Solving Einstein's Equation Numerically on Manifolds with Arbitrary Spatial Topologies

Lee Lindblom

Collaborators: Béla Szilágyi and Nicholas W. Taylor
Theoretical Astrophysics, Caltech

Center for Gravitation and Cosmology Seminar
University of Wisconsin at Milwaukee — 24 May 2013

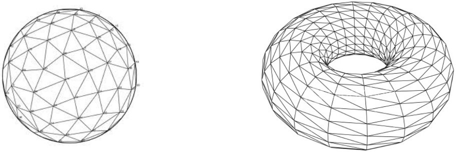
- Multi-cube representations of arbitrary three-manifolds.
- Boundary conditions for elliptic, parabolic and hyperbolic PDEs.
- Numerical tests for solutions of simple PDEs.
- Covariant first-order representation of Einstein's equation.
- Simple numerical Einstein evolutions.

Representations of Arbitrary Three-Manifolds

- Goal: Develop numerical methods that are easily adapted to solving elliptic PDEs on three-manifolds Σ with arbitrary topology, and parabolic or hyperbolic PDEs on manifolds $R \times \Sigma$.

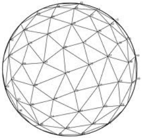
Representations of Arbitrary Three-Manifolds

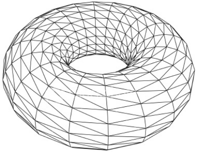
- Goal: Develop numerical methods that are easily adapted to solving elliptic PDEs on three-manifolds Σ with arbitrary topology, and parabolic or hyperbolic PDEs on manifolds $R \times \Sigma$.
- Every two- and three-manifold admits a triangulation (Radó 1925, Moire 1952), i.e. can be represented as a set of triangles (or tetrahedra), plus a list of rules for gluing their edges (or faces) together.



Representations of Arbitrary Three-Manifolds

- Goal: Develop numerical methods that are easily adapted to solving elliptic PDEs on three-manifolds Σ with arbitrary topology, and parabolic or hyperbolic PDEs on manifolds $R \times \Sigma$.
- Every two- and three-manifold admits a triangulation (Radó 1925, Moire 1952), i.e. can be represented as a set of triangles (or tetrahedra), plus a list of rules for gluing their edges (or faces) together.

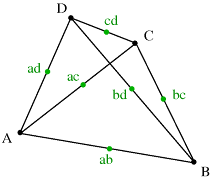




- Cubes make more convenient computational domains for finite difference and spectral numerical methods.
- Can arbitrary two- and three-manifolds be "cubed", i.e. represented as a set of squares or cubes plus a list of rules for gluing their edges or faces together?

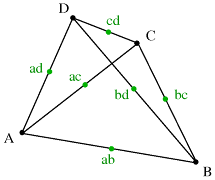
“Multi-Cube” Representations of Three-Manifolds

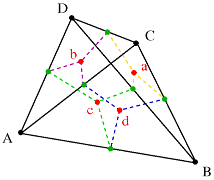
- Every two- and three-dimensional triangulation can be refined to a ”multi-cube” representation: For example, in three-dimensions divide each tetrahedron into four “distorted” cubes:



“Multi-Cube” Representations of Three-Manifolds

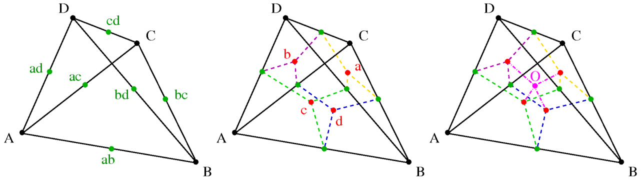
- Every two- and three-dimensional triangulation can be refined to a ”multi-cube” representation: For example, in three-dimensions divide each tetrahedron into four “distorted” cubes:





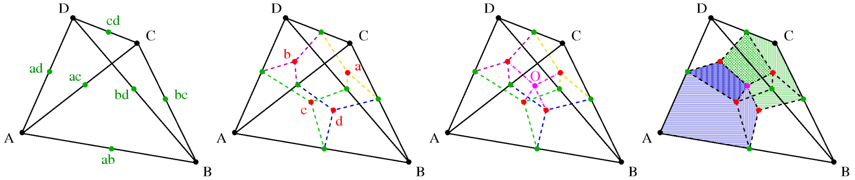
“Multi-Cube” Representations of Three-Manifolds

- Every two- and three-dimensional triangulation can be refined to a ”multi-cube” representation: For example, in three-dimensions divide each tetrahedron into four “distorted” cubes:



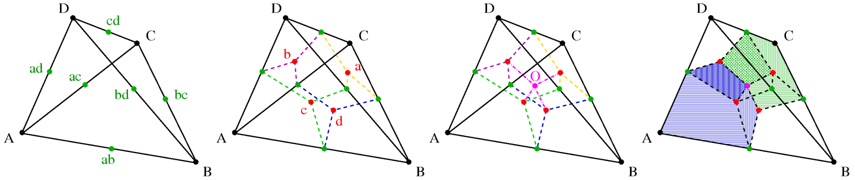
"Multi-Cube" Representations of Three-Manifolds

- Every two- and three-dimensional triangulation can be refined to a "multi-cube" representation: For example, in three-dimensions divide each tetrahedron into four "distorted" cubes:

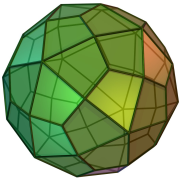


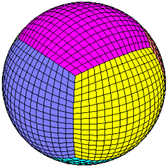
“Multi-Cube” Representations of Three-Manifolds

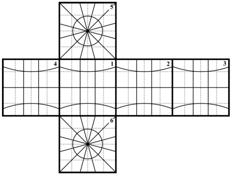
- Every two- and three-dimensional triangulation can be refined to a ”multi-cube” representation: For example, in three-dimensions divide each tetrahedron into four “distorted” cubes:



- Every two- or three-manifold can be represented as a set of squares or cubes, plus maps that identify their edges or faces.







Boundary Maps: Fixing the Topology

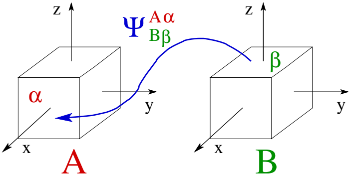
- Multi-cube representations of topological manifolds consist of a set of cubic regions, $\mathcal{B}_A$, plus maps that identify the faces of neighboring regions, $\Psi^{A\alpha}_{B\beta}(\partial_\beta \mathcal{B}_B) = \partial_\alpha \mathcal{B}_A$.

Boundary Maps: Fixing the Topology

- Multi-cube representations of topological manifolds consist of a set of cubic regions, $\mathcal{B}_A$, plus maps that identify the faces of neighboring regions, $\Psi^{A\alpha}_{B\beta}(\partial_\beta\mathcal{B}_B) = \partial_\alpha\mathcal{B}_A$.
- Choose cubic regions to have uniform size and orientation.

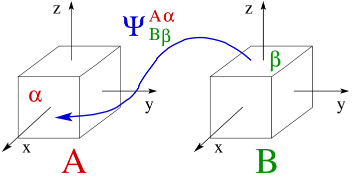
Boundary Maps: Fixing the Topology

- Multi-cube representations of topological manifolds consist of a set of cubic regions, $\mathcal{B}_A$, plus maps that identify the faces of neighboring regions, $\Psi^{A\alpha}_{B\beta}(\partial_\beta \mathcal{B}_B) = \partial_\alpha \mathcal{B}_A$.
- Choose cubic regions to have uniform size and orientation.
- Choose linear interface identification maps $\Psi^{A\alpha}_{B\beta}$: $x^i_A = c^i_{A\,\alpha} + C^{A\,\alpha\,i}_{B\,\beta\,k}(x^k_B - c^k_{B\,\beta})$, where $C^{A\,\alpha\,i}_{B\,\beta\,k}$ is a rotation-reflection matrix, and $c^i_{A\,\alpha}$ is center of α face of region A.

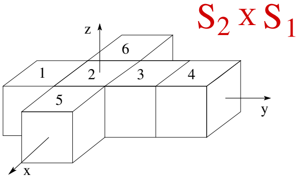


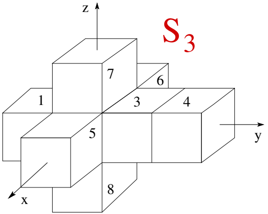
Boundary Maps: Fixing the Topology

- Multi-cube representations of topological manifolds consist of a set of cubic regions, $\mathcal{B}_A$, plus maps that identify the faces of neighboring regions, $\Psi^{A\alpha}_{B\beta}(\partial_\beta \mathcal{B}_B) = \partial_\alpha \mathcal{B}_A$.
- Choose cubic regions to have uniform size and orientation.
- Choose linear interface identification maps $\Psi^{A\alpha}_{B\beta}$:
  $x^i_A = c^i_{A\alpha} + C^{A\alpha\, i}_{B\beta\, k}(x^k_B - c^k_{B\beta})$,
  where $C^{A\alpha\, i}_{B\beta\, k}$ is a rotation-reflection matrix, and $c^i_{A\alpha}$ is center of α face of region A.



