Introduction to the Einstein Toolkit

Steven R. Brandt, Frank Löffler, Roland Haas, Peter Diener, Erik Schnetter, and others

Perimeter Institute for Theoretical Physics
Waterloo, Ontario, Canada

Sep 18, 2020
Einstein Toolkit

- Collection of scientific software components and tools to simulate and analyze general relativistic astrophysical systems
- Freely available as open source at http://einsteintoolkit.org
- Supported by NSF 1550551/1550461/1550436/1550514, NSF 1212401/1212426/1212433/1212460, NSF 0903973/0903782/0904015 (CIGR), 0701566/0855892 (XiRel), 0721915 (Alpaca), 0905046/0941653 (PetaCactus/PRAC)
- State-of-the-art set of tools for numerical relativity, open source
- Currently 259 members from 172 sites and 39 countries
- > 200 publications, > 30 theses building on these components (as of 2013)
- Regular, tested releases
- User support through various channels
Science

- Binary Black Hole Mergers
- Neutron Star Mergers
- Supernovae
- Accretion Disks
- Boson Stars
- Hairy Black Holes
- Cosmic Censorship
Community Effort!
Why?
Computational Challenges
Computational Challenges
More and more diverse hardware
Computational Challenges

- Simulate cutting edge science
- Use latest numerical methods
- Make use of latest hardware
  - Cache
Computational Challenges

- Simulate cutting edge science
- Use latest numerical methods
- Make use of latest hardware
  - Cache
  - Vector

Schnetter and Others
The Einstein Toolkit
2020-09-18
Computational Challenges

- Simulate cutting edge science
- Use latest numerical methods
- Make use of latest hardware
  - Cache
  - Vector
  - Scale to many cores
Computational Challenges

- Simulate cutting edge science
- Use latest numerical methods
- Make use of latest hardware
  - Cache
  - Vector
  - Scale to many cores
  - Scale to many nodes

Schnetter and Others
The Einstein Toolkit
2020-09-18
Computational Challenges

- Simulate cutting edge science
- Use latest numerical methods
- Make use of latest hardware
  - Cache
  - Vector
  - Scale to many cores
  - Scale to many nodes
  - Algorithms

Schnetter and Others

The Einstein Toolkit

2020-09-18
Efficient use of all hardware is complex and tedious.
Requires experts from different disciplines
Requires good data layouts and APIs
To ensure correctness, need good modularization on a number of levels and understanding of advanced programming concepts.
Design and implementation needs to be carefully thought out in order to ensure extensibility and portability.
Domain Decomposition

Without Ghostzones:

- Processor 0
- Processor 1
- Insufficient data available to update field at these locations
- Boundary of physical domain

With Ghostzones:

- Processor 0
- Processor 1
- Copy
- Ghostzones
- Boundary of physical domain

Schnetter and Others
The Einstein Toolkit
2020-09-18
Domain decomposition
Multiblock and refinement

$R_B$

$R_S$

$(\Delta \rho, \Delta \sigma)$

$\Delta R_1$

$\Delta R_2$
Computational Challenges

- Simulate cutting edge science
- Use latest numerical methods
- Make use of latest hardware
  - Vector (Kranc, NRPy+)

Schnetter and Others
The Einstein Toolkit
2020-09-18
Computational Challenges

- Simulate cutting edge science
- Use latest numerical methods
- Make use of latest hardware
  - Vector (Kranc, NRPy+)
  - Scale to many cores (openmp)
- Scale to many nodes (MPI, Carpet, CarpetX)
- Algorithms / AMR (Adaptive Mesh Refinement, Carpet, CarpetX, MOL)
- GPU (CarpetX)
- FPGA?
- ASIC?
- Neuromorphic processor?
- Q-bits?
Computational Challenges

- Simulate cutting edge science
- Use latest numerical methods
- Make use of latest hardware
  - Vector (Kranc, NRPy+)
  - Scale to many cores (openmp)
  - Scale to many nodes (MPI, Carpet, CarpetX)
Computational Challenges

- Simulate cutting edge science
- Use latest numerical methods
- Make use of latest hardware
  - Vector (Kranc, NRPy+)
  - Scale to many cores (openmp)
  - Scale to many nodes (MPI, Carpet, CarpetX)
  - Algorithms / AMR (Adaptive Mesh Refinement, Carpet, CarpetX, MOL)
Computational Challenges

- Simulate cutting edge science
- Use latest numerical methods
- Make use of latest hardware
  - Vector (Kranc, NRPy+)
  - Scale to many cores (openmp)
  - Scale to many nodes (MPI, Carpet, CarpetX)
  - Algorithms / AMR (Adaptive Mesh Refinement, Carpet, CarpetX, MOL)
  - GPU (CarpetX)
Computational Challenges

- Simulate cutting edge science
- Use latest numerical methods
- Make use of latest hardware
  - Vector (Kranc, NRPy+)
  - Scale to many cores (openmp)
  - Scale to many nodes (MPI, Carpet, CarpetX)
  - Algorithms / AMR (Adaptive Mesh Refinement, Carpet, CarpetX, MOL)
  - GPU (CarpetX)
  - FPGA?
Computational Challenges

- Simulate cutting edge science
- Use latest numerical methods
- Make use of latest hardware
  - Vector (Kranc, NRPy+)
  - Scale to many cores (openmp)
  - Scale to many nodes (MPI, Carpet, CarpetX)
  - Algorithms / AMR (Adaptive Mesh Refinement, Carpet, CarpetX, MOL)
  - GPU (CarpetX)
  - FPGA?
  - ASIC?
Computational Challenges

