GRHayL & IllinoisGRMHD: cross-infrastructure GRMHD code development

Samuel Cupp
In collaboration with
Leo Werneck, Terrence Pierre Jacques,
and Zachariah Etienne

einstein toolkit

NRPy

US ETK Meeting 2024
Baton Rouge, LA

Funding acknowledgements
NASA TCAN-80NSSC18K1488
NSF PHY-2110352

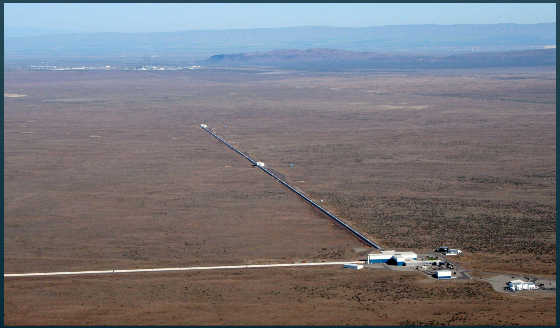
University of Idaho
Department of Physics

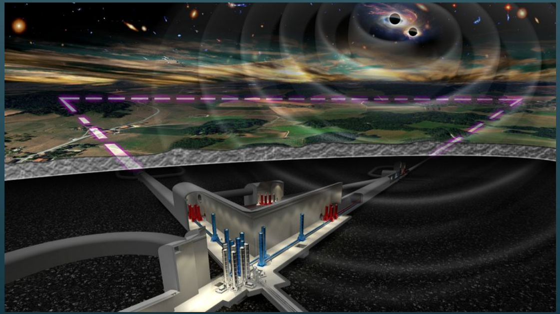
West Virginia University
DEPARTMENT OF PHYSICS AND ASTRONOMY

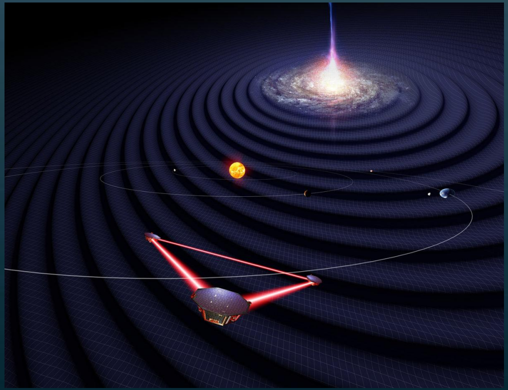
Challenges in Numerical Relativity

Improved gravitational wave detectors
New land- and space-based detectors

→ Higher accuracy requirements



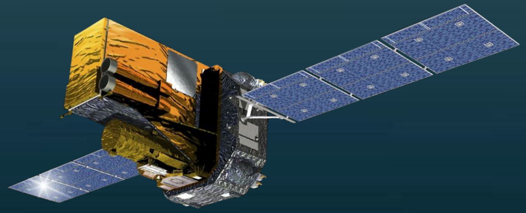


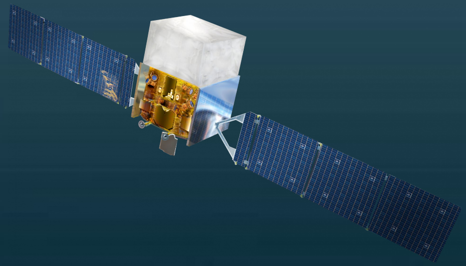


Challenges in Numerical Relativity

Improved gravitational wave detectors
New land- and space-based detectors → Higher accuracy requirements

Multi-messenger astronomy → Additional physics (e.g. neutrinos)





Challenges in Numerical Relativity

Improved gravitational wave detectors
New land- and space-based detectors
→ Higher accuracy requirements

Multi-messenger astronomy
→ Additional physics (e.g. neutrinos)

Higher computational cost!

Challenges in Numerical Relativity

Improved gravitational wave detectors
New land- and space-based detectors
→ Higher accuracy requirements

Multi-messenger astronomy → Additional physics (e.g. neutrinos)

Rapidly changing HPC landscape
CPU→GPU
→ Requires infrastructure changes

Challenges in Numerical Relativity

Additional physics

Higher accuracy requirements

Infrastructure changes

GRMHD codes must rapidly adapt to changing infrastructures and community needs

Challenges in Numerical Relativity

- Additional physics
- Higher accuracy requirements
- Infrastructure changes

GRMHD codes must rapidly adapt to changing infrastructures and community needs

Our solution: a modular, extensible GRMHD code library

What is *IllinoisGRMHD*?

- Numerical relativity code for simulating general relativistic magneto-hydrodynamics (GRMHD)
- Well-tested GRMHD evolution thorn in the Einstein Toolkit

- Refactored from closed-source code from the Illinois NR group
- Agrees with old code to round-off level
- Designed to be cleaner, better documented

What is *IllinoisGRMHD*?

Internal Illinois Group code
Duez *et al.* (2005)

What is *IllinoisGRMHD*?

Internal Illinois Group code
Duez *et al.* (2005)

Open-sourced, code cleanup/optimization

IllinoisGRMHD
Etienne *et al.* (2015)

What is *IllinoisGRMHD*?

Internal Illinois Group code
Duez *et al.* (2005)

