New Paradigms for Computational Astrophysics

Lawrence E. Kidder

Cornell Center for Astrophysics and Planetary Science
Cornell University

The Architecture of LISA Science Analysis: Imagining the Future
18 January 2018, Keck Institute for Space Studies, Pasadena

SIMULATING EXTREME SPACETIMES

Black holes, neutron stars, and beyond...

Payoffs from binary black hole simulations

What do we learn?

- Dynamics of strongly warped spacetime
- Gravitational waveforms

How can they be used?

- Directly compare theory to observations
- Improve analytic waveform models
- Inject into data analysis pipelines
- Construct surrogate models
- Determine remnant black hole properties
- Explore nonlinear behavior of gravity
- Produce visualizations for public outreach

Solving the vacuum Einstein equations on a computer

Goal: determine the spacetime metric describing the inspiral, merger, and ringdown of a binary black hole system

- Solve as an initial-boundary value problem
- Slice spacetime into spatial hypersurfaces
  - Constraint equations
  - Evolution equations
  - Coordinate freedom
- Specify initial conditions that describe a binary black hole system and satisfy the constraint equations
- Choose the computational domain on which to do the evolution
  - deal with singularities inside the black holes
  - introduce artificial outer boundary
- Choose a formulation of the evolution equations
- Choose a numerical algorithm
- Specify coordinate conditions
- Specify boundary conditions
- Decide how to control constraint violations

Running a BBH simulation

- Choose desired physical configuration q, S_1, S_2, e at some initial orbital parameters ω_{orb}, d, v_r
- Iterative initial data solve to get desired parameters
- Evolve for several orbits, measure eccentricity and adjust initial orbit parameters
- Also adjust physical parameters as black holes relax
- Once desired setup is achieved, evolve through merger and ringdown until waves reach extraction surfaces
- Extrapolate/Evolve extracted waves to null infinity
- Depending on parameters and desired accuracy, runs take days to months

What can be simulated?

- Number of orbits before merger
  - Desired orbits for testing analytic models?
  - Desired orbits for parameter estimation?
  - For low mass systems, need to hybridize to cover detector frequency band
  - Routinely do 20-40 orbits, can do hundreds, but ...
- Parameter space
  - Total mass m scales out of the problem
  - Mass ratio: $1 \leq q \lessapprox 20$ possible
    $q = 100$ demo; robust for $q \lessapprox 6$
  - Spins: $0 \leq \chi \lessapprox 0.8$ robust
    high spin $0.92 \lessapprox \chi$ requires improved initial data
  - high spin on small BH is very difficult
  - Precession: no problem
  - Eccentricity: no problem unless $e \approx 1$

LISA Sources that require Numerical Relativity

- Massive Black hole binaries
- EMRIs

- Some signals with SNRs over 1000, maybe 10000
- Expected mass ratios 10 - 100+
- Signals in band for many months

What are requirements for NR BBH Waveforms?

- What is the expected mass and spin distribution?
- How accurate does the waveform need to be?
- How many orbits to test/improve/build analytic models?

Changing landscape of high performance computing

- Processor speed no longer following Moore's Law
- More cores per node
- Problem: specialized languages needed to exploit GPUs
- Problem: Many threads limited by serial chunk size, memory contention, communication between nodes
- Want to hide latency of communication
- Want to avoid global synchronization points

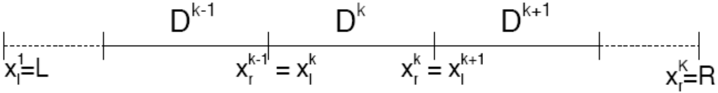
Parallelization strategies

- Domain-based parallelism
- Task-based parallelism

Task-based parallelism

- More natural to divide computation into tasks
- Functions define tasks, parameters define dependencies
- When dependencies are ready, task is put in a ready pool
- Scheduler then selects ready task and it is executed
- When task is finished, check if new tasks can be added to ready pool
- Need to take care of remote dependencies
- Load balancing, task-stealing

Task-based DG



- Cover the physical domain with elements and interfaces
- Method of lines, volume RHS plus fluxes through interfaces
- Limiting to handle shocks, needs info from nearest neighbors

Element:

- Send data to interfaces
- Compute volume RHS
- When fluxes arrive, full RHS
- Send limiting data to neighbors
- When limit data arrive, limit
- Send data to observers

