Basis Function Method for Numerical Loop Quantum Cosmology: The Schwarzschild Black Hole Interior

Alec Yonika[1] and Gaurav Khanna[1]

[1]*Department of Physics & Center for Scientific Computing and Visualization Research, University of Massachusetts Dartmouth, North Dartmouth, MA 02747, USA*

Loop quantum cosmology is a symmetry-reduced application of loop quantum gravity that has led to the resolution of classical singularities such as the big bang, and those at the center of black holes. This can be seen through numerical simulations involving the quantum Hamiltonian constraint that is a partial *difference* equation. The equation allows one to study the evolution of sharply-peaked Gaussian wave packets that generically exhibit a quantum "bounce" or a non-singular passage through the classical singularity, thus offering complete singularity resolution. In addition, von-Neumann stability analysis of the difference equation – treated as a stencil for a numerical solution that steps through the triad variables – yields useful constraints on the model and the allowed space of states. In this paper, we develop a new method for the numerical solution of loop quantum cosmology models using a set of basis functions that offer a number of advantages over computing a solution by stepping through the triad variables. We use the Corichi and Singh model for the Schwarzschild interior as the main case study in this effort. The main advantage of this new method is computational efficiency and the ease of parallelization. In addition, we also discuss how the stability analysis appears in the context of this new approach.

I. INTRODUCTION

Classical general relativity is plagued with singularities such as those at the center of black holes and also the big bang in cosmological models. In loop quantum gravity, the spacetime continuum is replaced by a discrete quantum structure wherein geometric operators such as areas and volumes have discrete eigenvalues with a non-zero minimum [1–3]. The discrete structure of quantum spacetime only plays a significant role when the curvature approaches Planck scale; otherwise, strong agreement with general relativity is found. Nearly two decades of study has been performed in the context of singularity resolution in symmetry-reduced cosmological models with fairly robust results that replace the big bang with a big bounce [4–7]. States sharply peaked on classical trajectories when the universe is large and expanding, can be evolved backward using the quantum Hamiltonian constraint; such states evolve in a stable and non-singular way, and bounce in the deep Planck regime into a contracting branch [11].

Several of the above mentioned results were obtained through the application of computational techniques in the field of loop quantum gravity (See Refs. [8–10] and references therein). Recently, similar numerical studies were performed in the context of the Corichi and Singh (CS) model of the Schwarzschild interior [12] resulting in a clear numerical demonstration of singularity resolution [13]. The CS model improves upon previous models, offering a consistent and correct infra-red limit and independence from fiducial structures used in the quantization procedure. The quantum Hamiltonian constraint is a partial *difference* equation in two discrete triad variables making it somewhat more complicated in comparison with the previously studied cosmological models. Von-Neumann stability analysis of this equation results

in a stability condition for black holes which have a very large mass compared to the Planck mass. In addition, further analysis for such large black holes leads to a constraint on the choice of the allowed states in numerical evolution. Evolution of a sharply peaked Gaussian wave packet yields a bounce in one of the triad variables, but for the other triad variable singularity resolution arises through a simple passage through the classical singularity. In addition, states are found to be peaked at the classical trajectory for a long time before and after the classical singularity [13]. These results support a *symmetric* quantum black hole to white hole transition paradigm that has received tremendous attention in recent years [14–16].

Needless to say that the semi-classical behavior of the model is expected in the regime where the triad variables have large values in comparison to the Planck scale. This implies that the numerical triad grid required must span a very large domain, and is often limited by the finite computational resources (memory, compute time, numerical precision, etc.). In practice, our previous efforts succeeded in performing numerical simulations only on the scale of a few hundred Planck units in each triad dimension [13]. The method used therein was a straightforward approach inspired by a finite-difference stencil computation – a recursive stepping through a 2D grid built using two discrete- valued triad variables. The method was difficult to parallelize owing to its intrinsically serial structure and was also constrained by finite floating-point numerical precision. In this paper, we develop a new numerical solution method inspired by the well-known spectral collocation approach for numerical solutions of partial differential equations that utilizes a set of basis functions in one triad variable, and performs an explicit stepping in the other variable. This allows for high computational efficiency, especially for the larger sized computations and reduced numerical precision requirements. Moreover, the

basis function approach is readily parallelizable on multi- and many- core processors like modern CPUs and GPUs.

An alternative method of solution to quantum Hamiltonian constraints could also offer a different perspective on the role of the von-Neumann stability analysis in loop quantum cosmology models. After all, one may ask, are the discovered instabilities through that analysis a true feature of the model and possibly the underlying physics, or simply an artifact of some sort of how one solves the equations? The von-Neumann stability analysis is most commonly used to evaluate finite-difference stencil-based computations, i.e. a local stepping approach on a grid built using the triad variables' discrete set of values. What if one took a more global approach towards solving the equation, that doesn't involve any stepping? Does the same instability manifest itself in some other way? If so, then indeed, that would be a strong indication of the inherent nature of the instability and its relevance to a property of the model and possibly even something physical. In this paper, we study this question in some detail and show that the previously discovered instabilities are not simply artifacts of the manner in which the equations where solved, rather they are indicative of the issues within the CS model itself or something else of significance (see Ref. [13] for different possibilities).

This paper is organized as follows: In Sec. II we introduce the loop quantum CS model and briefly describe its key features. In Sec. III we present the new numerical method to solve such models, and in Sec. III A we apply this technique to a variable-separated representation of the CS model. In Sec. III B the application is broadened to the full 2D form of the CS model. A discussion of some of the technical aspects of the implementation of the new numerical technique are also presented therein. A detailed discussion of the instability exhibited by the CS model is presented in Sec. III C. We conclude with a discussion of results in Sec. IV, and the novel benefit that the numerical technique offers is elaborated on in Sec. IV A and Sec. IV B.