Open-sourced, code cleanup/optimization

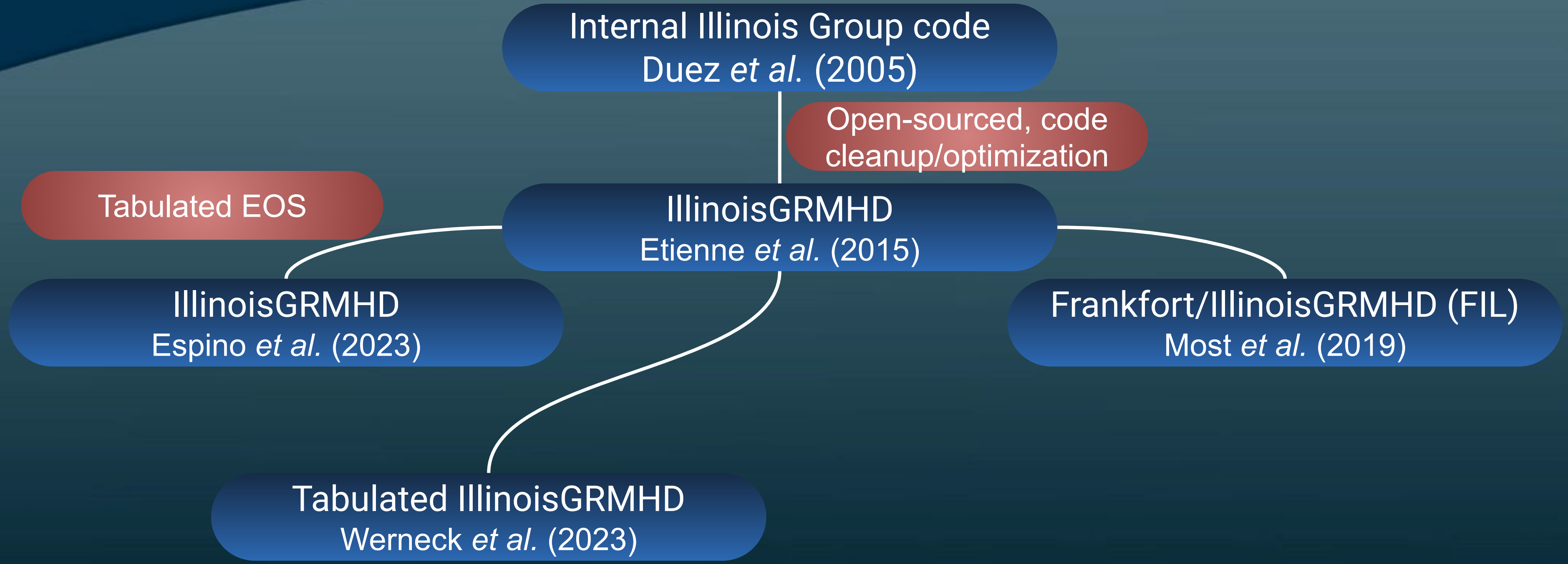
IllinoisGRMHD
Etienne *et al.* (2015)

IllinoisGRMHD
Espino *et al.* (2023)

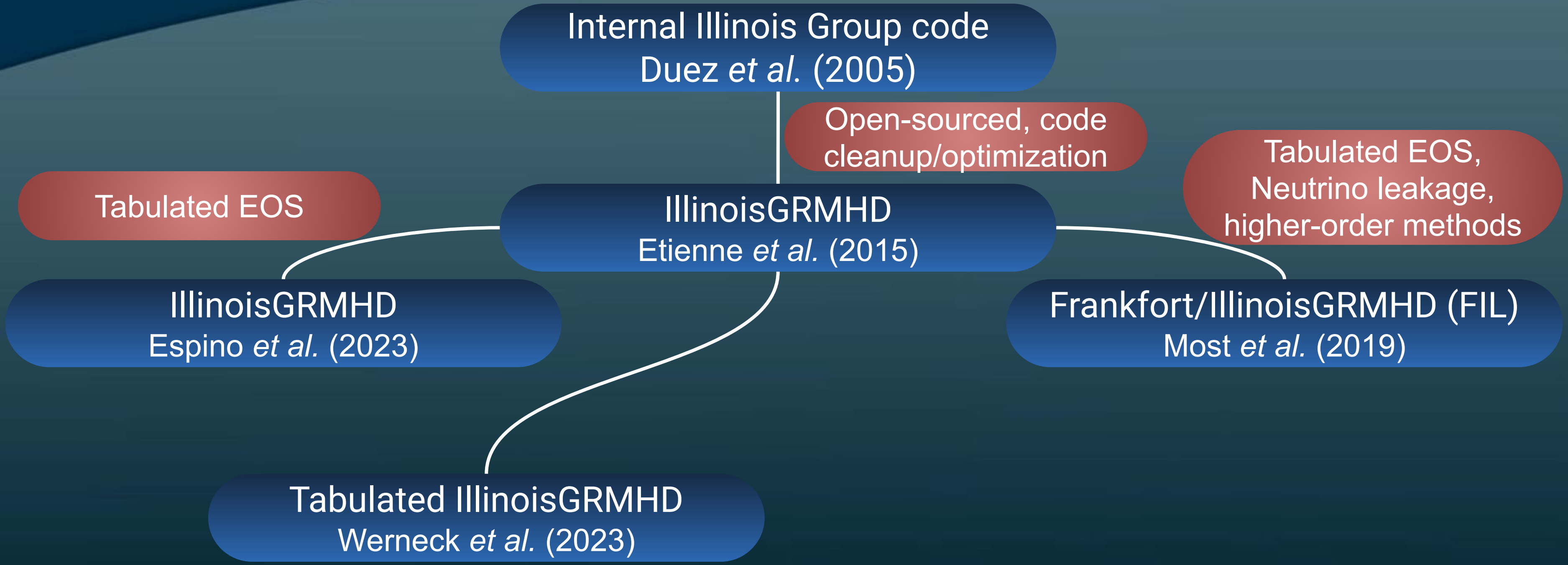
Frankfort/IllinoisGRMHD (FIL)
Most *et al.* (2019)

Tabulated IllinoisGRMHD
Werneck *et al.* (2023)