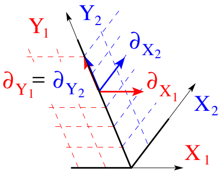
- Examples:





Fixing the Differential Structure

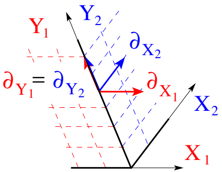
- The boundary identification maps, $\Psi^{A\alpha}_{B\beta}$, used to construct multi-cube topological manifolds are continuous, but typically are not differentiable at the interfaces.



- Smooth tensor fields expressed in multi-cube coordinates are not (in general) even continuous at the interfaces.

Fixing the Differential Structure

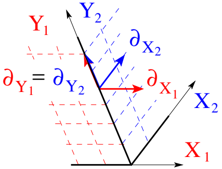
- The boundary identification maps, $\Psi^{A\alpha}_{B\beta}$, used to construct multi-cube topological manifolds are continuous, but typically are not differentiable at the interfaces.



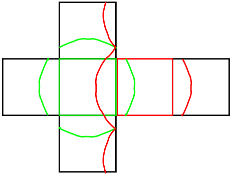
- Smooth tensor fields expressed in multi-cube coordinates are not (in general) even continuous at the interfaces.
- Differential structure provides the framework in which smooth functions and tensors are defined on a manifold.
- The standard construction assumes the existence of overlapping coordinate domains having smooth transition maps.

Fixing the Differential Structure

- The boundary identification maps, $\Psi^{A\alpha}_{B\beta}$, used to construct multi-cube topological manifolds are continuous, but typically are not differentiable at the interfaces.



- Smooth tensor fields expressed in multi-cube coordinates are not (in general) even continuous at the interfaces.
- Differential structure provides the framework in which smooth functions and tensors are defined on a manifold.
- The standard construction assumes the existence of overlapping coordinate domains having smooth transition maps.
- Multi-cube manifolds need an additional layer of infrastructure: e.g., overlapping domains $\mathcal{D}_A \supset \mathcal{B}_A$ with transition maps that are smooth in the overlap regions.



Fixing the Differential Structure II

- All that is needed to define continuous tensor fields at interface boundaries is the Jacobian $J^{A\alpha i}_{B\beta k}$ and its dual $J^{*B\beta k}_{A\alpha i}$ that transform tensors from one multi-cube coordinate region to another: for example, $v^i_A = J^{A\alpha i}_{B\beta k} v^k_B$ and $w_{Ai} = J^{*B\beta k}_{A\alpha i} w_{Bk}$.

Fixing the Differential Structure II

- All that is needed to define continuous tensor fields at interface boundaries is the Jacobian $J^{A\alpha i}_{B\beta k}$ and its dual $J^{*B\beta k}_{A\alpha i}$ that transform tensors from one multi-cube coordinate region to another: for example, $v^i_A = J^{A\alpha i}_{B\beta k} v^k_B$ and $w_{Ai} = J^{*B\beta k}_{A\alpha i} w_{Bk}$.
- A smooth reference metric $\tilde{g}_{ij}$ determines the needed Jacobians.
- Let $\tilde{g}_{Aij}$ and $\tilde{g}_{Bij}$ be the components of a smooth reference metric in the multi-cube coordinates of regions $\mathcal{B}_A$ and $\mathcal{B}_B$ that are identified at the faces $\partial_\alpha \mathcal{B}_A \leftrightarrow \partial_\beta \mathcal{B}_B$.

Fixing the Differential Structure II

- All that is needed to define continuous tensor fields at interface boundaries is the Jacobian $J^{A\alpha i}_{B\beta k}$ and its dual $J^{*B\beta k}_{A\alpha i}$ that transform tensors from one multi-cube coordinate region to another: for example, $v^i_A = J^{A\alpha i}_{B\beta k} v^k_B$ and $w_{Ai} = J^{*B\beta k}_{A\alpha i} w_{Bk}$.
- A smooth reference metric $\tilde{g}_{ij}$ determines the needed Jacobians.
- Let $\tilde{g}_{Aij}$ and $\tilde{g}_{Bij}$ be the components of a smooth reference metric in the multi-cube coordinates of regions $\mathcal{B}_A$ and $\mathcal{B}_B$ that are identified at the faces $\partial_\alpha \mathcal{B}_A \leftrightarrow \partial_\beta \mathcal{B}_B$.
- Use the reference metric to define the outward directed unit normals: $n_{A\alpha i}$, $n^i_{A\alpha}$, $n_{B\beta i}$, and $n^i_{B\beta}$.

Fixing the Differential Structure II

- All that is needed to define continuous tensor fields at interface boundaries is the Jacobian $J^{A\alpha i}_{B\beta k}$ and its dual $J^{*B\beta k}_{A\alpha i}$ that transform tensors from one multi-cube coordinate region to another: for example, $v^i_A = J^{A\alpha i}_{B\beta k} v^k_B$ and $w_{Ai} = J^{*B\beta k}_{A\alpha i} w_{Bk}$.
- A smooth reference metric $\tilde{g}_{ij}$ determines the needed Jacobians.
- Let $\tilde{g}_{Aij}$ and $\tilde{g}_{Bij}$ be the components of a smooth reference metric in the multi-cube coordinates of regions $\mathcal{B}_A$ and $\mathcal{B}_B$ that are identified at the faces $\partial_\alpha\mathcal{B}_A \leftrightarrow \partial_\beta\mathcal{B}_B$.
- Use the reference metric to define the outward directed unit normals: $n_{A\alpha i}$, $n^i_{A\alpha}$, $n_{B\beta i}$, and $n^i_{B\beta}$.
- The needed Jacobians are given by

$$J^{A\alpha i}_{B\beta k} = C^{A\alpha i}_{B\beta\ell}\left(\delta^\ell_k - n^\ell_{B\beta} n_{B\beta k}\right) - n^i_{A\alpha} n_{B\beta k},$$
$$J^{*B\beta k}_{A\alpha i} = \left(\delta^\ell_i - n_{A\alpha i} n^\ell_{A\alpha}\right) C^{B\beta k}_{A\alpha\ell} - n_{A\alpha i} n^k_{B\beta}.$$

Fixing the Differential Structure II

- All that is needed to define continuous tensor fields at interface boundaries is the Jacobian $J^{A\alpha i}_{B\beta k}$ and its dual $J^{*B\beta k}_{A\alpha i}$ that transform tensors from one multi-cube coordinate region to another: for example, $v^i_A = J^{A\alpha i}_{B\beta k} v^k_B$ and $w_{Ai} = J^{*B\beta k}_{A\alpha i} w_{Bk}$.
- A smooth reference metric $\tilde{g}_{ij}$ determines the needed Jacobians.
- Let $\tilde{g}_{Aij}$ and $\tilde{g}_{Bij}$ be the components of a smooth reference metric in the multi-cube coordinates of regions $\mathcal{B}_A$ and $\mathcal{B}_B$ that are identified at the faces $\partial_\alpha \mathcal{B}_A \leftrightarrow \partial_\beta \mathcal{B}_B$.
- Use the reference metric to define the outward directed unit normals: $n_{A\alpha i}$, $n^i_{A\alpha}$, $n_{B\beta i}$, and $n^i_{B\beta}$.
- The needed Jacobians are given by

$$J^{A\alpha i}_{B\beta k} = C^{A\alpha i}_{B\beta \ell}\left(\delta^\ell_k - n^\ell_{B\beta} n_{B\beta k}\right) - n^i_{A\alpha} n_{B\beta k},$$
$$J^{*B\beta k}_{A\alpha i} = \left(\delta^\ell_i - n_{A\alpha i} n^\ell_{A\alpha}\right) C^{B\beta k}_{A\alpha \ell} - n_{A\alpha i} n^k_{B\beta}.$$

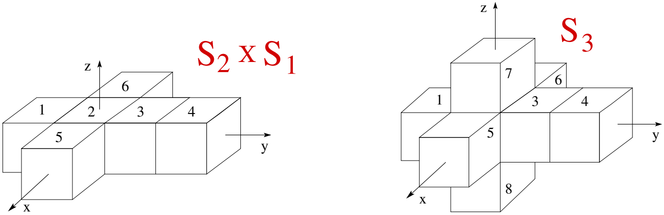
- These Jacobians satisfy:

$$n^i_{A\alpha} = -J^{A\alpha i}_{B\beta k} n^k_{B\beta},$$
$$n_{A\alpha i} = -J^{*B\beta k}_{A\alpha i} n_{B\beta k},$$
$$t^i_{A\alpha} = J^{A\alpha i}_{B\beta k} t^k_{B\beta} = C^{A\alpha i}_{B\beta k} t^k_{B\beta},$$
$$\delta^{Ai}_{Ak} = J^{A\alpha i}_{B\beta \ell} J^{*B\beta \ell}_{A\alpha k}.$$