- Simulate cutting edge science
- Use latest numerical methods
- Make use of latest hardware
  - Vector (Kranc, NRPy+)
  - Scale to many cores (openmp)
  - Scale to many nodes (MPI, Carpet, CarpetX)
  - Algorithms / AMR (Adaptive Mesh Refinement, Carpet, CarpetX, MOL)
  - GPU (CarpetX)
  - FPGA?
  - ASIC?
  - Neuromorphic processor?
Computational Challenges

- Simulate cutting edge science
- Use latest numerical methods
- Make use of latest hardware
  - Vector (Kranc, NRPy+)
  - Scale to many cores (openmp)
  - Scale to many nodes (MPI, Carpet, CarpetX)
  - Algorithms / AMR (Adaptive Mesh Refinement, Carpet, CarpetX, MOL)
  - GPU (CarpetX)
  - FPGA?
  - ASIC?
  - Neuromorphic processor?
  - Q-bits?
Computational Challenges

More Mundane Challenges
Computational Challenges

More Mundane Challenges

- Efficient I/O
More Mundane Challenges

- Efficient I/O
- HDF5
Computational Challenges

More Mundane Challenges

- Efficient I/O
- HDF5
- Checkpoint/Restart
More Mundane Challenges

- Efficient I/O
- HDF5
- Checkpoint/Restart
- Parameter Parsing
Computational Challenges

More Mundane Challenges
- Efficient I/O
- HDF5
- Checkpoint/Restart
- Parameter Parsing
- Visualization
More Mundane Challenges

- Efficient I/O
- HDF5
- Checkpoint/Restart
- Parameter Parsing
- Visualization
- Steering
Collaborative Challenges
Collaborative Challenges

How can we work together?

- Researchers in the USA
  - Louisiana
  - Illinois
  - Virginia
  - Pennsylvania
  - Georgia
  - California

- Researchers in Other countries
  - Italy
  - Spain
  - Portugal
  - Canada
  - Germany
Goals:
- Community Driven
- Core computational tool for GR
- General purpose tool!

Components:
- Cactus
- Simulation Factory
- Kranc
- NRPy+
- Science Modules

Guiding Principles
- Open
- Community Driven
- Good interfaces
- Separation of physics from computational infrastructure
- Code reviews
Initially: some infrastructure, some application code
Growing application suite
Growing infrastructure “return”
Einstein Toolkit as growing project

- Users from more fields of science
Einstein Toolkit as growing project

- Most modules open-source, but not necessarily all
Base Modules
The Einstein Equations

\[ G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu} \]
spacetime curvature

\[ G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu} \]
spacetime curvature

\[ G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu} \]

constants
The Einstein Toolkit

Schnetter and Others

$G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu}$

spacetime curvature

constants

matter
\[ G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu} \]

- Spacetime curvature
- Constants
- Hydrodynamics
- Matter
- El.-magnetism
- Particle radiation
\[ G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu} \]
ADMBase

\[ G_{\mu \nu} = \frac{8\pi G}{c^4} T_{\mu \nu} \]
\[ G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu} \]
\[ G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu} \]
\[ G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu} \]
$G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu}$

ADMBase → TmunuBase → HydroBase

ML_BSSN → TmunuBase → GRHydro
Initial Data / Analysis

\[ G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu} \]

ADMBase → HydroBase

ρ, p, ε, T

ML_BSSN → TmunuBase

GRHydro

Schnetter and Others

The Einstein Toolkit

2020-09-18
$G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu}$

ADMBase → ID / Analysis → HydroBase

ADMBase → TmunuBase

TmunuBase → ML BSSN LeanBSSN MoL

TmunuBase → GR Hydro Illinois GRMHD
Guiding Principles

- Open, community-driven software development
- Separation of **physics** software and **computational** infrastructure
- Stable interfaces, allowing extensions
- Simplify usage where possible:
  - Doing science >> Running a simulation
  - Students need to know a lot about physics (meaningful initial conditions, numerical stability, accuracy/resolution, have patience, have curiosity, develop a “gut feeling” for what is right ...)
  - Einstein Toolkit **cannot** give that, **however**:
  - Open codes that are easy to use allow to concentrate on these things!
In academics: citations, citations, citations!

For Einstein Toolkit:

- Open and free source
- No **requirement** to cite anything
- However: **requested** to cite
  - Maybe the ET or Cactus papers
  - Some papers for the components list a few as well
  - List published on website and manage through publication database
Vision

Cutting Edge / Future

- New Driver Thorn: AMReX
- New Declarative Synchronization: Presync
- New Spherical Coordinates Thorn (RIT)
- New Python Code Generator: NRPy+

Recent

- Proca Thorns
- LEAN Thorns
- GiRaFFE thorns
Einstein Toolkit

- http://einsteintoolkit.org/
- Tools for high-performance computing in numerical relativity
- Open Source
- World-wide, open Community
- Used in high-end research
Supported By

The Einstein Toolkit has been supported by
NSF 2004157/2004044/2004311/2004879/2003893,
NSF 1550551/1550461/1550436/1550514,
NSF 1212401/1212426/1212433/1212460,
NSF 0903973/0903782/0904015 (CIGR), 0701566/0855892 (XiRel),
0721915 (Alpaca), 0905046/0941653(PetaCactus/PRAC). Any opinions,
findings, and conclusions or recommendations expressed in this material
are those of the author(s) and do not necessarily reflect the views of the
National Science Foundation.