SAND2015-3169C

Multi-scale, Multi-objective Behavioral Emulation of Future-gen Applications and Systems

Nalini Kumar, Chris Hajas, Alan George, Herman Lam, Greg Stitt

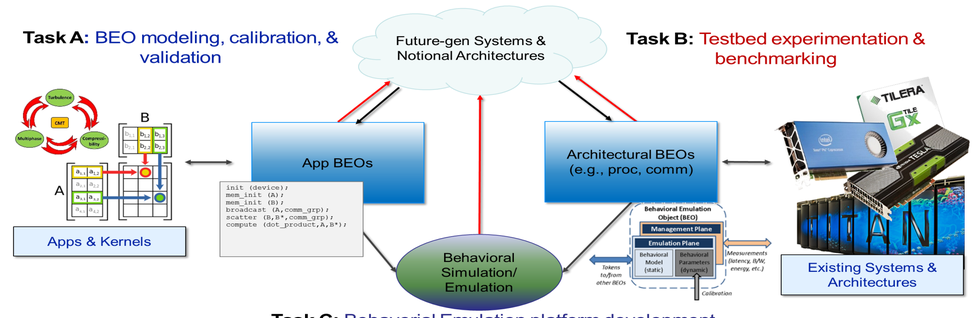
PSAAP II Center for Compressible Multiphase Turbulence, NSF Center for High Performance Reconfigurable Computing
University of Florida, Gainesville

Motivation

Iterative Modeling & Simulation is needed to enable effective HW/SW co-design

- Traditional simulation methods are either too slow (cycle-accurate simulation) or not accurate enough (analytical machine models)
- There is need for a fast, scalable, and reasonably accurate simulation methodology and simulator

Behavioral Emulation aims to address this problem by *mimicking or emulating* behavior, not cycle-accurate functionality, of apps running on future devices & systems



We propose to handle exascale complexity via:

- Coarse-grained component models: Behavioral Emulation Objects
- Multi-scale simulation at ***micro-***, ***meso-***, and ***macro-***scale

BEO Design & Calibration

BEOs are simulation constructs that mimic behavior of app (AppBEO) & arch (ArchBEO) under study

- **AppBEOs:** High-level app representation understood by simulator
  - List of instructions processed by ProcBEOs
  - Each instruction represents a comp/comm block
  - Simple, extensible, platform-independent, high-level API

AppBEO (high-level description)

```
// Define group as nodes 0-3
VAR commGrp=0:3
// Broadcast matrix A
(dataSize=64*64/2) to group
Bcast(int32,2048,0,commGrp)
// Barrier sync
Barrier(commGrp)
// Scatter 1/4 of matrix B
(dataSize=(64*64)/(4*2)) to each node
Scatter(int32,512,0,commGrp)
// Perform dot product of vector size 64
of int32
DotProduct(int32,64)
// Gather solutions from matrices
(dataSize=(64*64)/(4*2))
Gather(int32,512,commGrp)
Done
```

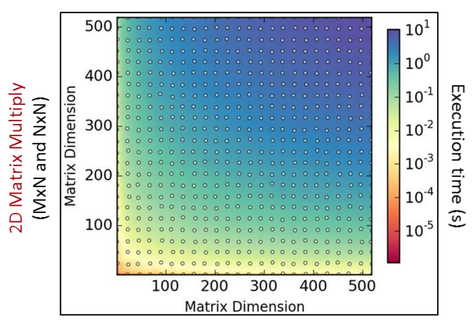
Intermediate format

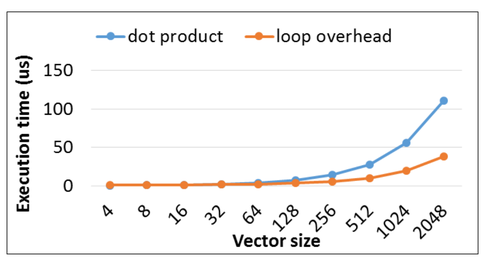
```
send 1 1 129971 1
recv 4
send 2 2 129971 1
recv 8
send 13 1 381 1
recv 12
send 16 1 32420 1
recv 17
send 18 2 32420 1
recv 19
send 20 3 32420 1
recv 21
advt 5753856
```

Script-generation tool

- AppBEO script for VisualSim
- AppBEO script for SMP simulator
- AppBEO script for HW Emulator → FPGA memory initialization file

- **ArchBEOs:** Mimic functionality of hardware components
  - Just as app SW drives the CPU, AppBEOs drive the ArchBEOs
  - ProcBEO mimic the processing core
  - CommBEO for the communication switch