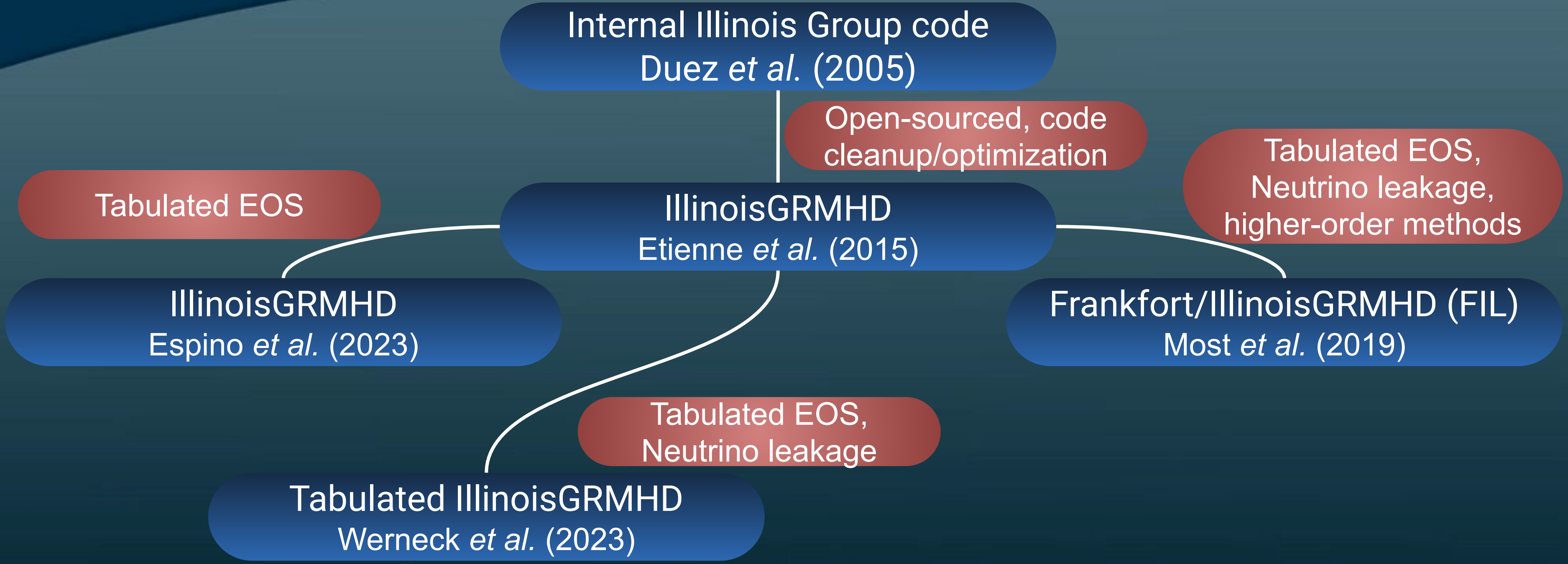
What is *IllinoisGRMHD*?



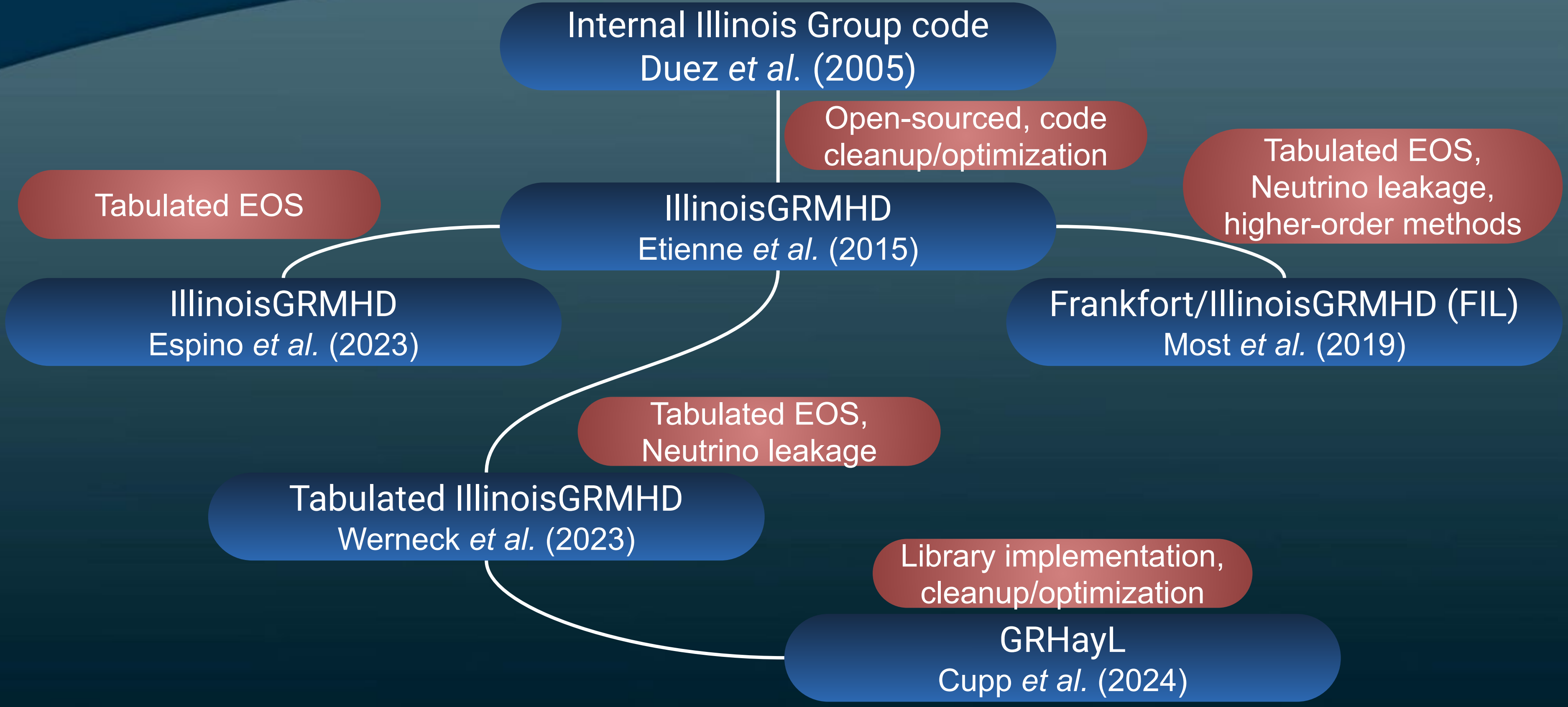
What is *IllinoisGRMHD*?



What is *IllinoisGRMHD*?



What is *IllinoisGRMHD*?



What is *GRHayL*?

General Relativistic Hydrodynamics Library (GRHayL)

- Refactors IllinoisGRMHD into modular components
- Library functions are purely pointwise or stencil-wise, ensuring infrastructure-agnosticism

What is GRHayL?

General Relativistic Hydrodynamics Library (GRHayL)

- Refactors IllinoisGRMHD into modular components
- Library functions are purely pointwise or stencil-wise, ensuring infrastructure-agnosticism

GRHayL is the *future* of IllinoisGRMHD

- Better separation of code into independent modules
- Better documentation
- Not bound to a specific infrastructure!

What is *GRHayL*?

General Relativistic Hydrodynamics Library (GRHayL)

- Refactors IllinoisGRMHD into modular components
- Library functions are purely pointwise or stencil-wise, ensuring infrastructure-agnosticism

GRHayL is the *future* of IllinoisGRMHD

- Better separation of code into independent modules
- Better documentation
- Not bound to a specific infrastructure!