II. BACKGROUND

The loop quantization of the Schwarzschild interior is performed using a Kantowski-Sachs vacuum spacetime with a phase space expressed in terms of holonomies of Ashtekar-Barbero connection components b and c, and the two conjugate triad variables p_b and p_c. The spacetime metric in terms of these variables is given by

$$\mathrm{d}s^2 = -N^2\mathrm{d}t^2 + \frac{p_b^2}{|p_c|L_o^2}\mathrm{d}x^2 + |p_c|(\mathrm{d}\theta^2 + \sin^2\theta\mathrm{d}\phi^2) \ . \ (1)$$

Here L_o is a fiducial length scale in the x-direction of the spatial manifold. To relate this with the usual Schwarzschild metric variables, p_b and p_c satisfy

$$\frac{p_b^2}{p_c} = \frac{2m}{t} - 1 \quad |p_c| = t^2 \ , \qquad (2)$$

where $m = GM$, with M as the ADM mass of the black hole space-time. In the classical theory, the horizon at $t = 2m$ is identified with $p_b = 0$ and $p_c = 4m^2$ while the central singularity is where both p_b and p_c vanish. In the quantum theory, the eigenvalues of triad operators are given by

$$\hat{p}_b\,|\mu,\tau\rangle = \frac{\gamma\ell_{\mathrm{Pl}}^2}{2}\,\mu\,|\mu,\tau\rangle, \quad \hat{p}_c\,|\mu,\tau\rangle = \gamma\ell_{\mathrm{Pl}}^2\,\tau\,|\mu,\tau\rangle \quad (3)$$

where γ is the Immirzi parameter and ℓ_{Pl} is the Planck length. Loop quantization of the classical Hamiltonian constraint using the holonomies of the connection components b and c yields the following quantum difference equation [12]

$$\delta_b(\sqrt{|\tau|} + \sqrt{|\tau + 2\delta_c|})\,(\Psi_{\mu+2\delta_b,\tau+2\delta_c} - \Psi_{\mu-2\delta_b,\tau+2\delta_c})$$
$$+\tfrac{1}{2}\,(\sqrt{|\tau + \delta_c|} - \sqrt{|\tau - \delta_c|})\bigg[(\mu + 2\delta_b)\Psi_{\mu+4\delta_b,\tau}$$
$$+ (\mu - 2\delta_b)\Psi_{\mu-4\delta_b,\tau} - 2\mu(1 + 2\gamma^2\delta_b^2)\Psi_{\mu,\tau}\bigg]$$
$$+\delta_b(\sqrt{|\tau|} + \sqrt{|\tau - 2\delta_c|})\,(\Psi_{\mu-2\delta_b,\tau-2\delta_c} - \Psi_{\mu+2\delta_b,\tau-2\delta_c})$$
$$= 0 \ . \qquad (4)$$

Here, for a Schwarzschild black hole interior corresponding to mass m,

$$\delta_b = \frac{\sqrt{\Delta}}{2m}, \quad \text{and} \quad \delta_c = \frac{\sqrt{\Delta}}{L_o} \qquad (5)$$

where Δ denotes the minimum area eigenvalue in loop quantum gravity, i.e. $\Delta = 4\sqrt{3}\pi\gamma\ell_{\mathrm{Pl}}^2$.

III. BASIS FUNCTION METHOD

In this section, we demonstrate the use of our newly developed basis function method (BFM) to solve discrete quantum Hamiltonian constraints in loop quantum cosmology. As mentioned earlier, this approach is inspired by the well-known spectral collocation method use commonly for numerical solutions of partial differential equations. The main advantage of the basis function method is its high computational efficiency and ease of parallelization on modern computer hardware. We also compare the basis method based solutions with the previous approach of recursively stepping (RSM) through a grid of triad variable values.

We use the example of the CS model of the Schwarschild interior throughout. We set $\gamma\delta_b \to 0$ (a requirement for stable solutions [13]) and set $\delta_c = 2\delta_b = 1$ without loss of generality.

A. Separable Solutions

Let us begin with performing a separation-of-variables solution of the CS model under consideration. To demon-

strate the viability of the basis function method solution for this model, Ψ is taken to be of the form $\Psi_{\mu,\tau} \to A(\mu)B(\tau)$, which reduces the CS model to

$$(\mu + 2\delta_b)A(\mu + 4\delta_b) \; + \; (\mu - 2\delta_b)A(\mu - 4\delta_b) = 2\mu A(\mu)$$
$$2\delta_b\lambda(A(\mu - 2\delta_b) \; - \; A(\mu + 2\delta_b)) + 4\mu\gamma^2\delta_b^2 A(\mu) \quad (6)$$

and

$$(\sqrt{|\tau|} + \sqrt{|\tau + 2\delta_c|})B(\tau + 2\delta_c)$$
$$-(\sqrt{|\tau|} + \sqrt{|\tau - 2\delta_c|})B(\tau - 2\delta_c)$$
$$= -\lambda(\sqrt{|\tau + \delta_c|} - \sqrt{|\tau - \delta_c|})B(\tau) \quad (7)$$

where λ is the separation parameter.

The basic idea behind the basis function method is to start with a set of appropriate functions, and then solve the difference equation on a linear span of this set. The chosen basis should allow for highly variable behavior for small values of the triad variables μ and τ, but only capture very smooth behavior for much larger values. This allows for very sharp quantum fluctuations in the deep Planck regime, and yet very smooth semi-classical behavior for large values of the triad variables. One approach to build such a basis is to seek inspiration from a Fourier basis as used in spectral collocation methods, but with a non-constant wavelength, i.e. $k(x)$. Note that our requirement of smoother behavior for larger values of x, then becomes the condition that $k(x)$ be a monotonically decreasing function of x. Specifically, we let k drop rapidly (exponentially) with x. Thus our desired modified Fourier basis would take the form, $\exp(in\theta(x))$ where $\theta'(x) = k(x)$ drops exponentially with x. Given such a form for a basis, we can then expand any solution as a linear combination of these basis elements and obtain the values of the coefficients by imposing the difference equation. Thus, ultimately we solve a linear system of equations for which a wide variety of efficient numerical algorithms and solvers are readily available.

More specifically, we choose the basis functions to take the form,

$$\Phi_n(\tau) = \exp(ine^{-\frac{|\tau|}{N}\exp(\frac{2n}{N})}) \quad (8)$$

which is a slightly modified version of a basis that was proposed by one of us in Ref. [17] many years ago. Some sample basis elements are depicted in Fig. 1.

