NRPy+: Python-based C code generation framework for NR

Tensorial expressions in Einstein-like notation ⇒ Highly optimized C-code kernels (with FDs)

This talk (Google Slides): https://tinyurl.com/icermzach

BlackHoles@Home
Status Report
Zach Etienne

https://nrpyplus.net
https://github.com/zachetienne/nrpytutorial

BlackHoles@Home aims to fit numerical-relativity-based binary black hole (BBH) calculations on a consumer-grade desktop computer, enabling gravitational waveform follow-ups and catalogs at unprecedentedly large scales using volunteer computers.
Importance of modeling gravitational wave and multimessenger sources

- Example: LIGO detects a gravitational wave from a black hole or neutron star binary
Importance of modeling gravitational wave and multimessenger sources

- $1B+ Question: What *exactly* caused this and *how*?
  - Answer can provide deep insights into extreme gravity & extreme matter, testing theories beyond current limits
  - To advance science, must compare observations with theoretical predictions
    - Theoretical predictions need to span observ. & theor. uncertainties
Anatomy of a Binary Black Hole Merger, as seen in gravitational waves

- Gravitational-wave driven “Relativistic death spiral”
Anatomy of a Binary Black Hole Merger, as seen in gravitational waves

- Gravitational-wave driven “Relativistic death spiral”

Time axis ➞
(spans ~200ms)
Wave amplitude ⇑
(wave strain, arb. units)
Anatomy of a Binary Black Hole Merger, as seen in gravitational waves

- Gravitational-wave driven “Relativistic death spiral”

These waves encode info about masses, spins, and eccentricity of orbiting black holes

Time axis ➡ (spans ~200ms)
Wave amplitude ⬆️ (wave strain, arb. units)
Anatomy of a Binary Black Hole Merger, as seen in gravitational waves

- Gravitational-wave driven "Relativistic death spiral"

(Very) early inspiral:
Perturbative solutions to Einstein gravity (GR)

Time axis ➤
(spans ~200ms)
Wave amplitude ➪
(wave strain, arb. units)
Anatomy of a Binary Black Hole Merger, as seen in gravitational waves

- Gravitational-wave driven "Relativistic death spiral"

(Very) early inspiral: Perturbative solutions to Einstein gravity (GR)

Late inspiral: Perturb. theory breaks down; Only full GR solutions

Time axis ➩
(spas ~200ms)
Wave amplitude ⇧
(wave strain, arb. units)
Modeling Challenges

Reformulate Einstein’s theory of gravity for the computer

1. Stability, even when simulating BHs
2. Reliability: numerical errors small and well-understood
Modeling Challenges

Reformulate Einstein’s theory of gravity for the computer

1. **Stability**, even when simulating BHs
2. **Reliability**: numerical errors small and well-understood

GR Equations are complex;
Solving the 2-body problem
(two orbiting point masses) in
GR took 90 years (1915-2005)
Modeling Challenges

Reformulate Einstein’s theory of gravity for the computer
1. **Stability**, even when simulating BHs
2. **Reliability**: numerical errors small and well-understood

GR Equations are complex;
Solving the 2-body problem (two orbiting point masses) in GR took 90 years (1915-2005)

\[ G_{\mu\nu} = 8\pi T_{\mu\nu} \]
Modeling Challenges

Reformulate Einstein’s theory of gravity for the computer

1. **Stability**, even when simulating BHs
2. **Reliability**: numerical errors small and well-understood

GR Equations are complex; Solving the 2-body problem (two orbiting point masses) in GR took 90 years (1915-2005)