Automated continuous integration (CI) testing

- Validates output against old IllinoisGRMHD when possible
- Unit tests cover 97% of the GRHayL codebase

Modularity and Extensibility

https://www.teepublic.com/tapestry/3141846-tangled-octopus

Modularity and Extensibility

https://www.teepublic.com/tapestry/3141846-tangled-octopus

https://www.deviantart.com/sylviaritter/art/Cosmic-Cuttlefish-766515479

Modularity and Extensibility

Core Code Infrastructure

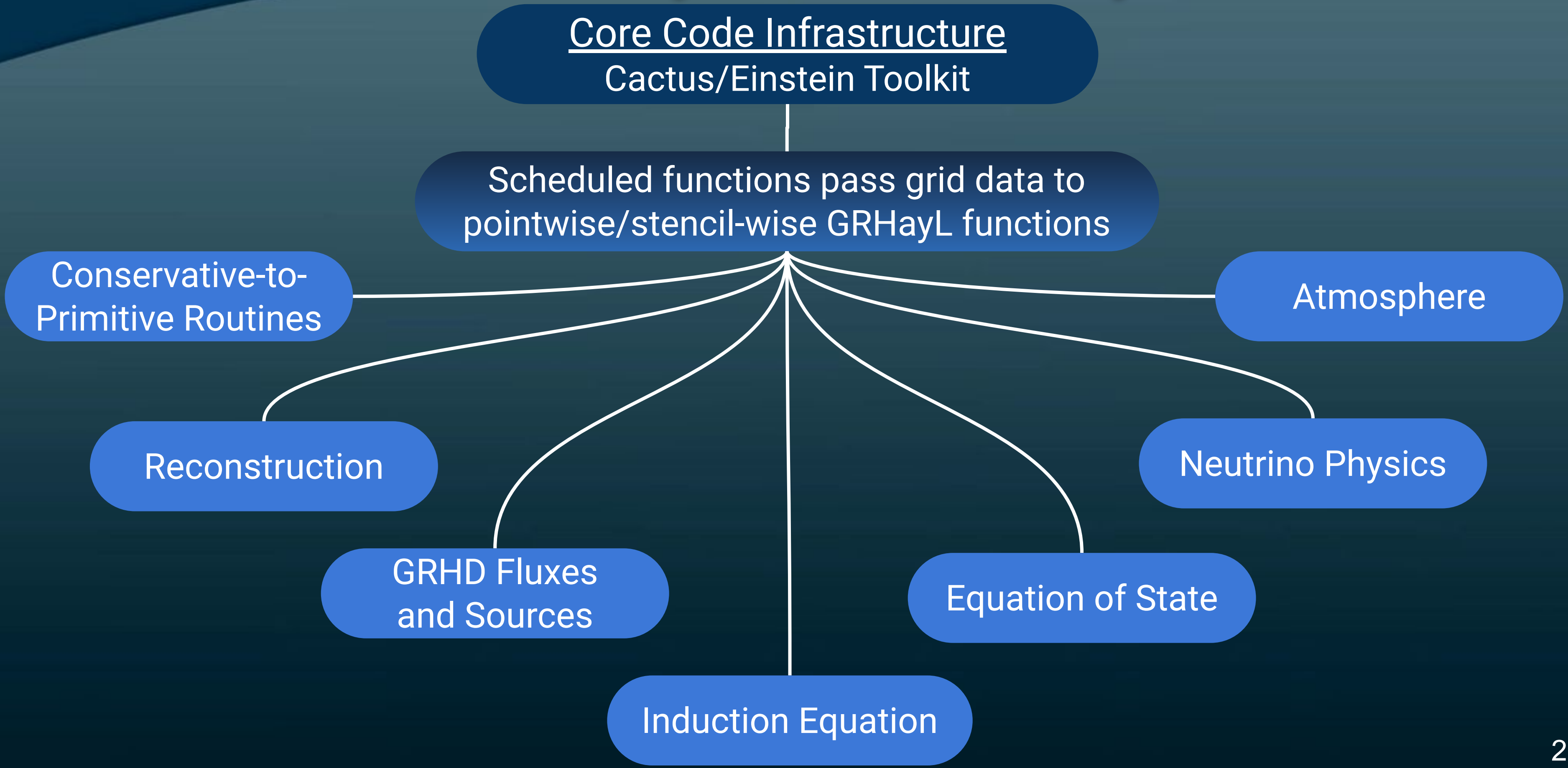
Cactus/Einstein Toolkit

Modularity and Extensibility

Core Code Infrastructure
Cactus/Einstein Toolkit

Scheduled functions pass grid data to pointwise/stencil-wise GRHayL functions

Modularity and Extensibility



Modularity and Extensibility

GRHayL Features

- *Atmosphere* module currently only provides constant atmosphere prescription
- Radial falloff prescription is in progress

GRHayL Features

- *Atmosphere* module currently only provides constant atmosphere prescription
- Radial falloff prescription is in progress

- *Neutrinos* module (to be renamed *Radiation*) provides core algorithms for neutrino leakage
- Neutrino M1-closure scheme planned, as well as photon transport schemes

GRHayL Features

- *Atmosphere* module currently only provides constant atmosphere prescription
- Radial falloff prescription is in progress

- *Neutrinos* module (to be renamed *Radiation*) provides core algorithms for neutrino leakage
- Neutrino M1-closure scheme planned, as well as photon transport schemes

- *Flux_Source* module provides functions for computing
- Characteristic speeds
- Source terms and fluxes for conservative variables, including entropy and electron fraction

GRHayL Features

- *EOS* module provides simple Gamma-law, hybrid piecewise polytrope, and tabulated equations of state
- Function calls abstracted to allow for swapping of functions for more modularity

GRHayL Features

- *EOS* module provides simple Gamma-law, hybrid piecewise polytrope, and tabulated equations of state
- Function calls abstracted to allow for swapping of functions for more modularity

- *Con2Prim* module provides several options for gamma-law/hybrid and tabulated EOS
- Up to 3 backups can be selected for use by built-in backup routine

GRHayL Features

- *EOS* module provides simple Gamma-law, hybrid piecewise polytrope, and tabulated equations of state
- Function calls abstracted to allow for swapping of functions for more modularity

- *Con2Prim* module provides several options for gamma-law/hybrid and tabulated EOS
- Up to 3 backups can be selected for use by built-in backup routine

- *Induction* module provides function to facilitate computing the right-hand sides for evolving the staggered vector potential and scalar potential

GRHayL Features

- *Reconstruction* module provides methods for computing the face values of cell-centered quantities
- Currently includes minmod, mc, superbee, and PPM methods

