ij} = -\frac{1}{2\alpha}(\partial_t \gamma_{ij} - \xi^k \gamma_{kj}). $$

With these definitions, Einstein’s equations split into the evolution equations:

$$ \partial_t \gamma_{ij} = -2\alpha K_{ij} + \xi^k \gamma_{kj}, $$

$$ \partial_t K_{ij} = \alpha(R_{ij} - 2K_{ij}K^k_k + K^j_iK^k_k) - D_j \alpha + \xi^k K_{kj} - 8\pi \xi S_{ij} + 4\pi \gamma_{ij}(S - \rho), $$

and the Hamiltonian and momentum constraint equations:

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

$$ D_j (K^{ij} - \gamma^{ij}K) = 8\pi j^i. $$

Here $R_{ij}$ and $R$ are the Ricci tensor and scalar computed from $\gamma_{ij}$, $D_i$ is the derivative operator compatible with $\gamma_{ij}$ and all indices here are raised and lowered with the 3-metric $\gamma_{ij}$. The source terms $\rho$, $j^i$, $S_{ij}$ and $S = \gamma^{ij} S_{ij}$ are projections of the stress–energy tensor $T_{\mu\nu}$, given by

$$ \rho = T_{\mu\nu} n^\mu n^\nu, $$

$$ j^i = -T_{\mu\nu} n^\mu \gamma^{\nu i}, $$

$$ S^{ij} = T_{\mu\nu} \gamma^{\mu i} \gamma^{\nu j}, $$

and correspond to the energy density, flux and stress tensor. The vector $n^\mu$ appearing here is the 4-vector normal to a $t = \text{const}$ slice.
2.2. Decomposition of 3-metric and extrinsic curvature

As in [9, 10], the 3-metric $\gamma_{ij}$ is decomposed into a conformal factor $\psi$ and a conformal metric $\bar{\gamma}_{ij}$ such that

$$\gamma_{ij} = \psi^4 \bar{\gamma}_{ij}. \quad (6)$$

The extrinsic curvature is split into its trace $K$ and its tracefree part $A_{ij}$ by writing it as

$$K_{ij} = A_{ij} + \frac{1}{3} \psi \gamma_{ij} K. \quad (7)$$

2.3. Quasiequilibrium assumptions

We now make some additional simplifying assumptions. First we assume that our binary is in an approximately circular orbit and that the stars are corotating. This implies the existence of an approximate helical Killing vector $\xi^\mu$. In a coordinate system where this helical symmetry is manifest and the time evolution vector lies along $\xi^\mu$, all time derivatives should approximately be zero. Here we only assume that the time derivative $\partial_t \bar{\gamma}_{ij}$ of the conformal metric and the time derivative $\partial_t K$ of the trace of the extrinsic curvature vanish. The former allows us to express the extrinsic curvature in terms of the shift and results in

$$A_{ij} = \frac{1}{2 \psi^4 \alpha} (L \beta)^{ij}, \quad (8)$$

where

$$(L \beta)^{ij} = D^i \beta^j + D^j \beta^i - \frac{2}{3} D_k \beta^k,$$ 

and $D_k$ is the derivative operator compatible with $\bar{\gamma}_{ij}$. The assumption $\partial_t K = 0$ together with the evolution equation of $K$ (derived from equation (3)) implies

$$\psi^{-5} [D_j D^k (\alpha \psi) - \alpha D_j D^k \psi] = \alpha (R + K)^2 + \beta^i D_i K + 4 \pi \alpha (S - 3 \rho). \quad (10)$$

2.4. Further simplifications and boundary conditions

Next we also choose a maximal slice and thus $K = 0$, and assume that the conformal 3-metric is flat and given by [9, 10]

$$\bar{\gamma}_{ij} = \delta_{ij}. \quad (11)$$

This latter assumption merely simplifies our equations and could, in principle, be improved by, e.g., choosing a post-Newtonian expression for $\bar{\gamma}_{ij}$ as in [25, 26]. Using equation (11) the Hamiltonian and momentum constraints in equations (4) and (10) simplify, and we obtain

$$\bar{D}^2 \psi = - \frac{\psi^5}{32 \alpha^2} (\bar{L} B)^{ij} \bar{B}_{ij} - 2 \pi \psi^5 \rho,$$

$$\bar{D}_j (\bar{L} B)^{ij} = (\bar{L} B)^{ij} \bar{D}_j \ln (\alpha \psi^{-6}) + 16 \pi \alpha \psi^4 j^i,$$

$$\bar{D}^2 (\alpha \psi) = \alpha \psi \left[ \frac{7 \psi^4}{32 \alpha^2} (\bar{L} B)^{ij} \bar{B}_{ij} + 2 \pi \psi^4 (\rho + 2 \delta) \right], \quad (12)$$

where $(\bar{L} B)^{ij} = \bar{D}^i B^j + \bar{D}^j B^i - \frac{2}{7} \bar{D}^k B^k \delta_i^j$, $\bar{D}_j = \partial_i$, and