Fixing the Differential Structure II

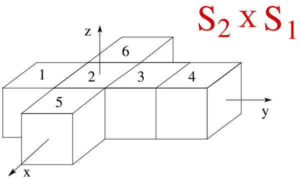
- All that is needed to define continuous tensor fields at interface boundaries is the Jacobian $J^{A\alpha i}_{B\beta k}$ and its dual $J^{*B\beta k}_{A\alpha i}$ that transform tensors from one multi-cube coordinate region to another: for example, $v^i_A = J^{A\alpha i}_{B\beta k} v^k_B$ and $w_{Ai} = J^{*B\beta k}_{A\alpha i} w_{Bk}$.
- A smooth reference metric $\tilde{g}_{ij}$ determines the needed Jacobians.
- Let $\tilde{g}_{Aij}$ and $\tilde{g}_{Bij}$ be the components of a smooth reference metric in the multi-cube coordinates of regions $\mathcal{B}_A$ and $\mathcal{B}_B$ that are identified at the faces $\partial_\alpha \mathcal{B}_A \leftrightarrow \partial_\beta \mathcal{B}_B$.
- Use the reference metric to define the outward directed unit normals: $n_{A\alpha i}$, $n^i_{A\alpha}$, $n_{B\beta i}$, and $n^i_{B\beta}$.
- The needed Jacobians are given by
$$J^{A\alpha i}_{B\beta k} = C^{A\alpha i}_{B\beta \ell}\left(\delta^\ell_k - n^\ell_{B\beta} n_{B\beta k}\right) - n^i_{A\alpha} n_{B\beta k},$$
$$J^{*B\beta k}_{A\alpha i} = \left(\delta^\ell_i - n_{A\alpha i} n^\ell_{A\alpha}\right) C^{B\beta k}_{A\alpha \ell} - n_{A\alpha i} n^k_{B\beta}.$$
- Use continuity of the covariant derivatives of tensors, e.g. $\tilde{\nabla}_{Ai} v^k_A$, to define their differentiability.
- These Jacobians satisfy:
$$n^i_{A\alpha} = -J^{A\alpha i}_{B\beta k} n^k_{B\beta},$$
$$n_{A\alpha i} = -J^{*B\beta k}_{A\alpha i} n_{B\beta k},$$
$$t^i_{A\alpha} = J^{A\alpha i}_{B\beta k} t^k_{B\beta} = C^{A\alpha i}_{B\beta k} t^k_{B\beta},$$
$$\delta^{Ai}_{Ak} = J^{A\alpha i}_{B\beta \ell} J^{*B\beta \ell}_{A\alpha k}.$$

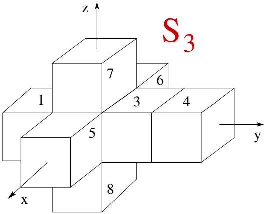
Solving PDEs on Multi-Cube Manifolds



- Solve PDEs in each cubic region separately.
- Use boundary conditions on cube faces to select the correct smooth global solution.

Solving PDEs on Multi-Cube Manifolds

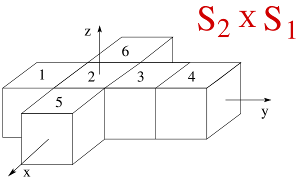


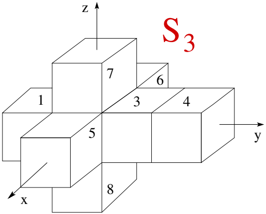


- Solve PDEs in each cubic region separately.
- Use boundary conditions on cube faces to select the correct smooth global solution.
- For second-order strongly-elliptic systems: enforce continuity on one face and continuity of normal derivatives on neighboring face,

$$u_A \simeq u_B \qquad\qquad \tilde{\nabla}_{n_B} u_B \simeq -\tilde{\nabla}_{n_A} u_A.$$

Solving PDEs on Multi-Cube Manifolds





- Solve PDEs in each cubic region separately.
- Use boundary conditions on cube faces to select the correct smooth global solution.
- For second-order strongly-elliptic systems: enforce continuity on one face and continuity of normal derivatives on neighboring face,

$$u_A \simeq u_B \qquad\qquad \tilde{\nabla}_{n_B} u_B \simeq -\tilde{\nabla}_{n_A} u_A.$$

- For first-order symmetric hyperbolic systems whose dynamical fields are tensors: set incoming characteristic fields with outgoing characteristics from neighbor,

$$\hat{u}_A^- \simeq \hat{u}_B^+ \qquad\qquad \hat{u}_B^- \simeq \hat{u}_A^+.$$

Numerical Methods

- Represent each component of each tensor function as a (finite) sum of spectral basis functions, $\mathbf{u} = \sum_{pqr} \mathbf{u}_{pqr}\, T_p(x)\, T_q(y)\, T_r(z)$, in each cubic region.

Numerical Methods

- Represent each component of each tensor function as a (finite) sum of spectral basis functions, $\mathbf{u} = \sum_{pqr} \mathbf{u}_{pqr}\, T_p(x)\, T_q(y)\, T_r(z)$, in each cubic region.
- Evaluate derivatives of the functions using the known derivatives of the basis functions: $\partial_x \mathbf{u} = \sum_{pqr} \mathbf{u}_{pqr}\, \partial_x T_p(x)\, T_q(y)\, T_r(z)$.

Numerical Methods

- Represent each component of each tensor function as a (finite) sum of spectral basis functions, $\mathbf{u} = \sum_{pqr} \mathbf{u}_{pqr}\, T_p(x)\, T_q(y)\, T_r(z)$, in each cubic region.
- Evaluate derivatives of the functions using the known derivatives of the basis functions: $\partial_x \mathbf{u} = \sum_{pqr} \mathbf{u}_{pqr}\, \partial_x T_p(x)\, T_q(y)\, T_r(z)$.
- Evaluate the PDEs and BCs on a set of collocation points, $\{x_i, y_j, z_k\}$, chosen so that $\mathbf{u}(x_i, y_j, z_k)$ can be mapped efficiently onto the spectral coefficients $\mathbf{u}_{pqr}$. Derivatives become linear combinations of the fields: $\partial_x \mathbf{u}(x_i, y_j, z_k) = \sum_\ell D_i{}^\ell\, \mathbf{u}(x_\ell, y_j, z_k)$.

Numerical Methods

- Represent each component of each tensor function as a (finite) sum of spectral basis functions, $\mathbf{u} = \sum_{pqr} \mathbf{u}_{pqr}\, T_p(x)\, T_q(y)\, T_r(z)$, in each cubic region.
- Evaluate derivatives of the functions using the known derivatives of the basis functions: $\partial_x \mathbf{u} = \sum_{pqr} \mathbf{u}_{pqr}\, \partial_x T_p(x)\, T_q(y)\, T_r(z)$.
- Evaluate the PDEs and BCs on a set of collocation points, $\{x_i, y_j, z_k\}$, chosen so that $\mathbf{u}(x_i, y_j, z_k)$ can be mapped efficiently onto the spectral coefficients $\mathbf{u}_{pqr}$. Derivatives become linear combinations of the fields: $\partial_x \mathbf{u}(x_i, y_j, z_k) = \sum_\ell D_i{}^\ell\, \mathbf{u}(x_\ell, y_j, z_k)$.
- For elliptic systems, these pseudo-spectral equations become a system of algebraic equations for $\mathbf{u}(x_i, y_j, z_k)$. Solve these algebraic equations using standard numerical methods.
- For hyperbolic systems these equations become a system of ordinary differential equations for $\mathbf{u}(x_i, y_j, z_k, t)$. Solve these equations by the method of lines using standard ode integrators.

Testing the Elliptic PDE Solver

- Solve the elliptic PDE, $\nabla^i\nabla_i\psi - c^2\psi = f$ where c^2 is a constant, and f is a given function.

Testing the Elliptic PDE Solver

- Solve the elliptic PDE, $\nabla^i\nabla_i\psi - c^2\psi = f$ where c^2 is a constant, and f is a given function.
- Use the co-variant derivative ∇_i for the round metric on $S^2 \times S^1$:

$$\begin{aligned} ds^2 &= R_1^2 d\chi^2 + R_2^2\left(d\theta^2 + \sin^2\theta\, d\varphi^2\right), \\ &= \left(\frac{2\pi R_1}{L}\right)^2 dz_A^2 + \left(\frac{\pi R_2}{2L}\right)^2 \frac{(1+X_A^2)(1+Y_A^2)}{(1+X_A^2+Y_A^2)^2} \\ &\quad \times\left[(1+X_A^2)\,dx_A^2 - 2X_A Y_A\, dx_A\, dy_A + (1+Y_A^2)\,dy_A^2\right]. \end{aligned}$$

where $X_A = \tan\left[\pi(x_A - c_A^x)/2L\right]$ and $Y_A = \tan\left[\pi(y_A - c_A^y)/2L\right]$ are "local" Cartesian coordinates in each cubic region.