The basis function method ultimately involves solving a linear system of equations for a vector of weights ω_n, where n is an index spanning the number of basis elements. The system matrix is represented as an "interpolation matrix", with each entry being the sequence in Eqn. 7, and with each B replaced with an appropriate representation of Eqn. 8. Each row in this matrix corresponds to an increasing value of τ in the domain of the
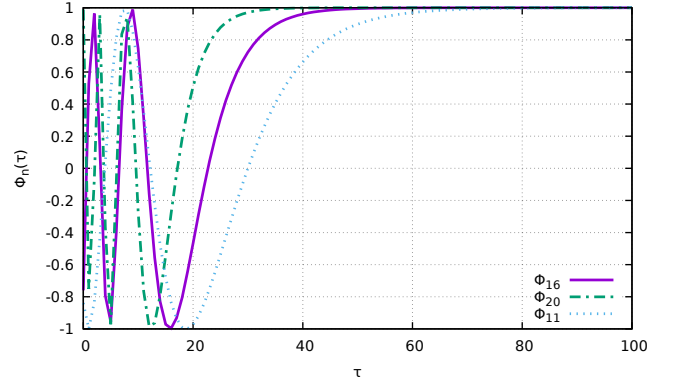

FIG. 1: Sample basis elements utilized throught this work. Note how they allow for highly oscillatory behavior at small values of the triad and then smoothen out as the values get larger.

computation. Thus, Eqn. 7 would be represented as

$$\sum_{n=0}^{N} \omega_n \big[(\sqrt{|\tau|} + \sqrt{|\tau + 2\delta_c|})\Phi_n(\tau + 2\delta_c)$$
$$-(\sqrt{|\tau|} + \sqrt{|\tau - 2\delta_c|})\Phi_n(\tau - 2\delta_c)$$
$$+\lambda(\sqrt{|\tau + \delta_c|} - \sqrt{|\tau - \delta_c|})\Phi_n(\tau)\big] = 0 \quad (9)$$

To solve the above system for the function $B(\tau)$, other than the trivial $B(\tau) = 0$, inspiration was taken from a standard technique for numerically solving partial differential equations. A row can be injected into the system matrix, as long as there is an accompanying column, to disrupt the homogeneity of the system without fundamentally changing the computation. This row can be interpreted as numerically incorporating initial or boundary conditions; and in our approach is treated as a restriction on the solution. For our results in this section, we used $\sum_n \omega_n \Phi_n(\tau = T) = 1$ for a large value of T.

Using this approach, a reconstructed solution can be generated, which is in precise agreement with the recursively computed solution. In Fig. 2 we show a sample basis solution for the $B(\tau)$ equation as compared with a recursion based solution computed by stepping through the values of the τ variable. There, we set $\gamma\delta_b \to 0$ (a requirement for stable solutions) and set $\delta_c = 2\delta_b = \lambda = 1$ for simplicity.

Similarly to solve Eqn. 6 we first make the substitution $C(\mu) \equiv A(\mu + 2\delta_b) - A(\mu - 2\delta_b)$ which results in

$$\mu\big(C(\mu + 2\delta_b) - C(\mu - 2\delta_b)\big) \; +$$
$$2\delta_b\big(C(\mu + 2\delta_b) + C(\mu - 2\delta_b)\big) \; = \; -2\lambda C(\mu) \; . \quad (10)$$

This allows us to easily solve Eqn. 6 using both the basis function method and the recursive approach. The outcome is similar as that shown previously for the $B(\tau)$ function.
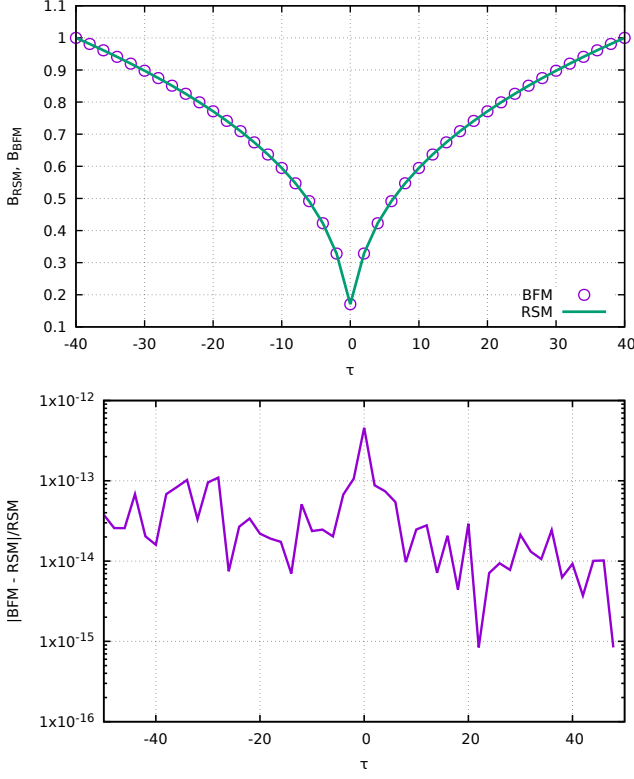
FIG. 2: Solution of the $B(\tau)$ equation using the basis function method as compared with a solution computed by stepping through the τ values using a recursive approach. The upper panel shows both solutions plotted together. The lower panel depicts the absolute difference between the two solutions, which is on the scale of machine precision.
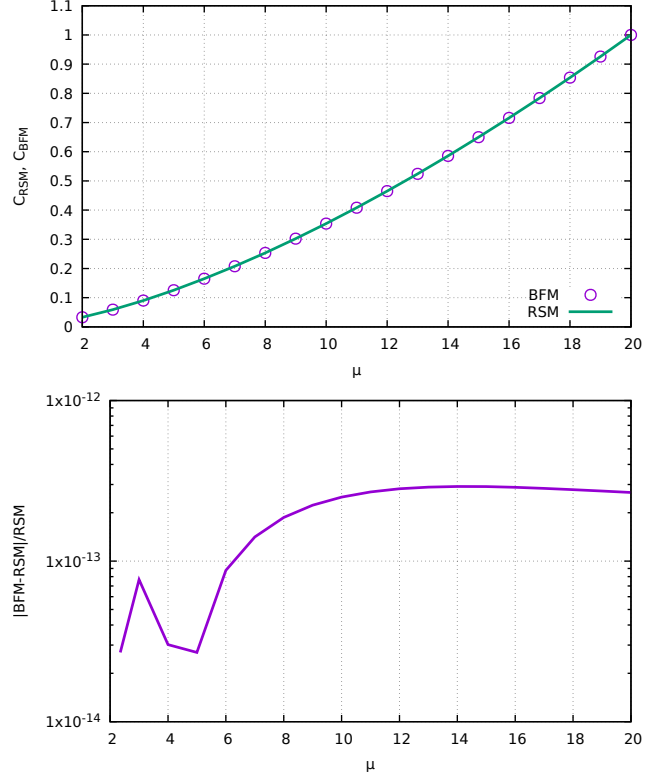

FIG. 3: Solution of the $C(\mu)$ equation using the basis function method as compared with a solution computed by stepping through the μ values using a recursive approach. The upper panel shows both solutions plotted together. The lower panel depicts the absolute difference between the two solutions.