Most popular formulation

\[ G_{\mu\nu} = 8\pi T_{\mu\nu} \]

\[
\begin{align*}
\partial_t \tilde{g}_{ij} &= [\beta^k \partial_k \tilde{g}_{ij} + \partial_i \beta^k \tilde{g}_{kj} + \partial_j \beta^k \tilde{g}_{ik}] + \frac{2}{3} \tilde{g}_{ij} \left( \alpha \tilde{A}_k - D_k \beta^k \right) - 2\alpha \tilde{A}_{ij} , \\
\partial_t \tilde{A}_{ij} &= [\beta^k \partial_k \tilde{A}_{ij} + \partial_i \beta^k \tilde{A}_{kj} + \partial_j \beta^k \tilde{A}_{ik}] - \frac{2}{3} \tilde{A}_{ij} D_k \beta^k - 2\alpha \tilde{A}_{ik} \tilde{A}^k_j + \alpha \tilde{A}_{ij} K \\
&\quad + e^{-4\phi} \left\{ -2\alpha \tilde{D}_i \tilde{D}_j \phi + 4\alpha \tilde{D}_i \phi \tilde{D}_j \phi + 4 \tilde{D}_i \alpha \tilde{D}_j \phi - \tilde{D}_i \tilde{D}_j \alpha + \alpha \tilde{R}_{ij} \right\}^{TF} , \\
\partial_t \phi &= [\beta^k \partial_k \phi] + \frac{1}{6} (\tilde{D}_k \beta^k - \alpha K) , \\
\partial_t K &= [\beta^k \partial_k K] + \frac{1}{3} \alpha K^2 + \alpha \tilde{A}_{ij} \tilde{A}^{ij} - e^{-4\phi} \left( \tilde{D}_i \tilde{D}_j \alpha + 2\tilde{D}_i \alpha \tilde{D}_j \phi \right) , \\
\partial_t \tilde{A}^i &= \left[ \beta^k \partial_k \tilde{A}^i - \partial_k \beta^i \tilde{A}^k \right] + \tilde{g}^{jk} \tilde{D}_j \tilde{D}_k \beta^i + \frac{2}{3} \Delta^i \tilde{D}_j \beta^j + \frac{1}{3} \tilde{D}^i \tilde{D}_j \beta^j \\
&\quad - 2\tilde{A}^{ij} (\partial_j \alpha - 6\partial_j \phi) + 2\alpha \tilde{A}^j \Delta^i_{jk} - \frac{4}{3} \alpha \tilde{g}^{ij} \partial_j K \\
\partial_t \alpha &= [\beta^j \partial_j \alpha] - 2\alpha K \\
\partial_t \beta^i &= [\beta^j \partial_j \beta^i] + B^i \\
\partial_t B^i &= [\beta^j \partial_j B^i] + \frac{3}{4} \partial_0 \tilde{A}^i - \eta B^i
\end{align*}
\]
NRPy+: Automated code generation for numerical relativity

Most popular formulation
NRPy+: Automated code generation for numerical relativity

How to code this up?!
NRPy+: Automated code generation for numerical relativity

Tensorial expressions in Einstein-like notation ⇒ Highly optimized C-code kernels (with FDs)