Testing the Elliptic PDE Solver

- Solve the elliptic PDE, $\nabla^i\nabla_i\psi - c^2\psi = f$ where c^2 is a constant, and f is a given function.
- Use the co-variant derivative ∇_i for the round metric on $S^2\times S^1$:

$$
\begin{aligned}
ds^2 &= R_1^2 d\chi^2 + R_2^2\left(d\theta^2 + \sin^2\theta\, d\varphi^2\right),\\
&= \left(\frac{2\pi R_1}{L}\right)^2 dz_A^2 + \left(\frac{\pi R_2}{2L}\right)^2 \frac{(1+X_A^2)(1+Y_A^2)}{(1+X_A^2+Y_A^2)^2}\\
&\quad\times\left[(1+X_A^2)\,dx_A^2 - 2X_AY_A\,dx_A\,dy_A + (1+Y_A^2)\,dy_A^2\right].
\end{aligned}
$$

where $X_A = \tan\left[\pi(x_A - c_A^x)/2L\right]$ and $Y_A = \tan\left[\pi(y_A - c_A^y)/2L\right]$ are "local" Cartesian coordinates in each cubic region.

- Let $f = -(\omega^2 + c^2)\psi_E$, where $\psi_E = \Re\left[e^{ik\chi}Y_{\ell m}(\theta,\varphi)\right]$. The angles χ, θ and φ are functions of the coordinates x, y and z.

Testing the Elliptic PDE Solver

- Solve the elliptic PDE, $\nabla^i\nabla_i\psi - c^2\psi = f$ where c^2 is a constant, and f is a given function.
- Use the co-variant derivative ∇_i for the round metric on $S^2\times S^1$:

$$\begin{aligned} ds^2 &= R_1^2 d\chi^2 + R_2^2\left(d\theta^2 + \sin^2\theta\, d\varphi^2\right), \\ &= \left(\frac{2\pi R_1}{L}\right)^2 dz_A^2 + \left(\frac{\pi R_2}{2L}\right)^2 \frac{(1+X_A^2)(1+Y_A^2)}{(1+X_A^2+Y_A^2)^2} \\ &\quad \times\Big[(1+X_A^2)\,dx_A^2 - 2X_A Y_A\,dx_A\,dy_A + (1+Y_A^2)\,dy_A^2\Big]. \end{aligned}$$

  where $X_A = \tan\left[\pi(x_A - c_A^x)/2L\right]$ and $Y_A = \tan\left[\pi(y_A - c_A^y)/2L\right]$ are "local" Cartesian coordinates in each cubic region.

- Let $f = -(\omega^2 + c^2)\psi_E$, where $\psi_E = \Re\left[e^{ik\chi}Y_{\ell m}(\theta,\varphi)\right]$. The angles χ, θ and φ are functions of the coordinates x, y and z.
- The unique, exact, analytical solution to this problem is $\psi = \psi_E$, when $\omega^2 = \ell(\ell+1)/R_2^2 + k^2/R_1^2$.

Testing the Elliptic PDE Solver II

- Measure the accuracy of the numerical solution ψ_N as a function of numerical resolution N (grid points per dimension) in two ways:
  - First, with the residual $R_N \equiv \nabla^i\nabla_i\psi_N - c^2\psi_N - f$, and its norm:
$$\mathcal{E}_R = \sqrt{\frac{\int R_N^2\sqrt{g}d^3x}{\int f^2\sqrt{g}d^3x}}.$$
  - Second, with the solution error, $\Delta\psi = \psi_N - \psi_E$, and its norm:
$$\mathcal{E}_\psi = \sqrt{\frac{\int \Delta\psi^2\sqrt{g}d^3x}{\int \psi_E^2\sqrt{g}d^3x}}.$$

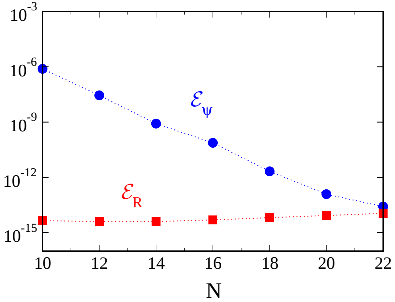
Testing the Elliptic PDE Solver II

- Measure the accuracy of the numerical solution ψ_N as a function of numerical resolution N (grid points per dimension) in two ways:
  - First, with the residual $R_N \equiv \nabla^i\nabla_i\psi_N - c^2\psi_N - f$, and its norm:

$$\mathcal{E}_R = \sqrt{\frac{\int R_N^2\sqrt{g}d^3x}{\int f^2\sqrt{g}d^3x}}.$$

  - Second, with the solution error, $\Delta\psi = \psi_N - \psi_E$, and its norm:

$$\mathcal{E}_\psi = \sqrt{\frac{\int \Delta\psi^2\sqrt{g}d^3x}{\int \psi_E^2\sqrt{g}d^3x}}.$$



- All these numerical tests were performed by implementing the ideas described here into the Spectral Einstein Code (SpEC) developed originally by the Caltech/Cornell numerical relativity collaboration.

Testing the Hyperbolic PDE Solver

- Solve the equation $\partial_t^2\psi = \nabla_i\nabla^i\psi$ with given initial data.
- Convert the second-order equation into an equivalent first-order system: $\partial_t\psi = -\Pi$, $\partial_t\Pi = -\nabla^i\Phi_i$ and $\partial_t\Phi_i = -\nabla_i\Pi$ with constraint $\mathcal{C}_i = \nabla_i\psi - \Phi_i$.

Testing the Hyperbolic PDE Solver

- Solve the equation $\partial_t^2\psi = \nabla_i\nabla^i\psi$ with given initial data.
- Convert the second-order equation into an equivalent first-order system: $\partial_t\psi = -\Pi$, $\partial_t\Pi = -\nabla^i\Phi_i$ and $\partial_t\Phi_i = -\nabla_i\Pi$ with constraint $\mathcal{C}_i = \nabla_i\psi - \Phi_i$.
- Use the co-variant derivative ∇_i for the round metric on S^3:

$$
\begin{aligned}
ds^2 &= R_3^2\left[d\chi^2 + \sin^2\chi\left(d\theta^2 + \sin^2\theta\, d\varphi^2\right)\right], \\
&= \left(\frac{\pi R_3}{2L}\right)^2 \frac{(1+X_A^2)(1+Y_A^2)(1+Z_A^2)}{(1+X_A^2+Y_A^2+Z_A^2)^2}\left[\frac{(1+X_A^2)(1+Y_A^2+Z_A^2)}{(1+Y_A^2)(1+Z_A^2)}dx^2 + \frac{(1+Y_A^2)(1+X_A^2+Z_A^2)}{(1+X_A^2)(1+Z_A^2)}dy^2\right. \\
&\qquad \left. + \frac{(1+Z_A^2)(1+X_A^2+Y_A^2)}{(1+X_A^2)(1+Y_A^2)}dz^2 - \frac{2X_AY_A}{1+Z_A^2}dx\,dy - \frac{2X_AZ_A}{1+Y_A^2}dx\,dz - \frac{2Y_AZ_A}{1+X_A^2}dy\,dz\right].
\end{aligned}
$$

Testing the Hyperbolic PDE Solver

- Solve the equation $\partial_t^2\psi = \nabla_i\nabla^i\psi$ with given initial data.
- Convert the second-order equation into an equivalent first-order system: $\partial_t\psi = -\Pi$, $\partial_t\Pi = -\nabla^i\Phi_i$ and $\partial_t\Phi_i = -\nabla_i\Pi$ with constraint $\mathcal{C}_i = \nabla_i\psi - \Phi_i$.
- Use the co-variant derivative ∇_i for the round metric on S^3:

$$
\begin{aligned}
ds^2 &= R_3^2\left[d\chi^2 + \sin^2\chi\left(d\theta^2 + \sin^2\theta\, d\varphi^2\right)\right],\\
&= \left(\frac{\pi R_3}{2L}\right)^2 \frac{(1+X_A^2)(1+Y_A^2)(1+Z_A^2)}{(1+X_A^2+Y_A^2+Z_A^2)^2}\left[\frac{(1+X_A^2)(1+Y_A^2+Z_A^2)}{(1+Y_A^2)(1+Z_A^2)}dx^2 + \frac{(1+Y_A^2)(1+X_A^2+Z_A^2)}{(1+X_A^2)(1+Z_A^2)}dy^2\right.\\
&\qquad \left. + \frac{(1+Z_A^2)(1+X_A^2+Y_A^2)}{(1+X_A^2)(1+Y_A^2)}dz^2 - \frac{2X_AY_A}{1+Z_A^2}dx\,dy - \frac{2X_AZ_A}{1+Y_A^2}dx\,dz - \frac{2Y_AZ_A}{1+X_A^2}dy\,dz\right].
\end{aligned}
$$

- Choose initial data with $\psi_{t=0} = \Re[Y_{k\ell m}(\chi,\theta,\varphi)]$, $\Pi_{t=0} = -\Re[i\omega Y_{k\ell m}(\chi,\theta,\varphi)]$ and $\Phi_{i\,t=0} = \Re[\nabla_i Y_{k\ell m}(\chi,\theta,\varphi)]$ where $\omega^2 = k(k+2)/R_3^2$.

