einstein toolkit

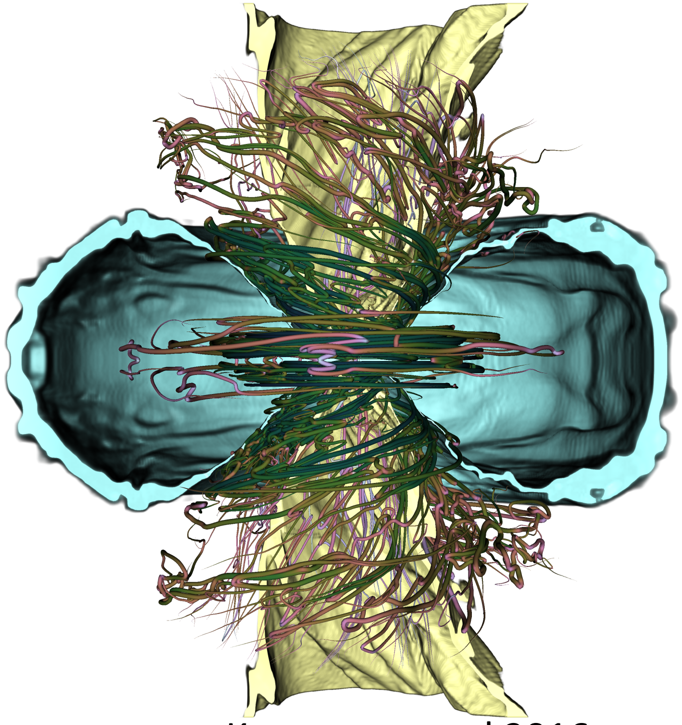
Bruno Giacomazzo

University of Trento and INFN-TIFPA

www.brunogiacomazzo.org

General Relativity and Astrophysics

- Binary Black Hole Mergers
- Binary Neutron Star Mergers
- Neutron Star – Black Hole Mergers
- Supernovae
- Accretion Disks
- Boson Stars
- Cosmology



Kawamura et al 2016

In all these scenarios general relativity plays a fundamental role.

Developing a code that solves the full set of GR and (Magneto)Hydrodynamic equations is not an easy task.

Einstein Toolkit
einsteintoolkit.org

- Set of publicly available tools for relativistic astrophysics
- Latest release on July 17 2017 (codename “Hack”)
- More than 150 users on 6 continents
- Tested on several HPC infrastructures around the world
- Includes over 100 Cactus thorns, including:
  - McLachlan (space-time evolution)
  - GRHydro and IllinoisGRMHD (GRMHD equations)
  - Several initial data and analysis routines
- Data can be read and visualized by open source codes (e.g., Visit, PostCactus, yt)

- Open source framework
  - Decentralized code development
  - Active and friendly user community
  - Module based approach

- Infrastructure modules
  - Parameter file handling
  - Parallelization (MPI + OpenMP)
  - Adaptive Mesh refinement (Carpet)
  - IO (ASCII and HDF5) + checkpointing

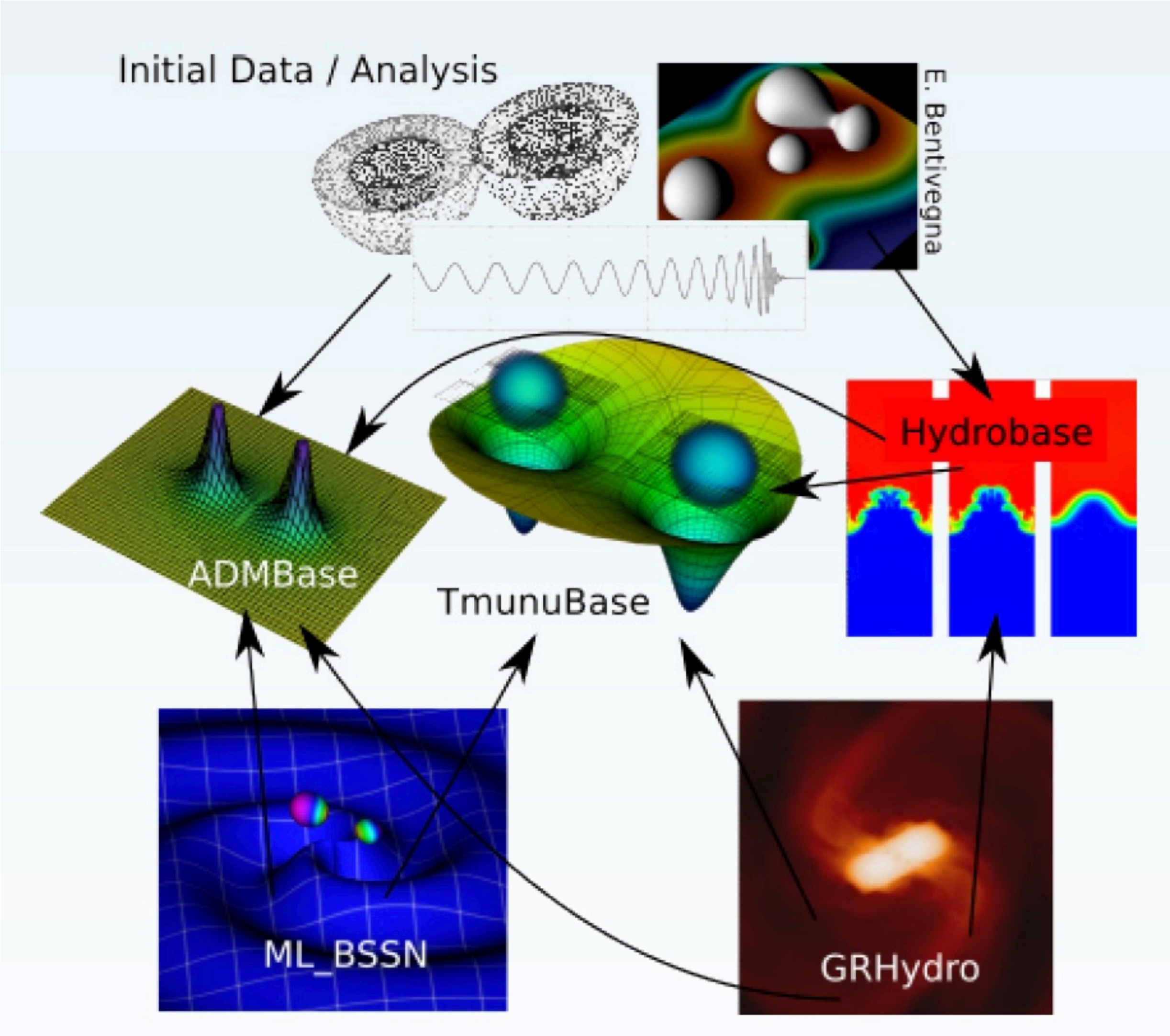
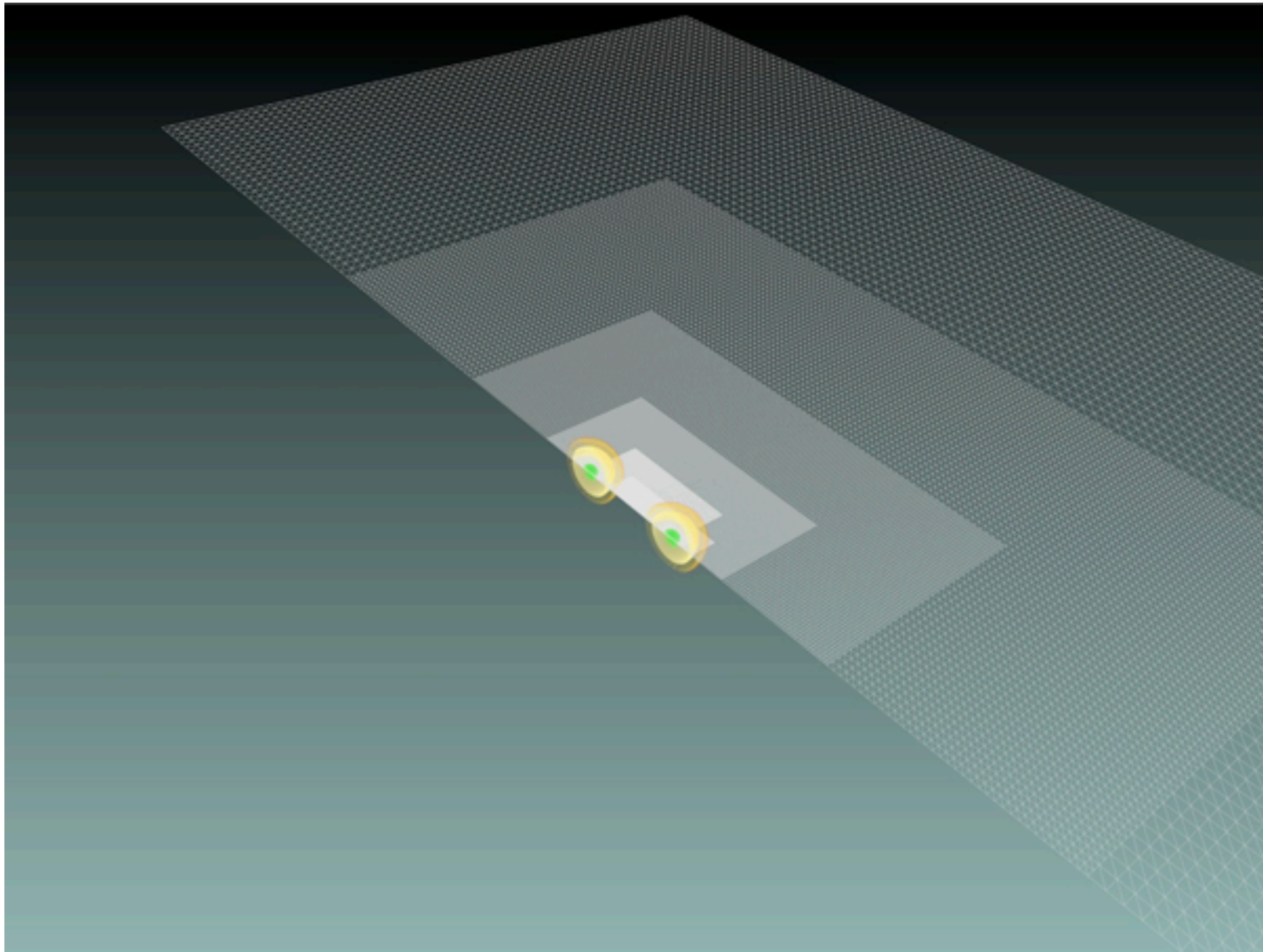