B. Full 2D Solution: Evolution with Basis Functions

The numerical solution computation for the full 2D non-separable case, takes a similar approach. The solution is again taken to be a weighted-sum of the basis functions Eqn. 8, but with τ dependent weights. This leads to Eqn. 4 taking the form

$$\sum_{n=0}^{N} \omega_n(\tau - 2\delta_c)(\Phi_n(\mu + 2\delta_b) - \Phi_n(\mu - 2\delta_b)) =$$

$$\frac{\sqrt{|\tau|} + \sqrt{|\tau + 2\delta_c|}}{\sqrt{|\tau|} + \sqrt{|\tau - 2\delta_c|}} \left(\Psi_{\mu + 2\delta_b, \tau + 2\delta_c} - \Psi_{\mu - 2\delta_b, \tau + 2\delta_c} \right)$$

$$+ \frac{1}{2\delta_b} \frac{\sqrt{|\tau + \delta_c|} - \sqrt{|\tau - \delta_c|}}{\sqrt{|\tau|} + \sqrt{|\tau - 2\delta_c|}} \left[(\mu + 2\delta_b)\Psi_{\mu + 4\delta_b, \tau} + \right.$$

$$\left. (\mu - 2\delta_b)\Psi_{\mu - 4\delta_b, \tau} - 2\mu(1 + 2\gamma^2\delta_b^2)\Psi_{\mu, \tau} \right] . \quad (11)$$

The solution can then be reconstructed on the full range of μ, τ values via

$$\Psi_{\mu, \tau} = \sum_{n=0}^{N} \omega_n(\tau)\Phi_n(\mu) . \quad (12)$$

The τ dependence is captured by stepping through a τ-valued grid of this reconstructed solution.

Similar to the 1D variable-separated case, the basis function method ultimately leads to a linear system of equations. The system involves finding the weights ω_n, while the invertible system matrix represents the left-hand-side of Eqn. 11. A boundary condition is implemented in the manner of row-column injection used previously, where $\lim_{\mu \to \infty} \Psi_{\mu, \tau} = 0$ is represented as $\sum_n \omega_n \Phi_n(\mu = M) = 0$. As typically done in such models, we impose a constraint that represents an initial "semi-classical" wave-packet, i.e. a Gaussian profile for the solution at large τ, and then step backwards in τ, ultimately evolving the system deep into the quantum regime and beyond.

The error denoted by the L_∞-norm as the maximum value of the residual between the recursive step method and basis method solutions at slices of constant τ is depicted in Fig. 5. It is clear that the error stays low for the larger values of τ, however it increases in the neighborhood of $\tau = 0$. This is likely due to the use of a relatively small number of basis elements. In fact, as seen in Fig. 5, it is clear that overall error reduces dra-
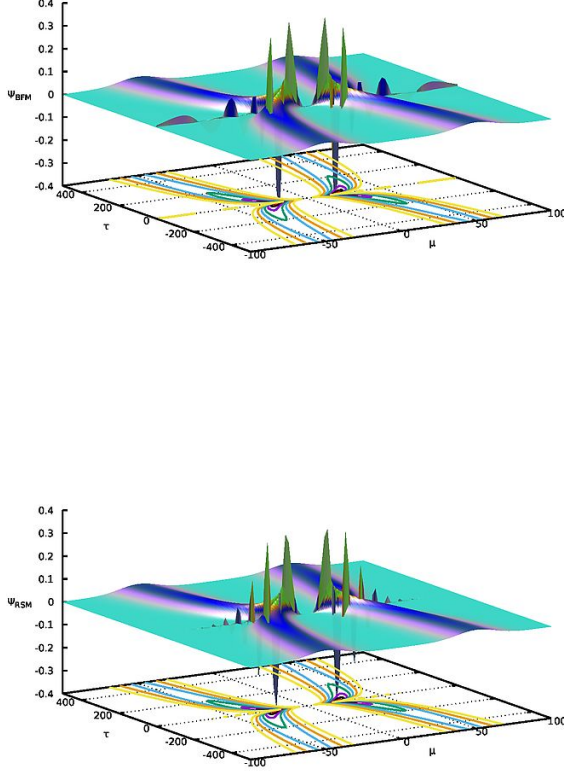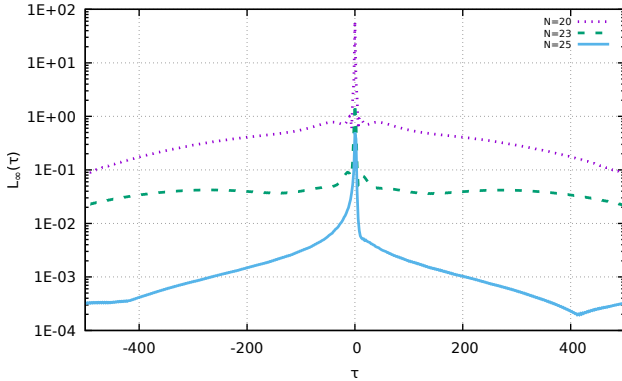
basis elements. Of course, one can also envision a minor tweak in the form of the basis to reduce the error in the small τ regime further. We do not attempt to do that in this work.