Testing the Hyperbolic PDE Solver

- Solve the equation $\partial_t^2\psi = \nabla_i\nabla^i\psi$ with given initial data.
- Convert the second-order equation into an equivalent first-order system: $\partial_t\psi = -\Pi$, $\partial_t\Pi = -\nabla^i\Phi_i$ and $\partial_t\Phi_i = -\nabla_i\Pi$ with constraint $\mathcal{C}_i = \nabla_i\psi - \Phi_i$.
- Use the co-variant derivative ∇_i for the round metric on S^3:

$$
\begin{aligned}
ds^2 &= R_3^2\left[d\chi^2 + \sin^2\chi\left(d\theta^2 + \sin^2\theta\, d\varphi^2\right)\right],\\
&= \left(\frac{\pi R_3}{2L}\right)^2 \frac{(1+X_A^2)(1+Y_A^2)(1+Z_A^2)}{(1+X_A^2+Y_A^2+Z_A^2)^2}\left[\frac{(1+X_A^2)(1+Y_A^2+Z_A^2)}{(1+Y_A^2)(1+Z_A^2)}dx^2 + \frac{(1+Y_A^2)(1+X_A^2+Z_A^2)}{(1+X_A^2)(1+Z_A^2)}dy^2\right.\\
&\qquad\left. + \frac{(1+Z_A^2)(1+X_A^2+Y_A^2)}{(1+X_A^2)(1+Y_A^2)}dz^2 - \frac{2X_AY_A}{1+Z_A^2}dx\,dy - \frac{2X_AZ_A}{1+Y_A^2}dx\,dz - \frac{2Y_AZ_A}{1+X_A^2}dy\,dz\right].
\end{aligned}
$$

- Choose initial data with $\psi_{t=0} = \Re[Y_{k\ell m}(\chi,\theta,\varphi)]$, $\Pi_{t=0} = -\Re[i\omega Y_{k\ell m}(\chi,\theta,\varphi)]$ and $\Phi_{i\,t=0} = \Re[\nabla_i Y_{k\ell m}(\chi,\theta,\varphi)]$ where $\omega^2 = k(k+2)/R_3^2$.
- The unique, exact, analytical solution to this problem is $\psi = \psi_E = \Re[e^{i\omega t}Y_{k\ell m}(\chi,\theta,\varphi)]$, $\Pi = -\partial_t\psi_E$, and $\Phi_i = \nabla_i\psi_E$.

Testing the Hyperbolic PDE Solver II

- Measure the accuracy of the numerical solution ψ_N as a function of numerical resolution N (grid points per dimension) in two ways:
  - First, with the solution error, $\Delta\psi = \psi_N - \psi_E$, and its norm:
  $$\mathcal{E}_\psi = \sqrt{\frac{\int \Delta\psi^2 \sqrt{g} d^3x}{\int \psi^2 \sqrt{g} d^3x}}.$$
  - Second, with the constraint error, $\mathcal{C}_i = \Phi_i - \nabla_i\psi$, and its norm:
  $$\mathcal{E}_\mathcal{C} = \sqrt{\frac{\int g^{ij}\mathcal{C}_i\mathcal{C}_j \sqrt{g} d^3x}{\int g^{ij}(\Phi_i\Phi_j + \nabla_i\psi\nabla_j\psi)\sqrt{g} d^3x}}.$$

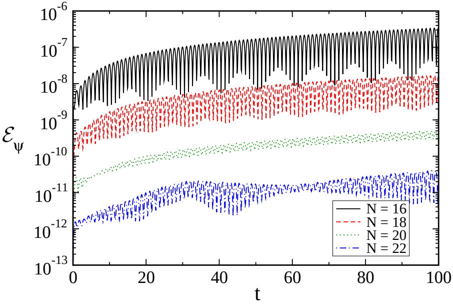
Testing the Hyperbolic PDE Solver II

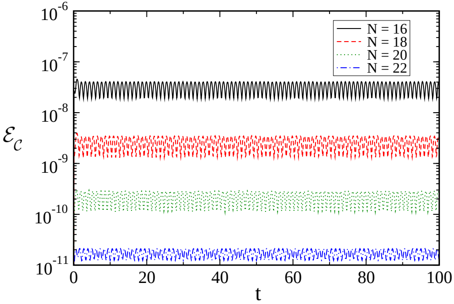
- Measure the accuracy of the numerical solution ψ_N as a function of numerical resolution N (grid points per dimension) in two ways:
  - First, with the solution error, $\Delta\psi = \psi_N - \psi_E$, and its norm:

$$\mathcal{E}_\psi = \sqrt{\frac{\int \Delta\psi^2 \sqrt{g} d^3x}{\int \psi^2 \sqrt{g} d^3x}}.$$

  - Second, with the constraint error, $\mathcal{C}_i = \Phi_i - \nabla_i\psi$, and its norm:

$$\mathcal{E}_\mathcal{C} = \sqrt{\frac{\int g^{ij}\mathcal{C}_i\mathcal{C}_j\sqrt{g} d^3x}{\int g^{ij}(\Phi_i\Phi_j+\nabla_i\psi\nabla_j\psi)\sqrt{g} d^3x}}.$$





Solving Einstein's Equation on Multi-Cube Manifolds

- Multi-cube methods were designed to solve first-order hyperbolic systems, $\partial_t u^\alpha + A^{k\,\alpha}{}_\beta(u)\tilde\nabla_k u^\beta = F^\alpha(u)$, where the dynamical fields u^α are tensors that can be transformed across interface boundaries using the Jacobians $J^{A\alpha i}_{B\beta k}$, etc.

Solving Einstein's Equation on Multi-Cube Manifolds

- Multi-cube methods were designed to solve first-order hyperbolic systems, $\partial_t u^\alpha + A^{k\,\alpha}{}_\beta(u)\tilde\nabla_k u^\beta = F^\alpha(u)$, where the dynamical fields u^α are tensors that can be transformed across interface boundaries using the Jacobians $J^{A\alpha i}_{B\beta k}$, etc.
- The usual first-order representations of Einstein's equation fail to meet these conditions in two important ways:
  - The usual choice of dynamical fields, $u^\alpha = \{\psi_{ab}, \Pi_{ab} = -t^c\partial_c\psi_{ab}, \Phi_{iab} = \partial_i\psi_{ab}\}$ are not tensor fields.
  - The usual first-order evolution equations are not covariant: i.e., the one that comes from the definition of Π_{ab}, $\Pi_{ab} = -t^c\partial_c\psi_{ab}$, and the one that comes from preserving the constraint $C_{iab} = \Phi_{iab} - \partial_i\psi_{ab}$, $t^c\partial_c C_{iab} = -\gamma_2 C_{iab}$.

Solving Einstein's Equation on Multi-Cube Manifolds

- Multi-cube methods were designed to solve first-order hyperbolic systems, $\partial_t u^\alpha + A^{k\,\alpha}{}_\beta(u)\tilde\nabla_k u^\beta = F^\alpha(u)$, where the dynamical fields u^α are tensors that can be transformed across interface boundaries using the Jacobians $J^{A\alpha i}_{B\beta k}$, etc.
- The usual first-order representations of Einstein's equation fail to meet these conditions in two important ways:
  - The usual choice of dynamical fields, $u^\alpha = \{\psi_{ab}, \Pi_{ab} = -t^c\partial_c\psi_{ab}, \Phi_{iab} = \partial_i\psi_{ab}\}$ are not tensor fields.
  - The usual first-order evolution equations are not covariant: i.e., the one that comes from the definition of Π_{ab}, $\Pi_{ab} = -t^c\partial_c\psi_{ab}$, and the one that comes from preserving the constraint $C_{iab} = \Phi_{iab} - \partial_i\psi_{ab}$, $t^c\partial_c C_{iab} = -\gamma_2 C_{iab}$.
- Our attempts to construct the transformations for non-tensor quantities like $\partial_i\psi_{ab}$ and Φ_{iab} across the non-smooth multi-cube interface boundaries failed to result in stable numerical evolutions.
- A spatially covariant first-order representation of the Einstein evolution system seems to be needed.

Covariant Representations of Einstein's Equation

- Let $\tilde{\psi}_{ab}$ denote a smooth reference metric on the manifold $R \times \Sigma$. For convenience we choose $ds^2 = \tilde{\psi}_{ab} dx^a dx^b = -dt^2 + \tilde{g}_{ij} dx^i dx^j$, where $\tilde{g}_{ij}$ is the smooth multi-cube reference three-metric on Σ.