$$B^i = \beta^i + \omega \epsilon^{ijkl} (x^l - x^l_{CM}). \quad (13)$$

Here $x^l_{CM}$ denotes the center-of-mass position, and $\omega$ is the orbital angular velocity, which we have chosen to lie along the $z$-direction. The elliptic equations (12) have to be solved subject to the boundary conditions

$$\lim_{r \to \infty} \psi = 1, \quad \lim_{r \to \infty} B^i = 0, \quad \lim_{r \to \infty} \alpha \psi = 1, \quad (14)$$

at spatial infinity.
2.5. Matter equations

We assume that the matter in both stars is a perfect fluid with a stress–energy tensor:

\[
T^{\mu\nu} = \rho_0 (1 + \epsilon) + P u^\mu u^\nu + P g^{\mu\nu}.
\] (15)

Here \(\rho_0\) is the mass density (which is proportional to the number density of baryons), \(P\) is the pressure, \(\epsilon\) is the internal energy density divided by \(\rho_0\), \(u^\mu\) is the 4-velocity of the fluid, and \(g^{\mu\nu}\) is the spacetime metric. The matter variables in equation (5) are then

\[
\rho = \alpha^2 \rho_0 (1 + \epsilon) + P u^0 u^0 - P,
\]
\(j^i = \alpha \rho_0 (1 + \epsilon) + P u^0 u^0 (u^i / u^0 + \beta^i),\)
\(S^{ij} = \rho_0 (1 + \epsilon) + P u^0 u^0 (u^i / u^0 + \beta^i) (u^j / u^0 + \beta^j) + P \gamma^{ij}.\) (16)

The fact that \(\nabla_\nu T^{\mu\nu} = 0\) yields the relativistic Euler equation:

\[
[\rho_0 (1 + \epsilon) + P] u^\nu \nabla_\nu u^\mu = -(g^{\mu\nu} + u^\mu u^\nu) \nabla_\nu P,
\] (17)

which together with the continuity equation,

\[
\nabla_\nu (\rho_0 u^\nu) = 0,
\] (18)

governs the fluid.

For corotating stars, we can show that the continuity equation (18) is identically satisfied. Furthermore, one can show that the Euler equation leads to (see, e.g., problem 16.17 in [27])

\[
[\rho_0 (1 + \epsilon) + P] d \ln (u_\mu \xi^\mu) = -dP,
\] (19)

where \(\xi^\mu\) is the assumed helical Killing vector. With the help of the first law of thermodynamics (\(d[\rho_0 (1 + \epsilon)] = [\rho_0 (1 + \epsilon) + P] d\rho_0 / \rho_0\)), this equation can be integrated to yield

\[
u_\mu \xi^\mu = \frac{C_{1/2} \rho_0}{\rho_0 (1 + \epsilon) + P},
\] (20)

where \(C_{1/2}\) are constants of integration for each star. We will later choose them such that the rest mass of each star has a prescribed value. In corotating coordinates and taking into account our conformally flat 3-metric, \(u_\mu \xi^\mu\) can be written as

\[
u_\mu \xi^\mu = -1 / u^0 = -[\alpha^2 \psi^4 \delta_{ij} \beta^i \beta^j]^{1/2}.\) (21)

In order to simplify the problem we assume a polytropic equation of state:

\[
P = k \rho_0^{1+1/n}.
\] (22)

It is then convenient to introduce the dimensionless ratio

\[
q = P / \rho_0,
\] (23)

which we use to write

\[
\rho_0 = k^{-n} q^n, \quad P = k^{-n} q^{n+1}, \quad \epsilon = n q.\] (24)

3. Numerical method

In order to construct binary NS initial data we have to solve the five elliptic equations in equation (12), with the matter terms given by equations (16), (21) and (24). In addition, our data also have to satisfy equation (20), which can be expressed as

\[
q = \frac{1}{n+1} \left( \frac{C_{1/2}}{u_\mu \xi^\mu} - 1 \right).
\] (25)
for each star. We will solve the whole set of equations by iterating over the following steps: (i) we first come up with an initial guess for \( q \) in each star, in practice we simply choose Tolman–Oppenheimer–Volkoff (TOV) solutions (see, e.g., chapter 23 in [24]) for each. (ii) Next, we solve the five coupled elliptic equations (12) for this given \( q \). (iii) Then we use equation (25) to update \( q \) in each star. The constants \( C_{1/2} \) in general, have different values for each star. We adjust the value for each star such that it has a prescribed rest mass. After updating \( q \) we go back to step (ii) and iterate until all equations are satisfied up to a given tolerance.

### 3.1. Coordinates adapted to star surfaces

Note that the matter is smooth inside the stars. However, at the surface (at \( q = 0 \)), \( \rho, P \) and \( \epsilon \) are not differentiable. This means that if we want to take advantage of a spectral method, the star surfaces should be domain boundaries. A difficulty with our iterative approach, however, is that each time we update \( q \) the matter distributions change, so that the stars change shape or even move. Hence the domain boundaries have to be changed as well. In order to address this problem, we introduce several domains each with its own coordinates. These coordinates depend on two freely specifiable functions which will allow us to vary the location of the domain boundaries, so that we can always adapt our domains to the current star surfaces in each iteration. As in the initial data approaches in [28, 29], our aim was to introduce as few domains as possible.

