Guest Lecture: Numerical Relativity

Lecture 4: Gravitational Waves MSc Course
Why do we need numerical relativity?

\[ G_{\mu\nu} = R_{\mu\nu} - \frac{1}{2} R g_{\mu\nu} = 8\pi T_{\mu\nu} \]

only a limited number of analytical solutions e.g.:

- **1916 Schwarzschild metric**
- **1963 Kerr metric**
- **1939 TOV equation**
- **1995 Rotating Disk of Dust**
- ...  

*We are far away from having any kind of solution of two orbiting compact object!*
Structure of the Lecture

1.) 3+1-decomposition
2.) requirements for successful simulations
3.) simple computational methods
4.) applications
   - binary black hole simulations
   - binary neutron star simulations
3+1- decomposition
Foliation of four-dimensional spacetime

\[ x^i = \text{const.} \]
Foliation of four-dimensional spacetime

normal vector:

\[ n_\mu = -\alpha \nabla_\mu t \]

induced metric:

\[ \gamma_{\mu\nu} = g_{\mu\nu} + n_\mu n_\nu \]

shift and lapse:

\[ n^\mu = \left( \frac{1}{\alpha}, -\frac{\beta^i}{\alpha} \right) \]

\[ ds^2 = (-\alpha^2 + \beta_i \beta^i) dt^2 + 2\beta_i dt dx^i + \gamma_{ij} dx^i dx^j \]

extrinsic curvature:

\[ K_{\mu\nu} = -P^\sigma_\mu \nabla_\sigma n_\nu \]

projection operator

\[ P^\mu_\nu = \delta^\mu_\nu + n^\mu n_\nu \]
Foliation of four-dimensional spacetime

\[ \gamma_{ij} = \begin{pmatrix} \gamma_{11} & \gamma_{12} & \gamma_{13} \\ \gamma_{21} & \gamma_{22} & \gamma_{23} \\ \gamma_{31} & \gamma_{32} & \gamma_{33} \end{pmatrix} \]

\[ K_{ij} = \begin{pmatrix} K_{11} & K_{12} & K_{13} \\ K_{21} & K_{22} & K_{23} \\ K_{31} & K_{32} & K_{33} \end{pmatrix} \]

\[ \partial_t \gamma_{ij} = -2\alpha K_{ij} + D_i \beta_j + D_j \beta_i \]
Splitting the Field Equations

We have split the 4D spacetime, but need to do the same for the field equation:

\[ G_{\mu\nu} = R_{\mu\nu} - \frac{1}{2} R g_{\mu\nu} = 8\pi T_{\mu\nu} \]

idea: use projections of 4D quantities to compute 3D variables e.g.:

\[ P^\delta_{\alpha} P^\kappa_{\beta} P^\lambda_{\mu} P^\sigma_{\nu} \mathcal{R}^{(4)}_{\delta\kappa\lambda\sigma} = (3) R_{\alpha\beta\mu\nu} + K_{\alpha\mu} K_{\beta\nu} - K_{\alpha\nu} K_{\beta\mu} \]

\[ P^\delta_{\alpha} P^\kappa_{\beta} P^\lambda_{\mu} n^{(4)}_{\sigma} R_{\delta\kappa\lambda\sigma} = D_{\beta} K_{\alpha\mu} - D_{\alpha} K_{\beta\mu}, \]
Splitting the Field Equations

Constraint Equations:

\[(3) \quad R + K^2 - K_{\alpha\beta} K^{\alpha\beta} = 16\pi E\]

Hamiltonian Constraint

\[D_j K^{ij} - D^i K = 8\pi S^i\]

Momentum Constraint

Question: Which other constraint equations do you know in classical physics?
Splitting the Field Equations

Evolution Equations:

\[ \partial_t \gamma_{ij} = -2\alpha K_{ij} + D_i \beta_j + D_j \beta_i \]

\[ \partial_t K_{ij} = -D_i D_j \alpha + \beta^k \partial_k K_{ij} + K_{ik} \partial_j \beta^k + K_{kj} \partial_i \beta^k + \alpha \left( \langle 3 \rangle R_{ij} + KK_{ij} - 2K_{ik}K^k_j \right) + 4\pi\alpha \left( \gamma_{ij} (S - E) - 2S_{ij} \right) \]