The agreement between the basis and recursive method can be even further inspected by examining the volume expectation value $\langle v \rangle$. This is shown in Fig. 6.
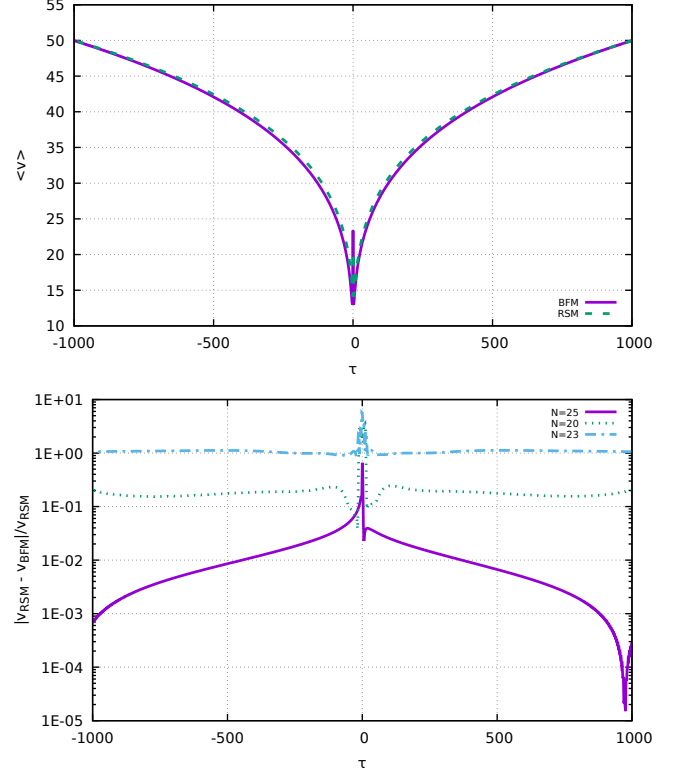

FIG. 4: Solution of Eqn. 4 using the basis function method (upper panel) as compared with a solution computed by stepping through the τ, μ values using a recursive approach (lower panel).

FIG. 6: $\langle v \rangle$ computed using both the basis function method and the recursive step method, and the associated relative difference.

FIG. 5: L_∞-norm of the error at each value of τ for different number of basis elements $N = 20, 23, 25$.

matically with even a modest increase in the number of

1. Optimization

To reduce computational cost, we make use of a key property of the solutions, i.e. they *must* get smoother for the larger triad values. When the solutions are smooth, they can be interpolated very effectively, thus allowing for a significant reduction in the "sampling rate". In order to take advantage of this, we only solve over a subset of triad values as computed through this expression

$$\mu_{i+1} = \mu_i + \lfloor 1 + \left(\frac{2\mu_i}{25}\right)^2 \rfloor \qquad (13)$$

where $\lfloor \rfloor$ denotes the "integer part" or the *gint* function. Visually, this set can be represented as shown in Fig. 7. Through some experimentation, we discovered that this

form yields significant computational benefit with relatively little loss of accuracy. This allows us to reduce the effective grid size by a factor of 10, offering us a tremendous speed-up!
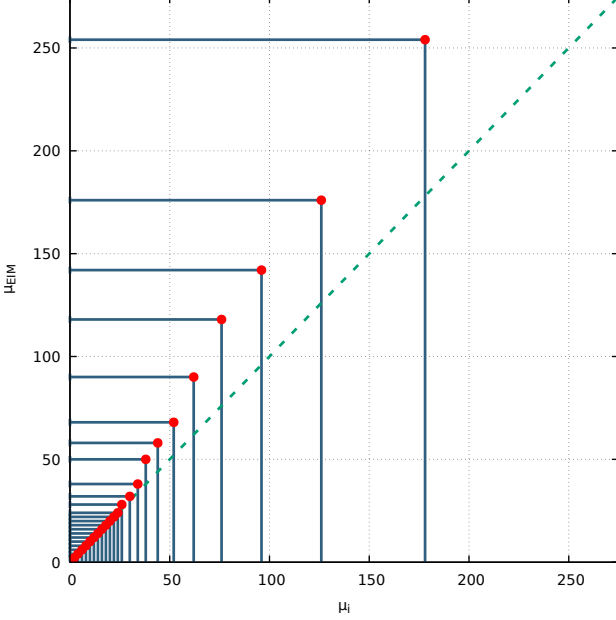

FIG. 7: Subset of μ variable grid values compared with optimal μ variable grid values obtained through the empirical interpolation method.

In addition, while the choice of basis etc. was made largely based on physical considerations and numerical experimentation, we can show that this choice is reasonably optimal using a *reduced basis* approach [18] with an *empirical interpolation method* framework [19], calculated via the *greedy algorithm* [20]. This was done using the open source code *rompy* [21, 22] which demonstrated that the optimal sampling nodes have a similar distribution to the nodes generated with Eqn. 13. This is shown graphically in Fig. 7. Utilizing the chosen basis function in Eqn. 8, one can generate an associated Vandermonde matrix, with the elements

$$V_{i,j} = \Phi_j(\mu_i) \ . \tag{14}$$

If the basis elements $\Phi_j(\mu_i)$ are considered as a reduced basis, one can then perform a singular value decomposition (SVD) of $V_{i,j}$. This demonstrates that the chosen basis is approximately orthonormal and well-conditioned. This was further verified by performing Gram-Schmidt orthonormalization on the basis, which led to no perceived difference in the orthogonality of the basis, and only an insignificant reduction in the conditioning. Thus, we observe that while certain techniques may be used to improve our basis function method (e.g. Gram-Schmidt,

Empirical Interpolation), they are not necessarily required.

C.   Instability

After performing von-Neumann stability analysis on Eqn. 4, it was noted previously [13] that the model is subject to an instability condition,

$$\mu \ > \ 4\tau \ . \tag{15}$$

This condition $\mu > 4\tau$ is of particular interest as it only appears through use of von-Neumann stability analysis in the context of the full 2D system Eqn. 4, and not in the variable-separable case. This is indicative of the fact that the $\mu > 4\tau$ condition is a consequence of the full 2D form of the quantum Hamiltonian constraint. In this section, we attempt to understand how exactly this instability appears in the 2D equation – both, in the context of the stencil-based, finite-difference-like stepping approach and, of course, the basis function method.

1.   The Large μ, τ Limit

Beginning with the original Eqn. 4 and taking the approximation $(\mu, \tau) >> (\delta_b, \delta_c)$ and setting $\gamma\delta_b \to 0$, we obtain

$$(1 + \sqrt{|1 + 2\frac{\delta_c}{\tau}|}) \left(\Psi_{\mu+2\delta_b,\tau+2\delta_c} - \Psi_{\mu-2\delta_b,\tau+2\delta_c} \right)$$
$$+ \frac{1}{2\delta_b} (\sqrt{|1 + \frac{\delta_c}{\tau}|} - \sqrt{|1 - \frac{\delta_c}{\tau}|}) \Big[(\mu)\Psi_{\mu+4\delta_b,\tau} +$$
$$(\mu)\Psi_{\mu-4\delta_b,\tau} - 2\mu\Psi_{\mu,\tau} \Big] +$$
$$(1 + \sqrt{|1 - 2\frac{\delta_c|}{\tau}}) \left(\Psi_{\mu-2\delta_b,\tau-2\delta_c} - \Psi_{\mu+2\delta_b,\tau-2\delta_c} \right)$$
$$= 0 \ . \tag{16}$$