How to code this up?!
NRPy+: Automated code generation for numerical relativity

\[
\begin{align*}
\partial_i \tilde{\gamma}_{ij} &= [\beta^k \partial_k \tilde{\gamma}_{ij} + \partial_i \beta^k \tilde{\gamma}_{kj} + \partial_j \beta^k \tilde{\gamma}_{ik}] + \frac{2}{3} \tilde{\gamma}_{ij} \left( \alpha \tilde{\Lambda}^k_k - \tilde{D}_k \beta^k \right) - 2a \tilde{\Lambda}_{ij} , \\
\partial_i \tilde{\Lambda}_{ij} &= [\beta^k \partial_k \tilde{\Lambda}_{ij} + \partial_i \beta^k \tilde{\Lambda}_{kj} + \partial_j \beta^k \tilde{\Lambda}_{ik}] - \frac{2}{3} \tilde{\Lambda}_{ij} \tilde{D}_k \beta^k - 2a \tilde{\Lambda}_{ik} \tilde{\Lambda}^k_j + a \tilde{\Lambda}_{ij} K \\
&+ e^{-\phi} \left\{ -2a \tilde{D}_j \tilde{D}_j \phi + 4a \tilde{D}_i \phi \tilde{D}_j \phi + 4 \tilde{D}_i \alpha \tilde{D}_j \phi - \tilde{D}_i \tilde{D}_j \alpha + a \tilde{R}_{ij} \right\}^{TF} , \\
\partial_i \phi &= [\beta^k \partial_k \phi] + \frac{1}{6} \left( \tilde{D}_k \beta^k - aK \right) , \\
\partial_i K &= [\beta^k \partial_k K] + \frac{1}{3} aK^2 + a \tilde{\Lambda}_{ij} \tilde{\Lambda}^{ij} - e^{-\phi} \left( \tilde{D}_i \tilde{D}_j \alpha + 2 \tilde{D}_i \alpha \tilde{D}_j \phi \right) , \\
\partial_i \tilde{\Lambda}^i &= \left[ \beta^k \partial_k \tilde{\Lambda}^i - \partial_k \beta^i \tilde{\Lambda}^k \right] + \tilde{\gamma}^{jk} \tilde{D}_j \partial_k \beta^i + \frac{2}{3} \Delta^i \tilde{D}_j \beta^i + \frac{1}{3} \tilde{D}_j \tilde{D}_j \beta^i \\
&- 2 \tilde{\Lambda}^{ij} (\partial_j \alpha - 6 \partial_j \phi) + 2a \tilde{\Lambda}^{jk} \Delta^i_{jk} - \frac{4}{3} \tilde{\gamma}^{ij} \partial_i K \\
\partial_i \alpha &= [\beta^i \partial_i \alpha] - 2aK \\
\partial_i \beta^i &= [\beta^i \partial_i \beta^i] + B' \\
\partial_i B^i &= [\beta^i \partial_i B^i] + \frac{3}{4} \partial_0 \tilde{\Lambda}^i - \eta B' ,
\end{align*}
\]

How to code this up?!

NRPy+: Python-based C code generation framework for NR

Tensorial expressions in Einstein-like notation ⇒ Highly optimized C-code kernels (with FDs)

"Nerpy", the NRPy+ mascot. Photo CC2.0 Pacific Environment (modified).

https://nrpyplus.net
https://github.com/zachetienne/nrpytutorial

ICERM NRPy+ tutorial
Thursday, Oct 15; time TBD
Modeling Challenges

Reformulate Einstein’s theory of gravity for the computer

1. **Stability**, even when simulating BHs
2. **Reliability**: numerical errors small and well-understood
Reformulate Einstein’s theory of gravity for the computer

1. **Stability**, even when simulating BHs
2. **Reliability**: numerical errors small and well-understood

Address (~5 orders of mag) disparity in physical scales

1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales

1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)
Address (~5 orders of mag) disparity in physical scales
1. Resolve *sharp*, rapidly changing grav fields near BHs and NSs
2. Model *long-wavelength* gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales
1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales
1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

- Simulations performed on **numerical grids**
  - Numerical solution stored at each grid point
  - Need **denser** grids to model **sharper** features
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales
1. Resolve sharp, rapidly changing grav fields near BHs and NSs
2. Model long-wavelength gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

- Simulations performed on numerical grids
  - Numerical solution stored at each grid point
  - Need denser grids to model sharper features

Fewer grid points = lower computational cost

Less computational cost unlocks
- More simulations, and/or
- More physical realism
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales
1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

- Simulations performed on **numerical grids**
  - Numerical solution stored at each grid point
  - Need **denser** grids to model **sharper** features

Fewer grid points = **lower computational cost**

Less computational cost unlocks
- More simulations, and/or
- More physical realism
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales
1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

**AMR**
Adaptive Mesh Refinement
(Most Popular Method in NR)
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales

1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

**AMR**

*Adaptive Mesh Refinement*  
*(Most Popular Method in NR)*

Resolve disparate lengthscales with nested Cartesian cubes of differing grid spacings
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales

1. Resolve sharp, rapidly changing grav fields near BHs and NSs
2. Model long-wavelength gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

