SF2A 2016
C. Reylé, J. Richard, L. Cambrésy, M. Deleuil, E. Pécontal, L. Tresse and I. Vauglin (eds)

NUMERICAL RELATIVITY AND SPECTRAL METHODS

P. Grandclement [1]

Abstract. The term numerical relativity denotes the various techniques that aim at solving Einstein's equations using computers. Those computations can be divided into two families: temporal evolutions on the one hand and stationary or periodic solutions on the other one. After a brief presentation of those two classes of problems, I will introduce a numerical tool designed to solve Einstein's equations: the KADATH library. It is based on the the use of spectral methods that can reach high accuracy with moderate computational resources. I will present some applications about quasicircular orbits of black holes and boson star configurations.

Keywords: Gravitation, Methods: numerical

1 Introduction

In the strong field regime, gravitation must be studied in the framework of general relativity. The geometry of the spacetime is described by the metric which gives the distance between neighboring points. It is a four-dimensional, second order, symmetric tensor and so consists of ten components. This metric depends on the energy content of the spacetime and the link between geometry and energy is given by the famous Einstein's equations. They consist of a set of ten highly coupled non-linear equations. Those equations are not easy to solve and analytic solutions are known in only very few cases. Most of the time one relies on semi-analytically methods (like the famous post-Newtonian expansion) or on computers to find solutions. This proceeding is concerned with the latter case. In particular, I will briefly introduce a class of numerical techniques known as spectral methods. Then I will present a numerical tool that enables the use of those methods: the KADATH library.

The fields of astrophysics where general relativity must be taken into account are numerous. One can think about coalescing compact binaries, especially with the first direct detection of the gravitational waves in September 2015. General relativity in also important in supernovae simulations and for studying the structure of neutron stars. There are also a lot of other applications that concerns more theoretical physics than classical astrophysics. One can mention the study of critical phenomena or the stability of ADS spacetime. I will present two applications, one about quasicircular configurations of compact binaries and one about objects that could be a viable alternative to black holes: the boson stars.

2 3+1 formalism

The 3+1 formalism is a rewriting of Einstein's equations in order to make them suitable for numerical resolution (see Gourgoulhon (2012) for a review). It is essentially a splitting of the four spacetime dimensions into space (the 3) and time (the 1). In order to do so one has to introduce a family of spatial hypersurfaces Σ_t such that the full spacetime is given by the union of all those hypersurfaces. At each point of Σ_t one introduces the normal $\mathbf{n}$ which is a timelike vector. The choice of Σ_t is not unique but is merely a choice of time coordinate. Under those assumptions, the metric reads :

$$\mathrm{d}s^2 = -\left(N^2 - N^i N_i\right)\mathrm{d}t^2 + 2N_i \mathrm{d}t\mathrm{d}x^i + \gamma_{ij}\mathrm{d}x^i\mathrm{d}x^j,$$

[1] LUTH, Observatoire de Paris, 5 place J. Janssen, 92195 Meudon

where N is the lapse, N^i the shift vector and γ_{ij} the spatial metric. All those objects are purely three dimensional objects that live on the hypersurfaces Σ_t. It implies that the Latin indices range from 1 to 3 only. However those quantities do depend on the hypersurface considered, meaning they do depend on time. So, to summarize, instead of working with four-dimensional quantities, the 3+1 formalism describes the spacetime by temporal sequences of purely spatial quantities.

The next step is then to project Einstein's equations onto the hypersurfaces and on the normal $\mathbf{n}$. Doing so, one translates the four dimensional equations into a set of equations involving only spatial quantities. This leads to the 3+1 equations of general relativity. They are given below, along with the Maxwell ones, for comparison purposes.