Next, we insert the following Taylor approximation, making implicit use of the expectation that the solution tends towards smooth behavior for large values of μ, τ

$$\Psi_{\mu\pm l\delta_b,\tau\pm m\delta_c} \approx \Psi(\mu \pm l\delta_b, \tau \pm m\delta_c)$$
$$\to \Psi \pm l\delta_b\frac{\partial}{\partial\mu}\Psi \pm m\delta_c\frac{\partial}{\partial\tau}\Psi$$
$$+ \frac{(l\delta_b)^2\partial^2}{2!\partial\mu^2}\Psi + \frac{(m\delta_c)^2\partial^2}{2!\partial\tau^2}\Psi$$
$$\pm 2lm\delta_b\delta_c\frac{\partial^2}{\partial\mu\partial\tau}\Psi + \dots$$
$$\sqrt{|1 \pm 2\frac{\delta_c}{\tau}|} \approx 1 \pm \frac{\delta_c}{\tau} \ .$$

This substitution reduces Eqn. 16, to the expression

$$4\tau\frac{\partial^2}{\partial\mu\partial\tau}\Psi + \frac{\partial}{\partial\mu}\Psi + \mu\frac{\partial^2}{\partial\mu^2}\Psi = 0 \tag{17}$$

which can be further reduced into an equation of the form of the *advection* partial differential equation,

$$\dot{\beta} + \frac{\mu}{4\tau}\beta' = 0$$

$$\beta \equiv \mu \frac{\partial}{\partial \mu}\Psi \ . \tag{18}$$

Now, it is relatively easy to see why an instability appears for $\mu > 4\tau$ part of the computational domain. Making an analogy with the so-called, "CFL condition" [23] that commonly appears in numerical partial differential equation methods, Eqn. 18 presents an advection equation with a locally defined speed of $v = \frac{\mu}{4\tau}$. This defines, at each point in (τ, μ), a local light cone extending into the past defined by the region traced out by the characteristic speed v. However, the finite-difference stencil as defined by the Hamiltonian constraint, has a triad stepping ratio of $\delta_b/2\delta_c = 1$. This results in an instability when $v > 1$, since the local physical domain of dependence is no longer fully contained within the local computational domain as determined by the quantum Hamiltonian constraint.
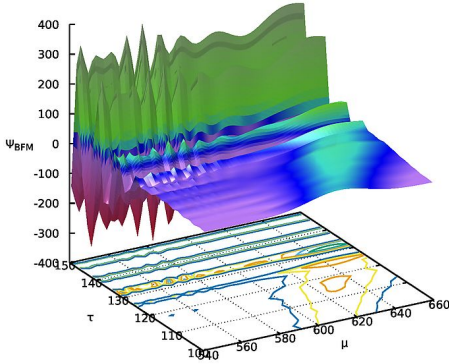

FIG. 8: Evolution using the basis function method into the unstable regime of μ, τ.

As mentioned before, the instability condition of $\mu > 4\tau$ is a result of recursive stepping through both μ and τ as in Eqn. 4. With a global method like the basis method, it could be reasoned that such a local instability condition may be avoided. However, it does not; this can be seen by performing a "shift" of coordinates, $\mu \rightarrow \mu + 555, \tau \rightarrow \tau + 100$ and allowing part of the Gaussian wavepacket to evolve into the unstable regime. Fig. 8 depicts such a sample evolution.

We find that the instability condition as shown is still applicable even in the basis function method. It is suspected that this is due to an intrinsic issue with the CS model, or perhaps due to the lack of a complete Hilbert space of solutions or possibly even a reflection of some underlying physics. Thus, von-Neumann analysis continues to be beneficial towards providing insight on the viability of models in loop quantum cosmology. More specifically, it is able to provide various constraints on such models based on the requirement that evolutions must be stable [8, 13, 25].

2. Stability in the Basis Function Method

To understand the instability in the context of the basis function approach, we take Eqns. 4 and 12 together, while noting that the only time-varying component in Eqn. 12 is $\vec{\omega}$. We then rewrite Eqn. 4 using vector and matrix notation

$$\Psi_{\mu \pm k\delta_b, \tau \pm n\delta_c} \rightarrow \sum_{i=0}^{N} \Phi_i(\mu \pm k\delta_b)\omega_i(\tau \pm n\delta_c)$$

$$= \mathbf{\Phi}(\mu \pm k\delta_b)\vec{\omega}_{\tau \pm n\delta_c} \ . \tag{19}$$

where $\mathbf{\Phi}$ is a square matrix with the different basis elements arranged in columns and each row representing a different μ value. In this notation the CS model can be rewritten as

$$\left(\frac{\sqrt{|\tau|}+\sqrt{|\tau-2\delta_c|}}{\sqrt{|\tau|}+\sqrt{|\tau+2\delta_c|}}\right)\left[\mathbf{\Phi}(\mu-2\delta_b) - \mathbf{\Phi}(\mu+2\delta_b)\right]\vec{\omega}_{\tau-2\delta_c}$$

$$+\frac{1}{2}\left(\frac{\sqrt{|\tau+\delta_c|}-\sqrt{|\tau-\delta_c|}}{\sqrt{|\tau|}+\sqrt{|\tau+2\delta_c|}}\right)\left[(\mu+2\delta_b)\mathbf{\Phi}(\mu+4\delta_b) + (\mu-2\delta_b)\mathbf{\Phi}(\mu-4\delta_b) - 2\mu\mathbf{\Phi}(\mu)\right]\vec{\omega}_\tau$$

$$-\left[\mathbf{\Phi}(\mu-2\delta_b) - \mathbf{\Phi}(\mu+2\delta_b)\right]\vec{\omega}_{\tau+2\delta_c} = \vec{0} \ . \tag{20}$$