Figure by R. Haas

Adaptive Mesh Refinement Driver (Carpet)
https://carpetcode.org/

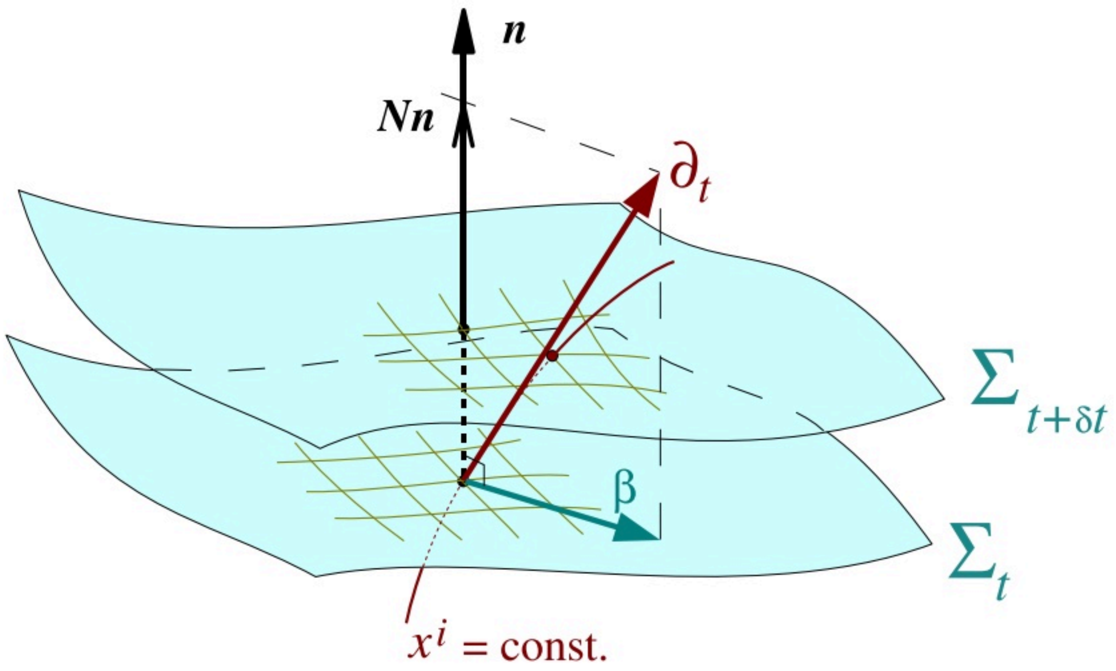
- Carpet provides box-in-box mesh refinement
- It is included in the toolkit



Visualization by Giacomazzo, Kaehler, Rezzolla

Einstein Equations

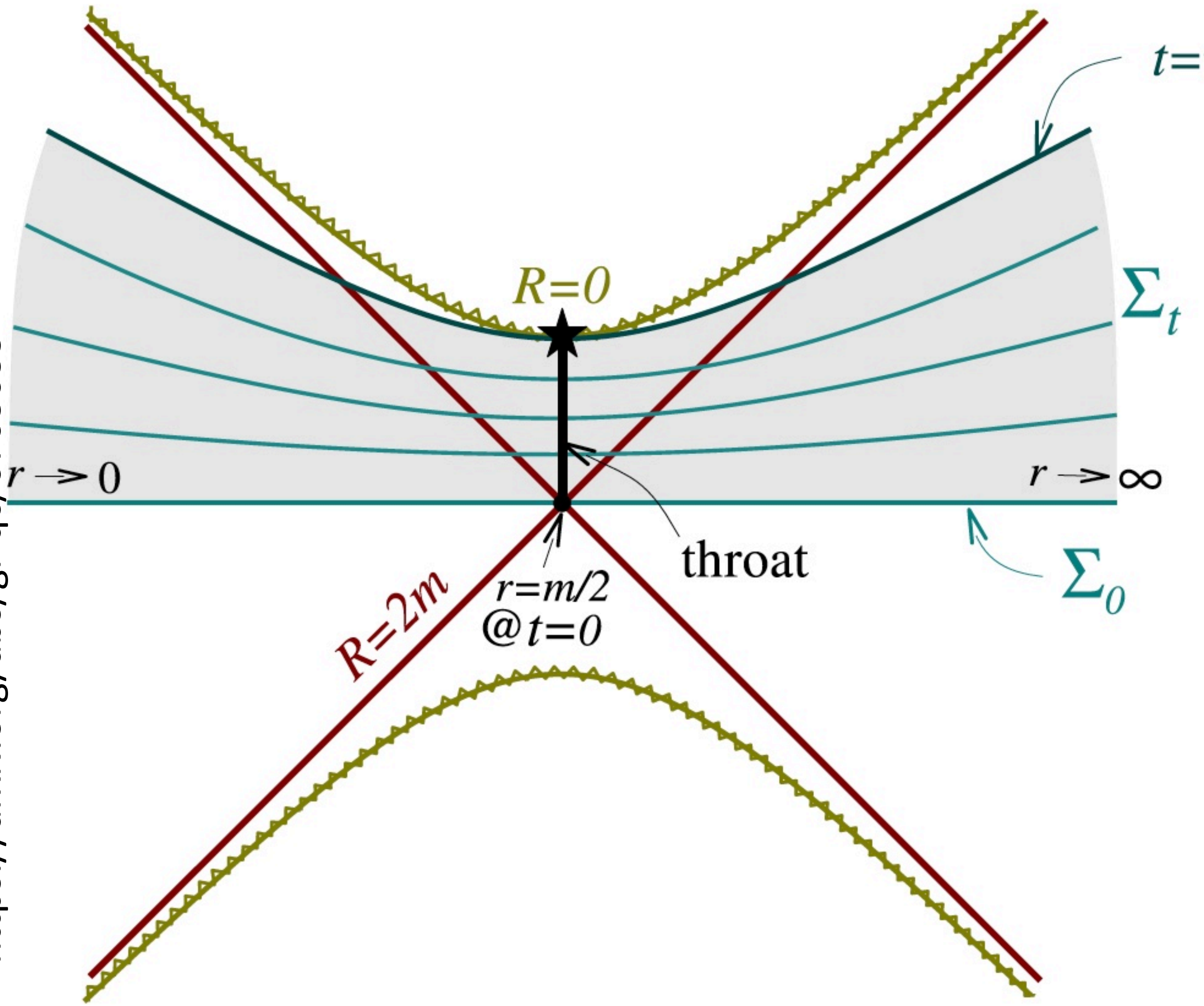
Space-Time Foliation



https://arxiv.org/abs/gr-qc/0703035

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

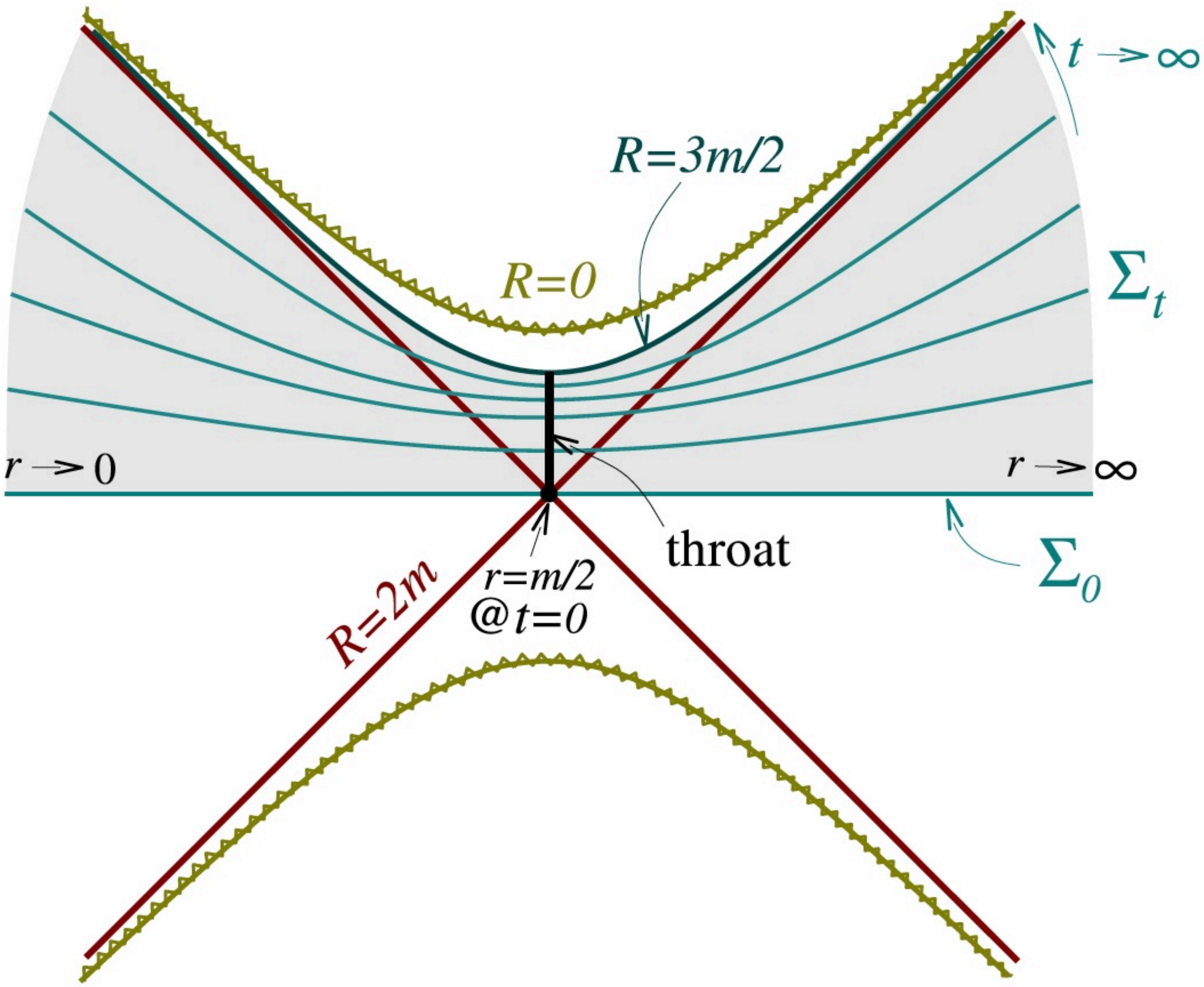
Choice of Foliation: geodesic slicing



https://arxiv.org/abs/gr-qc/0703035

The simplest choice could be to just set the lapse to be constant.