The coordinates we will use are very similar to those introduced by Ansorg [30]. We place both stars on the \( x \)-axis and write down the necessary coordinate transformations in two steps. First we express the standard Cartesian coordinates as

\[
\begin{align*}
x &= b \left[ \frac{1}{(X^2 + R^2)^2} + 1 \right] (X^2 - R^2), \\
y &= b \left[ \frac{1}{(X^2 + R^2)^2} - 1 \right] XR \cos \phi, \\
z &= b \left[ \frac{1}{(X^2 + R^2)^2} - 1 \right] XR \sin \phi,
\end{align*}
\]

(26)

where \( b \) is a parameter related to the distance between the stars, and \( X, R \) are functions of the new coordinates \( (A, B, \phi) \) we will use in each domain. Note that spatial infinity is located at the point where \( X = R = 0 \) and that in order to cover all \((x, y, z)\) it is sufficient the restrict \( X, R, \phi \) to the ranges \( 0 \leq X \leq 1 \), \( 0 \leq R \leq \sqrt{1 - X^2} \) and \( 0 \leq \phi \leq 2\pi \). In order to complete the coordinate transformation between the Cartesian \((x, y, z)\) and the new coordinates \((A, B, \phi)\), we now write down \( X \) and \( R \) as functions of \((A, B, \phi)\). Inside star1 we use

\[
\begin{align*}
X &= (1 - A)\Re[C_+(B, \phi)] - B\Im[C_+(1, \phi)] + B \cos([1 - A] \arg[C_+(1, \phi)]) + (1 - B)A, \\
R &= (1 - A)\Im[C_+(B, \phi)] - B\Re[C_+(1, \phi)] + B \sin([1 - A] \arg[C_+(1, \phi)]),
\end{align*}
\]

(27)

where the strictly positive function \( \sigma_+(B, \phi) \) in

\[
\sigma_+(B, \phi) = \sqrt{\frac{\tanh \left( \frac{\sigma_+(B, \phi) + i\pi B}{2} \right)}{4}}
\]

determines the shape of the star surface. The surface is always located at \( A = 0 \) but depending on the choice for \( \sigma_+(B, \phi) \) it will be at different \((x, y, z)\), e.g. for \( \sigma_+(B, \phi) = \text{const} \) we will
get a spherical surface in \((x, y, z)\). Note that this star is located around \(x = b \). Inside star 2 we use a similar transformation given by

\[
X = (1 - A)[\Re[C_-(B, \phi)] - B\Re[C_-(1, \phi)]] + B \cos \left(\frac{\pi}{4}A + [1 - A] \arg[C_-(1, \phi)]\right),
\]

\[
R = (1 - A)[\Re[C_+(B, \phi)] - B\Re[C_+(1, \phi)]] + B \sin \left(\frac{\pi}{4}A + [1 - A] \arg[C_+(1, \phi)]\right) + (1 - B)A,
\]

where the strictly negative function \(\sigma_-(B, \phi)\) in

\[
C_-(B, \phi) = \sqrt{\frac{\sigma_-(B, \phi) + i\pi B}{4}}
\]
determines where the star surface \((A = 0)\) is located in \((x, y, z)\) coordinates. Star 2 is located around \(x = -b \). Note that the \(A, B, \phi\) coordinates are different inside each star, but in order to cover each star their ranges are \(0 \leq A \leq 1, 0 \leq B \leq 1\) and \(0 \leq \phi \leq 2\pi\) in each case.

The outside of both stars is covered by two additional domains. The first one covers the region outside star 1 for all positive \(x\), while the second one covers the region outside star 2 for all negative \(x\). Both coordinate transformations can be written as

\[
X = (1 - A)[\Re[C_+(B, \phi)] - B\Re[C_+(1, \phi)]] + B \cos \left(\frac{\pi}{4}A + [1 - A] \arg[C_+(1, \phi)]\right),
\]

\[
R = (1 - A)[\Re[C_-(B, \phi)] - B\Re[C_-(1, \phi)]] + B \sin \left(\frac{\pi}{4}A + [1 - A] \arg[C_-(1, \phi)]\right),
\]

where we use \(C_+(B, \phi)\) in the former and \(C_-(B, \phi)\) in the latter. In each case, the star surface is at \(A = 0\) and spatial infinity is at \((A, B) = (1, 0)\). Figure 1 shows the coordinate lines in the \(z = 0\) plane.

