

Data Collection for Performance Analysis and Visualization of ROSS/CODES Simulations

Caitlin Ross and Christopher Carothers
Rensselaer Polytechnic Institute

Kelvin Li and Kwan-Liu Ma
University of California, Davis

Misbah Mubarak and Robert Ross
Argonne National Laboratory



Rensselaer

ROSS data collection

- Reverse computation adds complexity to development of simulations
 - Need help tuning and debugging
- Add data collection to ROSS that will give us more insight to the behavior of the simulation engine
- Connect simulation-level data with model-level data to understand the correlations between the two

4 Views of PDES

**Local
Speculative**

**Global
Speculative**

**Local
Committed**

**Global
Committed**



4 Views of PDES

Local Speculative

- PDES written from this view
- Collect during batch processing of events
- Model can reach invalid states

Global Speculative

Local Committed

Global Committed



4 Views of PDES

Local Speculative

- PDES written from this view
- Collect during batch processing of events
- Model can reach invalid states

Global Speculative

- Max, min, avg, mode
- Helpful for performance analysis and debugging
- Observe load balance of system, impact of rollbacks

Local Committed

Global Committed



4 Views of PDES

Local Speculative

- PDES written from this view
- Collect during batch processing of events
- Model can reach invalid states

Global Speculative

- Max, min, avg, mode
- Helpful for performance analysis and debugging
- Observe load balance of system, impact of rollbacks

Local Committed

- Helpful for understanding model behavior
- Collect just after GVT computation

Global Committed

4 Views of PDES

Local Speculative

- PDES written from this view
- Collect during batch processing of events
- Model can reach invalid states

Global Speculative

- Max, min, avg, mode
- Helpful for performance analysis and debugging
- Observe load balance of system, impact of rollbacks

Local Committed

- Helpful for understanding model behavior
- Collect just after GVT computation

Global Committed

- Max, min, avg, mode
- Progress tracking
- Global synch to allow for in situ analysis of global state

What's been done so far

- GVT collection
 - Collect immediately after GVT computation
 - # events processed, rollbacks, ties, remote events, remote sends and recvs
- Real time collection
 - User specifies how often to sample
 - Collect during batch processing
 - How far KPs are ahead of GVT, cycle counters, forward events, reverse events, rollbacks

What's been done so far

- These can be run independently or together
- Vis2 branch on ROSS GitHub repo
- Turned off by default
 - `--enable-gvt-stats=1`
 - `--real-time-samp=n`, where n is time in milliseconds
- Collected data stored in a buffer
 - Configurable size and when to dump

I/O

- All writing takes place just after GVT
- Output is in binary
- Uses MPI I/O
- One file per simulation
- Right now using our own format
 - Switch to an established format in the future?

Challenges

- Don't want to perturb simulation
 - Too much could slow down the simulation enough that the rollback behavior changes drastically
- Too much data to store/transfer
 - Long running simulations, large-scale models means we can't collect everything at a fine granularity

How to deal with these challenges?

- Perform as much work as possible immediately after GVT
- Store collected data in a buffer and only write out when it's full
- Only collect fine grained data when 'something interesting happens'



In Progress

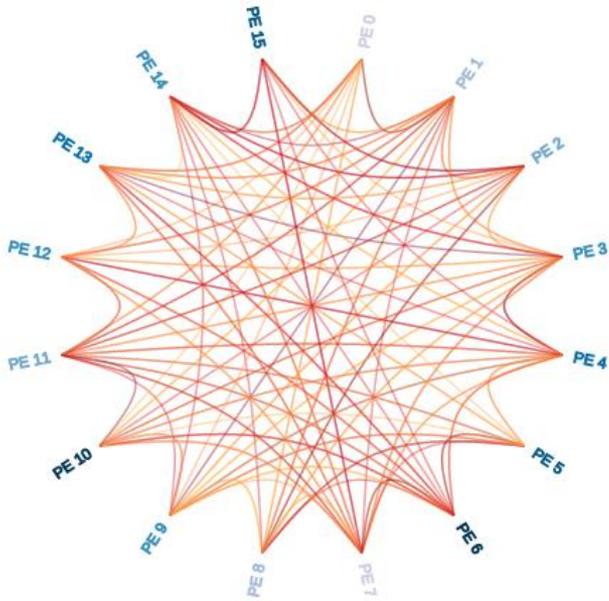
- Preliminary testing of data collection
 - Compare to non-instrumented ROSS
 - So far we've seen minimal decrease (1%) of rollbacks in most cases we've tested
- Developing use cases for this data collection for the dragonfly model
 - How to best show the 4 views discussed previously?