GRHayL in the Toolkit: GRHayLib

- GRHayL is provided by the GRHayLib thorn
- GRHayL options controlled through GRHayLib parameters

GRHayL in the Toolkit: GRHayLib

- GRHayL is provided by the GRHayLib thorn
- GRHayL options controlled through GRHayLib parameters

- Thorns using GRHayL simply inherit GRHayLib and add “USES INCLUDE: GRHayLib.h” in the interface.ccl
- Any files using GRHayL functions include the header file

GRHayL in the Toolkit: GRHayLib

- GRHayL is provided by the GRHayLib thorn
- GRHayL options controlled through GRHayLib parameters

- Thorns using GRHayL simply inherit GRHayLib and add "USES INCLUDE: GRHayLib.h" in the interface.ccl
- Any files using GRHayL functions include the header file

- Works for Carpet and CarpetX, but functions currently only available on the host

GRHayL in the Toolkit: IllinoisGRMHD

- IllinoisGRMHD refactored to use GRHayLib

GRHayL in the Toolkit: IllinoisGRMHD

- IllinoisGRMHD refactored to use GRHayLib
- Removed helper thorns ID_converter_ILGRMHD and Convert_to_HydroBase

GRHayL in the Toolkit: IllinoisGRMHD

- IllinoisGRMHD refactored to use GRHayLib
- Removed helper thorns ID_converter_ILGRMHD and Convert_to_HydroBase
- Combined official release code (simple polytrope hybrid EOS) with Werneck *et al.* code (piecewise polytropic EOS & tabulated EOS)

GRHayL in the Toolkit: IllinoisGRMHD

- IllinoisGRMHD refactored to use GRHayLib
- Removed helper thorns ID_converter_ILGRMHD and Convert_to_HydroBase
- Combined official release code (simple polytrope hybrid EOS) with Werneck *et al.* code (piecewise polytropic EOS & tabulated EOS)

- Con2Prim backup options and ordering now user-controlled

GRHayL in the Toolkit: IllinoisGRMHD

- IllinoisGRMHD refactored to use GRHayLib
- Removed helper thorns ID_converter_ILGRMHD and Convert_to_HydroBase
- Combined official release code (simple polytrope hybrid EOS) with Werneck *et al.* code (piecewise polytropic EOS & tabulated EOS)

- Con2Prim backup options and ordering now user-controlled
- Higher-order metric derivatives → demonstrably more accurate simulations
- Code less sensitive to small changes to initial data

GRHayL in the Toolkit: IllinoisGRMHD

- IllinoisGRMHD refactored to use GRHayLib
- Removed helper thorns ID_converter_ILGRMHD and Convert_to_HydroBase
- Combined official release code (simple polytrope hybrid EOS) with Werneck *et al.* code (piecewise polytropic EOS & tabulated EOS)

- Con2Prim backup options and ordering now user-controlled
- Higher-order metric derivatives → demonstrably more accurate simulations
- Code less sensitive to small changes to initial data
- Code improvements removed ~40% of grid functions

GRHayL in the Toolkit: GRHayLHD(X)

- GRHayLHD provides a purely hydrodynamic version of IllinoisGRMHD
- Reduces number of grid functions by ~80% compared to old IllinoisGRMHD

GRHayL in the Toolkit: GRHayLHD(X)

- GRHayLHD provides a purely hydrodynamic version of IllinoisGRMHD
- Reduces number of grid functions by ~80% compared to old IllinoisGRMHD

- GRHayLHDX implements an identical evolution code for CarpetX
- Currently only runs on the host
- Once GRHayL supports GPUs, this thorn can immediately turn on GPU capability

GRHayL in the Toolkit: GRHayLID(X)

- GRHayLID provides simple initial data options
  - 1D sound wave, shock tube, and Balsara tests
  - Constant density sphere
  - Isotropic gas

GRHayL in the Toolkit: GRHayLID(X)

- GRHayLID provides simple initial data options
  - 1D sound wave, shock tube, and Balsara tests
  - Constant density sphere
  - Isotropic gas

- Provides option to enforce beta equilibrium on tabulated EOS initial data (including other ID thorns)
- Can compute entropy from initial data (including other ID thorns)

GRHayL in the Toolkit: GRHayLID(X)

- GRHayLID provides simple initial data options
  - 1D sound wave, shock tube, and Balsara tests
  - Constant density sphere
  - Isotropic gas

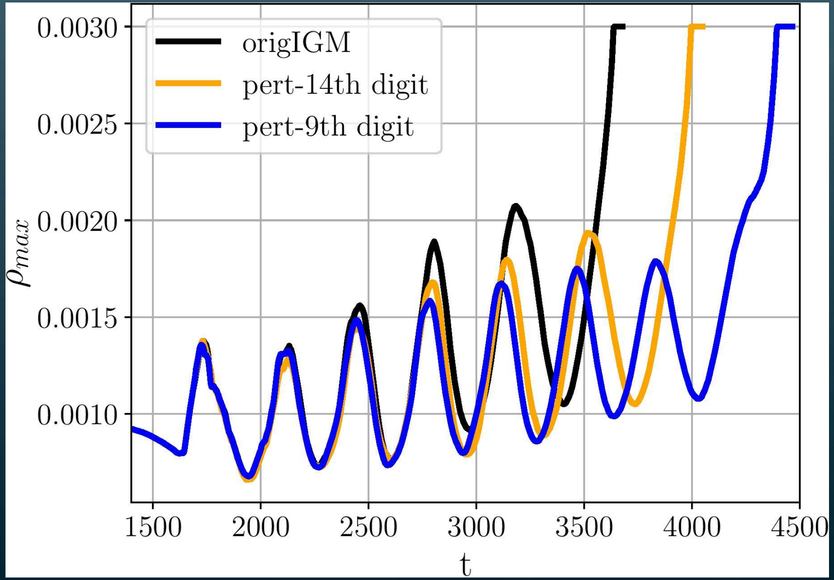
- Provides option to enforce beta equilibrium on tabulated EOS initial data (including other ID thorns)
- Can compute entropy from initial data (including other ID thorns)