Type	Einstein	Maxwell
Constraints	Hamiltonian $R + K^2 - K_{ij}K^{ij} = 0$	$\nabla \cdot \vec{E} = 0$
	Momentum : $D_j K^{ij} - D^i K = 0$	$\nabla \cdot \vec{B} = 0$
Evolution	$\frac{\partial \gamma_{ij}}{\partial t} - \mathcal{L}_{\vec{N}} \gamma_{ij} = -2NK_{ij}$	$\frac{\partial \vec{E}}{\partial t} = \frac{1}{\varepsilon_0 \mu_0} \left(\vec{\nabla} \times \vec{B} \right)$
	$\frac{\partial K_{ij}}{\partial t} - \mathcal{L}_{\vec{N}} K_{ij} = -D_i D_j N + N \left(R_{ij} - 2K_{ik}K^k_j + KK_{ij} \right)$	$\frac{\partial \vec{B}}{\partial t} = -\vec{\nabla} \times \vec{E}$

D_i and R_{ij} denote, respectively the covariant derivative and Ricci tensor associated to γ_{ij}. K_{ij} is the so-called extrinsic curvature tensor and can be seen as being the velocity of the metric, in the sense that it is closely related to the first time derivative of the metric. Indeed the first evolution equation is not one of Einstein's equation but merely the kinematic definition of K_{ij}.

Solving this set of equations proceeds in two steps. First one needs to solve the *initial value problem*, meaning one needs to find the values of $\gamma_{ij}\,(t=0)$ and $K_{ij}\,(t=0)$ that fulfill the constraint equations and that describe accurately the physical situation one wants to study. Mathematically speaking it involves solving a set of four elliptic coupled equations. The second step is the *evolution problem* where one uses the evolution equations to get the values of γ_{ij} and K_{ij} at later times. Let us mention that the second order system is written as a set of two first order equations, as it can be done in Newtonian dynamics if one rewrites Newton's equation as $\partial_t x = v$ and $\partial_t v = f/m$. If the constraint equations are fulfilled at $t = 0$ and if the evolution equations are solved properly, it can be demonstrated that the constraints equations are going to be true for all times. One says that they are transported by the evolution equations. Solving the evolution equations means having the ability to maintain stability and accuracy and that requires good choice of coordinates, choice that is being done via the lapse and shift.

3 Spectral methods

Spectral methods are a class of numerical techniques where the various fields are described by finite sums of known functions called the *basis functions*. An introduction to those methods can be found in Grandclément & Novak (2009).

In one dimension, consider an interval Λ and a set of orthogonal basis functions Φ_i on Λ. The spectral theory gives then a recipe to approximate any function f of Λ by its interpolant of degree N :

$$f \approx I_N f \equiv \sum_{i=0}^{N} a_i \Phi_i,$$

where the a_i are called the coefficients of f. Standard choices for the basis functions include orthogonal polynomials like Legendre of Chebyshev or trigonometrical functions. In this second case, the spectral expansion is nothing else than the usual discrete Fourier transform of f.

An important feature of the spectral expansion is the existence of the so-called *collocation points*. One can show that there exist $N + 1$ points x_i in Λ such that f and its interpolant coincide at those points : $f(x_i) = I_N f(x_i)$. It follows from that property that one has two ways of describing a function on Λ: either by its coefficients a_i or by its values at the various collocations points $f(x_i)$. There is a bijection between the two descriptions and one can go back and forth between the two without any loss of precision. Depending on the mathematical operation that one needs to perform on f it may be easier to work with one description or the

other. For instance, derivation is much easier to perform using the coefficients description for it only requires to know the derivative of the basis functions.

The main reason for using spectral methods is the very fast convergence of $I_N f$ to the real function f. One can show that, if f is $\mathcal{C}^\infty$, then the convergence is faster than any power-law of N. This is called *spectral convergence*. This is to be compared with finite difference schemes where only a power-law convergence is achieved. For less regular functions however, spectral convergence is lost. In that case a multi-domain decomposition can be used: by setting the domains such that the discontinuities lie at the interface, one can hope to recover spectral convergence (the functions being $\mathcal{C}^\infty$ by parts).

Figure 1 shows an example of spectral expansion. The blue curve corresponds to the function $u(x) = \cos^3(\pi x/2) - (x+1)^3/8$, which is not polynomial but is $\mathcal{C}^\infty$. The red curve denote the true projection of u onto the set of Chebyshev polynomials and the green one the interpolant $I_N u$. All those three functions differ. The circles show the location of the collocation points where, indeed, u and $I_N u$ coincide. The left panel is for $N = 4$ and the right one for $N = 8$. The convergence of $I_N u$ to u is clearly illustrated.