Given that $\mathbf{\Phi}$ is a matrix of full rank, we can compute the inverse of matrix $\mathbf{A}$ (defined below) and obtain

$$\mathbf{I}\vec{\omega}_{\tau+2\delta_c} + \frac{1}{2}\Big(\frac{\sqrt{|\tau-\delta_c|}-\sqrt{|\tau+\delta_c|}}{\sqrt{|\tau|}+\sqrt{|\tau+2\delta_c|}}\Big)\mathbf{A}^{-1}\mathbf{B}\vec{\omega}_\tau - \Big(\frac{\sqrt{|\tau|}+\sqrt{|\tau-2\delta_c|}}{\sqrt{|\tau|}+\sqrt{|\tau+2\delta_c|}}\Big)\mathbf{I}\vec{\omega}_{\tau-2\delta_c} = \vec{0}$$

$$\mathbf{\Phi}(\mu-2\delta_b) - \mathbf{\Phi}(\mu+2\delta_b) \equiv \mathbf{A}$$

$$\big[(\mu+2\delta_b)\mathbf{\Phi}(\mu+4\delta_b) + (\mu-2\delta_b)\mathbf{\Phi}(\mu-4\delta_b) - 2\mu\mathbf{\Phi}(\mu)\big] \equiv \mathbf{B} \ . \tag{21}$$

This can be further reduced down to

$$\mathbf{I}\vec{\omega}_{\tau+2\delta_c} + \frac{1}{2}\alpha\mathbf{D}\vec{\omega}_\tau - \beta\mathbf{I}\vec{\omega}_{\tau-2\delta_c} = \vec{0}$$

$$\mathbf{D} \equiv \mathbf{A}^{-1}\mathbf{B}$$

$$\alpha \equiv \Big(\frac{\sqrt{|\tau-\delta_c|}-\sqrt{|\tau+\delta_c|}}{\sqrt{|\tau|}+\sqrt{|\tau+2\delta_c|}}\Big)$$

$$\beta \equiv \Big(\frac{\sqrt{|\tau|}+\sqrt{|\tau-2\delta_c|}}{\sqrt{|\tau|}+\sqrt{|\tau+2\delta_c|}}\Big) \ . \tag{22}$$

Finally, if the following definitions are made,

$$\vec{q}_\tau \equiv \begin{bmatrix} \vec{\omega}_\tau \\ \vec{\omega}_{\tau-2\delta_c} \end{bmatrix} \qquad \mathbf{\Pi} \equiv \begin{bmatrix} -\frac{1}{2}\alpha\mathbf{D} & \beta\mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \tag{23}$$

then one can use Eqn. 23 to express Eqn. 22 in the following manner,

$$\vec{q}_{\tau+2\delta_c} = \mathbf{\Pi}\vec{q}_\tau \ . \tag{24}$$

Now, we use the eigenvalues of matrix $\mathbf{\Pi}$ to test for stability; if any of the eigenvalue magnitudes exceed unity, that makes the solution of Eqn. 24 grow unboundedly. With some additional simplifications (in particular, $(\mu,\tau) >> (\delta_b,\delta_c)$) accompanied with some numerical computations it is not difficult to see that one obtains the same unstable region ($\mu > 4\tau$) as derived before in Ref. [13] using basic von Neumann stability analysis.

IV. DISCUSSION AND CONCLUSIONS

In this paper, we have developed a new numerical method that is particularly suitable for computing solutions to the quantum Hamiltonian constraint of models in loop quantum cosmology. The method is inspired by the spectral collocation method that is commonly used for numerically solving partial differential equations. It uses a special set of basis functions that take full advantage of the expected behavior of physical solutions of the model. We also compared the new basis function method with the previously used approach of a stencil-style computation using a recursive computation over a grid of allowed values for the triad variables. In addition, we also discussed how the stability analysis appears in the context of the basis approach.

In this section we will document how the basis function method improves upon previous approaches. As pointed out earlier, the main advantage is computational efficiency and ease of parallelization.

A. Parallelizability

The basis function method ultimately involves the computation of a solution of a linear system of equations of size proportional to the square of the number of basis elements in use. Since this is a very well-studied problem, there exist many highly optimized solvers available even for parallel hardware such as multi-core CPUs and many-core GPUs. On the other hand, the recursive step method is intrinsically serial and rather challenging to parallelize.

Cores	BFM Run-Time (s)
1	161.110
2	85.805
4	48.852
8	29.751

TABLE I: Run-Time by number of cores with simple OpenMP multi-threading on $\mu,\tau \in [-1600, 1600]$ grid

With a simple multi-threaded implementation for the BFM algorithm, the benefit of parallelization can be clearly seen. Increased benefit is expected when the number of basis elements gets larger; and we anticipate a more in-depth study of this feature for both multi-core CPUs and many-core GPUs in future work.

B. Precision

As mentioned above, the basis function method involves solving a linear system of equations. Performing an analysis on the backward error of matrix inversion techniques yields

$$\mathbf{A}\vec{x} = \vec{b}$$

$$\epsilon_{BFM} = ||\mathbf{A}^{-1}\vec{b} - \vec{x}|| = \mathcal{O}(\kappa_2\epsilon_{machine})$$

$$\kappa_2 \equiv ||\mathbf{A}^{-1}||_2||\mathbf{A}||_2 \propto N \ . \tag{25}$$

On the other hand it has been found that for a recursive step method with $\mu \in [\frac{-M}{2}, \frac{M}{2}]$ and $\tau \in [\frac{-T}{2}, \frac{T}{2}]$, the error $\epsilon_{RSM} \propto \mathcal{O}(T \times M\epsilon_{machine})$. Since standard backward error is proportional to the number of basis elements, we have shown that a small and constant number can compute solutions on a range of domains accurately; it stands to reason that as long as the non-zero regions of the computed solution are reasonably well sampled by

the basis, the BFM approach can calculate a larger range of solutions with a set level of computational precision.
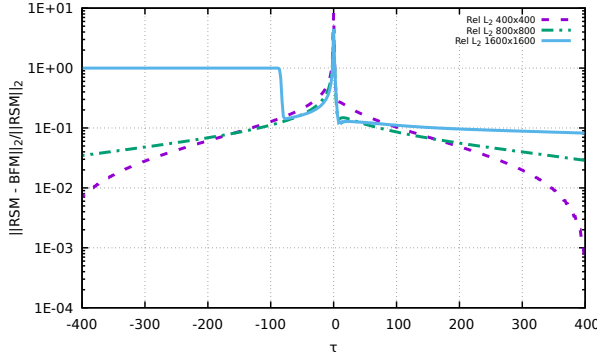

FIG. 9: Relative L_2 error for various size domains, and evolution with the basis function method over large domain, $\mu, \tau \in [-1600 : 1600]$

It can be seen in Fig. 9 that the recursive step method based solution experiences unbounded growth due to fixed, finite available precision; while the basis method computed solution maintains the correct overall behavior expected. For a closer analysis of the effects that finite precision can have on the recursive step method solutions a precision test can be performed. For such a test, we maintain a constant domain and solve *identical* simulations with varying precision. The error can then be analyzed by taking the highest precision solution as *most accurate*. Our standard error analysis follows as

$$\epsilon_P(\tau_i) = \frac{\sum_n |\Psi_{Quad}(\mu_n, \tau_i) - \Psi_P(\mu_n, \tau_i)|^2}{\sum_n \Psi_{Quad}^2(\mu_n, \tau_i)}$$

where P is a precision other than quadruple-precision (e.g. double, single). The results are depicted in Fig. 10. They suggest that the recursive step method is far more prone to suffer from precision limitations over the basis method. This ultimately translates into a major computational advantage in favor of the basis method.