Covariant Representations of Einstein's Equation

- Let $\tilde{\psi}_{ab}$ denote a smooth reference metric on the manifold $R \times \Sigma$. For convenience we choose $ds^2 = \tilde{\psi}_{ab} dx^a dx^b = -dt^2 + \tilde{g}_{ij} dx^i dx^j$, where $\tilde{g}_{ij}$ is the smooth multi-cube reference three-metric on Σ.
- A fully covariant expression for the Ricci tensor can be obtained using the reference covariant derivative $\tilde{\nabla}_a$:

$$\begin{aligned} R_{ab} &= -\tfrac{1}{2}\psi^{cd}\tilde{\nabla}_c\tilde{\nabla}_d\psi_{ab} + \nabla_{(a}\Delta_{b)} - \psi^{cd}\tilde{R}^e{}_{cd(a}\psi_{b)e} \\ &\quad +\psi^{cd}\psi^{ef}\left(\tilde{\nabla}_e\psi_{ca}\tilde{\nabla}_f\psi_{ab} - \Delta_{ace}\Delta_{bdf}\right), \end{aligned}$$

where $\Delta_{abc} = \psi_{ad}\left(\Gamma^d_{bc} - \tilde{\Gamma}^d_{bc}\right)$, and $\Delta_a = \psi^{bc}\Delta_{abc}$.

Covariant Representations of Einstein's Equation

- Let $\tilde\psi_{ab}$ denote a smooth reference metric on the manifold $R\times\Sigma$. For convenience we choose $ds^2=\tilde\psi_{ab}dx^a dx^b=-dt^2+\tilde g_{ij}dx^i dx^j$, where $\tilde g_{ij}$ is the smooth multi-cube reference three-metric on Σ.
- A fully covariant expression for the Ricci tensor can be obtained using the reference covariant derivative $\tilde\nabla_a$:

$$\begin{aligned} R_{ab} &= -\tfrac{1}{2}\psi^{cd}\tilde\nabla_c\tilde\nabla_d\psi_{ab}+\nabla_{(a}\Delta_{b)}-\psi^{cd}\tilde R^e{}_{cd(a}\psi_{b)e} \\ &\quad +\psi^{cd}\psi^{ef}\left(\tilde\nabla_e\psi_{ca}\tilde\nabla_f\psi_{ab}-\Delta_{ace}\Delta_{bdf}\right), \end{aligned}$$

  where $\Delta_{abc}=\psi_{ad}\left(\Gamma^d_{bc}-\tilde\Gamma^d_{bc}\right)$, and $\Delta_a=\psi^{bc}\Delta_{abc}$.
- A fully-covariant manifestly hyperbolic representation of the Einstein equations can be obtained by fixing the gauge with a covariant generalized harmonic condition: $\Delta_a=-H_a(\psi_{cd})$.
- The vacuum Einstein equations then become:

$$\begin{aligned} \psi^{cd}\tilde\nabla_c\tilde\nabla_d\psi_{ab} &= -2\nabla_{(a}H_{b)}+2\psi^{cd}\psi^{ef}\left(\tilde\nabla_e\psi_{ca}\tilde\nabla_f\psi_{ab}-\Delta_{ace}\Delta_{bdf}\right) \\ &\quad -2\psi^{cd}\tilde R^e{}_{cd(a}\psi_{b)e}+\gamma_0\left[2\delta^c_{(a}t_{b)}-\psi_{ab}t^c\right]\left(H_c+\Delta_c\right). \end{aligned}$$

Covariant Representations of Einstein's Equation II

- A first-order representation of the Einstein equations can be obtained from this covariant generalized harmonic representation by choosing dynamical fields:

  $$u^\alpha = \{\psi_{ab}, \Pi_{ab} = -t^c\tilde\nabla_c\psi_{ab}, \Phi_{iab} = \tilde\nabla_i\psi_{ab}\},$$

  which are tensors with respect to spatial coordinate transformations.

Covariant Representations of Einstein's Equation II

- A first-order representation of the Einstein equations can be obtained from this covariant generalized harmonic representation by choosing dynamical fields:
$$u^\alpha = \{\psi_{ab}, \Pi_{ab} = -t^c\tilde\nabla_c\psi_{ab}, \Phi_{iab} = \tilde\nabla_i\psi_{ab}\},$$
which are tensors with respect to spatial coordinate transformations.
- The first order equation that arises from the definition of Π_{ab}, $t^c\tilde\nabla_c\psi_{ab} = -\Pi_{ab}$ is now covariant, as is the equation for $t^c\tilde\nabla_c\Phi_{iab}$ that follows from the covariant constraint evolution equation, $t^c\tilde\nabla_c C_{iab} = -\gamma_2 C_{iab}$, where $C_{iab} = \Phi_{iab} - \tilde\nabla_i\psi_{ab}$.

Covariant Representations of Einstein's Equation II

- A first-order representation of the Einstein equations can be obtained from this covariant generalized harmonic representation by choosing dynamical fields:
$$u^\alpha = \{\psi_{ab}, \Pi_{ab} = -t^c\tilde\nabla_c\psi_{ab}, \Phi_{iab} = \tilde\nabla_i\psi_{ab}\},$$
which are tensors with respect to spatial coordinate transformations.
- The first order equation that arises from the definition of Π_{ab}, $t^c\tilde\nabla_c\psi_{ab} = -\Pi_{ab}$ is now covariant, as is the equation for $t^c\tilde\nabla_c\Phi_{iab}$ that follows from the covariant constraint evolution equation, $t^c\tilde\nabla_c C_{iab} = -\gamma_2 C_{iab}$, where $C_{iab} = \Phi_{iab} - \tilde\nabla_i\psi_{ab}$.
- The resulting first-order Einstein evolution system, $\partial_t u^\alpha + A^{k\,\alpha}{}_\beta(u)\tilde\nabla_k u^\beta = F^\alpha(u)$, is symmetric-hyperbolic and covariant with respect to spatial coordinate transformations.
- The characteristic speeds and fields of this covariant system have the same forms as the standard ones in terms of the dynamical fields ψ_{ab}, Π_{ab} and Φ_{iab}. These fields are now tensors, however, so the actual characteristic fields are somewhat different.

Testing the Einstein Solver: Static Universe on S^3

- Metric initial data is taken from the "Einstein Static Universe" geometry:

$$ds^2 = -dt^2 + R_3^2 \left[d\chi^2 + \sin^2\chi \left(d\theta^2 + \sin^2\theta \, d\varphi^2 \right) \right],$$

Testing the Einstein Solver: Static Universe on S^3

- Metric initial data is taken from the "Einstein Static Universe" geometry:

$$
\begin{aligned}
ds^2 &= -dt^2 + R_3^2\left[d\chi^2 + \sin^2\chi\left(d\theta^2 + \sin^2\theta\, d\varphi^2\right)\right],\\
&= -dt^2 + \left(\frac{\pi R_3}{2L}\right)^2 \frac{(1+X_A^2)(1+Y_A^2)(1+Z_A^2)}{(1+X_A^2+Y_A^2+Z_A^2)^2}\left[\frac{(1+X_A^2)(1+Y_A^2+Z_A^2)}{(1+Y_A^2)(1+Z_A^2)}dx^2 + \frac{(1+Y_A^2)(1+X_A^2+Z_A^2)}{(1+X_A^2)(1+Z_A^2)}dy^2\right.\\
&\qquad \left. + \frac{(1+Z_A^2)(1+X_A^2+Y_A^2)}{(1+X_A^2)(1+Y_A^2)}dz^2 - \frac{2X_AY_A}{1+Z_A^2}dx\,dy - \frac{2X_AZ_A}{1+Y_A^2}dx\,dz - \frac{2Y_AZ_A}{1+X_A^2}dy\,dz\right].
\end{aligned}
$$

Testing the Einstein Solver: Static Universe on S^3

- Metric initial data is taken from the "Einstein Static Universe" geometry:

$$
\begin{aligned}
ds^2 &= -dt^2 + R_3^2\left[d\chi^2 + \sin^2\chi\left(d\theta^2 + \sin^2\theta\, d\varphi^2\right)\right], \\
&= -dt^2 + \left(\frac{\pi R_3}{2L}\right)^2 \frac{(1+X_A^2)(1+Y_A^2)(1+Z_A^2)}{(1+X_A^2+Y_A^2+Z_A^2)^2}\left[\frac{(1+X_A^2)(1+Y_A^2+Z_A^2)}{(1+Y_A^2)(1+Z_A^2)}dx^2 + \frac{(1+Y_A^2)(1+X_A^2+Z_A^2)}{(1+X_A^2)(1+Z_A^2)}dy^2\right. \\
&\qquad \left. + \frac{(1+Z_A^2)(1+X_A^2+Y_A^2)}{(1+X_A^2)(1+Y_A^2)}dz^2 - \frac{2X_AY_A}{1+Z_A^2}dx\,dy - \frac{2X_AZ_A}{1+Y_A^2}dx\,dz - \frac{2Y_AZ_A}{1+X_A^2}dy\,dz\right].
\end{aligned}
$$

- This metric solves Einstein's equation with cosmological constant and complex scalar field source on a manifold with spatial topology S^3.