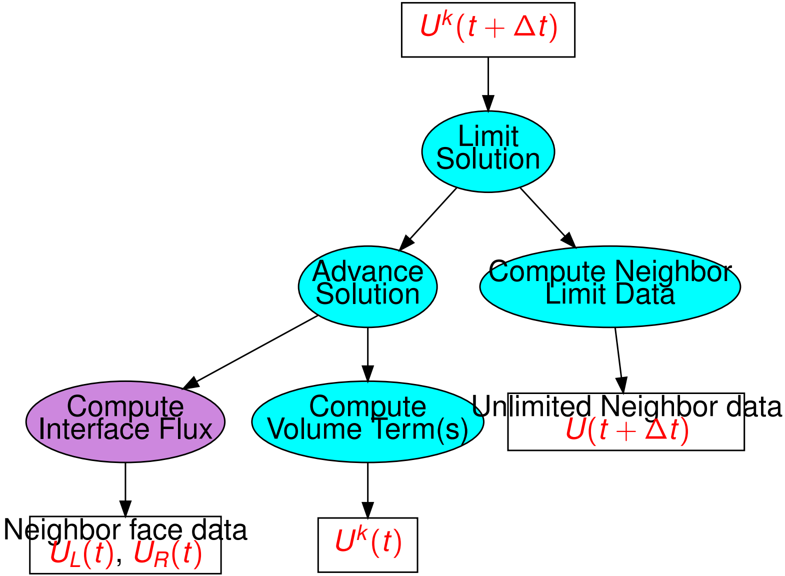
Interfaces:

- When neighbor data arrive, compute fluxes
- Send flux corrections to elements

Observers:

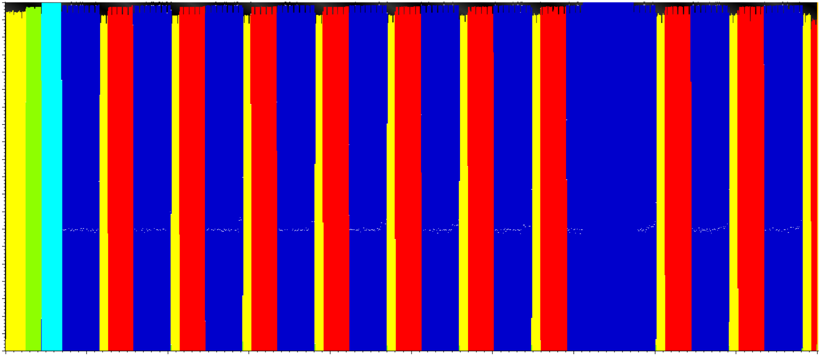
- When data arrives, process
- Write files

Task graph


$U^k(t+\Delta t)$
Limit Solution
Advance Solution
Compute Neighbor Limit Data
Compute Interface Flux
Compute Volume Term(s)
Unlimited Neighbor data
$U(t+\Delta t)$
Neighbor face data
$U_L(t)$, $U_R(t)$
$U^k(t)$

Time profile (1 core)

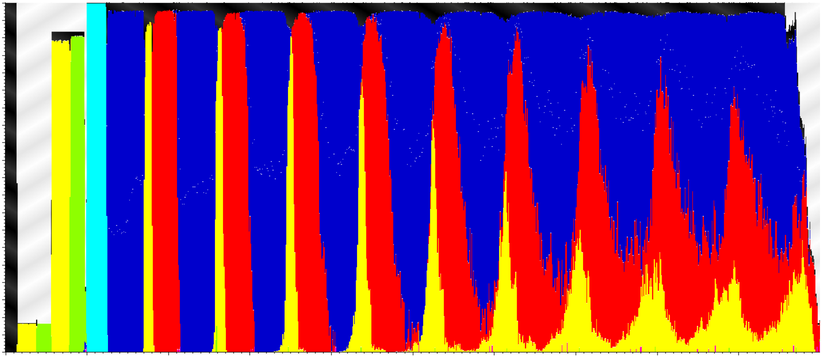
Ten time steps of a relativistic MHD test problem



Blue: volume terms Red: interface flux Yellow: slope limiting
Cyan: setup Black: Charm++ White: idle

Time profile (12 cores)

Ten time steps of a relativistic MHD test problem



Blue: volume terms Red: interface flux Yellow: slope limiting
Cyan: setup Black: Charm++ White: idle

Summary

- How accurate do model waveforms need to be?
- How long and accurate do NR waveforms need to be?
- New paradigms needed to meet modeling challenges for LISA
- Do we need a radically new method for high mass-ratio systems?