ADM-Equations

Richard Arnowitt, Stanley Deser, Charles Misner
1959

Question: Can we now evolve everything?
Splitting the Field Equations

Evolution Equations:

\[ \beta_j + D_j \beta_i \]

\[ j + K_{ik} \partial_j \beta^k + K_{kj} \partial_i \beta^k + \alpha \left( (\gamma_{ij}(S - E) - 2S_{ij}) \right) \]

**Question:** Can we now evolve everything?

ADM-Equations
Richard Arnowitt, Stanley Deser, Charles Misner
1959
Reformulation is necessary to obtain well posed problem (strongly hyperbolic)
An example: CTT + Z4c scheme

conformal transverse-traceless transformation for initial computation:

\[
\bar{\gamma}_{ij} = \psi^{-4} \gamma_{ij}, \quad \bar{A}_{ij} = \psi^2 A_{ij}
\]

Constraint Equations:

\[
\bar{D}^i \bar{D}_i \psi - \frac{1}{12} \psi^5 K^2 + \frac{1}{8} \psi^{-7} \bar{A}_{ij} \bar{A}^{ij} - \frac{1}{8} \psi^{(3)} R = -2\pi \psi^5 E,
\]

\[
\bar{D}_j \bar{A}^{ij} - \frac{2}{3} \psi^6 \bar{\gamma}^{ij} \bar{D}_j K = 8\pi \psi^{10} S^i.
\]

there are other alternatives to recast the equations
An example: CTT + Z4c scheme

transformation for evolution equations:

\[ R_{\alpha\beta} + \nabla_\alpha Z_\beta + \nabla_\beta Z_\alpha = 8\pi \left( T_{\alpha\beta} - \frac{1}{2} g_{\alpha\beta} T \right) + \kappa_1 (t_\alpha Z_\beta + t_\beta Z_\alpha - (1 + \kappa_2) g_{\alpha\beta} ) t_\gamma Z_\gamma \]

add constraints to evolution system

transformation:

\[
\begin{align*}
\tilde{\gamma}_{ij} &= \chi \gamma_{ij} \\
\tilde{A}_{ij} &= \chi (K_{ij} - \frac{1}{3} \gamma_{ij} K) \\
\hat{K} &= \gamma_{ij} K_{ij} - 2\Theta, \quad \Theta = -n_\alpha Z^\alpha
\end{align*}
\]
An example: CTT + Z4c scheme

Question: How do the coordinates evolve?

transformation for evolution equations:

\[
\begin{align*}
\partial_t \chi &= \frac{2}{3} \chi \left( \alpha (\dot{K} + 2\Theta) - D_i \beta^i \right) \\
\partial_t \tilde{\gamma}_{ij} &= -2\alpha \tilde{A}_{ij} + \beta^k \partial_k \tilde{\gamma}_{ij} + 2\tilde{\gamma}_{k(i} \partial_{j)} \beta^k - \frac{2}{3} \tilde{\gamma}_{ij} \partial_k \beta^k, \\
\partial_t \tilde{K} &= -D^i D_i \alpha + \alpha \left( \tilde{A}_{ij} \tilde{A}^{ij} + \frac{1}{3} (\dot{K} + 2\Theta)^2 \right) \\
&\quad + 4\pi \alpha (S + E) + \beta^k \partial_k \dot{K} + \alpha \kappa_1 (1 - \kappa_2) \Theta \\
\partial_t \tilde{A}_{ij} &= \chi \left( -D_i D_j \alpha + \alpha \left( ^{(3)} R_{ij} - 8\pi S_{ij} \right) \right)^{TF} + \alpha \left( \dot{K} + 2\Theta \right) \tilde{A}_{ij} - 2 \tilde{A}^k_j \tilde{A}_{kj} \\
&\quad + \beta^k \partial_k \tilde{A}_{ij} + 2 \tilde{A}_k \partial_k \beta^j - \frac{2}{3} \tilde{A}_{ij} \partial_k \beta^k \\
\partial_t \tilde{\Gamma}^i &= -2 \tilde{A}^{ik} \partial_k \alpha + 2\alpha \left( \tilde{\Gamma}^{ik}_{kl} \tilde{A}^{kl} - \frac{3}{2} \tilde{A}^{ik} \partial_k \ln(\chi) - \frac{1}{3} \tilde{\gamma}^{ik} \partial_k (\dot{K} + 2\Theta) - 8\pi \tilde{\gamma}^{ik} S_k \right) \\
&\quad + \tilde{\gamma}^{kl} \partial_k \partial_l \beta^i + \frac{1}{3} \tilde{\gamma}^{ik} \partial_t \partial_k \beta^l - 2\alpha \kappa_1 (\tilde{\Gamma}^i - \tilde{\Gamma}^i) \\
&\quad + \beta^k \partial_k \tilde{\Gamma}^i - \tilde{\Gamma}^k \partial_k \beta^i + \frac{2}{3} \tilde{\Gamma}^i \partial_k \beta^k, \\
\partial_t \Theta &= \frac{\alpha}{2} \left( ^{(3)} R - \tilde{A}_{ij} \tilde{A}^{ij} + \frac{2}{3} (\dot{K} + 2\Theta)^2 \right) \\
&\quad - \alpha \left( 8\pi E + \kappa_1 (2 + \kappa_2) \Theta \right) + \beta^i \partial_t \Theta,
\end{align*}
\]
Requirements for a successful simulation
Gauge/Coordinate choice