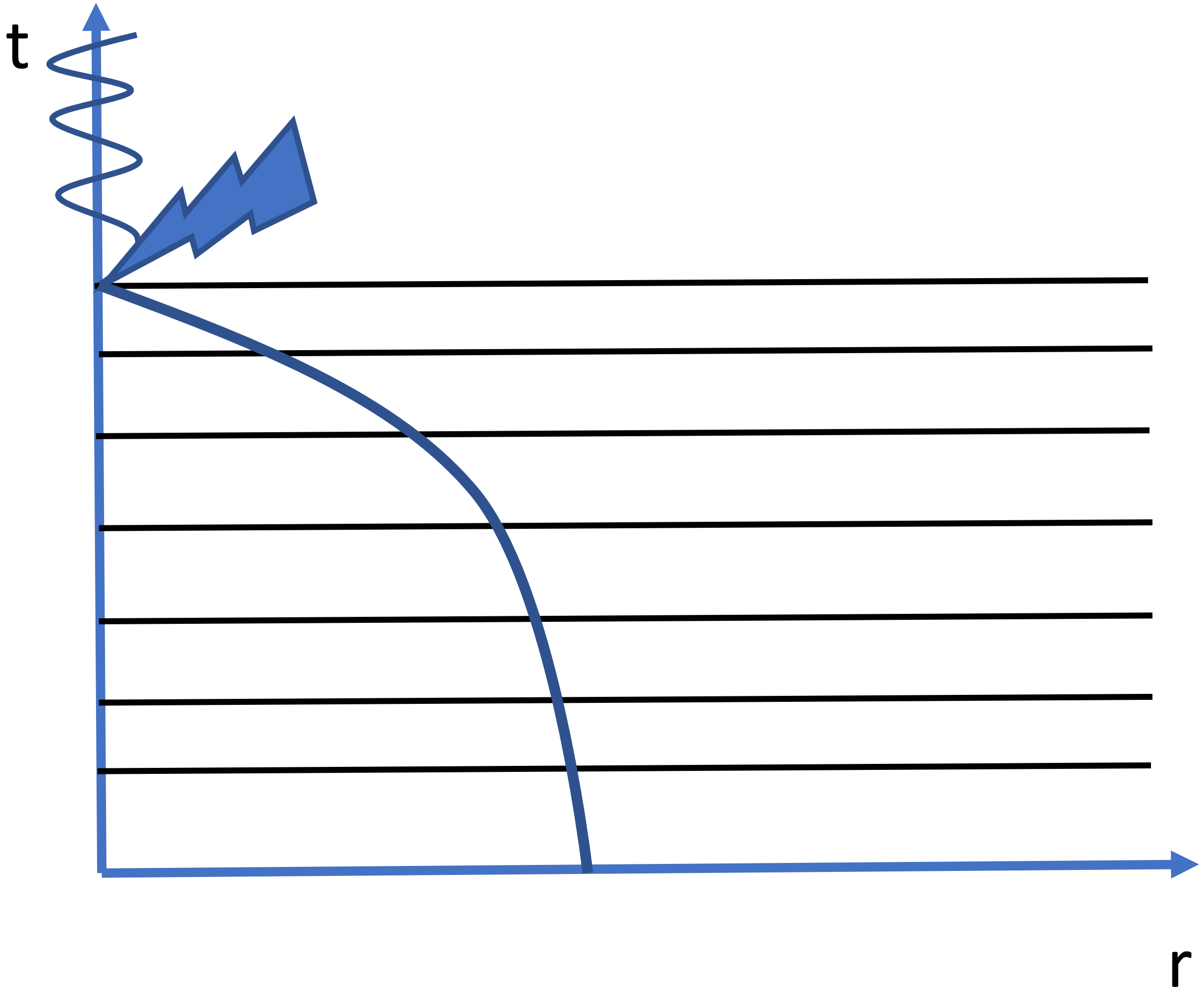
Choice of Foliation: maximal slicing



https://arxiv.org/abs/gr-qc/0703035

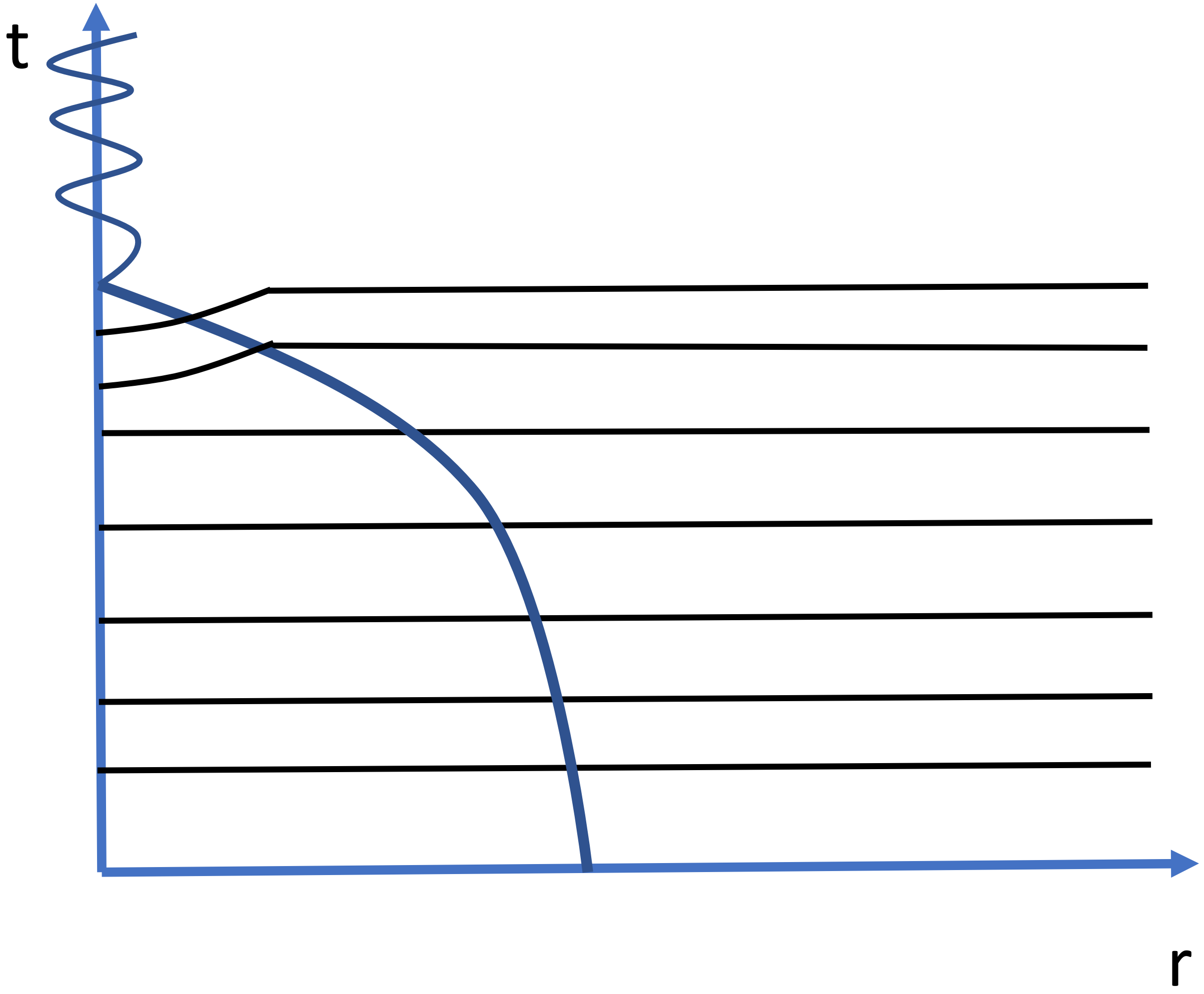
Better choices use evolution equations for lapse and shift such that the singularity can be avoided.

Choice of Foliation: geodesic slicing



The simplest choice could be to just set the lapse to be constant.

Choice of Foliation: singularity-avoding slicing



Better choices use evolution equations for lapse and shift such that the singularity can be avoided.

Einstein Equations

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

$R \equiv R^{\mu}_{\mu}$ Ricci scalar

$R_{\mu\nu} \equiv R^{\rho}_{\mu\rho\nu}$ Ricci tensor

$R^{\sigma}_{\mu\rho\nu} \equiv \partial_\rho \Gamma^{\sigma}_{\mu\nu} - \partial_\nu \Gamma^{\sigma}_{\mu\rho} + \Gamma^{\sigma}_{\tau\rho}\Gamma^{\tau}_{\mu\nu} - \Gamma^{\sigma}_{\tau\nu}\Gamma^{\tau}_{\mu\rho}$ Riemann tensor

$$\Gamma^{\sigma}_{\mu\rho} \equiv \frac{1}{2} g^{\sigma\tau} \left(\partial_\mu g_{\rho\tau} + \partial_\rho g_{\mu\tau} - \partial_\tau g_{\mu\rho} \right)$$

BSSN scheme

$$\left(\frac{\partial}{\partial t} - \mathcal{L}_{\boldsymbol{\beta}}\right)\Psi = \frac{\Psi}{6}\left(\tilde{D}_i\beta^i - NK\right) \tag{10.61}$$

$$\left(\frac{\partial}{\partial t} - \mathcal{L}_{\boldsymbol{\beta}}\right)\tilde{\gamma}_{ij} = -2N\tilde{A}_{ij} - \frac{2}{3}\tilde{D}_k\beta^k\,\tilde{\gamma}_{ij} \tag{10.62}$$

$$\left(\frac{\partial}{\partial t} - \mathcal{L}_{\boldsymbol{\beta}}\right)K = -\Psi^{-4}\left(\tilde{D}_i\tilde{D}^iN + 2\tilde{D}_i\ln\Psi\,\tilde{D}^iN\right) + N\left[4\pi(E+S) + \tilde{A}_{ij}\tilde{A}^{ij} + \frac{K^2}{3}\right] \tag{10.63}$$