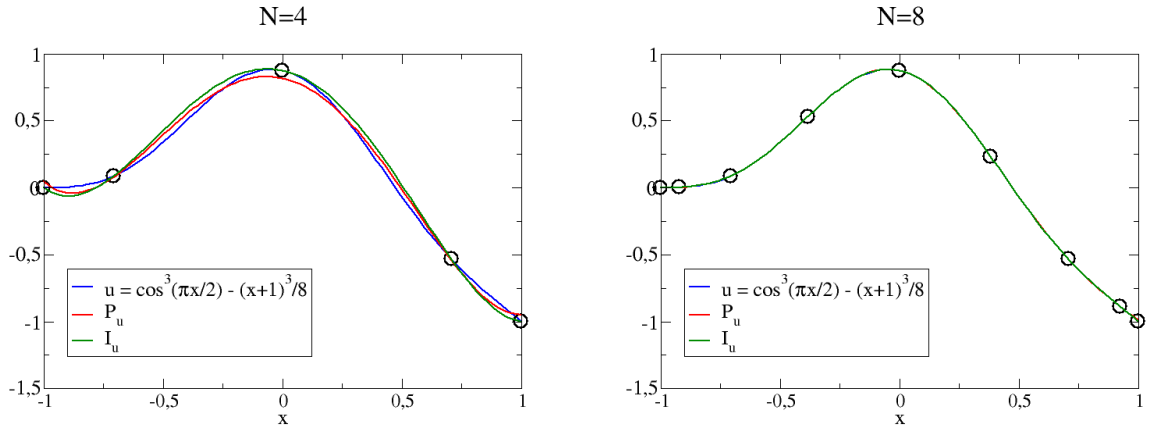

Fig. 1. Left: Function (blue curve), its true projection onto the basis functions (red curve) and its spectral representation (green curve). for $N = 4$. The circles denote the location of the collocation points.**Right:** Same thing for $N = 8$.

Figure 2 shows the maximum difference between u and its interpolant $I_N u$, as a function of N, for the same function u as in Fig. 1. The error decreases exponentially until it reaches a saturation of about 10^{-14}, coming from the fact that the code works in double precision.

In order to solve differential equations using spectral methods, one relies on a class of techniques known as the *weighted residual methods*. Consider a field equation written as $R = 0$, where R is given as a function of some unknown fields (for instance $R = \Delta f - S$ for solving a Poisson equation). The weighted residual methods provide a way of transforming this field equation into a set of discrete equations by demanding that $(R, \xi_i) = 0$. Here $(,)$ denotes the same scalar product as the one used for the spectral expansion. The ξ_i are called the test functions and various choices are possible. For instance, if one chooses as test functions the basis functions themselves, the weighted residual method is called the τ-method. Some of the discrete equations must usually be relaxed in order to enforce appropriate matching and boundary conditions.

4 The KADATH library

It is a library designed to enable the use of spectral methods in various context arising in the fields of astrophysics and theoretical physics. It is written in C++ and makes an extensive use of object programming. The library is intended to be very modular both in term of the geometry considered and the type of equations it can solve. The equations are passed to the code with a text interface inspired by LateX that should be easy to grasp for most researchers. A description of the library can be found in Grandclément (2010) and it can be downloaded freely at `http://luth.obspm.fr/~luthier/grandclement/kadath.html`.