- GRHayLIDX provides the same features for CarpetX

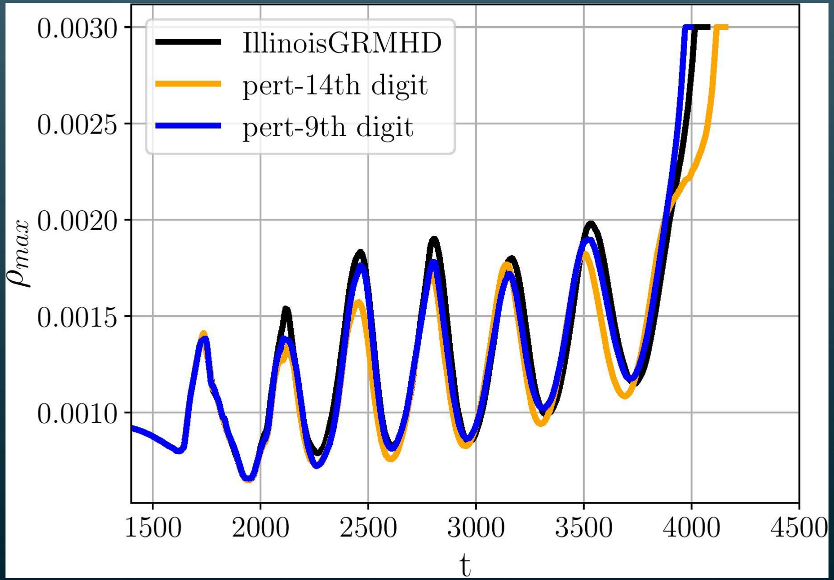
Hybrid EOS BNS comparison: central density

Effects of perturbing the initial data

Old IllinoisGRMHD



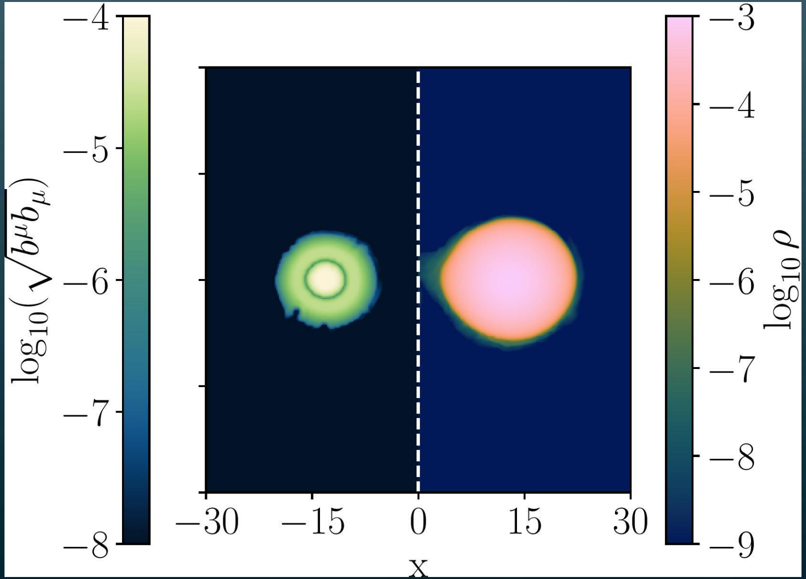
New (GRHayL-based) IllinoisGRMHD



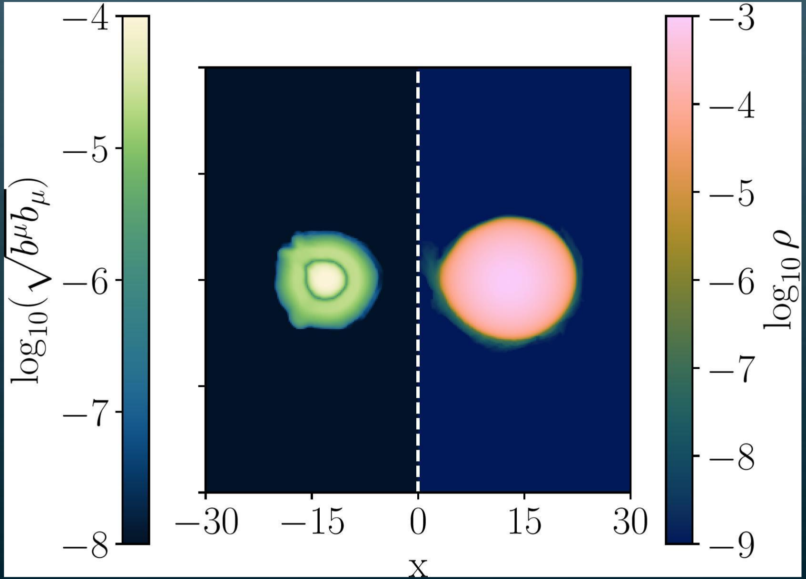
Hybrid EOS BNS comparison: density and magnetic field strength

Slice of z=0 plane during inspiral at t=944 (two full orbits)