Computational Space

Physical Space

Mapping
Gauge/Coordinate choice – geodesic slicing –

\[ \alpha = 1 , \ \beta^i = 0 \]

Question: What will happen when you evolve a Schwarzschild BH?

The value of result is: NaN

OK
Gauge/Coordinate choice
– How to evolve BHs –

moving puncture gauge, e.g.:

$$\partial_t \alpha = -2\alpha K$$

$$\partial_t \beta^i = B^i$$,
$$\partial_t B^i = \frac{3}{4} \partial_t \tilde{\Gamma}^i - \eta B^i$$
Gauge/Coordinate choice
– How to evolve BHs –

Question: Should the excision boundary be inside or outside of the horizon?

remove the points around the singularity
Artificial Atmosphere

Computation of pure Vacuum is not possible

\[ \rho \]

\[ r/M_\odot \]
Computational methods
Approximation of derivatives

Question: How to compute derivatives numerically?
Approximation of derivatives

- Finite differences
- Finite elements
- Finite volume
- Spectral method
- Discontinuous Galerkin
- ...
Finite differencing

Taylor expansion around central point

\[ f(x) = f(x_j) + (x - x_j)f'(x_j) + \frac{(x - x_j)^2}{2}f''(x_j) + \mathcal{O} \left( (x - x_j)^3 \right) \]

use neighboring gridpoints \( x_{j-1}, x_{j+1} \)

\[ f'(x_j) = \frac{f(x_{j+1}) - f(x_{j-1})}{x_{j+1} - x_{j-1}} + \mathcal{O} \left( (x_{j+1} - x_{j-1})^2 \right) \]
Spectral methods

approximate function by series expansion

\[ f(x) = \sum_n a_n \Phi_n \]

\[ a_n = \int_A^B f(x) \Phi_n(x) \omega(x) \, dx \]
Spectral methods

approximate function by series expansion

\[ f(x) = \sum_n a_n \Phi_n \]

\[ a_n = \int_A^B f(x) \Phi_n(x) \omega(x) \, dx \]

assuming a finite number of points

\[ f_{N,j} = \sum_{n=0}^{N-1} a_n \Phi_{n,j}, \quad \text{with} \quad a_n = \sum_{j=0}^{N-1} f_{N,j} \Phi_{n,j} \omega_j \]
Spectral methods

approximate function by series expansion

\[ f(x) = \sum_n a_n \Phi_n \]

\[ a_n = \int_A^B f(x) \Phi_n(x) \omega(x) \, dx \]

assuming a finite number of points

\[ f_{N,j} = \sum_{n=0}^{N-1} a_n \Phi_{n,j}, \quad \text{with} \quad a_n = \sum_{j=0}^{N-1} f_{N,j} \Phi_{n,j} \omega_j \]