$$\begin{aligned}\left(\frac{\partial}{\partial t} - \mathcal{L}_{\boldsymbol{\beta}}\right)\tilde{A}_{ij} \;=\; & -\frac{2}{3}\tilde{D}_k\beta^k\,\tilde{A}_{ij} + N\left[K\tilde{A}_{ij} - 2\tilde{\gamma}^{kl}\tilde{A}_{ik}\tilde{A}_{jl} - 8\pi\left(\Psi^{-4}S_{ij} - \frac{1}{3}S\tilde{\gamma}_{ij}\right)\right]\\ & +\Psi^{-4}\Bigg\{-\tilde{D}_i\tilde{D}_jN + 2\tilde{D}_i\ln\Psi\,\tilde{D}_jN + 2\tilde{D}_j\ln\Psi\,\tilde{D}_iN\\ & \quad +\frac{1}{3}\left(\tilde{D}_k\tilde{D}^kN - 4\tilde{D}_k\ln\Psi\,\tilde{D}^kN\right)\tilde{\gamma}_{ij}\\ & \quad +N\Bigg[\frac{1}{2}\left(-\tilde{\gamma}^{kl}\mathcal{D}_k\mathcal{D}_l\tilde{\gamma}_{ij} + \tilde{\gamma}_{ik}\mathcal{D}_j\tilde{\Gamma}^k + \tilde{\gamma}_{jk}\mathcal{D}_i\tilde{\Gamma}^k\right) + \mathcal{Q}_{ij}(\tilde{\boldsymbol{\gamma}},\boldsymbol{\mathcal{D}}\tilde{\boldsymbol{\gamma}})\\ & \quad -\frac{1}{3}\left(\mathcal{D}_k\tilde{\Gamma}^k + \mathcal{Q}(\tilde{\boldsymbol{\gamma}},\boldsymbol{\mathcal{D}}\tilde{\boldsymbol{\gamma}})\right)\tilde{\gamma}_{ij} - 2\tilde{D}_i\tilde{D}_j\ln\Psi + 4\tilde{D}_i\ln\Psi\,\tilde{D}_j\ln\Psi\\ & \quad\quad +\frac{2}{3}\left(\tilde{D}_k\tilde{D}^k\ln\Psi - 2\tilde{D}_k\ln\Psi\,\tilde{D}^k\ln\Psi\right)\tilde{\gamma}_{ij}\Bigg]\Bigg\}.\end{aligned} \tag{10.64}$$

$$\begin{aligned}\left(\frac{\partial}{\partial t} - \mathcal{L}_{\boldsymbol{\beta}}\right)\tilde{\Gamma}^i \;=\; & \frac{2}{3}\mathcal{D}_k\beta^k\,\tilde{\Gamma}^i + \tilde{\gamma}^{jk}\mathcal{D}_j\mathcal{D}_k\beta^i + \frac{1}{3}\tilde{\gamma}^{ij}\mathcal{D}_j\mathcal{D}_k\beta^k - 2\tilde{A}^{ij}\mathcal{D}_jN\\ & -2N\left[8\pi\Psi^4p^i - \tilde{A}^{jk}\Delta^i{}_{jk} - 6\tilde{A}^{ij}\mathcal{D}_j\ln\Psi + \frac{2}{3}\tilde{\gamma}^{ij}\mathcal{D}_jK\right],\end{aligned} \tag{10.65}$$

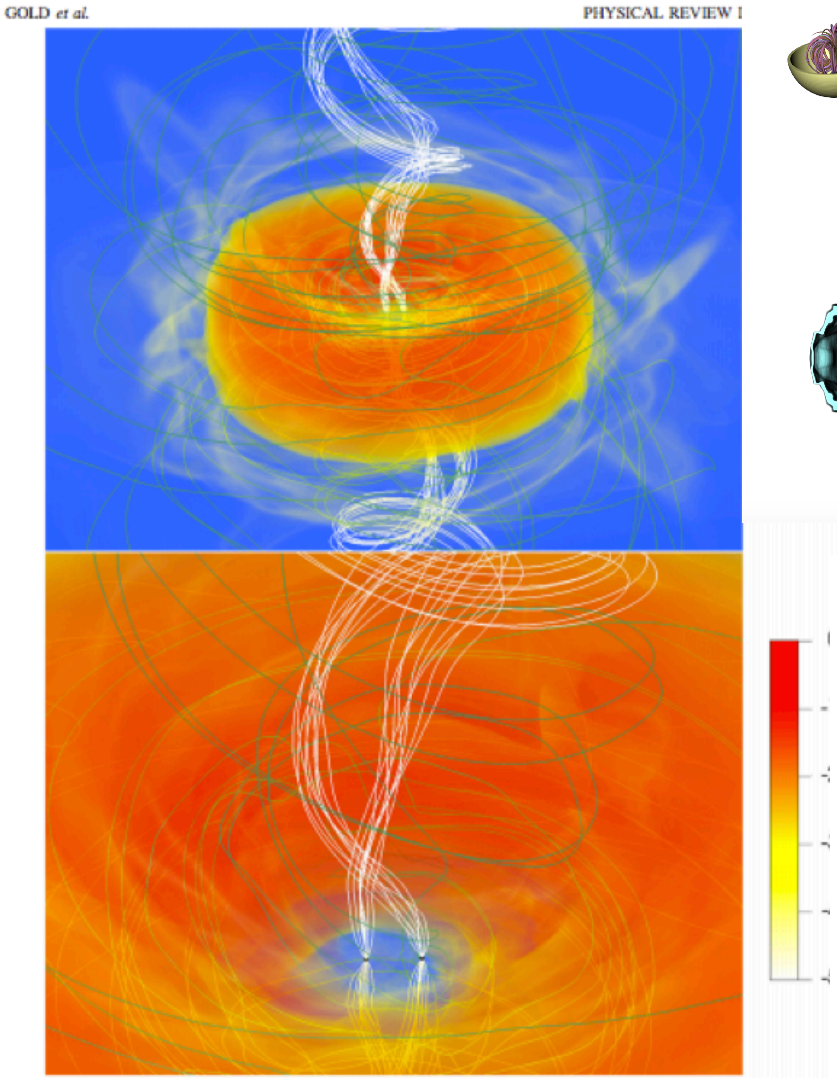
https://arxiv.org/abs/gr-qc/0703035

McLachlan

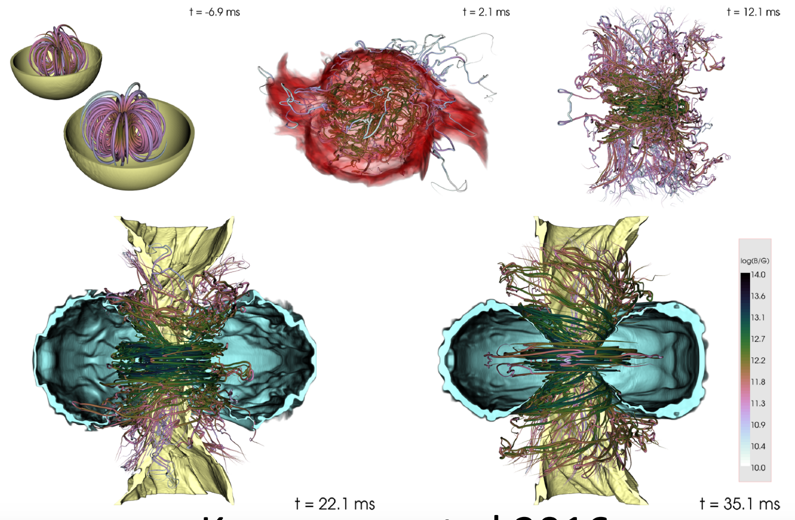
- Produced using the KRANC code: http://kranccode.org
- Can solve Einstein Equations using different formulations (BSSN and Z4)
- Implement finite difference schemes up to 8^{th} order
- Implements singularity-avoiding slicing conditions

GRMHD equations

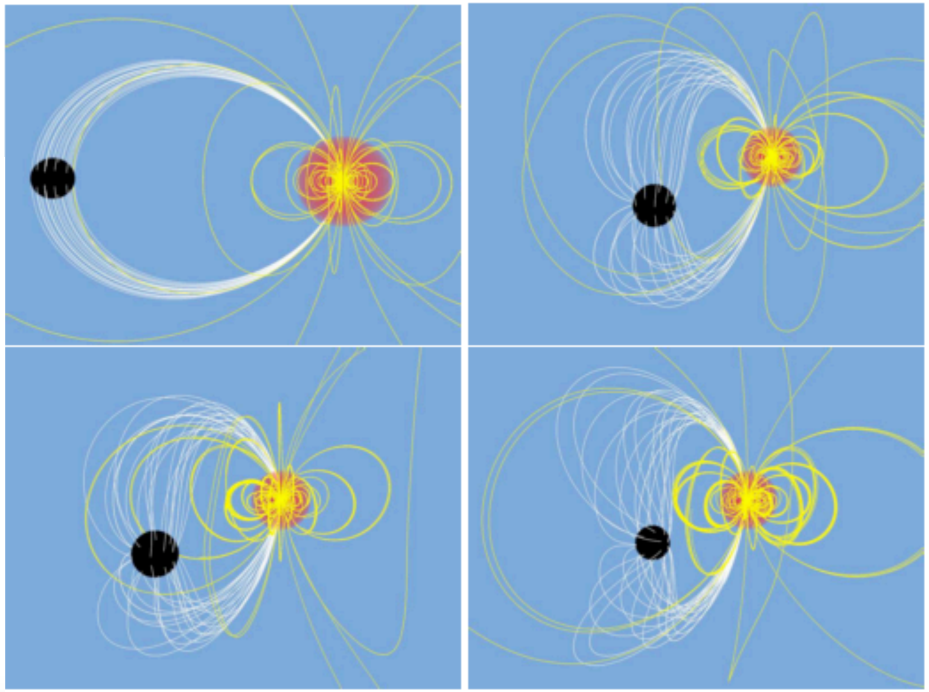
GRMHD APPLICATIONS



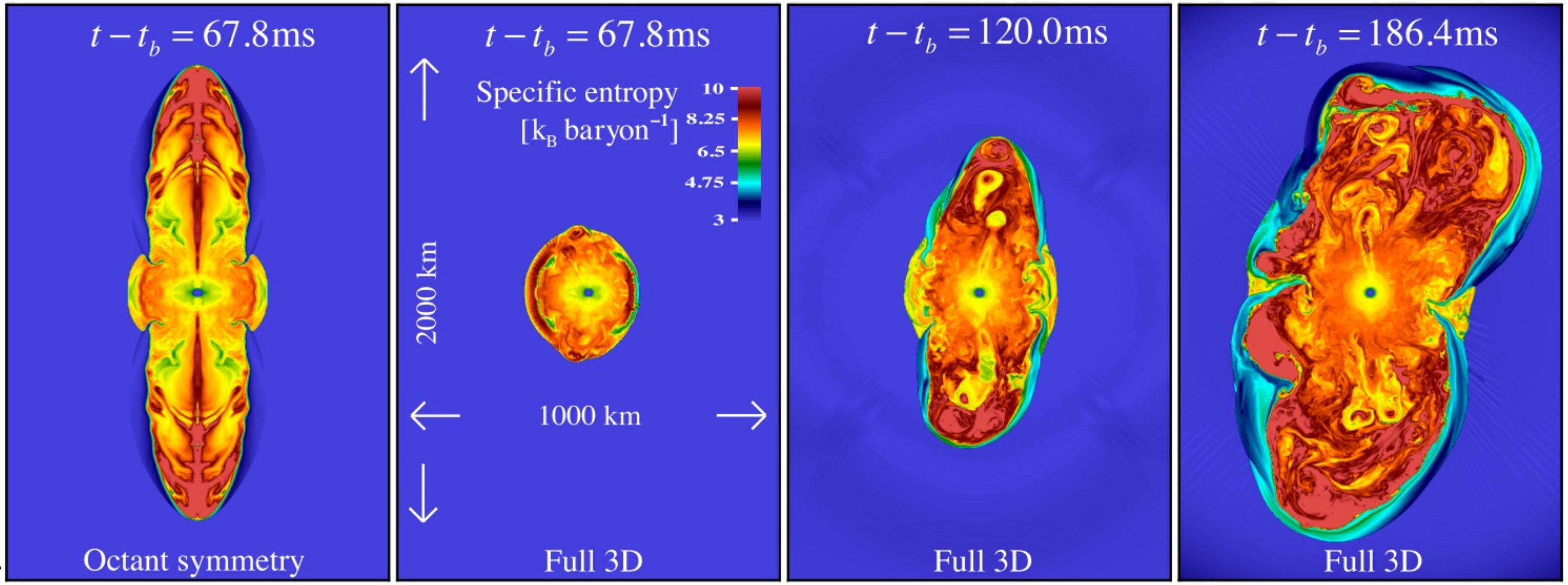
Gold et al 2014



Kawamura et al 2016



Paschalidis et al 2013



Moesta et al 2014

GRMHD equations

The evolution equations of the matter are given as usual by the conservation of energy-momentum and baryon number:

$$\begin{aligned}\nabla_\mu T^{\mu\nu} &= 0 \\ \nabla_\mu J^\mu &= 0 \\ J^\mu &\equiv \rho u^\mu\end{aligned}$$

$$T^{\mu\nu} = (\rho h + b^2) u^\mu u^\nu + \left(p + \frac{b^2}{2}\right) g^{\mu\nu} - b^\mu b^\nu$$

plus an Equation of State P=P(ρ,ε)

GRMHD equations

The system of equations is written in a conservative form (Valencia formulation, Anton et al 2006):

$$\left.\begin{array}{rcl}\nabla_\mu(\rho u^\mu) & = & 0 \\ \nabla_\mu T^{\mu\nu} & = & 0\end{array}\right\} \Rightarrow \frac{1}{\sqrt{-g}}\left(\frac{\partial\sqrt{\gamma}\mathbf{U}}{\partial t} + \frac{\partial\sqrt{-g}\mathbf{F}^i}{\partial x^i}\right) = \mathbf{S}$$

where **U** is the vector of conserved variables, $\mathbf{F}^i$ the fluxes, and **S** the source terms. They can then be solved using HRSC methods using approximate Riemann solvers.

Magnetic Field Evolution

The evolution of the magnetic field obeys **Maxwell's equations**:

$$\nabla_\nu {}^*F^{\mu\nu} = 0$$

that give the **divergence free condition**:

$$\nabla \cdot \left(\sqrt{\gamma}\vec{B}\right) = 0$$

and the equations for the **evolution of the magnetic field**:

$$\frac{\partial}{\partial t}\left(\sqrt{\gamma}\vec{B}\right) = \nabla \times \left[\left(\alpha\vec{v} - \vec{\beta}\right) \times \left(\sqrt{\gamma}\vec{B}\right)\right]$$

THE DIV(B)=0 PROBLEM

Several numerical techniques available to keep the magnetic field divergence-less:

1. Constrained Transport
   (Yee 1966, Evans & Hawley 1988, Balsara & Spicer 1999)

2. Hyperbolic Divergence Cleaning
   (Dedner et al 2002)

3. Vector Potential Evolution with Modified Lorenz Gauge
   (Etienne et al 2012, Farris et al 2012)

GAUGE ISSUES

$$B^i = \epsilon^{ijk}\partial_j A_k,$$

$$\partial_t A_i = \epsilon_{ijk} v^j B^k - \partial_i(\alpha\Phi - \beta^j A_j),$$

Algebraic gauge

$$\Phi = \frac{1}{\alpha}\beta^j A_j = -n^j A_j$$

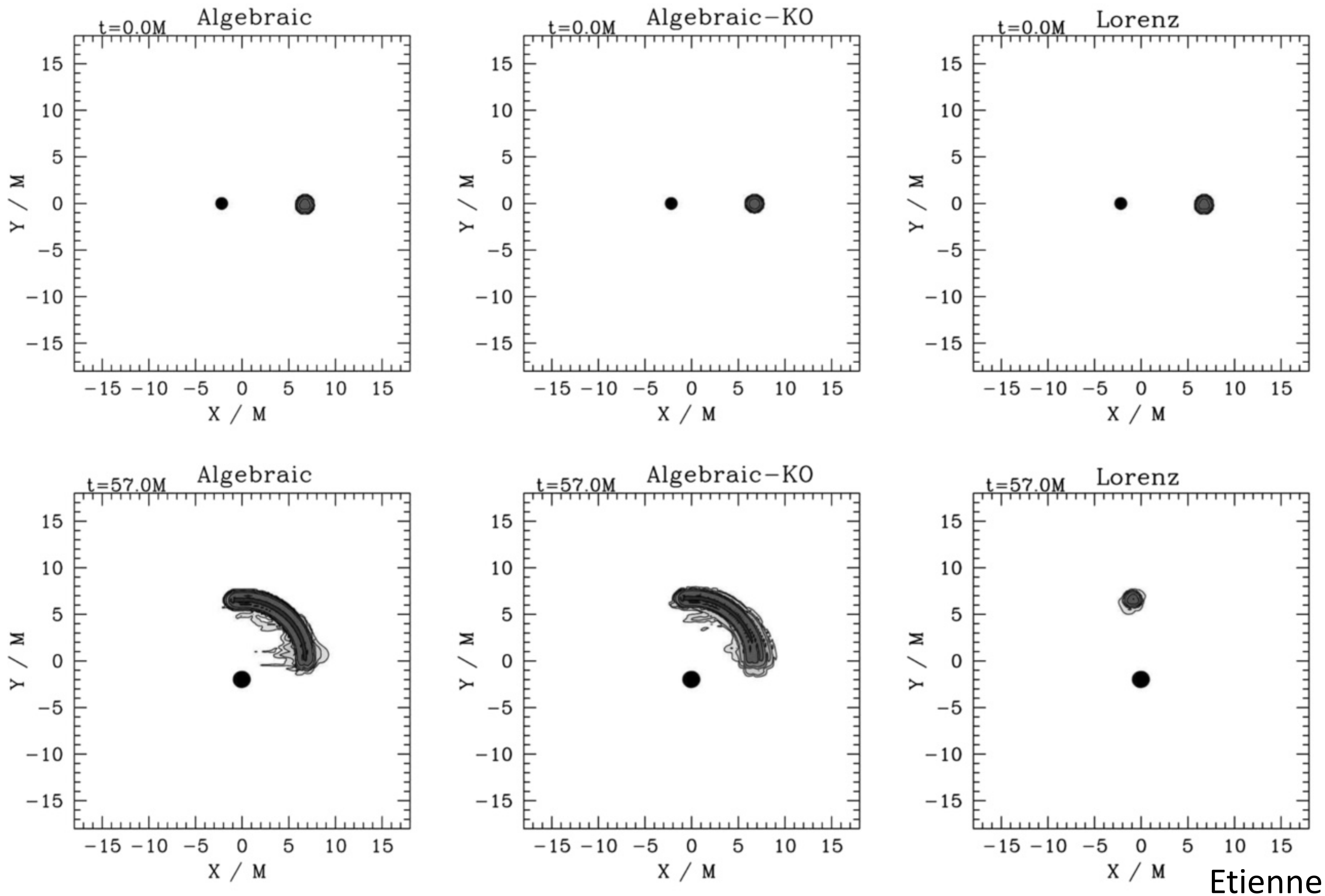
$$\partial_t A_i = \epsilon_{ijk} v^j B^k$$

Lorenz gauge

$$\nabla_\mu \mathcal{A}^\mu = 0,$$

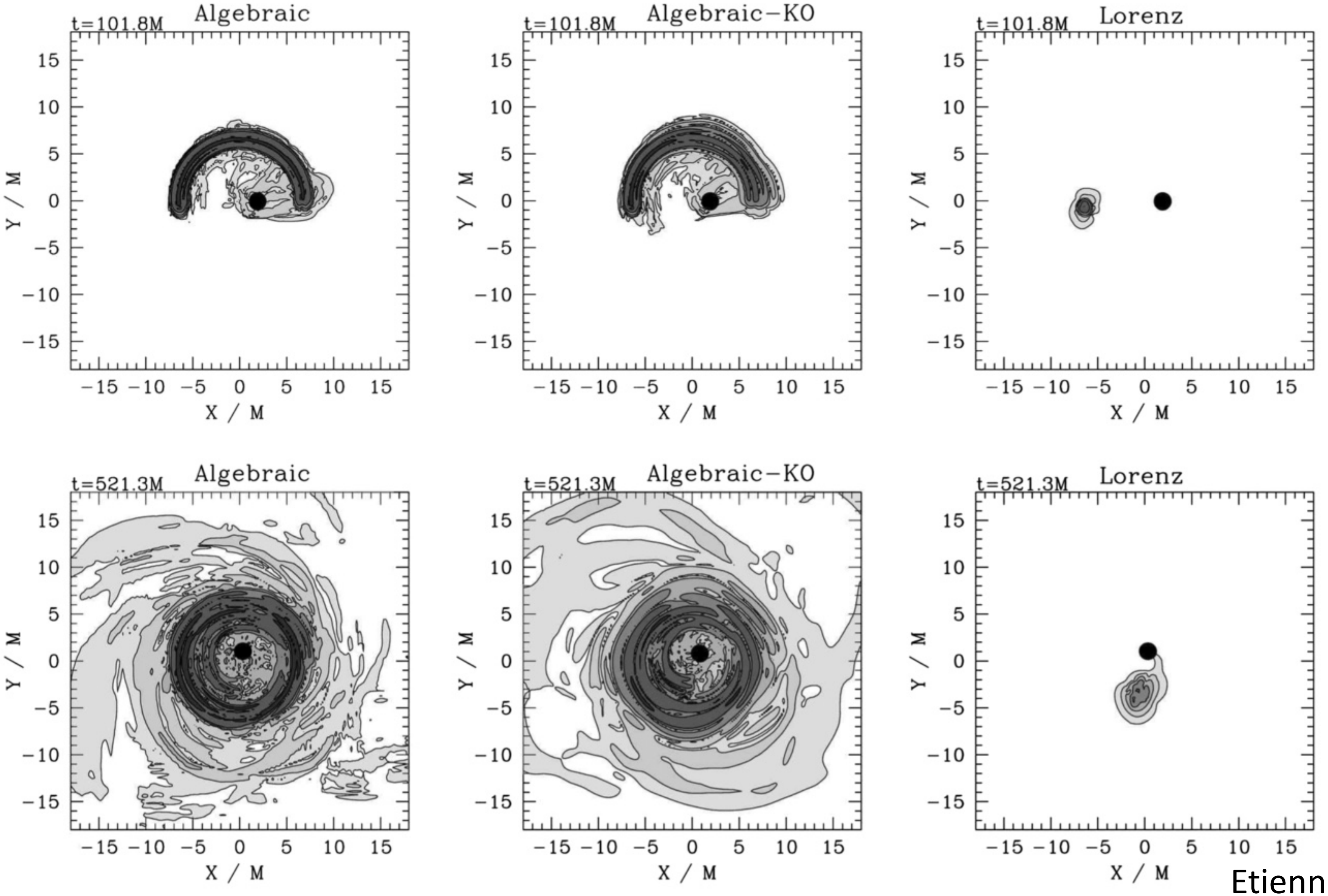
$$\partial_t(\sqrt{\gamma}\Phi) + \partial_j(\alpha\sqrt{\gamma}A^j - \sqrt{\gamma}\beta^j\Phi) = 0$$