Old IllinoisGRMHD



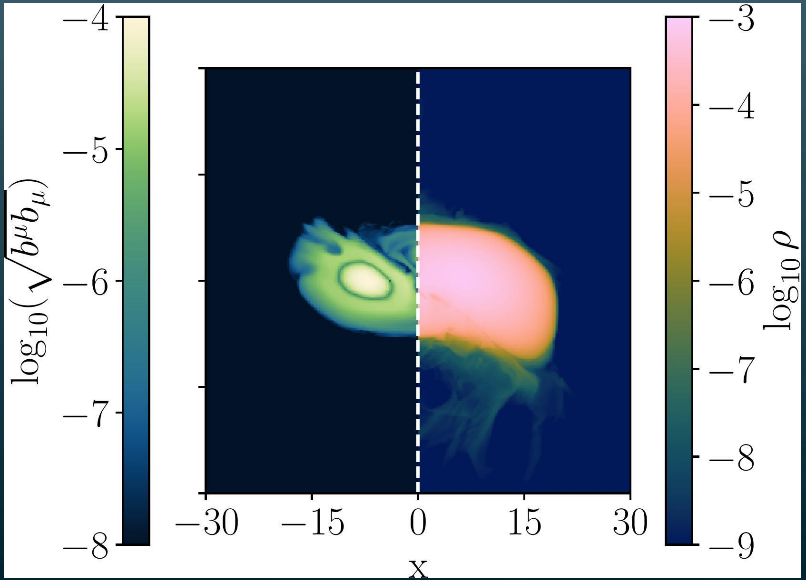
New (GRHayL-based) IllinoisGRMHD



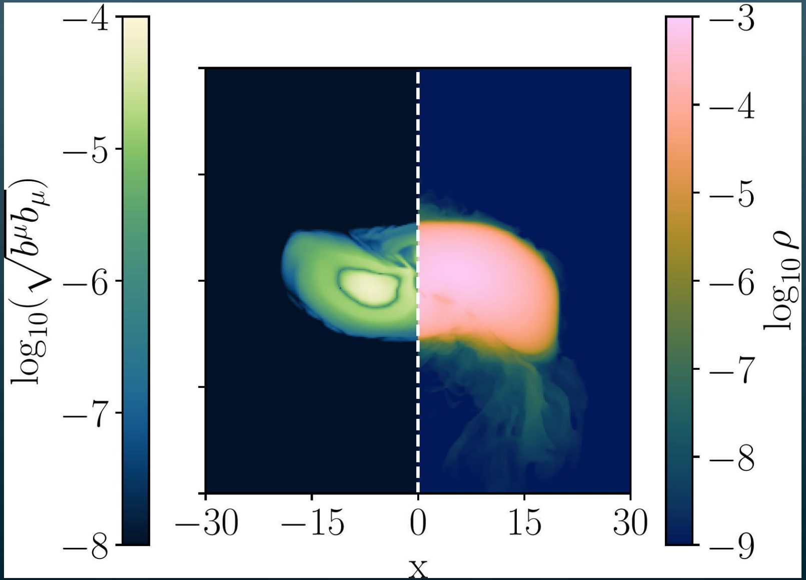
Hybrid EOS BNS comparison: density and magnetic field strength

Slice of z=0 plane during inspiral at t=1632 (first touch)

Old IllinoisGRMHD



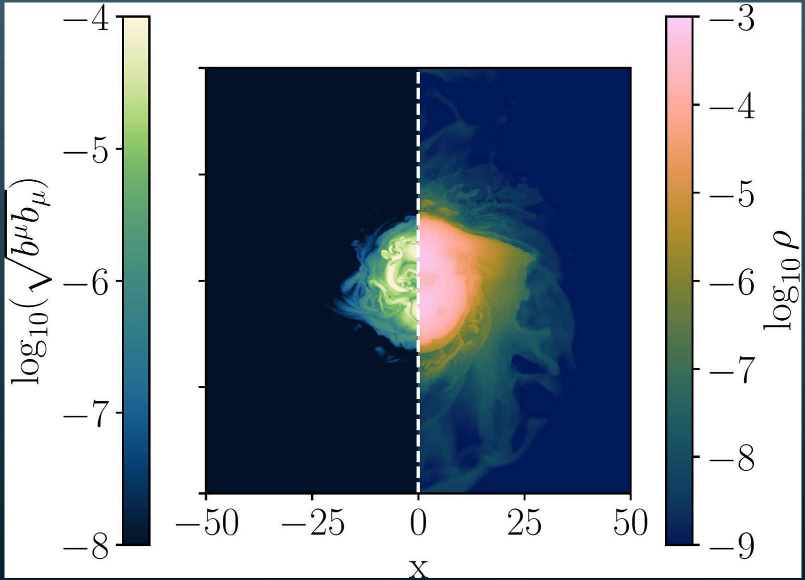
New (GRHayL-based) IllinoisGRMHD



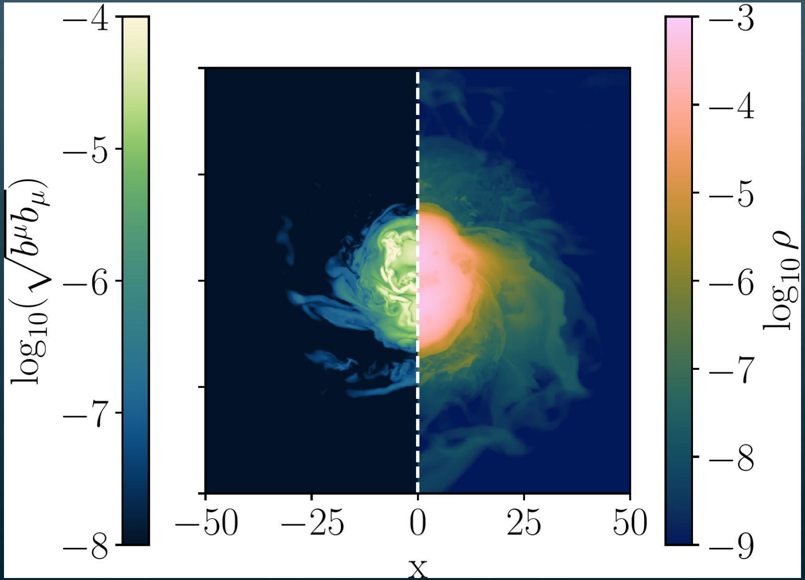
Hybrid EOS BNS comparison: density and magnetic field strength

Slice of z=0 plane at t=1920 (hypermassive neutron star)

