TraceR:
A Parallel Trace Replay Tool for Studying Interconnection Networks

Bilge Acun, PhD Candidate
Department of Computer Science
University of Illinois at Urbana-Champaign

*Contributors: Nikhil Jain, Abhinav Bhavele, Misbah Mubarak, Christopher D. Carothers, and Laxmikant V. Kale
Network Simulation

• **Motivation:**
  • Design of the future supercomputers
    • Node architecture
    • Interconnection network
  • Predict application performance
    • On existing – non existing architectures

• **State-of-the art:**
  • Discrete event based simulation
    • Not parallel or scalable
    • Large memory footprints
  • Cannot simulate real HPC workloads
    • Synthetic communication patterns
    • Skeletonized codes
TraceR: Trace Replay

• A trace-driven simulator
  • Optimistic parallel discrete-event simulation (PDES)
  • for real HPC traffic workloads
• Outperforms state-of-the-art simulators
  • BigNet-Sim, SST
• Scalable
  • simulate execution on half a million nodes in under 10 minutes using 512 cores
• Optimistic simulation parameter study
  • maximize performance for simulating real HPC traffic workloads
TraceR Components

Input

- PDES parameters
- Network Configuration
- Application Traces from BigSim

AMPI & Charm++

Output

- Dimensions, bandwidth, packet size, ...
- Accurate packet level network models; Torus, Dragonfly, ...
- PDES Framework

Performance Prediction
BigSim Simulator

• One of the earliest packet-level HPC network simulator
  • Around 2004

• Emulation framework
  • Can generate traces using much less cores than actual

• Built on **POSE** PDES framework
  • Cause of the slow performance
  • Poor scaling
TraceR Components

Input

- Application Traces from BigSim
- PDES parameters
- Network Configuration
- AMPI & Charm++

TraceR

- CODES
- ROSS

Output

- Performance Prediction
- Accurate packet level network models; Torus, Dragonfly, ...
- Dimensions, bandwidth, packet size, ...

PDES Framework
BigSim Trace Format

- Entry for each Sequential Execution Block (SEB)

**Time Stamp, Task ID, Name, Duration, ..., Msg ID, Source Node, ..., Back&Forward Dep.**

-1.000000 47 AMPI_Bcast--time:5960 0.000006 ... $B 46 $F 53
0.001148 48 start-broadcast--time:0 0.000000 ... $B $F 49
-1.000000 49 AMPI_generic--time:3099 0.000003 .. $B 48 $F 50 52
-1.000000 50 end-broadcast--time:0 0.000000 ... $B 49 $F
0.001151 51 msgep--time:953 0.000001 ... $B $F
0.001154 52 RECV_RESUME--time:953 0.000001 ... $B 49 $F 53
-1.000000 60 user_code--time:0 0.000000 ... $B 59 54 $F 61
Definitions and Evaluation Metrics

Definitions:
- **PE**: simulated process, logical process (LP) visible to ROSS
- **Task**: sequential execution block (SEB)
- **Event**: represents an action with a time-stamp in the PDES
  - Kickoff Event, Message Recv Event, Completion Event
- **Reverse Handler**: responsible for reversing the effect of an event

Metrics:
- **Execution time**: time spent in performing the simulation
- **Event rate**: number of events executed per second (excl. roll backs)
- **Event efficiency**: (or rollback efficiency)

\[
\text{Event efficiency}(\%) = \left(1 - \frac{\#\text{rolled back events}}{\#\text{committed events}}\right) \times 100
\]
TraceR: Execution flow

First task

Execute Task

Send message to other PEs

Remote Message

Receive message from other PEs

Schedule completion event

Completion Event

Message Recv Event

TraceR functions

ROSS Events
Experimental Results

• Scaling results are done with **Blue Waters at UIUC**
• Prediction study results are with **Vulcan at LLNL**

• Applications:
  • **3D Stencil:**
    • AMPI application
    • 7 point Jacobi relaxation on 3D grid
    • 128 x 128 x 128 grid points per MPI process -> 128KB msgs

• **LeanMD:**
  • Charm++ application
  • Mini-app version of NAMD molecular dynamics simulation
  • Mimics short-range force calculations of NAMD
  • 1.2 million atoms
Sequential Comparison of Simulators

Comparison of BigNetSim, SST and TraceR

Time (s)

Number of simulated 3D torus nodes

Skeletonized MPI code
Conservative vs. Optimistic

TraceR: 3D Stencil simulation of 4K nodes

Execution Time (s)

Number of cores

Cons. 3D TorusNet
Cons. SimpleNet
Opt. 3D TorusNet
Opt. SimpleNet
TraceR Scaling w/ AMPI app.

3D Stencil simulation using SimpleNet

Time (s)

Number of cores

512K
32K
128K
8K
TraceR Scaling w/ AMPI app.

3D Stencil simulation using 3D TorusNet

Time (s)

Number of cores

10000

1000

100

10
Event Efficiency

TraceR: 3D Stencil simulation

Event efficiency \(\%\) = \(\left(1 - \frac{\#\text{rolled back events}}{\#\text{committed events}}\right) \times 100\)
Trace Reading Time

- Insignificant overhead with increasing number of cores!
TraceR Performance Prediction w/ Charm++ app.

Prediction accuracy for LeanMD (5D TorusNet)

Time per step (s)

Actual Runtime

Predicted Runtime

Error %

Number of nodes

512

1024

2048

4096

8192

2%

2%

2%

9%

7%
Event Rate: million events/s

3D Stencil simulation on 8K nodes of 3D TorusNet

Event rate (million events/s)

Batch size (no. of events)
Efficiency

3D Stencil simulation on 8K nodes of 3D TorusNet

Event Efficiency (%) vs Batch size (no. of events)

- GVT 16
- GVT 64
- GVT 256
- GVT 1024
- GVT 4096
Ongoing Work and Summary

• **Ongoing & future work:**
  - Fat-tree network model
    - Integrated into CODES
  - Multiple job simulations
    - Effect of multiple jobs in the network
    - More realistic scenario
  - Switch to Charm++ based ROSS from MPI based ROSS

• **TraceR feature highlights:**
  - A parallel, trace-driven, scalable network simulator
  - Support for various topologies: Torus, Dragonfly, Fat-tree
  - Simulate AMPI, Charm++ applications
  - Can simulate half a million nodes in minutes
Thank you!

• Paper in progress:

Bilge Acun, Nikhil Jain, Abhinav Bhatle, Misbah Mubarak, Christopher D. Carothers, and Laxmikant V. Kale. TraceR: A Parallel Trace Replay Tool for Studying Interconnection Networks

• TraceR source code:
  • http://charm.cs.uiuc.edu/gerrit/#/admin/projects/tracer
TraceR Scaling w/ Charm++ app.

LeanMD simulation on 32K nodes of 5D TorusNet

Time (s)

Number of cores

TorusNet

SimpleNet