Testing the Einstein Solver: Static Universe on S^3

- Metric initial data is taken from the “Einstein Static Universe” geometry:

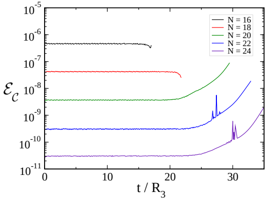
$$
\begin{aligned}
ds^2 &= -dt^2 + R_3^2\left[d\chi^2 + \sin^2\chi\left(d\theta^2 + \sin^2\theta\, d\varphi^2\right)\right],\\
&= -dt^2 + \left(\frac{\pi R_3}{2L}\right)^2 \frac{(1+X_A^2)(1+Y_A^2)(1+Z_A^2)}{(1+X_A^2+Y_A^2+Z_A^2)^2}\left[\frac{(1+X_A^2)(1+Y_A^2+Z_A^2)}{(1+Y_A^2)(1+Z_A^2)}dx^2 + \frac{(1+Y_A^2)(1+X_A^2+Z_A^2)}{(1+X_A^2)(1+Z_A^2)}dy^2\right.\\
&\qquad\left. + \frac{(1+Z_A^2)(1+X_A^2+Y_A^2)}{(1+X_A^2)(1+Y_A^2)}dz^2 - \frac{2X_AY_A}{1+Z_A^2}dx\,dy - \frac{2X_AZ_A}{1+Y_A^2}dx\,dz - \frac{2Y_AZ_A}{1+X_A^2}dy\,dz\right].
\end{aligned}
$$

- This metric solves Einstein’s equation with cosmological constant and complex scalar field source on a manifold with spatial topology S^3.
- Evolution of these initial data is the static universe geometry, if the cosmological constant is chosen to be $\Lambda = 1/R_3^2$, and the complex scalar field is $\varphi = \varphi_0 e^{i\mu t}$ with $\mu^2|\varphi_0|^2 = 1/4\pi R_3^2$.

Testing the Einstein Solver: Static Universe on S^3 II

- Monitor how well the numerical solutions satisfy the Einstein system by evaluating the norm of the various constraints:

$$\mathcal{E}_{\mathcal{C}} = \sqrt{\frac{\int \sum |\mathcal{C}|^2 \sqrt{g} d^3x}{\int \sum |\partial_i u|^2 \sqrt{g} d^3x}}.$$

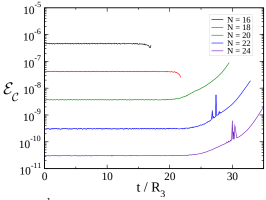


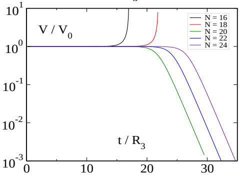
Testing the Einstein Solver: Static Universe on S^3 II

- Monitor how well the numerical solutions satisfy the Einstein system by evaluating the norm of the various constraints:

$$\mathcal{E}_{\mathcal{C}} = \sqrt{\frac{\int \sum |\mathcal{C}|^2 \sqrt{g} d^3x}{\int \sum |\partial_i u|^2 \sqrt{g} d^3x}}.$$

- Monitor the physical volume V of the S^3 in comparison with the Einstein Static Solution value $V_0 = 2\pi^2 R_3^3$:



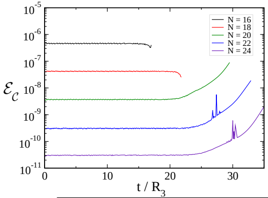


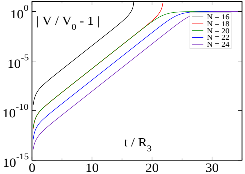
Testing the Einstein Solver: Static Universe on S^3 II

- Monitor how well the numerical solutions satisfy the Einstein system by evaluating the norm of the various constraints:

$$\mathcal{E}_{\mathcal{C}} = \sqrt{\frac{\int \sum |\mathcal{C}|^2 \sqrt{g} d^3x}{\int \sum |\partial_i u|^2 \sqrt{g} d^3x}}.$$

- Monitor the physical volume V of the S^3 in comparison with the Einstein Static Solution value $V_0 = 2\pi^2 R_3^3$:

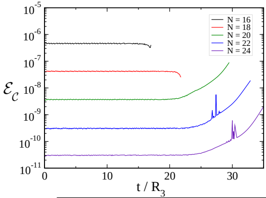




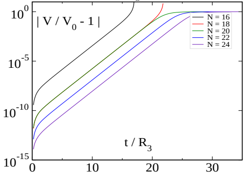
Testing the Einstein Solver: Static Universe on S^3 II

- Monitor how well the numerical solutions satisfy the Einstein system by evaluating the norm of the various constraints:

$$\mathcal{E}_\mathcal{C} = \sqrt{\frac{\int \sum |\mathcal{C}|^2 \sqrt{g} d^3x}{\int \sum |\partial_i u|^2 \sqrt{g} d^3x}}.$$

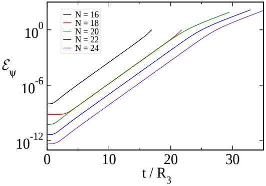


- Monitor the physical volume V of the S^3 in comparison with the Einstein Static Solution value $V_0 = 2\pi^2 R_3^3$:



- Monitor the accuracy of numerical metric solution by evaluating the norm of its error, $\Delta\psi_{ab} = \psi_{Nab} - \psi_{Aab}$:

$$\mathcal{E}_\psi = \sqrt{\frac{\int \sum_{ab} |\Delta\psi_{ab}|^2 \sqrt{g} d^3x}{\int \sum_{ab} |\psi_{ab}|^2 \sqrt{g} d^3x}}.$$



Testing the Einstein Solver: Static Universe on S^3 III

- The constraints are well satisfied for $t \lesssim 20$, so these are good (approximate) solutions for early times.

Testing the Einstein Solver: Static Universe on S^3 III

- The constraints are well satisfied for $t \lesssim 20$, so these are good (approximate) solutions for early times.
- The static Einstein-Klein-Gordon solution has unstable $k = 0$ and $k = 1$ modes with frequencies given by

$$\begin{aligned} \omega_0^2 R_3^2 &= 2\mu^2 R_3^2 - 2 - 2\sqrt{\mu^4 R_3^4 - \mu^2 R_3^2 + 1}, \\ \omega_1 R_3 &= \frac{i}{2}\left(\mu_G R_3 - \sqrt{4 + \mu_G^2 R_3^2}\right). \end{aligned}$$

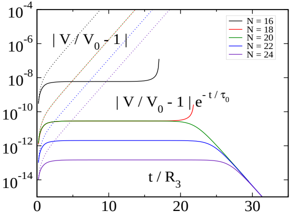
  For the mass and radius parameters used in these simulations $1/\tau_0 \equiv |\omega_0| \approx 1.100$ and $1/\tau_1 \equiv |\omega_1| \approx 0.618$.

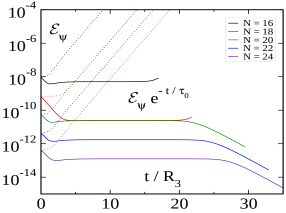
Testing the Einstein Solver: Static Universe on S^3 III

- The constraints are well satisfied for $t \lesssim 20$, so these are good (approximate) solutions for early times.
- The static Einstein-Klein-Gordon solution has unstable $k = 0$ and $k = 1$ modes with frequencies given by

$$\begin{aligned}
\omega_0^2 R_3^2 &= 2\mu^2 R_3^2 - 2 - 2\sqrt{\mu^4 R_3^4 - \mu^2 R_3^2 + 1}, \\
\omega_1 R_3 &= \frac{i}{2}\left(\mu_G R_3 - \sqrt{4 + \mu_G^2 R_3^2}\right).
\end{aligned}$$

  For the mass and radius parameters used in these simulations $1/\tau_0 \equiv |\omega_0| \approx 1.100$ and $1/\tau_1 \equiv |\omega_1| \approx 0.618$.





Testing the Einstein Solver: Static Universe on S^3 IV

- Test long term stability by damping out the unstable modes while leaving the other parts of the dynamics untouched.