Old IllinoisGRMHD

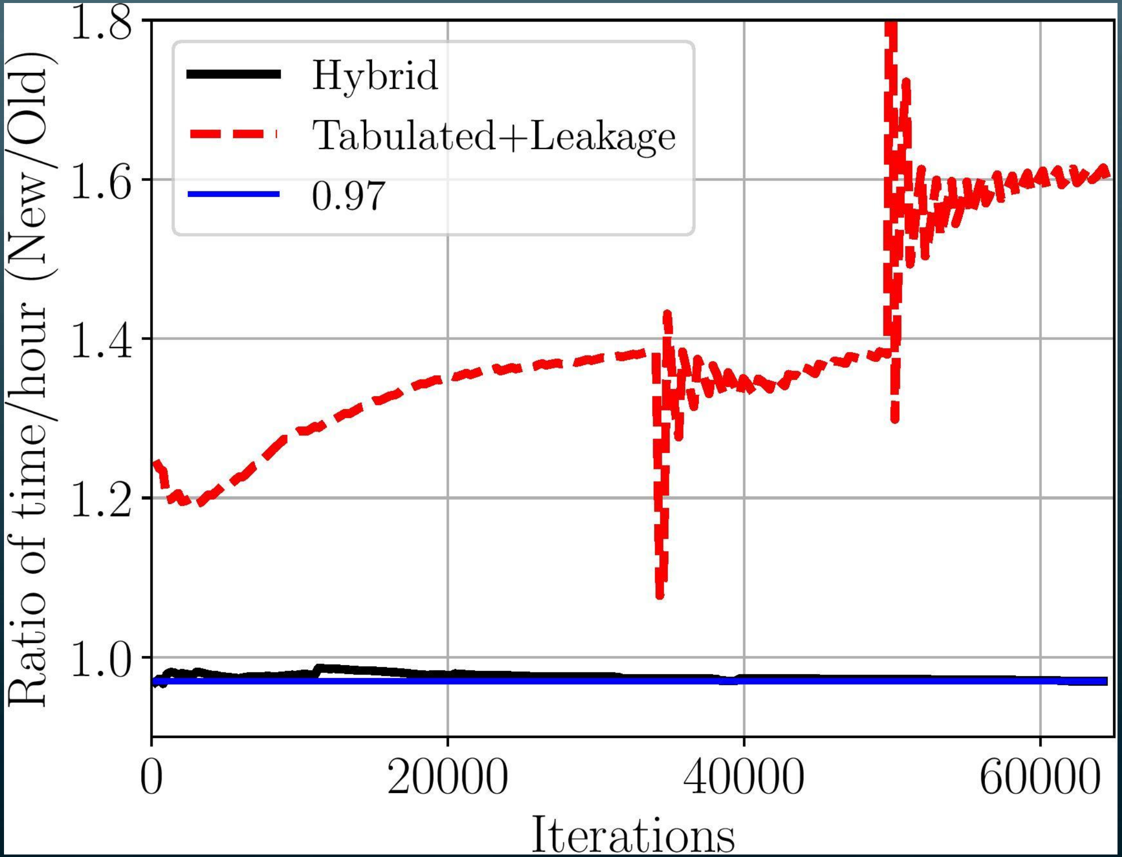


New (GRHayL-based) IllinoisGRMHD



BNS comparison: simulation runtimes

- Ratio of new/old simulation time per hour
- Hybrid code is slower by ~2.5% → 30 day simulation gains ~1 day
- Tabulated code is faster by 20-60%
- Spikes are from checkpoint/restarts



GRHayLib Parameter Setup

GRHayLib::con2prim_routine = “Noble2D”

GRHayLib::con2prim_backup_routines[0] = “Palenzuela1D”

GRHayLib::Lorenz_damping_factor = 0.1

GRHayLib::Psi6threshold = 1e+100

GRHayLib::rho_b_atm = 1.0e-10

GRHayLib::rho_b_min = 1.0e-10

GRHayLib::rho_b_max = 0.003

GRHayLib Parameter Setup: Hybrid EOS

GRHayLib::EOS_type = “Hybrid”

GRHayLib::Gamma_ppoly_in[0] = 2

GRHayLib::Gamma_th = 2

GRHayLib::k_ppoly0 = 1

GRHayLib::neos = 1

GRHayLib Parameter Setup: piecewise polytrope (SLy)

```
GRHayLib::EOS_type            = "Hybrid"
GRHayLib::Gamma_th            = 2
GRHayLib::neos                = 7
GRHayLib::k_ppoly0            = 168.57487497864866555
GRHayLib::rho_ppoly_in[0]     = 3.9514374600825099344e-11
GRHayLib::rho_ppoly_in[1]     = 6.1264330975269766832e-07
GRHayLib::rho_ppoly_in[2]     = 4.2549756827347083401e-06
GRHayLib::rho_ppoly_in[3]     = 0.00023677859688909009043
GRHayLib::rho_ppoly_in[4]     = 0.00081153036440410962048
GRHayLib::rho_ppoly_in[5]     = 0.0016192159535484849844
GRHayLib::Gamma_ppoly_in[0] = 1.5842499999999999361
GRHayLib::Gamma_ppoly_in[1] = 1.2873300000000000853
GRHayLib::Gamma_ppoly_in[2] = 0.62222999999999999498
GRHayLib::Gamma_ppoly_in[3] = 1.3569199999999999904
GRHayLib::Gamma_ppoly_in[4] = 3.0049999999999998934
GRHayLib::Gamma_ppoly_in[5] = 2.9879999999999999893
GRHayLib::Gamma_ppoly_in[6] = 2.8509999999999999787
```

GRHayLib Parameter Setup: Tabulated EOS

```
GRHayLib::EOS_tablepath = “path_to_table.h5”
GRHayLib::EOS_type = “tabulated”
GRHayLib::evolve_entropy = “yes”
GRHayLib::evolve_temperature = “yes”
GRHayLib::rho_b_atm = 1.29e-10
GRHayLib::rho_b_max = 0.004
GRHayLib::rho_b_min = 1.29e-10
GRHayLib::T_atm = 0.01
GRHayLib::T_max = 90
GRHayLib::T_min = 0.01
GRHayLib::Y_e_atm = 0.5
GRHayLib::Y_e_max = 0.5
```

Converting an IllinoisGRMHD parfile

1. Remove ID_converter_ILGRMHD and Convert_to_HydroBase thorns
2. If using ID_converter_ILGRMHD::pure_hydro_run, switch to using GRHayLHD
3. Change the following parameters:
   - ID_converter_ILGRMHD::Gamma_Initial→GRHayLib::Gamma_ppoly_in[0]
   - ID_converter_ILGRMHD::random_seed→IllinoisGRMHD::random_seed
   - ID_converter_ILGRMHD::random_pert→IllinoisGRMHD::random_pert
   - ID_converter_ILGRMHD::K_Initial→GRHayLib::k_ppoly0
   - Convert_to_HydroBase::Convert_to_HydroBase_every
     →IllinoisGRMHD::Convert_to_HydroBase_every
4. If you want to perturb the initial data, add
   IllinoisGRMHD::perturb_initial_data = yes
5. Remove the deprecated parameters IllinoisGRMHD::tau_atm and IllinoisGRMHD::conserv_to_prims_debug

Converting an IllinoisGRMHD parfile

6. Change the following parameters
   - IllinoisGRMHD::GAMMA_SPEED_LIMIT→GRHayLib::max_Lorentz_factor
   - IllinoisGRMHD::K_poly→GRHayLib::k_ppoly0
   - IllinoisGRMHD::rho_b_atm→GRHayLib::rho_b_atm
   - IllinoisGRMHD::rho_b_max→GRHayLib::rho_b_max
   - IllinoisGRMHD::Psi6threshold→GRHayLib::Psi6threshold
   - IllinoisGRMHD::neos→GRHayLib::neos
   - IllinoisGRMHD::gamma_th→GRHayLib::Gamma_th
   - IllinoisGRMHD::damp_lorenz→GRHayLib::Lorenz_damping_factor

Thanks for attending!

Questions?