Future work

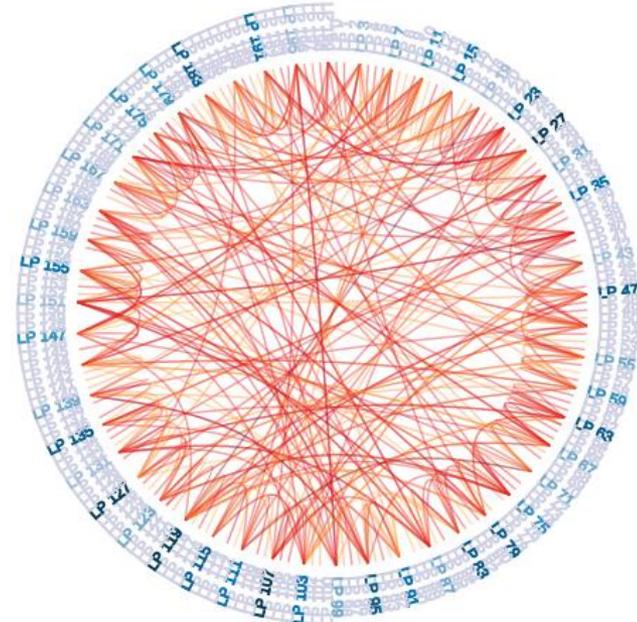
- Continue adding various metrics to what we're already collecting, such as memory usage
- Event level data collection
 - Decide which events are recorded



Radial View (PEs)



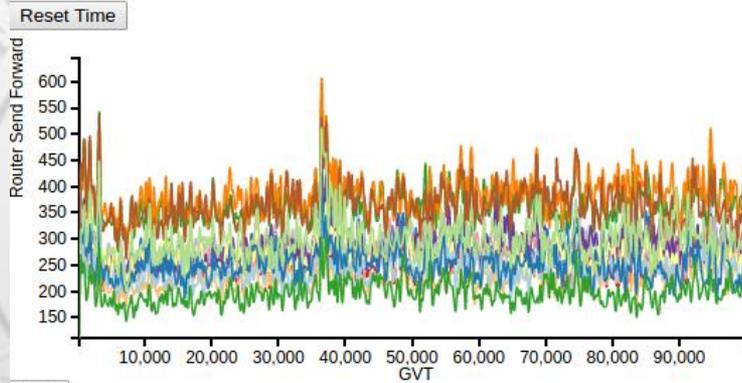
LP View (Terminals and Routers)



Help

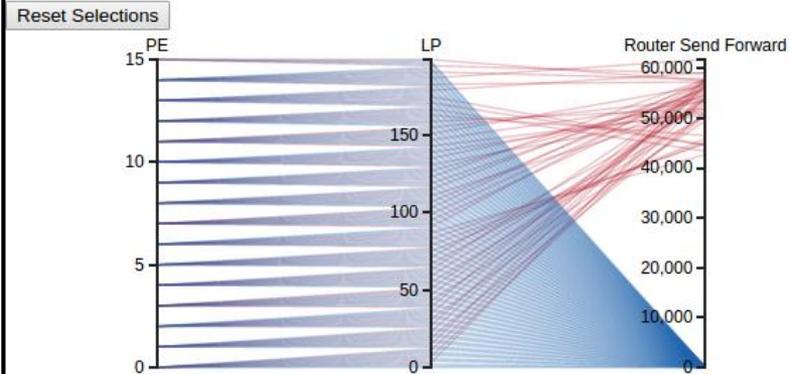
Help

Time Selector



Help

LP Selector



Help

Future work

- Continue adding various metrics to what we're already collection, such as memory usage
- Event level data collection
 - Decide which events are recorded
- Configuration file: collect only specific data the user wants
 - Allow the user to trade between performance and data capture
- Let the user define situations where they want to collect data at a finer granularity

Thank You!
Questions?