3.2. Spectral method

In order to solve the elliptic equations (12) we use the SGRID code [23] which employs pseudospectral methods to accurately compute spatial derivatives. We use Chebyshev expansions in the \(A\)- and \(B\)-directions and Fourier expansions in the \(\phi\)-direction. As collocation points we choose

\[
A_j = \frac{1}{2} \left[ 1 - \cos \left(\frac{\pi j}{n_A - 1}\right) \right], \quad B_j = \frac{1}{2} \left[ 1 - \cos \left(\frac{\pi j}{n_B - 1}\right) \right].
\]
\[ \phi_k = \frac{2\pi k}{n_{\phi}}, \]  

(32)

where \( l, j, k \) are integers obeying

\[ 0 \leq l < n_A, \quad 0 \leq j < n_B, \quad 0 \leq k < n_{\phi}. \]  

(33)

The number \( n_A, n_B \) and \( n_{\phi} \) of collocation points in each direction is chosen to be equal in all four domains, to ensure that the grid points on the boundaries of two adjacent domains will be at the same \((x, y, z)\) location. As in [23] we will solve equation (12) as written down in the Cartesian form and compute derivatives like \( \partial_x \psi \) using the chain rule:

\[ \partial_x \psi = \frac{\partial A}{\partial x} \partial_A \psi + \frac{\partial B}{\partial x} \partial_B \psi + \frac{\partial \phi}{\partial x} \partial_{\phi} \psi. \]  

(34)

Note that all points with \( B = 0 \) or \( B = 1 \) lie along the \( x \)-axis, with \( x \) independent of \( \phi \). Hence along this axis we have the standard coordinate singularity of polar coordinates. Furthermore, all points with \( A = 1 \) in the interior of each star correspond to just one point on the \( x \)-axis. Thus there is an additional coordinate singularity at \( A = 1 \) inside each star. Near these points, the Jacobian matrix \( \frac{\partial (A, B, \phi)}{\partial (x, y, z)} \) blows up so strongly that we cannot accurately compute derivatives using equation (34). This problem would also occur if we did use the exact same coordinates as proposed by Ansorg [30]. In [30], the problem is not addressed since for black holes one can use excision boundary conditions and does not need the inner domains. One way around this problem would be to construct different basis functions, which would have to be chosen such that they have vanishing \( B \) and \( \phi \) derivatives at \( A = 1 \). However, then we would not be able to use fast Fourier transforms anymore to compute derivatives. For this reason, we have chosen a different approach. We simply restrict the range of \( A \) inside each star so that inside \( 0 \leq A \leq A_{\text{max}} < 1 \). The collocation points in \( A \) inside the stars are then given by

\[ A_l = A_{\text{max}} \frac{1 - \cos \left( \frac{l\pi}{n_A - 1} \right)}{2}. \]  

(35)

We typically choose \( A_{\text{max}} = 0.85 \). In this way, we completely avoid the singularities at \( A = 1 \). Of course then, our inner domains do not cover the entire star interiors any more. Instead they leave out a small hole around \( A = 1 \). We simply cover this hole by placing two additional cubical domains inside each star. Each cube is chosen such that it completely covers the hole (described by \( A_{\text{max}} < A \leq 1 \)). In each cube, we use standard Cartesian coordinates so that it overlaps with part of the inner domain covered by the \( A, B, \phi \) coordinates. The collocation points inside each cube are then

\[ x_l = \frac{x_{\text{min}} - x_{\text{max}}}{2} \cos \left( \frac{l\pi}{n_c - 1} \right) + \frac{x_{\text{min}} + x_{\text{max}}}{2}, \]

\[ y_j = \frac{y_{\text{min}} - y_{\text{max}}}{2} \cos \left( \frac{j\pi}{n_c - 1} \right) + \frac{y_{\text{min}} + y_{\text{max}}}{2}, \]

\[ z_k = \frac{z_{\text{min}} - z_{\text{max}}}{2} \cos \left( \frac{k\pi}{n_c - 1} \right) + \frac{z_{\text{min}} + z_{\text{max}}}{2}, \]  

(36)

where the minima and maxima \( x_{\text{min}}, x_{\text{max}}, y_{\text{min}}, y_{\text{max}}, z_{\text{min}}, z_{\text{max}} \) are chosen such that the cube covers the hole, and where \( n_c = 6 \) is typically sufficient, since the cubes are very small. Figure 2 shows how a cube is fitted into star 1.

In order to numerically solve the elliptic equations in equation (12), we arrange the values of the fields \( \psi, B^i \) and \( \alpha \psi \) at each grid point in a vector \( w \). Since we solve for five fields,
In order to avoid the coordinate singularity at $A = 1$ inside the stars, we have restricted the range of $A$ to $0 \leq A \leq 0.85$. The remaining space is filled with a small cube of length $0.016b$. This figure is a blowup of the inside of NS1 in figure 1 where the small cubes are not visible.

the dimension of this vector is five times the total number of grid points. In order to find $w$ that satisfies equation (12), we impose equation (12) at all interior grid points. At adjacent domain boundaries we impose the conditions that fields and their normal derivatives are equal on both sides. At infinity we impose equation (14). In the domains covered by the $A, B, \phi$ coordinates, we impose the following regularity conditions along the $x$-axis: for $k > 0$ we demand

$$
\Psi(A_i, B_j, \phi_k) = \Psi(A_i, B_j, \phi_0),
$$

while for $k = 0$ we impose

$$
\partial_x \Psi(A_i, B_j, \phi_0) + \partial_x \partial_\phi \partial_\phi \Psi(A_i, B_j, \phi_0) = 0.
$$