GAUGE ISSUES



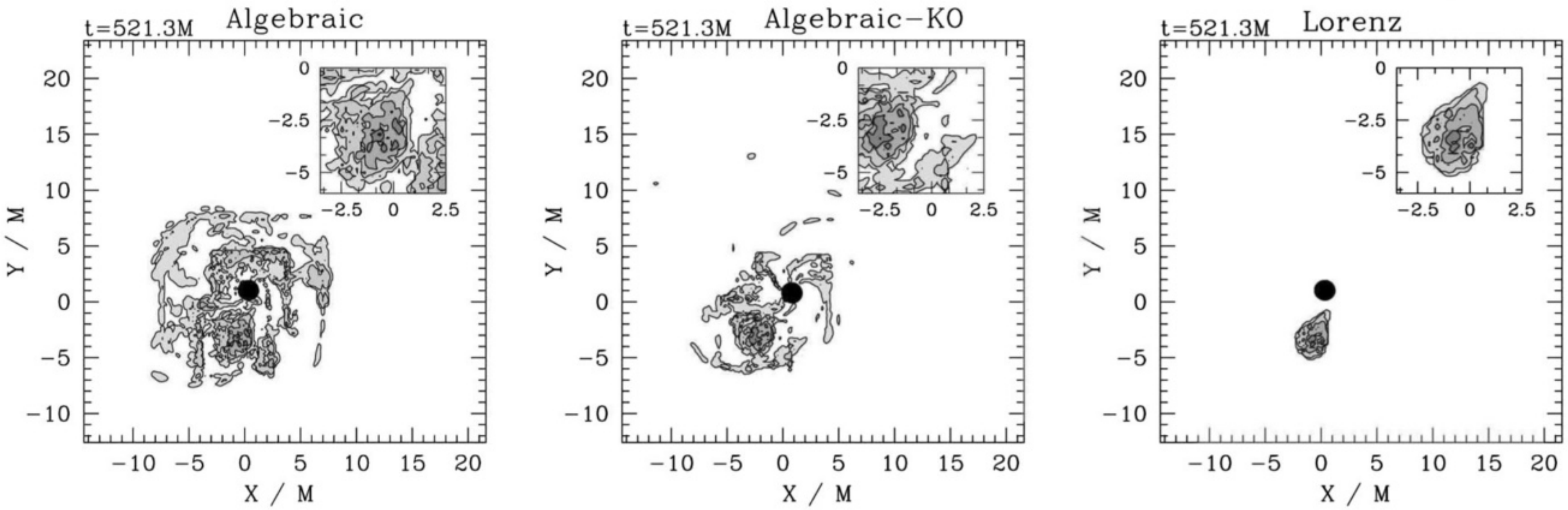
Etienne et al 2012

GAUGE ISSUES



Etienne et al 2012

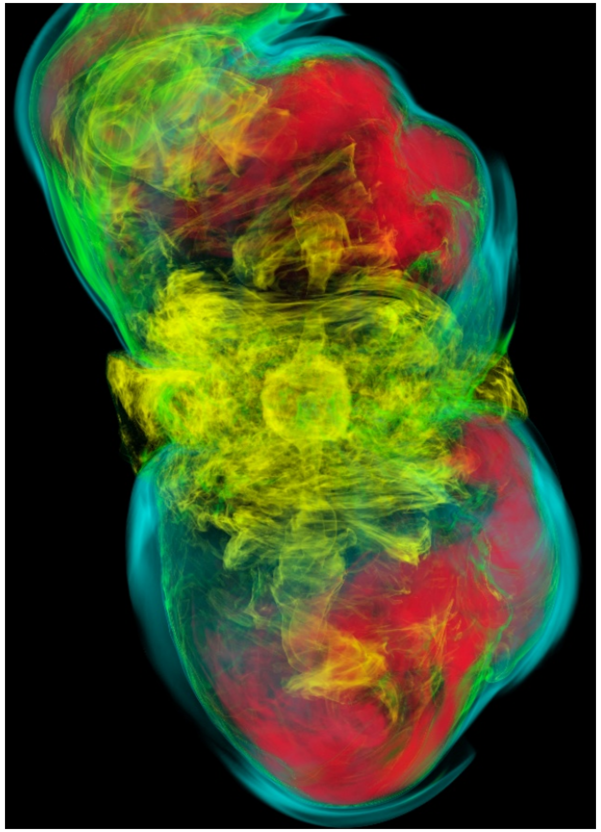
GAUGE ISSUES



Etienne et al 2012

Interpolation of vector potential "tail" at refinement level boundaries produces spurious magnetic field amplifications in the algebraic gauge.

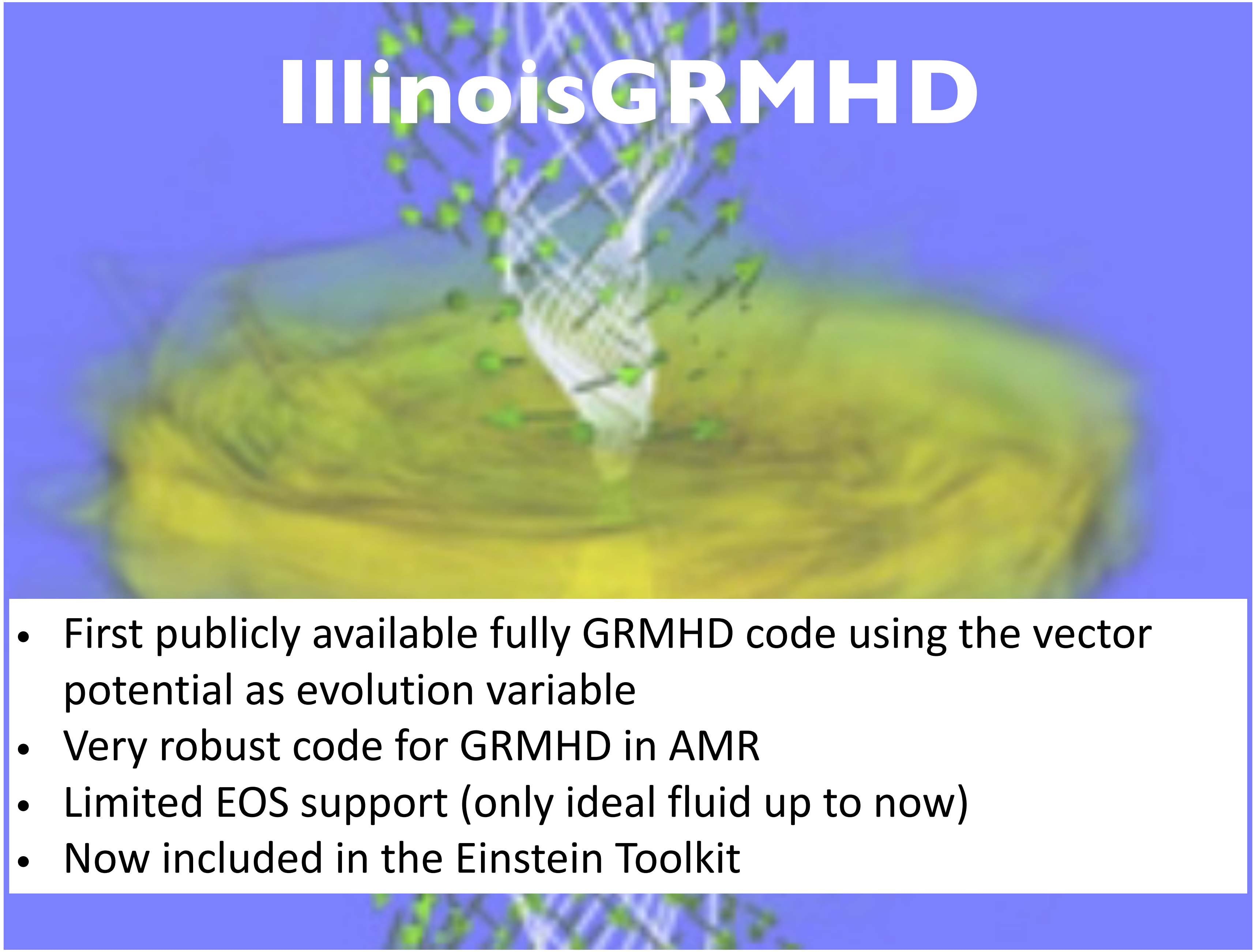
GRHYDRO



- First publicly available fully GRMHD code
- Based on the public version of the Whisky code
- Fully embedded in the Einstein Toolkit
- Uses Valencia formulation (as Whisky)