AMR
Adaptive Mesh Refinement (Most Popular Method in NR)
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales
1. Resolve sharp, rapidly changing grav fields near BHs and NSs
2. Model long-wavelength gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

AMR
Adaptive Mesh Refinement (Most Popular Method in NR)
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales

1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

**AMR**

*Adaptive Mesh Refinement (Most Popular Method in NR)*

Resolution highest where fields are sharpest -- near BHs for example
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales

1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

**AMR**

*Adaptive Mesh Refinement (Most Popular Method in NR)*

Resolution lower where gravitational waves modeled
Address (~5 orders of mag) disparity in physical scales

1. Resolve sharp, rapidly changing grav fields near BHs and NSs
2. Model long-wavelength gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

AMR
Adaptive Mesh Refinement
(Most Popular Method in NR)

Resolution lower where gravitational waves modeled
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales

1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

**AMR**

Adaptive Mesh Refinement
(Most Popular Method in NR)

Resolution lower where gravitational waves modeled
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales

1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

**AMR**

*Adaptive Mesh Refinement (Most Popular Method in NR)*

Add coarser grids to push outer boundary far away
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales

1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

**AMR**

*Adaptive Mesh Refinement (Most Popular Method in NR)*

Add coarser grids to push outer boundary far away
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales

1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

AMR
Adaptive Mesh Refinement
(Most Popular Method in NR)
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales
1. Resolve \textbf{sharp}, rapidly changing grav fields near BHs and NSs
2. Model \textbf{long-wavelength} gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

\textbf{AMR}
Adaptive Mesh Refinement
(Most Popular Method in NR)
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales

1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

What’s the problem?
A: Inefficient!
⇒ greater comp. cost

**AMR**
Adaptive Mesh Refinement
(Most Popular Method in NR)
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales
1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

**AMR Grid**

**Inefficiencies**
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales
1. Resolve sharp, rapidly changing grav fields near BHs and NSs
2. Model long-wavelength gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

AMR Grid Inefficiencies

1. Black holes & gravitational waves:
   nearly spherical/axisymmetric
   ○ grav fields vary most strongly in radial direction
   ➔ need high sampling in only one (r) direction
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales

1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

AMR Grid

Inefficiencies

1. Black holes & gravitational waves:
   - nearly spherical/axisymmetric
   - grav fields vary most strongly in *radial* direction
   - need high sampling in *only one* ($r$) direction

2. Gravitational fields are mostly smooth
   - Cartesian AMR grids:
     - 2x jumps in resolution between boxes
     - Boxes have sharp corners
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales
1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

**AMR Grids**
Adaptive Mesh Refinement (Most Popular Method in NR)

**New BiSphere Grids**
~20x more efficient sampling for compact binary simulations
Modeling Challenges

Address (~5 orders of mag) disparity in physical scales

1. Resolve **sharp**, rapidly changing grav fields near BHs and NSs
2. Model **long-wavelength** gravitational waves far away
3. Push outer boundary very far away (due to approx. BCs)

AMR Grids

*Adaptive Mesh Refinement (Most Popular Method in NR)*

- Exploits near-symmetries (~5x)
- Smooth transitions in resolution (~4x)

New BiSphere Grids

~20x more efficient sampling for compact binary simulations
BiSphere grids: Two overlapping numerical grids in Spherical coordinates
BiSphere Challenges

BiSphere grids: Two overlapping numerical grids in Spherical coordinates

- Challenge #1: Spherical coordinates have coordinate singularities
  - Tensors and vectors diverge or go to zero → numerically unstable
**BiSphere grids**: Two overlapping numerical grids in Spherical coordinates

- **Challenge #1**: Spherical coordinates have coordinate singularities
  - Tensors and vectors diverge or go to zero -> numerically **unstable**
- **Idea**: *Scale out singular parts*, (treat singular & nonsingular separately)
BiSphere Challenges