Here $\Psi$ stands for either $\psi, B^i$ or $\alpha \psi$, and $s = \sqrt{y^2 + z^2}$ is the distance from the $x$-axis. In order to deal with the cubes which overlap the $A, B, \phi$ covered domains inside each star, we impose the condition that the fields at points on the cube boundaries must be equal to the fields in the $A, B, \phi$ covered domain interpolated to these points. Correspondingly, we also demand that the fields at points on the $A = A_{\text{max}}$ boundaries must equal to the fields in the cube interpolated to these points. This interpolation is done with our given spectral accuracy. To compute a field at a point that is not a collocation point, we first compute the spectral expansion coefficients from the field values at the collocation points in the domain of interest. To interpolate to any point in this domain, we then compute the field value from a sum over coefficient times the basis functions evaluated at the point in question. This last interpolation step can be computationally expensive if we interpolate onto many points, while going between field values and collocation points can be done via fast Fourier transforms and is thus not very expensive. Note, however, that in our case these interpolations are not too costly, because our cubical domains have only $6^3$ grid points. Such a small number of points should always be sufficient because the cubic domains are so small that all fields are nearly constant inside the cubes, e.g. on the scale of figure 1 the cubes are not visible. This is an advantage over the domain decomposition used in [17, 31] where interpolations are needed between domains with many more grid points.

If we take all these conditions into account, we obtain $N = 4n_A n_B n_\phi + 2n_1^3$ nonlinear equations of the form

$$
f_m(w) = 0, \quad m = 1, 2, \ldots, N,
$$

Figure 2. In order to deal with the coordinate singularity at $A = 1$ inside the stars, we have restricted the range of $A$ to $0 \leq A \leq 0.85$. The remaining space is filled with a small cube of length $0.016b$. This figure is a blowup of the inside of NS1 in figure 1 where the small cubes are not visible.
for the $N$ unknowns comprising the solution vector $w$. We solve this system of equations by a Newton–Raphson scheme. This scheme requires an initial guess, for which we simply use two TOV solutions in conformally flat isotropic coordinates. In order to solve the linearized equations

$$\frac{\partial f_m(w)}{\partial w^n} x^n = -f_m(w)$$

(40)

in each Newton–Raphson step, we note that $f_m(w)$ contains spectral derivatives of $w$ in different directions, so that the $N \times N$ matrix $\frac{\partial f_m(w)}{\partial w^n}$ is sparse in the sense that it contains about 95% zeros. So in order to numerically solve the linearized equation (40) we use the sparse matrix solver UMFPACK [32–36].

3.3. Iteration scheme

As already mentioned, solving the elliptic equations in equation (12) once is not enough. After each solve, we have to adjust $q$ using equation (25). This adjustment presents the problem that the star surfaces (located at $q = 0$) change. Hence we have to also adjust $\sigma_+ (B, \phi)$ and $\sigma_- (B, \phi)$ in order to keep both star surfaces at $A = 0$. While this adjustment is not hard to implement, it incurs a high computational cost since adjusting $\sigma_+ (B, \phi)$ and $\sigma_- (B, \phi)$ amounts to changing our computational grid, and after each such adjustment we need to interpolate all relevant fields onto the new grid. In addition, the adjustment has to be carried out several times after each individual elliptic solve, since we were only able to achieve a stable iteration scheme if we pick the free constants $C_1, C_2, \omega$ and $x_{CM}$ as follows. Let us call the intersection points of the $x$-axis with the side of the star surface not facing the origin $x_{out1}$ and $x_{out2}$ (located at $(A, B) = (0, 0)$) for each star. We then determine $\omega$ and $x_{CM}$ by requiring that $x_{out1}$ and $x_{out2}$ remain constant. This task is accomplished by a root finder. In each iteration of this root finder, $C_1, C_2$ are adjusted such that the rest mass of each star remains constant (by another root finder). These root finders have to evaluate $q$ and thus need to adjust the domain shapes several times. If we do not adjust $\omega$ and $x_{CM}$ we find that the stars drift around too much for the iterations to converge. Instead of $x_{out1}$ and $x_{out2}$ it is also possible to fix the points $x_{max1}$ and $x_{max2}$ where the maximum values of $q$ occur.

In addition, we have observed that once a new $q$ is set, the elliptic solve often ‘overcorrects’ which again can result in an unstable iteration scheme. To overcome this problem, we typically do not take $\psi, B^i$ and $\alpha \psi$ coming from solving equation (12) as our new fields. Rather, we take the average of this solution and $\psi, B^i$ and $\alpha \psi$ from the previous iteration step as our new fields. In this way $\psi, B^i$ and $\alpha \psi$ change less from one iteration step to the next.