approximation of derivatives

\[ \partial_x f_{N,j} = \sum_{n=0}^{N-1} a_n \partial_x \Phi_{n,j} = \sum_{n=0}^{N-1} b_n \Phi_{n,j} \]
Spectral methods

\[ \Phi(t, x^i) = \sum_{N} c_N \phi_N \]

\[ \phi_N = \{x^0, x^1, x^2, x^3, x^4, \ldots\} \]

c0 = 1

c1 = 0

c2 = -1/2

c3 = 0

c4 = 1/24

c5 = 0

c6 = -1/720

c7 = 0

c8 = 1/40320

c9 = 0

c10 = -1/3268800
Explicit time integration

Euler method:

\[ y_{n+1} = y_n + h f(x_n, y_n) \]

Question: How to do better?
Explicit time integration

\[ k_1 = h f(x_n, y_n) \]

\[ k_2 = h f \left( x_n + \frac{1}{2} h, y_n + \frac{1}{2} k_1 \right) \]

\[ y_{n+1} = y_n + k_2 + O(h^3) \]

Convergence order improved by additional, intermediate step
4\textsuperscript{th} order explicit Runge-Kutta

\[ k_1 = hf(x_n, y_n) \]
\[ k_2 = hf(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}) \]
\[ k_3 = hf(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}) \]
\[ k_4 = hf(x_n + h, y_n + k_3) \]
\[ y_{n+1} = y_n + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6} + O(h^5) \]

Why not going to higher order?
Time integration (more advanced)

- Implicit methods
  - necessary for stiff ODEs

- Partially-implicit

- Adaptive time stepping
Method of lines

- principle idea: replace the spatial derivatives algebraic approximations
- only the initial value variable, typically time in a physical problem remains
- one obtains a system of ODEs that approximate the original PDE
- apply any integration algorithm for initial value ODEs to compute an approximate numerical solution to the PDE
Method of lines

Example: *heat equation*

\[
\frac{\partial u}{\partial t} - \beta^2 \Delta u = f
\]

\[
\frac{\partial u_j}{\partial t} - \frac{\beta^2}{h^2}(u_{j-1} - 2u_j + u_{j+1}) = f_j(t)
\]

Homework: wave equation
Some of the add ons for NR simulations

- Adaptive mesh refinement techniques
- Berger-Oliger/Collela time stepping
- high-order shock capturing schemes
- ...
Mesh refinement

Question: Can all level go with the same time step?

- Box-in-box mesh refinement
Mesh refinement

trade-off between efficiency, accuracy, and stability

- **Efficiency**: “Faster” evolution with larger time step
- **Accuracy**: Smaller time step decreases time integration error
- **Stability**: Courant-Friedrichs-Lewy condition needs to be fulfilled
Mesh refinement

**Stability:**

\[ \frac{dx}{dt} = \frac{1}{u} \]

![Numerical domain of dependence](image)

Stable

\[ \Delta x \]

Unstable

\[ \Delta x \]

e.g. 1D advection equation

\[ CFL \equiv \frac{|u| \Delta t}{\Delta x} \leq 1 \]
Berger – Oliger time stepping

2 (3) important steps:
- prolongation
- restriction
- correction (if matter is present)
Correction step:

- Correcting for flux differences within different refinement levels
High resolution shock capturing

Problem:
- What happens when function is not-smooth
High resolution shock capturing  
– example: WENO schemes –

\[
\begin{align*}
  u_t + \nabla \cdot f(u) &= 0 \\
  u(x, 0) &= u_0(x)
\end{align*}
\]

How to compute the flux?