**BiSphere grids**: Two overlapping numerical grids in **Spherical** coordinates

- **Challenge #1**: **Spherical** coordinates have **coordinate singularities**
  - Tensors and vectors diverge or go to zero -> numerically **unstable**
- **Idea**: *Scale out singular parts*, (treat singular & nonsingular separately)
- **Result**: Numerical stability & robustness **on par with Cartesian**
  - **Ordinary spherical polar**: *done!*
    Baumgarte, Montero, Cordero-Carrión, Müller (PRD 87, 044026, 2012),
    built upon covariant BSSN formulation of Brown (PRD 79, 104029, 2009)
  - **Generic-radius spherical polar (incl. log-radial)**: *done!*
    Ruchlin, Etienne, Baumgarte (PRD 97, 064036, 2018)
### BiSphere Challenges

**BiSphere grids**: Two overlapping numerical grids in **Spherical coordinates**

- **Challenge #1**: **Spherical** coordinates have **coordinate singularities**
  - Tensors and vectors diverge or go to zero -> numerically **unstable**
- **Idea**: *Scale out singular parts*, (treat singular & nonsingular separately)
- **Result**: Numerical stability & robustness *on par with Cartesian*
  a. **Ordinary spherical polar**: *done!*
     - Baumgarte, Montero, Cordero-Carrión, Müller (PRD 87, 044026, 2012),
       built upon covariant BSSN formulation of Brown (PRD 79, 104029, 2009)
  b. **Generic-radius spherical polar (incl. log-radial)**: *done!*
     - Ruchlin, Etienne, Baumgarte (PRD 97, 064036, 2018)

- **Challenge #2**: Develop new algorithm for
  - Interpolating between the two spherical grids;
    co-orbit grids with the binary system:
  - Changing basis between the two spherical grids:
BiSphere Challenges

**BiSphere grids:** Two overlapping numerical grids in **Spherical** coordinates

- **Challenge #1:** **Spherical** coordinates have **coordinate singularities**
  - Tensors and vectors diverge or go to zero -> numerically **unstable**
- **Idea:** **Scale out singular parts,** (treat singular & nonsingular separately)
- **Result:** Numerical stability & robustness **on par with Cartesian**
  - **Ordinary spherical polar:** **done!**
    - Baumgarte, Montero, Cordero-Carrión, Müller (PRD 87, 044026, 2012),
      built upon covariant BSSN formulation of Brown (PRD 79, 104029, 2009)
  - **Generic-radius spherical polar (incl. log-radial):** **done!**
    - Ruchlin, **Etienne,** Baumgarte (PRD 97, 064036, 2018)

- **Challenge #2:** Develop new algorithm for
  - **Interpolating** between the two spherical grids;
    - co-orbit grids with the binary system:
  - **Changing basis** between the two spherical grids:
2019 Results

Black Hole Head-on Collision (W Conf Factor)
Finding from BH collision test:

Numerical errors small and converge to zero at expected rate.

Post-merger num error, BH from x = -0.5 to +0.5
BiSphere Challenges

BiSphere grids: Two overlapping numerical grids in Spherical coordinates

- **Challenge #1**: Spherical coordinates have coordinate singularities
  - Tensors and vectors diverge or go to zero -> numerically unstable
- **Idea**: Scale out singular parts, (treat singular & nonsingular separately)
- **Result**: Numerical stability & robustness on par with Cartesian
  - (a) Ordinary spherical polar: done!
    Baumgarte, Montero, Cordero-Carrión, Müller (PRD 87, 044026, 2012), built upon covariant BSSN formulation of Brown (PRD 79, 104029, 2009)
  - (b) Generic-radius spherical polar (incl. log-radial): done!
    Ruchlin, Etienne, Baumgarte (PRD 97, 064036, 2018)

- **Challenge #2**: Develop new algorithm for
  - Interpolating between the two spherical grids; co-orbit grids with the binary system: done! (early 2019)
  - Changing basis between the two spherical grids: done!
BiSphere Challenges