In summary, we have developed a new numerical approach towards solving quantum Hamiltonian constraints in loop quantum cosmology. The approach makes use of a set of basis functions, specifically designed using key physical features of the solutions in mind. This basis method, borrows from the well-known spectral collocation method for solving partial differential equations. We demonstrate the efficacy of the method and compare
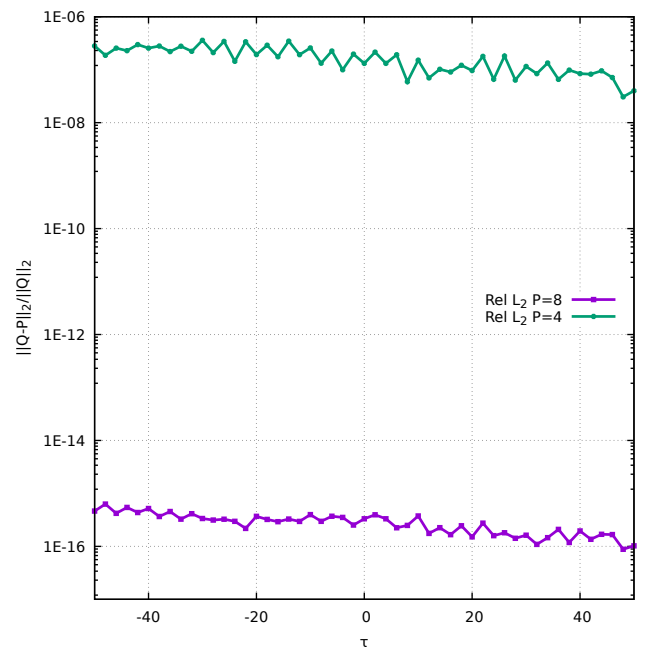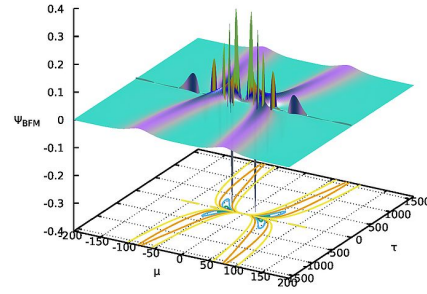

FIG. 10: Precision test for single (4-byte), and double (8-byte) precision. It is clear that error grows significantly as the evolution progresses (recall, the system evolves backward in τ).

it with the previously used recursive step method. We find that the basis method offers a number of benefits over the previously used approach, especially in the area

of computational efficiency. Throughout this paper, we use the Corichi and Singh loop quantum model of the Schwarzschild interior for demonstration purposes.

---

[1] T. Thiemann, *Modern Canonical Quantum General Relativity*, Cambridge Monographs on Mathematical Physics, Cambridge University Press (2008).

[2] C. Rovelli, *Quantum Gravity*, Cambridge University Press (2007).

[3] A. Ashtekar, J. Pullin, *Loop Quantum Gravity: The First 30 Years (100 Years of General Relativity)*, World Scientific (2017).

[4] A. Ashtekar, and P. Singh, Class. Quant. Grav. 28, 213001 (2012).

[5] M. Bojowald, Living Rev. Rel., 11, 4 (2008).

[6] A. Ashtekar, T. Pawlowski, and P. Singh, Phys. Rev. Lett. 96, 141301 (2006).

[7] A. Ashtekar, T. Pawlowski, and P. Singh, Phys. Rev. D 74, 084003 (2006).

[8] P. Singh, Computing in Science & Engineering 20(4), 26 (2018).

[9] D. Brizuela, D. Cartin, and G. Khanna, SIGMA 8, 001 (2012).

[10] P. Singh, Class. Quant. Grav. vol. 29, 244002 (2012).

[11] P. Diener, B. Gupt, M. Megevand, and P. Singh, Class.Quant. Grav., 31, 165006 (2014)

[12] A. Corichi, P. Singh, Class. Quantum Grav. 33, 055006 (2016).

[13] A. Yonika, G. Khanna, P. Singh, Class. Quantum Grav. 35, 045007 (2018).

[14] H. Haggard and C. Rovelli, Phys. Rev. D 92, 104020 (2015).

[15] E. Bianchi, M. Christodoulou, F. D'Ambrosio, H. Haggard and C. Rovelli, Class. Quantum Grav. 35, 225003 (2018).

[16] J. Olmedo, S. Saini and P. Singh, Class. Quantum Grav. 34, 225011 (2017).

[17] S. Connors, G. Khanna, Class. Quantum Grav. 23, 2919 (2006).

[18] S. Chaturantabut, D. Sorensen, SIAM J. of Sci. Comput. 32(5), (2010)

[19] Y. Maday, N. Nguyen, A. Patera, G. Pau, Communications on Pure and Applied Analysis 8(1), (2009)

[20] H. Antil, S. Field, F. Herrmann, R. Nochetto, M. Tiglio, Journal of Scientific Computing 57, (2013)

[21] S. Field, C. Galley, J. Hesthaven, J. Kaye, M. Tiglio, Phys. Rev. X 4, 031006 (2014)

[22] C. Galley, *rompy* package https://bitbucket.org/chadgalley/rompy/src/master

[23] R. Courant, K. O. Friedrichs, H. Lewy, IBM Journal 11, 215 (1967).

[24] P. Diener, B. Gupt and P. Singh, Class. Quant. Grav. 31, 025013 (2014)

[25] S. Saini and P. Singh, Class. Quant. Grav. 36, 105010 (2019)