\[
\begin{align*}
  \frac{2}{3} f_{j-1/2} + \frac{1}{3} f_{j+1/2} &= \frac{1}{6} f_{j-1} + \frac{5}{6} f_j \\
  \frac{1}{3} f_{j-1/2} + \frac{2}{3} f_{j+1/2} &= \frac{5}{6} f_j + \frac{1}{6} f_{j+1} \\
  \frac{2}{3} f_{j-1/2} + \frac{1}{3} f_{j+3/2} &= \frac{1}{6} f_j + \frac{5}{6} f_{j+1} \\
  \frac{3}{10} f_{j-1/2} + \frac{6}{10} f_{j+1/2} + \frac{1}{10} f_{j+3/2} &= \frac{1}{30} f_{j-1} + \frac{19}{30} f_j + \frac{10}{30} f_{j+1} \\
  \left(\frac{2}{3} \omega_1 + \frac{1}{3} \omega_2\right) f_{j-1/2} + \left(\frac{1}{3} \omega_1 + \frac{2}{3} (\omega_2 + \omega_3)\right) f_{j+1/2} + \frac{1}{3} \omega_3 f_{j+3/2} &= \frac{\omega_1}{6} f_{j-1} + \frac{5(\omega_1 + \omega_2)}{6} f_j + \frac{\omega_2 + 5 \omega_3}{6} f_{j+1}
\end{align*}
\]
HPC Computing

<table>
<thead>
<tr>
<th>Rank</th>
<th>Site</th>
<th>System Description</th>
<th>Cores</th>
<th>Rmax (TFlop/s)</th>
<th>Rpeak (TFlop/s)</th>
<th>Power (kW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>National Supercomputing Center in Wuxi, China</td>
<td>Sunway TaihuLight - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway NRCPC</td>
<td>10,649,600</td>
<td>93,014.6</td>
<td>125,435.9</td>
<td>15,371</td>
</tr>
<tr>
<td>2</td>
<td>National Super Computer Center in Guangzhou, China</td>
<td>Tianhe-2 (MilkyWay-2) - TH-IYB-FEP Cluster, Intel Xeon E5-2692 12C 2.20GHz, TH Express-2, Intel Xeon Phi 315P</td>
<td>3,120,000</td>
<td>33,862.7</td>
<td>54,902.4</td>
<td>17,808</td>
</tr>
<tr>
<td>3</td>
<td>Swiss National Supercomputing Centre (CSCS), Switzerland</td>
<td>Piz Daint - Cray XC50, Xeon E5-2690V3 12C 2.6GHz, Aries interconnect, NVIDIA Tesla P100</td>
<td>361,760</td>
<td>19,590.0</td>
<td>25,326.3</td>
<td>2,272</td>
</tr>
<tr>
<td>4</td>
<td>Japan Agency for Marine-Earth Science and Technology, Japan</td>
<td>Gyoukou - ZettaScaler-2.2 HPC system, Xeon D-1571 16C 1.3GHz, Infiniband EDR, PEZY-SC2 700MHz ExaScaler</td>
<td>19,860,000</td>
<td>19,135.8</td>
<td>28,192.0</td>
<td>1,350</td>
</tr>
<tr>
<td>5</td>
<td>DOE/SC/Oak Ridge National Laboratory, United States</td>
<td>Titan - Cray XK7, Opteron 6274 16C 2.20GHz, Cray Gemini interconnect, NVIDIA K20x</td>
<td>560,640</td>
<td>17,590.0</td>
<td>27,112.5</td>
<td>8,209</td>
</tr>
<tr>
<td>6</td>
<td>DOE/NNSA/LLNL, United States</td>
<td>Sequoia - BlueGene/Q, Power BQC 16C 1.60 GHz, Custom IBM</td>
<td>1,572,864</td>
<td>17,173.2</td>
<td>20,132.7</td>
<td>7,890</td>
</tr>
<tr>
<td>7</td>
<td>DOE/NNSA/LANL/SNL, United States</td>
<td>Trinity - Cray XC40, Intel Xeon Phi 7250 68C 1.40GHz, Aries interconnect Cray Inc.</td>
<td>979,768</td>
<td>14,137.3</td>
<td>43,902.6</td>
<td>3,664</td>
</tr>
<tr>
<td>8</td>
<td>DOE/SC/LBNL/NERSC, United States</td>
<td>Cori - Cray XC40, Intel Xeon Phi 7250 68C 1.46GHz, Aries interconnect Cray Inc.</td>
<td>622,336</td>
<td>14,014.7</td>
<td>27,880.7</td>
<td>3,939</td>
</tr>
<tr>
<td>9</td>
<td>Joint Center for Advanced High Performance Computing, Japan</td>
<td>Oakforest-PACS - PRIMERGY CX1660 M1, Intel Xeon Phi 7250 48C 1.46GHz, Intel Omni-Path</td>
<td>556,104</td>
<td>13,584.6</td>
<td>24,913.5</td>
<td>2,719</td>
</tr>
<tr>
<td>10</td>
<td>RIKEN Advanced Institute for Computational Science (AICS), Japan</td>
<td>K computer, SPARC64 VIIIfx 2.0GHz, Tohu interconnect</td>
<td>700,024</td>
<td>10,510.0</td>
<td>11,280.4</td>
<td>12,660</td>
</tr>
<tr>
<td>11</td>
<td>DOE/SC/Argonne National Laboratory, United States</td>
<td>Mira - BlueGene/Q, Power BQC 16C 1.60GHz, Custom IBM</td>
<td>786,432</td>
<td>8,586.6</td>
<td>10,066.3</td>
<td>3,945</td>
</tr>
<tr>
<td>12</td>
<td>Texas Advanced Computing</td>
<td>Stampede2 - PowerEdge</td>
<td>368,928</td>
<td>8,317.7</td>
<td>18,215.8</td>
<td></td>
</tr>
</tbody>
</table>