Moesta et al 2013: http://arxiv.org/abs/1304.5544

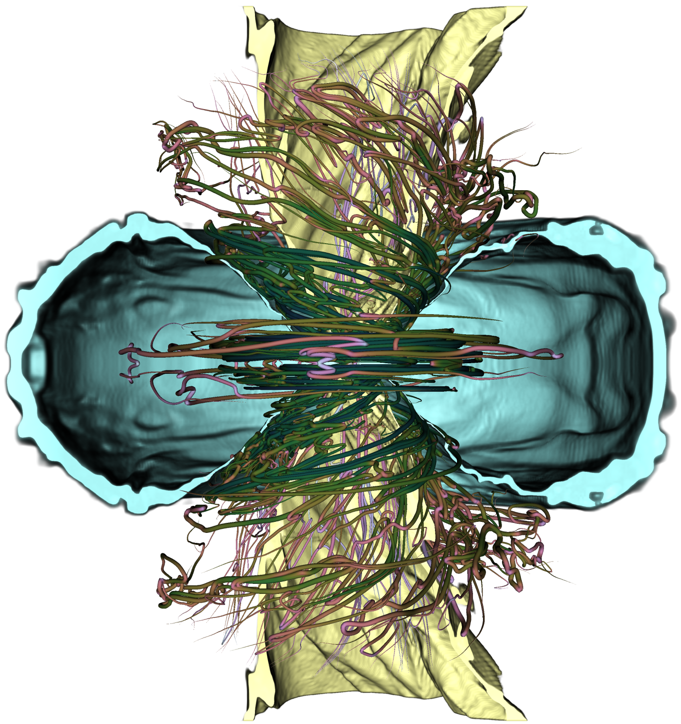
IllinoisGRMHD



- First publicly available fully GRMHD code using the vector potential as evolution variable
- Very robust code for GRMHD in AMR
- Limited EOS support (only ideal fluid up to now)
- Now included in the Einstein Toolkit

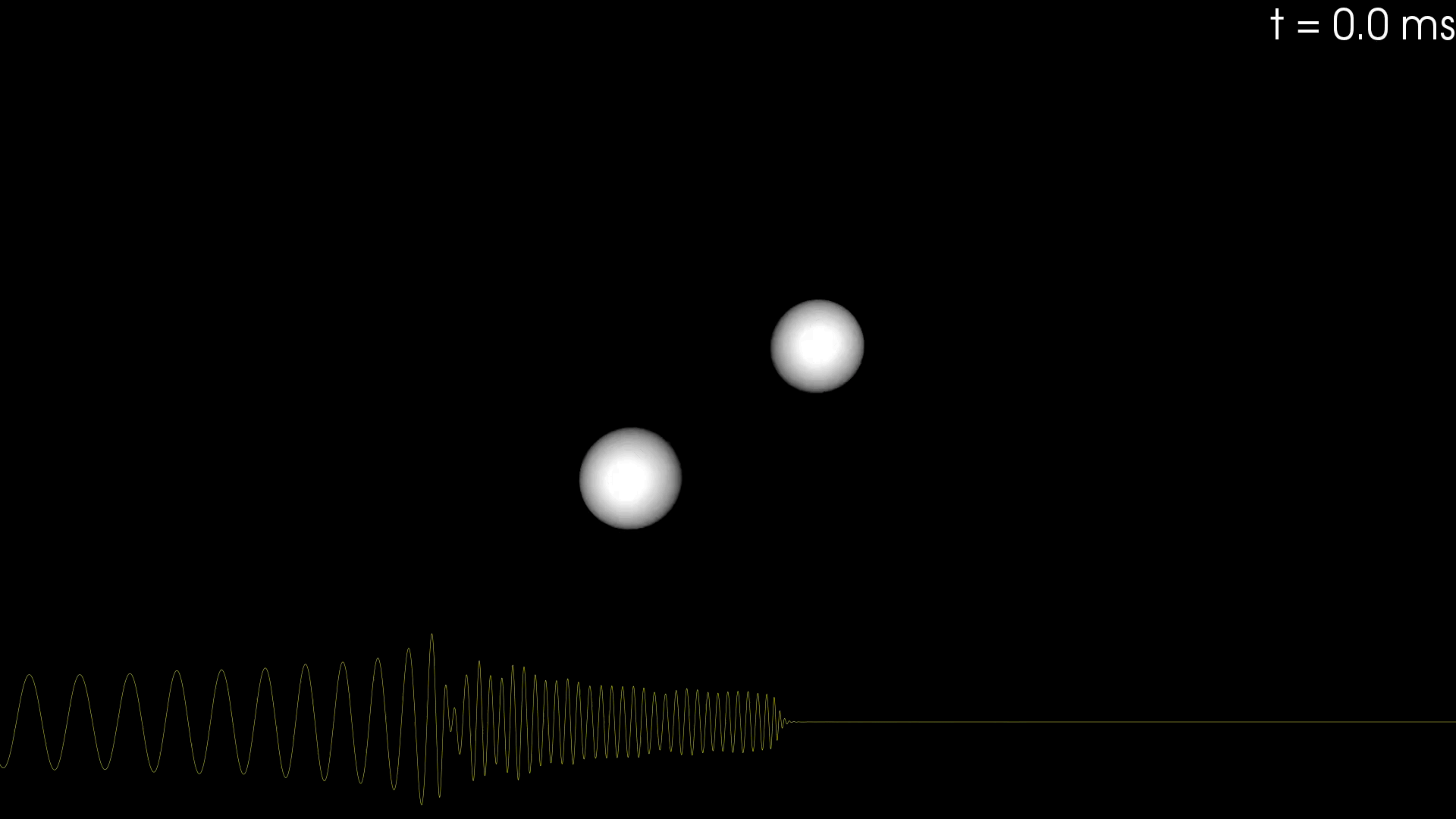
http://math.wvu.edu/~zetienne/ILGRMHD/index.html

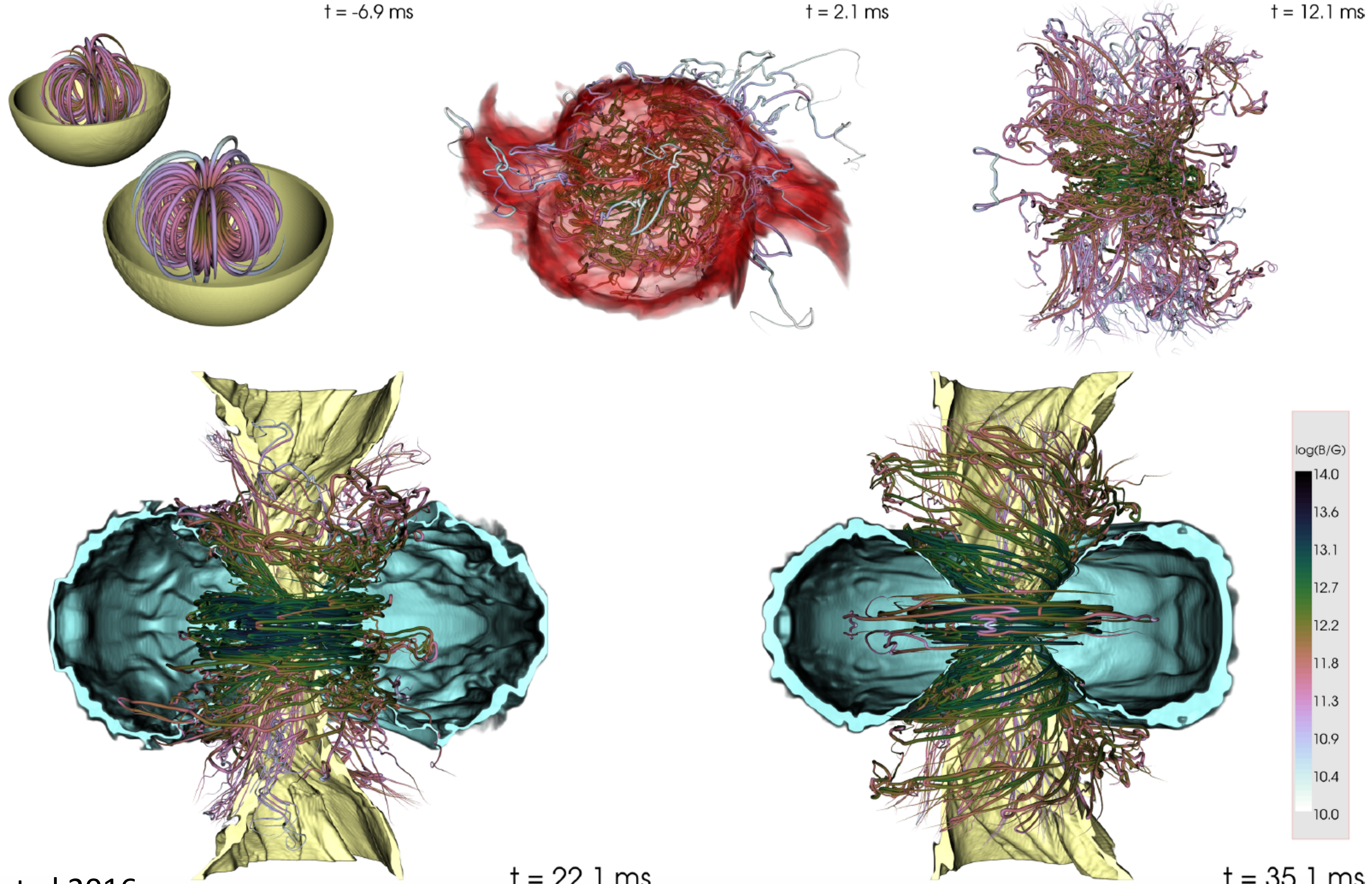
WhiskyMHD (private)



Kawamura et al 2016

- Currently maintained and developed in Trento (Giacomazzo)
- It makes use of the Einstein Toolkit
- Main applications are
  - NS-NS mergers (GWs and SGRBs)
  - Accretion onto Supermassive BH-BH mergers
- It supports piecewise-polytropic EOS
- Current developments include implementation of neutrino emission and finite temperature nuclear EOSs


t = 0.0 ms



Kawamura et al 2016

How to Download and Install the Einstein Toolkit

SimFactory
http://simfactory.org/

Command-line tools for setting up Cactus distribution and managing simulations on a variety of supercomputers, including PRACE machines.