Testing the Einstein Solver: Static Universe on S^3 IV

- Test long term stability by damping out the unstable modes while leaving the other parts of the dynamics untouched.
- Define the spherical harmonic projection $\bar{Q}^{k\ell m}$ of a quantity Q by
$$\bar{Q}^{k\ell m} = \int Y^{*k\ell m} Q \sqrt{\tilde{g}}\, d^3x.$$

Testing the Einstein Solver: Static Universe on S^3 IV

- Test long term stability by damping out the unstable modes while leaving the other parts of the dynamics untouched.
- Define the spherical harmonic projection $\bar{Q}^{k\ell m}$ of a quantity Q by $\bar{Q}^{k\ell m} = \int Y^{*k\ell m} Q\sqrt{\tilde{g}}\, d^3x$.
- Modify the Einstein-Klein-Gordon evolution system by adding terms that damp out this particular perturbation, e.g.

$$\partial_t \psi_{tt} \quad = \quad f_{tt} \quad - \quad \left[\bar{f}_{tt}^{k\ell m} + \eta \bar{\psi}_{tt}^{k\ell m}\right] Y^{k\ell m}.$$

Testing the Einstein Solver: Static Universe on S^3 IV

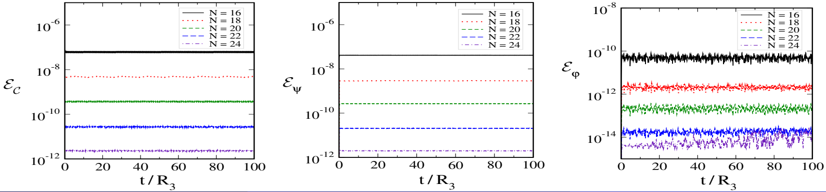
- Test long term stability by damping out the unstable modes while leaving the other parts of the dynamics untouched.
- Define the spherical harmonic projection $\bar{Q}^{k\ell m}$ of a quantity Q by $\bar{Q}^{k\ell m} = \int Y^{*k\ell m} Q \sqrt{\tilde{g}}\, d^3x$.
- Modify the Einstein-Klein-Gordon evolution system by adding terms that damp out this particular perturbation, e.g.

$$\partial_t \psi_{tt} \;=\; f_{tt} \;-\; \left[\bar{f}^{k\ell m}_{tt} + \eta \bar{\psi}^{k\ell m}_{tt}\right] Y^{k\ell m}.$$

- Check this equation by multiplying the modified evolution equations by $Y^{*k\ell m}$ and integrating to obtain the modified evolution of the damped mode, e.g., for the ψ_{tt} equation you get: $\partial_t \bar{\psi}^{k\ell m}_{tt} = -\eta \bar{\psi}^{k\ell m}_{tt}$.

Testing the Einstein Solver: Static Universe on S^3 IV

- Test long term stability by damping out the unstable modes while leaving the other parts of the dynamics untouched.
- Define the spherical harmonic projection $\bar{Q}^{k\ell m}$ of a quantity Q by
$$\bar{Q}^{k\ell m} = \int Y^{*k\ell m} Q \sqrt{\tilde{g}}\, d^3x.$$
- Modify the Einstein-Klein-Gordon evolution system by adding terms that damp out this particular perturbation, e.g.
$$\partial_t \psi_{tt} \quad = \quad f_{tt} \quad - \quad \left[\bar{f}_{tt}^{k\ell m} + \eta \bar{\psi}_{tt}^{k\ell m}\right] Y^{k\ell m}.$$
- Check this equation by multiplying the modified evolution equations by $Y^{*k\ell m}$ and integrating to obtain the modified evolution of the damped mode, e.g., for the ψ_{tt} equation you get: $\partial_t \bar{\psi}_{tt}^{k\ell m} = -\eta \bar{\psi}_{tt}^{k\ell m}$.



Testing the Einstein Solver: Perturbed Static S^3

- Find the normal modes of the perturbed Einstein-Klein-Gordon system analytically, e.g., $\delta\psi_{tt} = \Re\left(A_{tt} Y^{k\ell m} e^{i\omega t}\right)$, ...

Testing the Einstein Solver: Perturbed Static S^3

- Find the normal modes of the perturbed Einstein-Klein-Gordon system analytically, e.g., $\delta\psi_{tt} = \Re\left(A_{tt} Y^{k\ell m} e^{i\omega t}\right)$, ...
- The frequencies of the "scalar" modes of this system for $k \geq 2$ are given by $\omega_0^2 R_3^2 = k(k+2)$ and

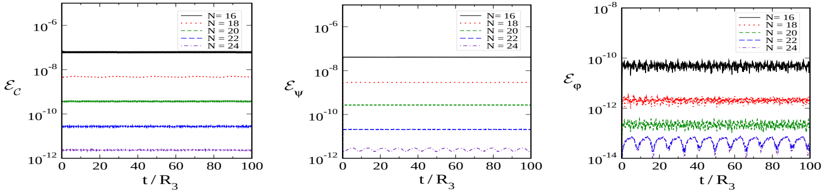
$$\begin{aligned} \omega_\pm^2 R_3^2 &= k(k+2) + 2(\mu^2 R_3^2 - 1) \\ &\qquad \pm\sqrt{(\mu^2 R_3^2 - 1)^2 + [k(k+2)+1]\mu^2 R_3^2}. \end{aligned}$$

Testing the Einstein Solver: Perturbed Static S^3

- Find the normal modes of the perturbed Einstein-Klein-Gordon system analytically, e.g., $\delta\psi_{tt} = \Re\left(A_{tt} Y^{k\ell m} e^{i\omega t}\right)$, ...
- The frequencies of the "scalar" modes of this system for $k \geq 2$ are given by $\omega_0^2 R_3^2 = k(k+2)$ and

$$\begin{aligned}\omega_\pm^2 R_3^2 \;=\; & k(k+2) + 2(\mu^2 R_3^2 - 1)\\ & \pm\sqrt{(\mu^2 R_3^2 - 1)^2 + [k(k+2)+1]\mu^2 R_3^2}.\end{aligned}$$

- Evolve initial data constructed from three superimposed normal modes (one from each frequency class ω_0 and $\omega_\pm$).
- Compare non-linear evolution with analytical perturbation solution:

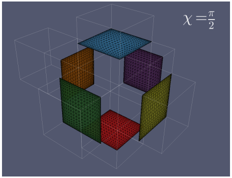


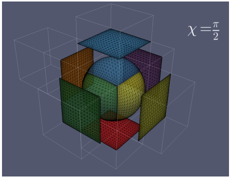
Testing the Einstein Solver: Perturbed Static S^3

- Find the normal modes of the perturbed Einstein-Klein-Gordon system analytically, e.g., $\delta\psi_{tt} = \Re\left(A_{tt} Y^{k\ell m} e^{i\omega t}\right)$, ...
- The frequencies of the "scalar" modes of this system for $k \geq 2$ are given by $\omega_0^2 R_3^2 = k(k+2)$ and

$$\begin{aligned}\omega_\pm^2 R_3^2 &= k(k+2) + 2(\mu^2 R_3^2 - 1) \\ &\qquad \pm\sqrt{(\mu^2 R_3^2 - 1)^2 + [k(k+2)+1]\mu^2 R_3^2}.\end{aligned}$$

- Evolve initial data constructed from fifteen superimposed normal modes (with modes from each frequency class ω_0 and $\omega_\pm$).
- Perturbed S^3 Evolution Movie.





Summary

- We have developed a simple and flexible multi-cube numerical method for solving partial differential equations on manifolds with arbitrary spatial topologies.

Summary

- We have developed a simple and flexible multi-cube numerical method for solving partial differential equations on manifolds with arbitrary spatial topologies.
- Each new topology requires:
  - A multi-cube representation of the topology, i.e. a list of cubic regions and a list of boundary identification maps.
  - A smooth reference metric $\tilde{g}_{ab}$ to define the global differential structure on this multi-cube representation of the manifold.

Summary

- We have developed a simple and flexible multi-cube numerical method for solving partial differential equations on manifolds with arbitrary spatial topologies.
- Each new topology requires:
  - A multi-cube representation of the topology, i.e. a list of cubic regions and a list of boundary identification maps.
  - A smooth reference metric $\tilde{g}_{ab}$ to define the global differential structure on this multi-cube representation of the manifold.
- These methods have been tested by solving simple elliptic and hyperbolic equations on several compact manifolds.

Summary

- We have developed a simple and flexible multi-cube numerical method for solving partial differential equations on manifolds with arbitrary spatial topologies.
- Each new topology requires:
  - A multi-cube representation of the topology, i.e. a list of cubic regions and a list of boundary identification maps.
  - A smooth reference metric $\tilde{g}_{ab}$ to define the global differential structure on this multi-cube representation of the manifold.
- These methods have been tested by solving simple elliptic and hyperbolic equations on several compact manifolds.
- A first-order symmetric-hyperbolic representation of the generalized harmonic Einstein evolution equations has been constructed that is covariant with respect to general spatial coordinate transformations.

Summary

- We have developed a simple and flexible multi-cube numerical method for solving partial differential equations on manifolds with arbitrary spatial topologies.
- Each new topology requires:
  - A multi-cube representation of the topology, i.e. a list of cubic regions and a list of boundary identification maps.
  - A smooth reference metric $\tilde{g}_{ab}$ to define the global differential structure on this multi-cube representation of the manifold.
- These methods have been tested by solving simple elliptic and hyperbolic equations on several compact manifolds.
- A first-order symmetric-hyperbolic representation of the generalized harmonic Einstein evolution equations has been constructed that is covariant with respect to general spatial coordinate transformations.
- These methods have been tested successfully for Einstein evolutions by finding simple solutions numerically on compact manifolds using our new covariant Einstein evolution system.