Fun Question: Which rank has Netherland’s fastest HPC cluster?
HPC Computing

Parallelization strategies:
- MPI [Message Passing Interface]
- OpenMP [Open Multi-Processing]
- pthread [POSIX Thread]
  
  ...

- CUDA [Compute Unified Device Architecture]
- OpenCL [Open Computing Language]
HPC Computing

- Hybrid MPI/OpenMP scaling

one box/ MPI – process each box has multiple OpenMP threads
HPC Computing

Performance

\[
\text{efficiency} = \frac{\text{speed}_2/\text{speed}_1}{\#\text{cores}_2/\#\text{cores}_1}
\]
Applications
– BBH simulations –
Numerical relativity simulation needed to understand the time around merger
Example: highly precessing BBH

https://www.youtube.com/watch?v=grA5KfDlsAY
Example: large mass ratio BBH

https://www.youtube.com/watch?v=Dxfx3NIanHk
The first BBH detection

https://www.youtube.com/watch?v=-vYJdh8wALg
The first BBH detection

https://www.youtube.com/watch?v=c-2XIuNFgD0&t=11s
What NR told us about BBHs

- Merger part is less violent than expected

Gravitational waves eject black hole from galaxy

- remnants can obtain a large kick due to GW emission
Applications
– BNS simulations –
Multimessenger Picture

Gravitational Waves
- inspiral signal: chirp
- postmerger signal

Electromagnetic Waves
- short GRB
- kilo/macronovae
- radio flares

Neutrinos
- high neutrino luminosity
The BNS merger

https://www.youtube.com/watch?v=V6cm-0bwJ98
Gravitational Waves

https://www.youtube.com/watch?v=B7DnRkQub-U
Gravitational Waves

LIGO detected several hundred of orbits for GW170817
Waveform variety

High masses

High mass ratio

Precessing

Highly eccentric

Realistic microphysics
An example: a high mass ratio BNS
Gravitational Waves

- Chirp signal during the inspiral

Determine the mass

Sky localization

28 sq-deg

Determine the EOS

Normalized amplitude

Frequency (Hz)

Time (seconds)
Gravitational Waves: Parameter Estimation

- Determine the Equation of State

GW observations favor Nss with smaller radii

Gravitational Waves: Postmerger

- postmerger signal at higher frequencies with low chances of detection

Bernuzzi, TD, Nagar PRL 115.091101
Gravitational Waves: Postmerger

- postmerger signal at higher frequencies with low chances of detection

no postmerger signal for GW170817 detected
Neutrinos
Neutrinos

- Heating during the NS merger virial temperature

\[ T_{\text{vir}} \approx 25 \text{ MeV} \left( \frac{M}{2.5 \, M_{\odot}} \right) \left( \frac{100 \, \text{km}}{R} \right) \]

Electron-positron production

\[ n + e^+ \rightarrow p + \bar{\nu}_e \]