KADATH implements various choices of geometry and coordinates (spherical, bispherical etc...) and additional cases are relatively easy to add. When the library is used to solve a system of equations, the unknowns

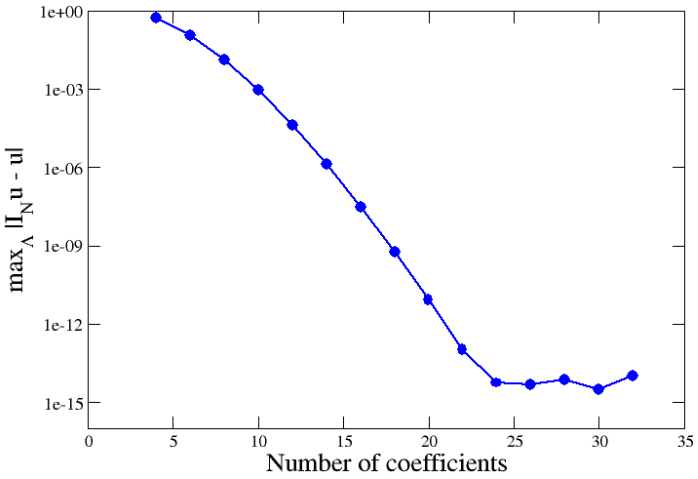

Fig. 2. Maximum difference between u and its interpolant, as a function of the number of coefficients.

are the coefficients of all the unknown fields in the whole space. Let us denote those coefficients by a vector $\vec{x}$. The equations are dealt with using the weighted residual method and they lead to a discrete system $\vec{F}(\vec{x}) = 0$.

In general, the system considered is non-linear and the solution must be sought be means of a Newton-Raphson method. One starts from an initial guess $\vec{x}_0$ and the true solution is approached by successive steps. Each iteration requires the inversion a linear approximation of the system. More precisely the approximation $\vec{x}_{i+1}$ relates to the previous one by $\vec{x}_{i+1} = \vec{x}_i - \vec{\delta x}$, where $\vec{\delta x}$ is the solution of $J\left(\vec{x^i}\right) \times \vec{\delta x} = \vec{F}(\vec{x}_i)$, where J is the Jacobian of the system (computed here at the position $\vec{x}_i$).

The Jacobian must be computed at each step of Newton-Raphson algorithm. In KADATH this is done via a numerical technique called *automatic differentiation*. Basically, each quantity (here each coefficient) is supplemented by its variation. One then talks about dual numbers. All the arithmetic is then redefined on those dual number. Let us consider the dual form of the unknown vector denoted by $\left\langle \vec{u}, \vec{\delta u} \right\rangle$. The action of $\vec{F}$ in its dual form leads to $\left\langle \vec{F}, \vec{\delta F} \right\rangle$. It is possible to show that the variation $\vec{\delta F}$ gives some information on the Jacobian. More precisely one has $\vec{\delta F} = J \times \vec{\delta u}$. So, the Jacobian itself can be obtained by taking all the possible values of $\vec{\delta u}$.

One of the main difficulty comes from the fact that the size of the Jacobian can be big. For three-dimensional problems, one can have to deal with a matrix of size $200,000 \times 200,000$. Because of this, KADATH has to be run in parallel. Thanks to the automatic differentiation, the Jacobian is obtained column by column, each computation being independent of the the others, so that this can be easily parallelized. The inversion of the Jacobian is also performed in a parallel manner using the library `scalapack`. KADATH has been successfully used with several thousands processors.

One of the main limitation of the library concerns explicit time evolutions. They are virtually non-existent and the library is only concerned with solutions having some symmetry with respect to time (either stationary or periodic solutions). The simplest way to deal with explicit time evolutions is to use a Runge-Kutta integration with respect to time, using the spectral approximation only for the spatial dimensions. This is widely used and seems to yield good results. An alternative is to use a spectral expansion also in time. For instance one can expand the fields onto Chebyshev polynomials in time and integrate the evolution equation on a given interval $[0, \Delta T]$, once the initial value of the fields are given (and possibly their first time derivative). The fields are then known on $[0, \Delta T]$ and the procedure can then be repeated. The choice of the numerical parameter ΔT is obviously important. Very preliminary tests indicate that this procedure works fine. However the resulting code is much longer than its counterpart based on Rugge-Kutta. Whether there are some cases where spectral expansion in time is really needed is still an open question.

