HPC-Driven Computational Reproducibility: A Use Case Study of Einstein Toolkit in Numerical Relativity and Astrophysics

Qian Zhang

CLIR postdoc fellow in software curation, University of Waterloo

Senior Analyst, National Digital Research Infrastructure (NDRIO), Canada

2021-02-10

Agenda of this talk

- Introduction: Why & what
- HPC-driven computational reproducibility: A case study in astrophysics
- HPC-driven research reproducibility and reuse: Challenges & opportunities
- Outlook
- Takeaways
- Acknowledgements

HPC-driven computational reproducibility: Why is it important?

- “Reproducibility is a Process, not an Achievement” (Lin & Zhang, 2020)
- Ensure quality science
- Help to “frame the agenda of digital curation” (Stodden, V., 2011. Reproducible Research: A Digital Curation Agenda)
- Central to scientific communication

HPC-driven computational reproducibility: What is it?

- ⇒ Obtain *consistent* (*qualitative)* research results
    - Different team
    - Same experimental setup
        - Same artifacts (input data, simulation/numerical model)
        - Same computational steps, methods
        - *Different* cluster environment: (formerly West Virginia University cluster; now Stampede2 and Comet)
            - Compiler (`bbox.cc`) ⇒ modify source code
            - Hardware (Intel Skylake and Haswell CPUs)
        - *Different* conditions of (post-processing) analysis
            - C++ && AWK && Bash && gnuplot ⇒ Python

HPC-driven computational reproducibility: A case study in Astrophysics

- We attempted to reproduce a study:
    - [IllinoisGRMHD: an open-source, user-friendly GRMHD code for dynamical spacetimes]() (Etienne et al., 2015)

Class. Quantum Grav. **32** (2015) 175009 (33pp) doi:10.1088/0264-9381/32/17/175009

IllinoisGRMHD: an open-source, user-friendly GRMHD code for dynamical spacetimes

Zachariah B Etienne[1,7], Vasileios Paschalidis[2], Roland Haas[3], Philipp Mösta[4] and Stuart L Shapiro[5,6]

[1] Department of Mathematics, West Virginia University, Morgantown, WV 26506, USA
[2] Department of Physics, Princeton University, Princeton, NJ 08544, USA
[3] Max-Planck-Institut für Gravitationsphysik, Albert-Einstein-Institut, D-14476 Golm, Germany
[4] TAPIR, Mailcode 350-17, California Institute of Technology, Pasadena, CA 91125, USA
[5] Department of Physics, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA
[6] Department of Astronomy and NCSA, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA

E-mail: zbetienne@mail.wvu.edu, vp16@princeton.edu, rhaas@aei.mpg.de, pmoesta@tapir.caltech.edu and slshapir@illinois.edu

Received 28 January 2015, revised 26 May 2015
Accepted for publication 23 June 2015
Published 10 August 2015

CrossMark

Abstract
In the extreme violence of merger and mass accretion, compact objects like black holes and neutron stars are thought to launch some of the most luminous outbursts of electromagnetic and gravitational wave energy in the Universe. Modeling these systems realistically is a central problem in theoretical astrophysics, but has proven extremely challenging, requiring the development of numerical relativity codes that solve Einstein's equations for the spacetime, coupled to the equations of general relativistic (ideal) magnetohydrodynamics (GRMHD) for the magnetized fluids. Over the past decade, the Illinois numerical relativity (ILNR) group's dynamical spacetime GRMHD code has proven itself as a robust and reliable tool for theoretical modeling of such GRMHD phenomena. However, the code was written 'by experts and for experts' of the code, with a steep learning curve that would severely hinder community adoption if it were open-sourced. Here we present IllinoisGRMHD, which is an open-source, highly extensible rewrite of the original closed-source

[7] Author to whom any correspondence should be addressed.