4. Results

All the results presented in this section were computed for $n = 1$ polytopes (see equation (22)) in units where $\kappa = 1$. In order to check that our SGRID code is working properly, we have checked the convergence of the constraints. For these tests, we have computed the constraints directly from equation (4) for different numbers of grid points. From figure 3 we see that the Hamiltonian constraint converges exponentially with the number of grid points, as expected for a spectral method. The momentum constraints as well as $\partial_t K$ converge to zero in a similar fashion. We have computed initial data for the configurations listed in table 1. Each configuration is described by the rest masses $m_{0i}$ and $m_{1i}$ of the two stars given by

$$m_0 = \int_{\text{star } i} \rho_0 u_0^{\alpha} \psi^6 d^3 x, \quad i = \{1, 2\}.$$

(41)
and the separation parameter $b$, which appears in the coordinate transformations and is approximately half the separation. For each configuration we have also computed the ADM mass and angular momentum given by

$$M_{\text{ADM}} = \int \left( \rho + \frac{1}{64\pi \alpha^2} (LB)^{(I)}(LB)_{(I)} \right) \psi^5 d^3 x \quad (42)$$

and

$$J_{\text{ADM}} = \int [ (x - x_{\text{CM}}) j^y - y j^x ] \psi^{10} d^3 x. \quad (43)$$

Our iterative scheme also yields the orbital angular velocity $\omega$ and the location of the center of mass $x_{\text{CM}}$. Furthermore, we list the maximum values of $q$ in each star along the $x$-axis, together with their $x$-coordinates, and also the locations of the inner and outer edges of each star. We also show the distance $d_{12}$ between the stars and star diameters $d_{1/2}$ defined by $d_{12} = |x_{\text{max1}} - x_{\text{max2}}|$ and $d_{1/2} = |x_{\text{out1/2}} - x_{\text{out1/2}}|$. The equal mass configuration in table 1 is very close to configurations already computed by Baumgarte et al [13] and also by Gourgoulhon et al [17] and agrees with them to better than 1% (see table 2 in [17]). Note that our code has no problems in handling unequal mass systems, as well as systems that are far apart. In addition, it is very memory efficient. A typical run with $n_A = n_B = 18$ points needs only about 80 MB of memory. The initial data can thus be generated on ordinary PCs. On a 2.3 GHz Linux PC it takes about 30 h to push the Hamiltonian constraint down to $10^{-4}$ if we use $n_A = n_B = 18$ points. The main reason for the low memory footprint is that our spectral code needs only very few grid points to achieve the quoted accuracies. Figure 4 shows $q$ in the $xy$-plane for a binary with rest masses $m_{01} = 0.14$ and $m_{02} = 0.06$. We can see how the domain boundaries are adapted such that $q$ is non-zero only in the inner domains. Note that $q$ is the rest mass density for $\kappa = n = 1$. In figures 5 and 6, we show the conformal factor $\psi$ and the largest shift component $B_x$ for the same configuration. Note that unlike $q$ both are smooth ($C^1$) across the domain boundaries.
Table 1. Properties of initial data for different parameters $m_{01}, m_{02}$ and $b$. The numbers are first given in units of $G = c = \kappa = 1$. The total ADM mass $M_{\text{ADM}}$ is also given in solar masses where $\kappa_0 = 5 \times 10^8$ m$^2$ and $n = 1$ (note that $\kappa^{-n/2}G M_{\text{ADM}}/c^2$ is dimensionless). After that we also list some quantities in geometric units ($G = c = 1$) in terms of the total ADM mass.

| Parameter | $m_{01}$ | $m_{02}$ | $b$ | $M_{\text{ADM}}$ | $J_{\text{ADM}}$ | $\omega$ | $d_{12}$ | $d_1$ | $d_2$ | $x_{\text{CM}}$ | $q_{\text{max}1}$ | $q_{\text{max}2}$ | $x_{\text{in}1}$ | $x_{\text{max}1}$ | $x_{\text{out}1}$ | $x_{\text{in}2}$ | $x_{\text{max}2}$ | $x_{\text{out}2}$ | $\frac{M_{\text{ADM}}}{M_\odot}$ | $(\frac{\omega}{c})^{1/2}$ |
|-----------|----------|----------|-----|-------------------|-------------------|---------|---------|------|------|-------------|----------------|----------------|--------------|-------------|---------------|-------------|----------------|----------------|----------------|----------------|----------------|-----------------|-----------------|
|           | 0.059    | 0.059    | 1.8412 | 0.1153            | 0.0231           | 0.038   | 4.224   | 2.276| 2.276| 0           | 0.0285         | 0.0285         | +0.975       | +2.112       | +3.251        | -0.975       | -2.112       | -3.251        | 1.7524       | 2.8578         |
| $m_{01}$  | 0.140    | 0.060    | 1.8400 | 0.1887            | 0.0423           | 0.048   | 4.174   | 1.711| 2.398| 0.74        | 0.106          | 0.0282         | +1.174       | +2.029       | +2.885        | -0.927       | -2.145       | -3.325        | 3.427         | 2.849          |
| $m_{02}$  | 0.140    | 0.100    | 10.00  | 0.2263            | 0.122            | 0.0052  | 20.09   | 1.70  | 1.96  | 1.6          | 0.108          | 0.0588         | +9.19        | +10.04       | +10.89        | -9.07        | -10.05       | -11.03        | 2.382         | 1.489          |
| $b$       | 0.150    | 0.050    | 5.00   | 0.1881            | 0.0527           | 0.013   | 10.20   | 1.63  | 2.25  | 2.4          | 0.127          | 0.0235         | +4.25        | +5.07        | +5.88         | -4.00        | -5.13        | -6.25         | 2.849         | 2.849          |