As seen before, most of the computational time is spent when calculating the Jacobian and solving the associated linear system. There exists a class of alternative methods that could reduce this cost. They aim at finding the solution of a linear system iteratively. There are various versions of those techniques (GMRES, Bicgstab ...) but they are all based on the notion of Krylov subspace. Essentially the solution of the system is assumed to be given by successive powers of J : $\sum_n J^n \times S$. Using this property, one can show that each step of the iteration requires only to be able to compute products like $J \times f$ where f is given (by the method and the iteration...). It follows that, if the procedure converges with a number of iteration much smaller than the size of the Jacobian, those iterative techniques would be much faster than the direct inversion. Also they do not require to store the matrix J and so are usually less expensive in terms of memory usage. However the main limitation of those iterative techniques is that convergence is far from being guarantied. Most of the time it requires a carefully preconditioning of the system and q fine-tuning of the various computational parameters. Those methods are probably not general enough to be used for all the applications of KADATH. Nevertheless, for some particular cases, it may be useful to allow the user to have access to those techniques and there are plans to include them in the KADATH library. A detailed presentation of various iterative methods can be found in Saad (2003).

5 Some applications

5.1 Boson star models

Boson stars are one of the alternative to black holes, especially in the context of supermassive objects at the center of galaxies. By this one means that they can have a great mass inside a small radius, without the presence of an event horizon. Boson stars are described by a complex scalar field coupled to gravity. The structure of those objects is then given by the solution of Einstein's equations (for the gravity) coupled to the Klein-Gordon one (for the scalar field).

Boson stars are obtained when considering the following ansatz for the scalar field $\phi = \phi_0 \exp\left[i\left(\omega t - k\varphi\right)\right]$. Doing so, the quantity ϕ_0 is real and only depends on (r, θ). The metric fields are also axisymmetric and stationary. The various boson star models are labelled by the winding number k which is an integer, and the real pulsation ω. InGrandclément et al. (2014) we solved the equations using a two-dimensional setting in KADATH (i.e. the *Polar* space). The left panel of Fig. 3 shows the configuration of the field ϕ_0 in the case $k = 2$ and $\omega = 0.8$.

Numerical models can be used to deduce observational constraints on the existence of boson stars. For instance, Vincent et al. (2016) have simulated the image of an accretion disk around such an object and see if there are some differences with accretion disk around a black hole. This is what is done on the right panel of Fig. 3. In this particular case however, it was noted that the two images were very close, even if the geometry of spacetime is different. Nevertheless, the study of various physical effects around boson stars should lead to several observational tests that could be used, in the future, to rule out or confirm the existence of such objects.

5.2 Quasi-circular compact binaries

This application is concerned with the computation of binary black hole configurations. It is assumed that the two holes are on closed circular orbit. This cannot be exact: due to gravitational wave emission the orbit is rather a spiral. However, in the early stages of the binary this is probably a good approximation. From the practical point of view it greatly simplifies the problem by removing any explicit time evolution (the problem becomes three-dimensional only). An additional approximation that is used is the so-called *conformal flatness approximation* which assumes that the spatial metric is conformally flat. Not only does it simplify the equations but it also kills the gravitational waves.

Under those assumptions, the mathematical problem reduces to solving five coupled non-linear elliptic equations for five unknown fields. Non-trivial boundary conditions must be enforced on the horizon of the holes. The system is solved using the bispherical coordinates of the KADATH library (see Uryũ et al. (2012)). Figure 4 shows one particular configuration where the mass ratio of the two objects is two. The upper panel shows the value of the surface gravity on the two holes. As expected, this quantity is inversely proportional to the mass of each hole. Also, as seen on the lower panel, it is almost constant on each horizon. This is a numerical confirmation of what is know as the zeroth law of black hole thermodynamics.