0264-9381/15/175009+33$33.00 © 2015 IOP Publishing Ltd Printed in the UK

HPC-driven reproducibility experiment setup

- Link to the code: [IllinoisGRMHD]
  - Instructions for downloading, compiling, and using IllinoisGRMHD: http://astro.phys.wvu.edu/zetienne/ILGRMHD/getstarted.html"
- HPC resources: XSEDE
  - Stampede2's Skylake (SKX) @Texas Advanced Computing Center (TACC)
  - Comet @San Diego Supercomputer Center (SDSC)

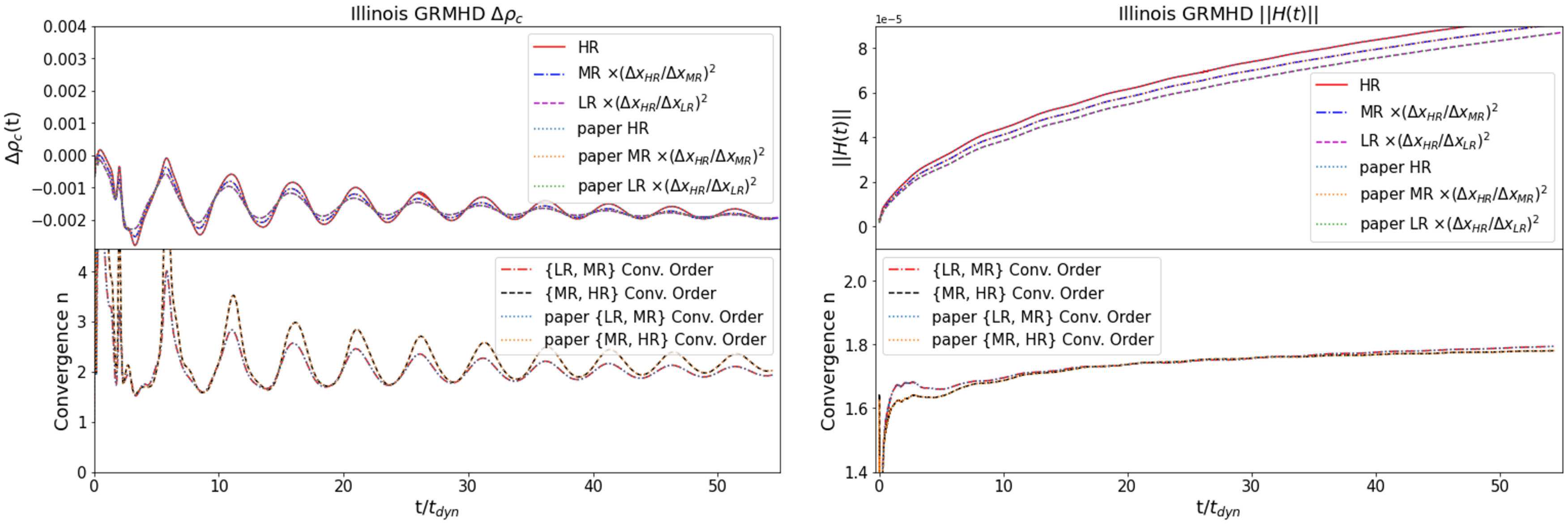

IllinoisGRMHD
An Open-Source, User-Friendly GRMHD Code for Dynamical Spacetimes
Introduction | Guide to Getting Started | Get Involved!
OPEN SOURCE FRIENDS
Alpaca Code
Cactus Code
Carpet AMR
Einstein Toolkit
GRHydro
Kranc Code
Llama Code
Sim. Factory
SimulationTools
Whisky Code
Add to this list
INTRODUCTION
In the extreme violence of merger and mass accretion, compact objects like black holes and neutron stars are thought to launch some of the most luminous outbursts of electromagnetic and gravitational wave energy in the Universe. Modeling these systems realistically is a central problem in theoretical astrophysics, but has proven extremely challenging, requiring the development of numerical relativity codes that solve Einstein's equations for the spacetime, coupled to the equations of general relativistic (ideal) magnetohydrodynamics (GRMHD) for the magnetized fluids.
Over the past decade, the Illinois Numerical Relativity (ILNR) Group's dynamical spacetime GRMHD code has proven itself as one of the most robust and reliable tools for theoretical modeling of such GRMHD phenomena. Despite the code's outstanding reputation, it was written "by experts and for experts" of the code, with a steep learning curve that would severely hinder community adoption if it were open-sourced.
IllinoisGRMHD is an open-source rewrite of the original closed-source GRMHD code of the ILNR Group, aimed at minimizing the learning curve, with the goal of facilitating community involvement in the code's development, as well as the minimization of human effort in generating new science. IllinoisGRMHD also saves computer time, generating roundoff-precision identical output to the original code on adaptive-mesh grids, but nearly twice as fast at scales of hundreds to thousands of processor cores.
ILLINOISGRMHD: GUIDE TO GETTING STARTED
homepage | contact | html | css
Creative Commons Attribution 3.0 License
TO GETTING STARTED