Figure 4. $q$ in the $xy$-plane for a binary with rest masses $m_{01} = 0.14, m_{02} = 0.06$ and $b = 1.84$. 
In order to further verify our code, we have performed a comparison with previous results from Taniguchi et al. [21]. In figures 7 and 8, we show how $M_{\text{ADM}}$ and $J_{\text{ADM}}$ vary as a function of $\omega$ for a binary with rest masses $m_{01} = 0.1461$ and $m_{02} = 0.1299$. As we can see our results (squares) approach the expected post-Newtonian results (taken from [25, 37–39]) for point particles (dotted line) for small $\omega$. At intermediate $\omega$, our results differ from point particle results and instead agree with previous results obtained by Taniguchi et al [21]. Note that, while the agreement in $M_{\text{ADM}}$ does not look as good as for $J_{\text{ADM}}$, the values for $M_{\text{ADM}}$ from both methods still agree to better than 0.05%. With our current code we can construct initial data only up to $\omega \sim 0.07$. Beyond that point, already the first iteration of our elliptic solver fails. We suspect that our initial guess of simply using two spherical TOV stars with $B^i = 0$ is not good enough for close configurations, and that the solver would succeed if we provided a guess that is closer to the true solution. Note that while Taniguchi et al [21] can extend their sequence to higher $\omega$ they also did not record a turning point in either curve for this configuration.
5. Discussion

The purpose of this paper is to introduce a new numerical method for the computation of binary NS initial data with the SGRID code [23]. The method uses six domains with different coordinate systems in each domain. The coordinates in four of these domains are closely related to those suggested in [30]. We have, however, added two extra domains with Cartesian coordinates to remove coordinate singularities. All our fields are $C^\infty$ inside each domain. This allows us to use an efficient pseudospectral collocation method to solve the elliptic equations (12) associated with the initial data construction. Note that we directly solve the five equations (12), i.e. we do not split the shift in the momentum constraint into a vector and a gradient of a scalar, which would introduce an additional elliptic equation. Thus we have one less equation to solve than that in the original Wilson–Mathews approach [9, 10]. Since two
of our domains extend to spatial infinity we are able to easily impose the boundary conditions in equation (14) without the need of any approximations such as Robin boundary conditions. At present we have only considered corotating configurations. The numerical method, however, could be easily extended to configurations with arbitrary spins, e.g. by following the approach in [20]. This approach involves the addition of one more elliptic equation for a velocity potential. The boundary conditions for this extra equation need to be imposed at the star surface, which is somewhat involved if one uses cubical domains as in [20]. In our new method, however, such boundary conditions can easily be imposed, since each star surface is a domain boundary.

It is well known that the star surfaces of close configurations can develop cusps due to tidal forces [17, 19, 20]. We have not investigated this issue with our new method yet, because our elliptic solver currently fails already during the first step for close configurations. We suspect that we need an initial guess that is better than two spherical TOV stars with vanishing shift $B'$. We would like to point out, however, that our method should not have any additional problems with such cusps if they occur only along the $x$-axis (the line connecting the two stars). The reason is that $\sigma_{\pm}(B, \phi)$, which appear in the coordinate transformations and describe the star surfaces, can easily be chosen such that the domain boundaries have arbitrary cusps on the $x$-axis. Note that such cusp producing $\sigma_{\pm}(B, \phi)$ are themselves perfectly smooth in $A, B, \phi$ coordinates, so that we do not expect to loose spectral accuracy.

Acknowledgments

It is a pleasure to thank Pedro Marronetti for useful discussions about NS initial data. This work was supported by NSF grant PHY-0652874.