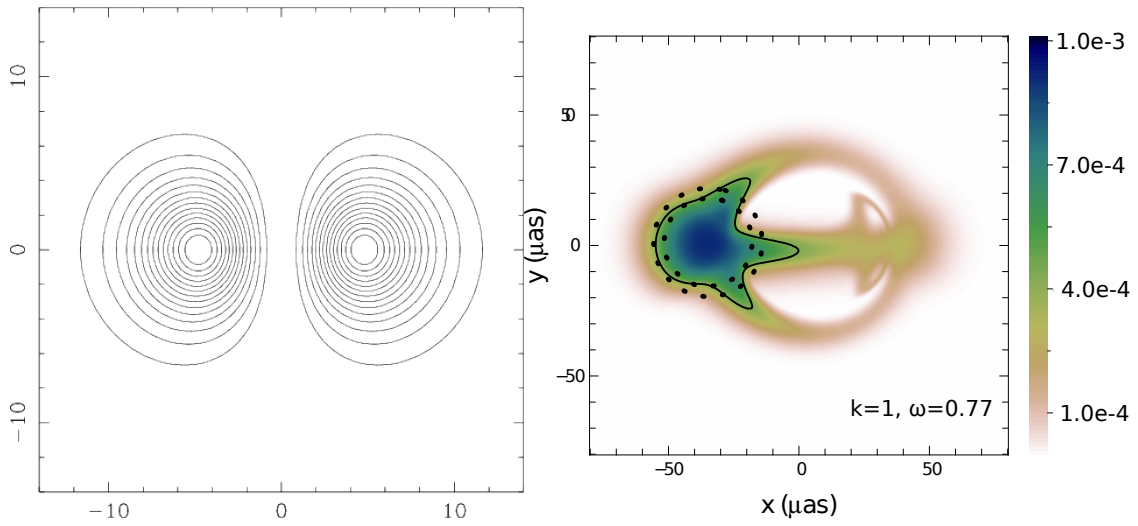

Fig. 3. Left: Isocountours of ϕ_0 in the (x, z) plane. The field is symmetric around the z-axis.**Right:** Image of an accretion disk around a $k = 4$, $\omega = 0.77$ boson star.

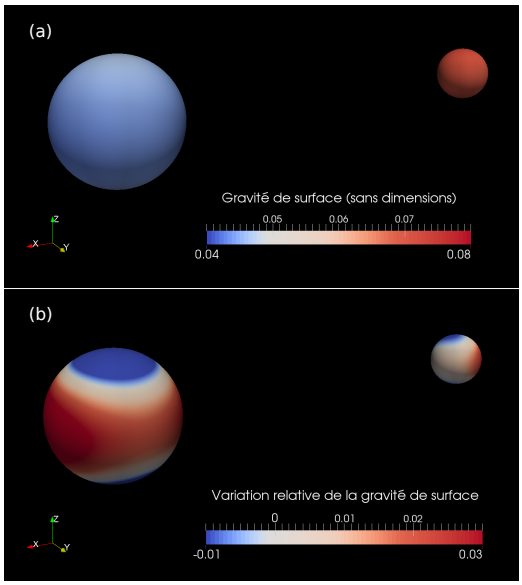

Fig. 4. Surface gravity of black holes in circular orbit. The mass ratio is two. The upper panel shows the surface gravity itself and the lower one its relative variation on each hole.

6 Conclusions

After years of struggles numerical relativity is able to produce meaningful results. Nevertheless there are still some work that needs to be done. One can mention the computation of more realistic initial data or the study of alternative models to black holes. Spectral methods are a powerful tool to do so. I have presented a library that enables the use of such methods, in a very modular manner: the KADATH library. It has already produced results in various fields, from compact objects to theoretical physics. Future developments of the library are planed, for instance by providing alternative mathematical methods to solve the discrete system resulting from the spectral approximation. An effort should also be made to make this tool easier to use, by providing better documentation, examples and tutorials.

References

Gourgoulhon, E. 2012, 3+1 Formalism in General Relativity ; Bases of Numerical Relativity (berlin: springer)
Grandclément, P. 2010, J. Comp. Phys., 229, 3334
Grandclément, P. & Novak, J. 2009, Liv. Rev. Relat., 12
Grandclément, P., Somé, C., & Gourgoulhon, E. 2014, Phys. Rev. D, 90, 024068
Saad, Y. 2003, Iterative methods for sparse linear systems (Society of Industral and Applied Mathematics)
Uryũ, K., Tsokaros, A., & Grandclément, P. 2012, Phys. Rev. D, 86, 104001
Vincent, F., Meliani, Z., Grandclḿent, P., Gourgoulhon, E., & Straub, O. 2016, Class. Quant. Grav., 33, 105015