Read the Requirements section of this document to make you have installed all software packages on which GRMHD and Einstein Toolkit's Cactus/Carpet adaptive-mesh (AMR) infrastructure depend. In addition, to run the test simulation, you will need a computer with at least 6--8GB of RAM. To visualize the output, please install gnuplot.
Step 1: Click here to download the September 2015 version of IllinoisGRMHD and Einstein Toolkit software package.
Step 2: Unpackage the software via
tar zxfv IllinoisGRMHD_Sept_1_2015_public_release__based_on_ET_2015_05.tar.gz
Step 3: Enter the directory containing the uncompressed package:
cd IllinoisGRMHD_Sept_1_2015_public_release__based_on_ET_2015_05
Step 4: Edit the
make_config-runmefirst
script to specify your computer's configuration. Then run the script via
./make_config-runmefirst
Step 5: If the above step is unsuccessful, install needed software packages based on the error messages and repeat Step 4. Otherwise, continue to Step 6.
Step 6: Type
make etilgrmhd
and wait for the code to compile. The full compile will take around 30 minutes or so on a fast (ca. 2015) desktop. So relax and grab a hot drink, but be quick! Compilations are fun to watch... [next step]
Step 7: ... until they fail. If the compilation fails, be careful not to spill your hot drink, check Step 0 again, then return to Step 4 and verify that your computer's configuration is consistent.

Workflow

- Download
- Compile & build
- Execute simulation
- Post-analysis

Reproducibility Demo

Preliminary results of the reproducibility



Observations & lessons learned

- Link rot (to code) in the paper
- Missing key parameter (file) in the paper
- Lack of *cluster-specific* documentation
- ⇒ Compilation errors
  - Unstoppable upgrades: cluster (SW stack, libraries, compilers)

Observations & lessons learned (cont.)

- Building issues
  - If installed on local laptops
    - Have to be clean slate
  - If installed on local institutional cluster platform
    - ⇒ Setup issues
- ⇒ Provide **instructions** on building
    - Documentation
      - Checklist

- Execution complications at other clusters: Comet
  - Cluster (SW + hardware)
- Issues when submitting jobs (shell script vs. command line) to queuing system

Why are HPC-driven research reproducibility and reuse so difficult?

- Model/code
  - Model/code availability/ease of use
  - Platform/system availability
  - Where/how was this run?
  - Model interoperability, re-usability (setup, etc.)
- Human efforts
- Data
  - Simulation inputs
  - Output usability

Why are HPC-driven research reproducibility and reuse so difficult? (cont.)