BiSphere grids: Two overlapping numerical grids in Spherical coordinates

- **Challenge #1:** Spherical coordinates have coordinate singularities
  - Tensors and vectors diverge or go to zero -> numerically unstable
- **Idea:** Scale out singular parts, (treat singular & nonsingular separately)
- **Result:** Numerical stability & robustness on par with Cartesian
  - **Ordinary spherical polar:** done!
    - Baumgarte, Montero, Cordero-Carrión, Müller (PRD 87, 044026, 2012), built upon covariant BSSN formulation of Brown (PRD 79, 104029, 2009)
  - **Generic-radius spherical polar (incl. log-radial):** done!
    - Ruchlin, Etienne, Baumgarte (PRD 97, 064036, 2018)

- **Challenge #2:** Develop new algorithm for
  - Interpolating between the two spherical grids; co-orbit grids with the binary system: done! (early 2019)
  - Changing basis between the two spherical grids: done!

- **Challenge #3:** Calculations 50x too slow: small timesteps!
BH binary fits on desktop now (6GB), but ~50x too slow
  ○ Recent: About 1.5x gain through software optimz!
Problem:
  ○ Simulation timestep $\propto$ min dist between gridpoints
  ○ Spherical coords focus gridpoints at $r=0$, z-axis
How to solve?
Small Timesteps, The Last BiSphere Challenge

- BH binary fits on desktop now (6GB), but ~50x too slow
  - Recent: About 1.5x gain through software optimz!
- Problem:
  - Simulation timestep $\propto$ min dist between gridpoints
  - Spherical coords focus gridpoints at $r=0$, z-axis
- How to solve?
  - *Add a large Cartesian cube* (binary inside)
    - ~100x larger timesteps!
    - Memory usage?!
Small Timesteps, The Last BiSphere Challenge

- BH binary fits on desktop now (6GB), but ~50x too slow
  - Recent: About 1.5x gain through software optimz!
- Problem:
  - Simulation timestep \( \propto \min\ \text{dist between gridpoints} \)
  - Spherical coords focus gridpoints at \( r=0, z\)-axis
- How to solve?
  - Add a large Cartesian cube (binary inside)
    - ~100x larger timesteps!
    - Memory usage?!

Will it work?!
Gravitational Wave Comparison

Wave amplitude

Technical details: Dominant mode (l=m=2) of spin-weight -2 spherical harmonic of im(\psi_4) = \ddot{h}_x

Red: new simulation
Blue: trusted result

Early 2020 Results

Results
Gravitational Wave Comparison

Early 2020 Results

**Wave amplitude**

- **Red**: new simulation
- **Blue**: trusted result

Technical details:
Dominant mode (l=m=2) of spin-weight -2 spherical harmonic of \(\text{im}(\psi_4) = \ddot{h}_x\)

**4 HPC nodes, 28GB RAM**
Gravitational Wave Comparison

Wave amplitude vs Time

Red: new simulation
- 2016 Mobile CPU (OnePlus 5 phone),
- 2GB RAM

Blue: trusted result
- 4 HPC nodes,
- 28GB RAM

Technical details: Dominant mode (l=m=2) of spin-weight -2 spherical harmonic of im(psi_4) = \ddot{h}_x
Gravitational Wave Comparison

Wave amplitude
Time

Red: new simulation
Blue: trusted result

Technical details:
Dominant mode (l=m=2) of spin-weight -2 spherical harmonic of \( \text{im}(\psi_4) = \ddot{h}_x \)

Early 2020 Results

Results

4 HPC nodes, 28GB RAM
2016 Mobile CPU (OnePlus 5 phone), 2GB RAM
Problem with this grid structure:
- Only works well for two orbiting black holes very close to merger
- Larger separations -- Cartesian grid too large -- too much memory!
- Narrow Cartesian grid & rotate grids? Nope; resolution drop too large

What to do?!
Latest:
5-grid Bispheres
Latest: 5-grid Bispheres

Schematic
Schematic

Latest: 5-grid Bispheres

Benefits
1. Cartesian grids maximize timestep where Sph grids would focus
2. z-axis collinear across all three SinhSpherical grids, Nphi fixed
   a. $\Rightarrow$ 2D interpolations!
3. Large SinhSpherical perfect for GW extraction!
4. Low memory footprint
5. Intergrid surface area $<<$ Cartesian AMR $\Rightarrow$ Could scale well!
Latest: 5-grid Bispheres