Download (latest version) and install, e.g. on Ubuntu
(see https://docs.einsteintoolkit.org/et-docs/Simplified_Tutorial_for_New_Users):

1. curl -kLO https://raw.githubusercontent.com/gridaphobe/CRL/ET_2017_06/GetComponents

2. chmod a+x GetComponents

3. #make sure git and svn are installed on your system
   ./GetComponents –parallel https://bitbucket.org/einsteintoolkit/manifest/raw/ET_2017_06/einsteintoolkit.th

Thornlist

```
# CactusBase thorns
!TARGET   = $ARR
!TYPE     = git
!URL      = https://bitbucket.org/cactuscode/cactusbase.git
!REPO_PATH= $2
!REPO_BRANCH = $ET_RELEASE
!CHECKOUT =
CactusBase/Boundary
CactusBase/CartGrid3D
CactusBase/CoordBase
CactusBase/Fortran
CactusBase/InitBase
CactusBase/IOASCII
CactusBase/IOBasic
CactusBase/IOUtil
CactusBase/SymBase
CactusBase/Time
```

Thornlist

```
# EinsteinEOS
!TARGET   = $ARR
!TYPE     = git
!URL      = https://bitbucket.org/einsteintoolkit/einsteineos.git
!REPO_PATH= $2
!CHECKOUT =
EinsteinEOS/EOS_Hybrid
EinsteinEOS/EOS_IdealFluid
EinsteinEOS/EOS_Omni
EinsteinEOS/EOS_Polytrope

# EinsteinEvolve
!TARGET   = $ARR
!TYPE     = git
!URL      = https://bitbucket.org/einsteintoolkit/einsteinevolve.git
!REPO_PATH= $2
!REPO_BRANCH = $ET_RELEASE
!CHECKOUT =
EinsteinEvolve/GRHydro
EinsteinEvolve/GRHydro_InitData
```

```
# EinsteinInitialData
!TARGET   = $ARR
!TYPE     = git
!URL      = https://bitbucket.org/einsteintoolkit/einsteininitialdata.git
!REPO_PATH= $2
!REPO_BRANCH = $ET_RELEASE
!CHECKOUT =
EinsteinInitialData/DistortedBHIVP
EinsteinInitialData/Exact
EinsteinInitialData/Hydro_InitExcision
EinsteinInitialData/IDAnalyticBH
EinsteinInitialData/IDAxiBrillBH
EinsteinInitialData/IDAxiOddBrillBH
EinsteinInitialData/IDBrillData
EinsteinInitialData/IDConstraintViolate
EinsteinInitialData/IDFileADM
EinsteinInitialData/IDLinearWaves
EinsteinInitialData/Meudon_Bin_BH
EinsteinInitialData/Meudon_Bin_NS
EinsteinInitialData/Meudon_Mag_NS
EinsteinInitialData/NoExcision
EinsteinInitialData/RotatingDBHIVP
EinsteinInitialData/TOVSolver
EinsteinInitialData/TwoPunctures
```

Preparing your machine (e.g., Ubuntu)

```
# Configuration for an Ubuntu installation, assuming the following
# list of packages is installed:
#
#    build-essential perl python gfortran g++ libmpich-dev
#
# In addition, installing the following list of packaed will prevent
# Cactus from compiling its own versions of these libraries:
#
#   libfftw3-dev libgsl-dev libatlas-base-dev libjpeg-dev libssl-dev
#   libhdf5-dev hdf5-tools libnuma-dev libltdl-dev libhwloc-dev zlib1g-dev
#
# Tools like GetComponents and Simfactory like to have the following list
# installed too
#
#   python subversion git
```

Download (latest version) and install, e.g. on Ubuntu
(see https://docs.einsteintoolkit.org/et-docs/Simplified_Tutorial_for_New_Users):

1. curl -kLO https://raw.githubusercontent.com/gridaphobe/CRL/ET_2017_06/GetComponents

2. chmod a+x GetComponents

3. #make sure git and svn are installed on your system
   ./GetComponents –parallel https://bitbucket.org/einsteintoolkit/manifest/raw/ET_2017_06/einsteintoolkit.th

4. cd Cactus

5. ./simfactory/bin/sim setup-silent --optionlist=ubuntu.cfg --runscript debian.sh

6. ./simfactory/bin/sim build --mdbkey make 'make -j2' --thornlist=manifest/einsteintoolkit.th

7. # start simulation
   ./simfactory/bin/sim submit static_tov --parfile=par/static_tov_small.par --procs=1 --walltime=8:0:0

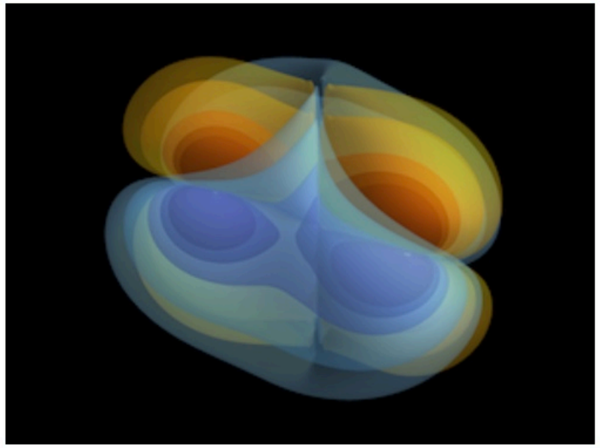
Current List of HPC machines supported

angel.ini
bethe.ini
bluewaters-cray.ini
bluewaters-pgi.ini
bluewaters.ini
carver.ini
cfermi.ini
comet.ini
compute.ini
compute1.ini
compute2.ini
compute20.ini
compute3.ini
compute4.ini
marconiA1.ini
marconiA2.ini
marconi_17.ini
marenostrum.ini
mars.ini
mike.ini
minerva.ini
mp2.ini
nvidia.ini
nvidia1.ini
oliver.ini
orca.ini
osx-homebrew.ini
osx-macports.ini
painter.ini
cori.ini
datura-gpu.ini
datura.ini
debian.ini
draco.ini
edison.ini
eric.ini
fedora.ini
fermi.ini
fionn.ini
fuchs.ini
galileo.ini
generic.ini
golub.ini
pandora.ini
philip.ini
pi0005009.ini
qb.ini
requin.ini
s-kraken.ini
saw.ini
sciama.ini
scinet.ini
shelob.ini
smic.ini
spine.ini
stampede-hybrid.ini
stampede-knl.ini
stampede-mic.ini
gpc.ini
guillimin.ini
gw.ini
hal1.ini
hal2.ini
hedges.ini
holodeck.ini
hopper.ini
hydra.ini
intrepid.ini
jacobi-uwm.ini
jyc.ini
loewe.ini
louie.ini
stampede.ini
sunnyvale.ini
supermuc.ini
surveyor.ini
tesla.ini
tezpur.ini
tianhe1a.ini
titan.ini
topf.ini
ubuntu.ini
vesta.ini
wheeler.ini
zwicky-intel14.ini
zwicky.ini

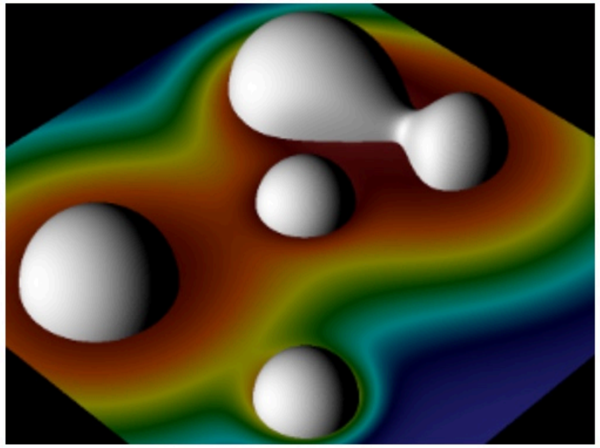
Einstein Toolkit Gallery

This page contains example simulations that can be run using the Einstein Toolkit, either exclusively or in combination with external codes. The parameter files and thornlists required to reproduce the simulations are provided. Some examples also include images and movies, analysis and visualisation scripts, example simulation data, and tutorials.

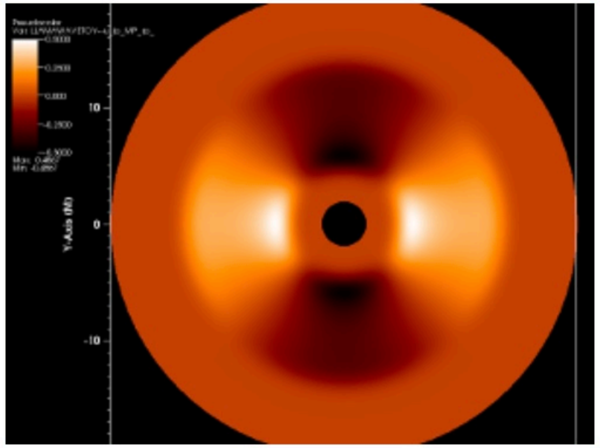
Binary black hole GW150914



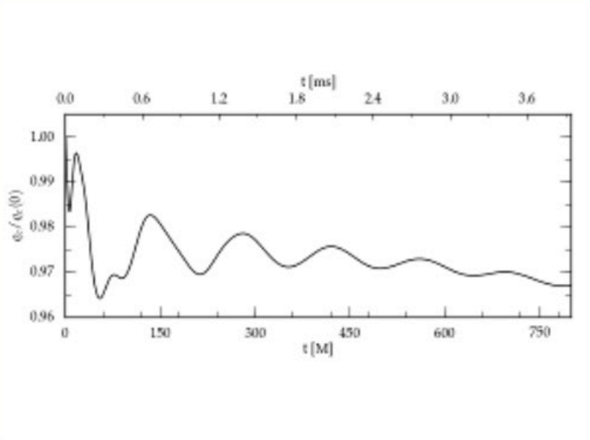
Poisson equation



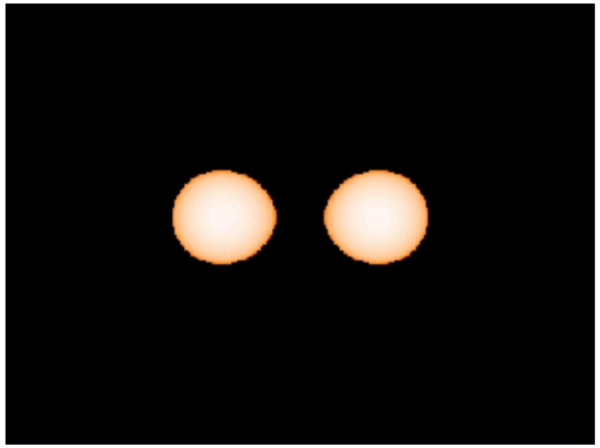
Multi Patch Energy Equation



Single, stable neutron star



Binary neutron star



The Einstein Toolkit has been supported by NSF 1550551/1550461/1550436/1550514, NSF 1212401/1212426/1212433/1212460, NSF 0903973/0903782/0904015 (CIGR), 0701566/0855892 (XiRel), 0721915

References

- Einstein Toolkit Webpage: http://einsteintoolkit.org
- Main Publications presenting the toolkit:
  - Loeffler et al 2012: http://arxiv.org/abs/1111.3344
  - Moesta et al 2013: http://arxiv.org/abs/1304.5544
  - Zilhao and Loeffler 2013: http://arxiv.org/abs/1305.5299
- Visualization Tools:
  - PostCactus & SimRep: https://bitbucket.org/DrWhat/pycactuset
  - Visit: https://visit.llnl.gov/
  - YT: http://yt-project.org/
- Every year workshops and (sometimes) schools are organized in EU and USA:
  - http://www.ncsa.illinois.edu/Conferences/ETK17/
  - http://grg.uib.es/EinsteinToolkit2017/