- Accessibility
  - Conformance to open or established standards
  - Archival accessibility
  - Longevity of the technology
- Cost
  - Computational cost
  - Storage cost

Opportunities of HPC-driven research reproducibility and reuse

- Ensure **transparency**, **reproducibility** and **reusability** of research results
- Provide effective **communication** of research outputs (publication, data and code) and advanced research computing resources
- Promote enhanced **access** to research outputs and resources
  - Policies and strategies
  - Network and collaborative initiatives
  - Research infrastructures
  - Research software as a primary output of research

Opportunities of HPC-driven research reproducibility and reuse (cont.)

- Develop standards for reproducibility **badges**
  - NISO's Draft Recommended Practice for Reproducibility Badging and Definitions
  - ACM Artifact Review and Badging *Version 1.1 - August 24, 2020*
- **Tools & platforms** for supporting computational science
  - Dissemination/reproducibility platforms (code ocean, Whole Tale)
  - Workflow tracking (Kurator)
  - Better documentation (Jupyter notebook)
- Practices & guidelines
- Training opportunities

https://www.acm.org/binaries/content/gallery/acm/publications/large-replication-badges/all-badges.png

Outlook

- Extensive re-use of data and code will become the norm
- Researcher competitiveness will be re-defined with multi-facet metrics
- Cultural change

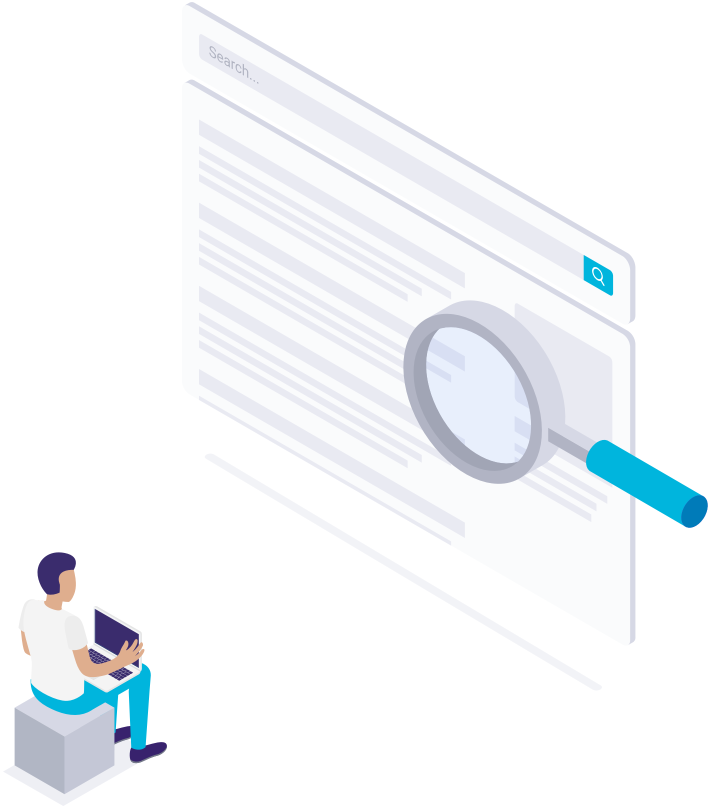
Takeaways

- “Reproducibility is a Process, not an Achievement”
- Greater clarity and guidance on dissemination of computational claims
- Code dissemination + full workflow documentation + instructions on the building and running environment
- Research community’s recommendations on good practices

Acknowledgement

- Thanks to Zachariah Etienne for providing his simulation result data files
- Team efforts:
    - University of Illinois at Urbana-Champaign (UIUC)
        - Roland Haas: National Center for Supercomputing Applications (NCSA), UIUC
        - Yufeng Luo: NCSA; Department of Astronomy, UIUC
- Grant information
    - This material is based upon work supported by the National Science Foundation under Grant No. 2004879 and No. 1550514.
    - This work and webinar is partially supported by the CLIR microgrant and Alfred P. Sloan Foundation.

Thank you for your attention!



Presentation template by SlidesCarnival; Illustrations by Sergei Tikhonov