- Cart.
- Small SinhSpherical
- Small SinhSpherical
- Large SinhSpherical
**Tough Validation Test**

- Fix grids in place (static grids)
- Match resolutions at red circles
  - (SinhSpherical/Cartesian intergrid boundary)
- Increase resolution near CoM
- Release BHs from rest, allowing BHs to cross intergrid boundaries
- BH merger remnant sits at intergrid boundary (SinhSpherical/SinhSpherical)
- Gravitational waves?!
5-grid Bispheres
*Tough* Validation Test
5-grid Bispheres
Tough Validation Test
5-grid Bispheres
*Tough* Validation Test
5-grid Bispheres
*Tough* Validation Test
5-grid Bispheres
*Tough* Validation Test
5-grid Bispheres
Tough Validation Test
5-grid Bispheres
*Tough* Validation Test
5-grid Bispheres: Two-BH, Head-on Collision

Brill-Lindquist evolution, $t=0.94642$ (W conformal factor)
5-grid Bispheres: Two-BH, Head-on Collision
5-grid Bispheres
Head-on Results

\[ \psi_4(s=-2, l=2, m=0) \]

\[ \begin{array}{c}
   \text{BH Pert Theory}
   \\
   \text{t/M}
   \\
   60 \quad 80 \quad 100 \quad 120 \quad 140 \quad 160 \quad 180 \quad 200 \quad 220
   \\
   -11 \quad -10 \quad -9 \quad -8 \quad -7 \quad -6 \quad -5 \quad -4 \quad -3 \quad -2
   \\
\end{array} \]
5-grid Bispheres
Head-on Results

\[ \Psi_4(s=2, l=2, m=0) \]

BH Pert Theory
New 5-grid NRPy+/SEN, 1.2 GB

60 80 100 120 140 160 180 200 220

\( t/M \)
5-grid Bispheres
Head-on Results

\[ \psi_4(s=-2, l=2, m=0) \]

\[ t/M \]

- BH Pert Theory
- New 5-grid NRPy+/SENRE, 1.2GB
- Trusted (ETK), Low-Res, 7.7GB
5-grid Bispheres
Head-on Results

BH Pert Theory
New 5-grid NRPy+/SENRE, 1.2GB
Trusted (ETK), Hi-Res, 33GB

$\psi_4(s=-2, l=2, m=0)$

$t/M$
5-grid Bispheres
Head-on Results

\[
\psi_4(s=2,l=2,m=0)
\]

-2
-3
-4
-5
-6
-7
-8
-9
-10
-11

\[
t/M
\]

60 80 100 120 140 160 180 200 220

BH Pert Theory
New 5-grid NRPy+/SENRE, 2.7GB
Trusted (ETK), ExHi-Res, 78GB
Summary
1. (close-separation) BBH mergers on a cellphone!
2. Comparable or superior waveforms, 5-grid Bispheres vs Cartesian AMR
   a. ~20x less memory usage!

Conclusions
Conclusions

Summary
1. (close-separation) BBH mergers on a cellphone!
2. Comparable or superior waveforms, 5-grid Bispheres vs Cartesian AMR
   a. ~20x less memory usage!

We’re so close now!
-={ Last Steps }=-
1. Current focus: regridding algorithm
   a. Hard part (months) was intergrid interpolations!
2. Small modifications:
   a. GW extraction
   b. TwoPunctures ID
3. Finished, not reported here
   a. PN parameters for quasicircular BBH ID
      i. NRPyPN
1. **Tensor components can be singular** ($\rightarrow 0$ or $\infty$) **at coord singularities**
   - Use cell-centered grids to avoid exact overlap with singularities
   - Singular pieces are multiplicative and known analytically:
     i. Scale out singular pieces & handle spatial derivs analytically
     ii. Promote rescaled tensors to evolved quantities
1. **Tensor components can be singular** ($\rightarrow 0$ or $\infty$) at coord singularities
   - Use cell-centered grids to avoid exact overlap with singularities
   - Singular pieces are multiplicative and known analytically:
     i. Scale out singular pieces & handle spatial derivs analytically
     ii. Promote rescaled tensors to evolved quantities