References

[1] Abbott B et al 2007 LIGO: The laser interferometer gravitational-wave observatory arXiv:0711.3041 [gr-qc]
[2] http://www.ligo.caltech.edu/
[3] Acernese F et al 2008 The Virgo 3 km interferometer for gravitational wave detection J. Opt. A: Pure Appl. Opt. 10 064009
[4] VIRGO—http://www.virgo.infn.it/
[5] http://geo600.aei.mpg.de/
[6] Schutz B 1999 Gravitational wave astronomy Class. Quantum Grav. 16 A131–56
[7] Peters P C and Mathews J 1963 Gravitational radiation from point masses in a Keplerian orbit Phys. Rev. 131 415–439
[8] Peters P C 1964 Gravitational radiation and the motion of two point masses Phys. Rev. 136 B1224–32
[9] Wilson J R and Mathews G J 1995 Instabilities in close neutron star binaries Phys. Rev. Lett. 75 4161
[10] Wilson J R, Mathews G J and Marronetti P 1996 Relativistic numerical method for close neutron star binaries Phys. Rev. D 54 1317–31
[11] York J W 1999 Conformal ‘thin-sandwich’ data for the initial-value problem of general relativity Phys. Rev. Lett. 82 1350–3
[12] Baumgarte T W, Cook G B, Scheel M A, Shapiro S L and Teukolsky S A 1997 Binary neutron stars in general relativity: quasi-equilibrium models Phys. Rev. Lett. 79 1182–5
[13] Baumgarte T W, Cook G B, Scheel M A, Shapiro S L and Teukolsky S A 1998 General relativistic models of binary neutron stars in quasiequilibrium Phys. Rev. D 57 299–7311
[14] Mathews G J, Marronetti P and Wilson J R 1998 Relativistic hydrodynamics in close binary systems: Analysis of neutron-star collapse Phys. Rev. D 58 043003
[15] Marronetti P, Mathews G J and Wilson J R 1998 Binary neutron star systems: from the Newtonian regime to the last stable orbit Phys. Rev. D 58 107503
[16] Bonazzola S, Gourgoulhon E and Marck J-A 1999 Numerical models of irrotational binary neutron stars in general relativity Phys. Rev. Lett. 82 892–5
[17] Gourgoulhon E, Grandeclent P, Taniguchi K, Marck J-A and Bonazzola S 2001 Quasiequilibrium sequences of synchronized and irrotational binary neutron stars in general relativity: I. Method and tests Phys. Rev. D 63 064029
[18] Marronetti P, Mathews G J and Wilson J R 1999 Irrotational binary neutron stars in quasi-equilibrium Phys. Rev. D 60 087301
[19] Uryu K and Eriguchi Y 2000 A new numerical method for constructing quasi-equilibrium sequences of irrotational binary neutron stars in general relativity Phys. Rev. D 61 124023
[20] Marronetti P and Shapiro S L 2003 Relativistic models for binary neutron stars with arbitrary spins Phys. Rev. D 68 104024
[21] Taniguchi K and Gourgoulhon E 2002 Quasiequilibrium sequences of synchronized and irrotational binary neutron stars in general relativity. III. Identical and different mass stars with gamma = 2 Phys. Rev. D 66 104019
[22] Taniguchi K and Gourgoulhon E 2003 Various features of quasiequilibrium sequences of binary neutron stars in general relativity Phys. Rev. D 68 124025
[23] Tichy W 2006 Black hole evolution with the bssn system by pseudo-spectral methods Phys. Rev. D 74 084005
[24] Misner C W, Thorne K S and Wheeler J A 1973 Gravitation (San Francisco: Freeman)
[25] Tichy W, Brügmann B, Campanelli M and Diener P 2003 Binary black hole initial data for numerical general relativity based on post-Newtonian data Phys. Rev. D 67 064008 (arXiv:gr-qc/0207011)
[26] Kelly B J, Tichy W, Campanelli M and Whiting B F 2007 Black hole puncture initial data with realistic gravitational wave content Phys. Rev. D 76 024008
[27] Lightman A P, Press W H, Price R H and Teukolsky S A 1975 Problem Book in Relativity and Gravitation (Princeton, NJ: Princeton University Press)
[28] Ansorg M, Brügmann B and Tichy W 2004 A single-domain spectral method for black hole puncture data Phys. Rev. D 70 064011
[29] Ansorg M 2005 A double-domain spectral method for black hole excision data Phys. Rev. D 72 024018
[30] Ansorg M 2007 Multi-domain spectral method for initial data of arbitrary binaries in general relativity Class. Quantum Grav. 24 S1–14
[31] Bonazzola S, Gourgoulhon E and Marek J-A 1998 Numerical approach for high precision 3-D relativistic star models Phys. Rev. D 58 104020
[32] Davis T A and Duff I S 1997 An unsymmetric-pattern multifrontal method for sparse LU factorization SIAM J. Matrix Anal. Appl. 18 140–58
[33] Davis T A and Duff I S 1999 A combined unifrontal/multifrontal method for unsymmetric sparse matrices ACM Trans. Math. Softw. 25 1–20
[34] Davis T A 2004 Algorithm 832: Umfpack v4.3—an unsymmetric-pattern multifrontal method ACM Trans. Math. Softw. 30 196–9
[35] Davis T A 2004 A column pre-ordering strategy for the unsymmetric-pattern multifrontal method ACM Trans. Math. Softw. 30 165–95
[36] Davis T A UMFPACK a sparse linear systems solver using the Unsymmetric MultiFrontal method http://www.cise.ufl.edu/research/sparse/umfpack/
[37] Schäfer G and Wex N 1993 Phys. Lett. A 174 196
[38] Tichy W, Brügmann B and Laguna P 2003 Gauge conditions for binary black hole puncture data based on an approximate helical killing vector Phys. Rev. D 68 064008
[39] Tichy W and Brügmann B 2004 Quasi-equilibrium binary black hole sequences for puncture data derived from helical killing vector conditions Phys. Rev. D 69 024006