- Also production of heavy leptons etc.
Neutrinos

- Detection of neutrinos very unlikely

<table>
<thead>
<tr>
<th>EoS</th>
<th>$q$</th>
<th>$t$ [ms]</th>
<th>$\langle E_{\nu_e} \rangle$ [MeV]</th>
<th>$\langle E_{\nu_\mu} \rangle$ [MeV]</th>
<th>$L_{\nu_e}^{10^53}$ erg/s</th>
<th>$R_{\nu}$ [#/ms]</th>
</tr>
</thead>
<tbody>
<tr>
<td>NL3</td>
<td>1.0</td>
<td>3.4</td>
<td>18.5 (22.4)</td>
<td>15.2 (18.3)</td>
<td>0.7</td>
<td>18</td>
</tr>
<tr>
<td>NL3</td>
<td>0.85</td>
<td>3.0</td>
<td>15.6 (18.7)</td>
<td>12.6 (15.1)</td>
<td>0.8</td>
<td>18</td>
</tr>
<tr>
<td>DD2</td>
<td>1.0</td>
<td>3.3</td>
<td>18.3 (22.1)</td>
<td>14.6 (17.4)</td>
<td>1.1</td>
<td>28</td>
</tr>
<tr>
<td>DD2</td>
<td>0.85</td>
<td>2.8</td>
<td>18.1 (21.7)</td>
<td>15.1 (18.0)</td>
<td>1.0</td>
<td>25</td>
</tr>
<tr>
<td>DD2</td>
<td>0.76</td>
<td>2.4</td>
<td>19.7 (23.9)</td>
<td>14.8 (17.9)</td>
<td>1.3</td>
<td>36</td>
</tr>
<tr>
<td>SFHo</td>
<td>1.0</td>
<td>3.5</td>
<td>24.6 (29.7)</td>
<td>23.5 (28.3)</td>
<td>3.5</td>
<td>121</td>
</tr>
<tr>
<td>SFHo</td>
<td>0.85</td>
<td>3.9</td>
<td>17.8 (21.3)</td>
<td>15.3 (17.9)</td>
<td>2.0</td>
<td>50</td>
</tr>
</tbody>
</table>

Lehner et al, arXiv:1603.00501

GW170817 @ 40Mpc

< 0.0003 neutrinos
EM signals

GW170817 DECam observation (0.5–1.5 days post merger)

GW170817 DECam observation (>14 days post merger)
EM Signals

Timeline
EM Signals – sGRBs

- BH + disk system
  - Neutrino & anti-neutrino annihilation
  - Magnetic field amplification and jet formation

Kuichi et al., PRD92, 124034

Ruiz et al., APJ. 824 (2016), L6
EM Signals – Kilonova

- pseudo-black body radiation from r-process elements
- formation of heavy elements

Metzger, arxiv:1710.05931
Incorporating NR results

- Predicting the ejecta mass

https://www.youtube.com/watch?v=qPL6ZHBLFil
Incorporating NR results

- Predicting the ejecta mass

https://www.youtube.com/watch?v=qPL6ZHBLFiI
Incorporating NR results

- Predictions about ejecta mass and compositions
  - dynamical ejecta:
    - tidal tail
    - shock heating
  - disk winds
    - neutrino driven winds
    - magnetic winds
    - secular

TD, Ujevic CQG. 34 (2017), 105014
Coughlin, TD et al, APJ. 849 (2017) no.1, 12
EM Signals – Kilonova

- two component model based on atomic structure and radiative transfer simulations
EM Signals

Kilonova

Coughlin, TD, et al., 2018 in prep.
Things to remember:

1.) Fundamental idea behind the 3+1 – decomposition and why it is needed

2.) Additional things one should worry about when doing numerical simulations (BH treatment, Gauge choice, …)

3.) Basics of finite differencing and spectral methods

4.) Method of lines

5.) Principle idea of: mesh refinement, Berger-Oliger time stepping, HPC Computing

6.) Applications of BBH simulations (computation of the waveforms close to merger, …)

7.) Applications of BNS simulations (predicting GWs close to merger and in the postmerger, measurement of the ejected material, estimation of neutrino luminosity, …)