*Example*: Smooth spacetime quantity $\Lambda^i$
   - **Cartesian**: all components regular; no coord singularities
Addressing Issues with Singular Coordinates
Baumgarte, Montero, Cordero-Carrión, Müller (PRD 87, 044026, 2012)

1. **Tensor components can be singular** \( (\rightarrow 0 \text{ or } \infty) \) **at coord singularities**
   - Use cell-centered grids to avoid exact overlap with singularities
   - Singular pieces are multiplicative and known analytically:
     i. Scale out singular pieces & handle spatial derivs analytically
     ii. Promote rescaled tensors to evolved quantities

- **Example**: Smooth spacetime quantity \( \Lambda^i \)
  - **Cartesian**: all components regular; no coord singularities
  - **Spherical**: e.g., \( \phi \) component diverges at coord singularity
    - **Idea**: where needed, only take numer. derivatives of smooth part, \( \Lambda^\phi \)
    - Perform **exact** differentiation on singular terms like \( 1/(r \sin \theta) \)

\[
\Lambda^x = [\text{smooth}]
\]
\[
\Lambda^y = [\text{smooth}]
\]
\[
\Lambda^z = [\text{smooth}]
\]
\[
\Lambda^\phi = \frac{1}{r \sin \theta} \times [\text{smooth part}]
\]
\[
= \frac{1}{r \sin \theta} \times \lambda^\phi
\]
Addressing Issues with Singular Coordinates
Baumgarte, Montero, Cordero-Carrión, Müller (PRD 87, 044026, 2012)

1. **Tensor components can be singular** \((\rightarrow 0 \text{ or } \infty)\) at coord singularities
   - Use cell-centered grids to avoid exact overlap with singularities
   - Singular pieces are multiplicative and known analytically:
     i. Scale out singular pieces & handle spatial derivs analytically
     ii. Promote rescaled tensors to evolved quantities

2. **Divergent multiplicative terms in RHSs of equations**
   - E.g., 1D scalar wave equation:
     \[
     \partial_t^2 u = \partial_r^2 u + \frac{2}{r} \partial_r u
     \]
   - \(2/r\) term “stiffens” the equation
   - Even with cell-centered grids, RK2 timestepping is unstable
     i. Can use PIRK2 (original formulation), but
     ii. **Ordinary** RK4 works just fine in 3+1 NR (discovered later)
Addressing Issues with Singular Coordinates

Baumgarte, Montero, Cordero-Carrión, Müller (PRD 87, 044026, 2012)

1. Tensor components can be **singular** (→0 or ∞) at coord singularities
   ○ Use cell-centered grids to avoid exact overlap with singularities
   ○ Singular pieces are multiplicative and known analytically:
     i. Scale out singular pieces & handle spatial derivs analytically
     ii. Promote rescaled tensors to evolved quantities

2. Divergent multiplicative terms in RHSs of equations
   ○ E.g., 1D scalar wave equation:
     \[ \partial_t^2 u = \partial_r^2 u + \frac{2}{r} \partial_r u \]
   ○ \(2/r\) term “stiffens” the equation
   ○ Even with cell-centered grids, RK2 timestepping is unstable
     i. Can use PIRK2 (original formulation), but
     ii. *Ordinary* RK4 works just fine in 3+1 NR (discovered later)

*Net result*: **Stability & convergence properties on par** with Cartesian grids
SENR/NRPy+: Code Validation

http://blackholesathome.net

- Black hole simulation
  - Wormhole initial data
  - Cylindrical coordinates
  - Fourth-order finite differencing

- Excellent convergence
  - at t = 5M, in region unaffected by outer boundary (at r=10M)