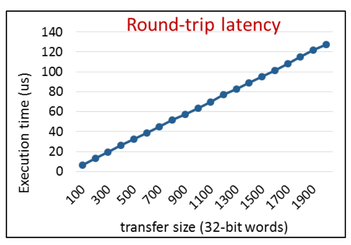
Network configuration parameters

- Topology: 2D mesh
- Mesh size: 6x6
- Routing policy: dim-order
- Routing policy: store and forward
- Arbitration: round-robin

Switching Time

Direction	Time (ns)
x-x	1
y-y	1
x-y	1

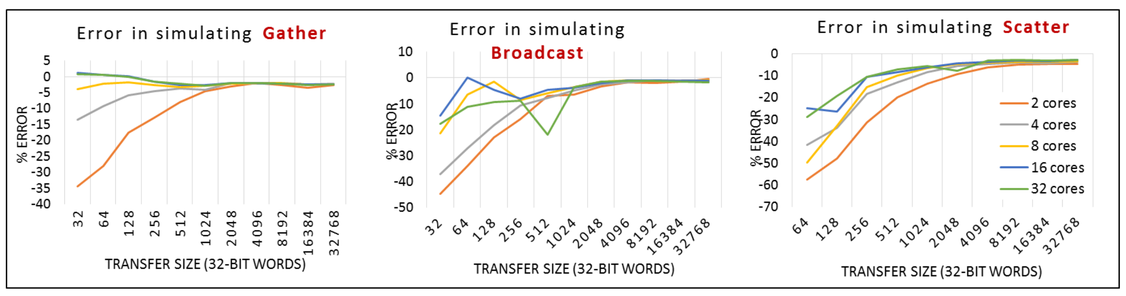
Hop Time: 1ns



Simulation Validation

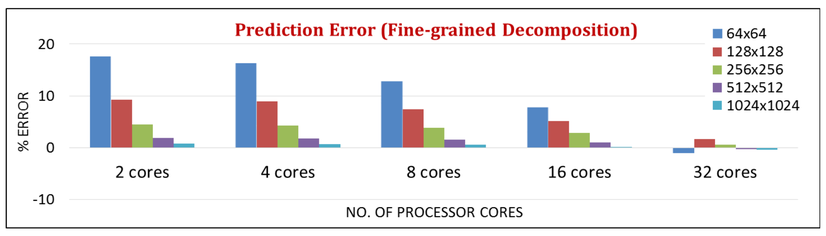
It is important to evaluate the accuracy of BE approach

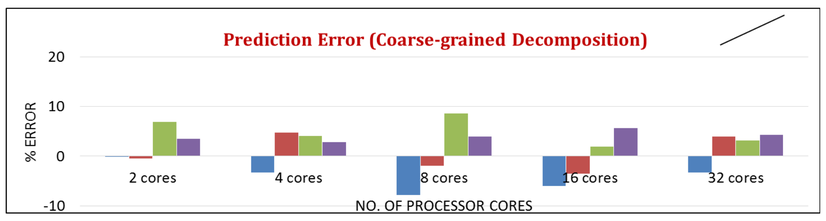
- Study contribution of communication & computation to error
- Evaluate use of fine vs. coarse-grained app decomposition
- **Device:** Many-core device with user accessible mesh network

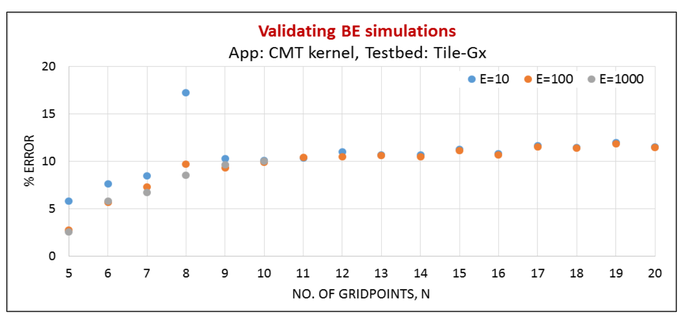


Application: Parallel 2D matrix multiply

- Fine-grained decomposition into dot products
- Coarse-grained decomposition into small matrix multiplies







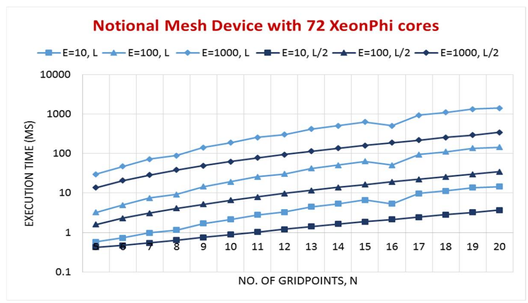
Application: Spectral element solver

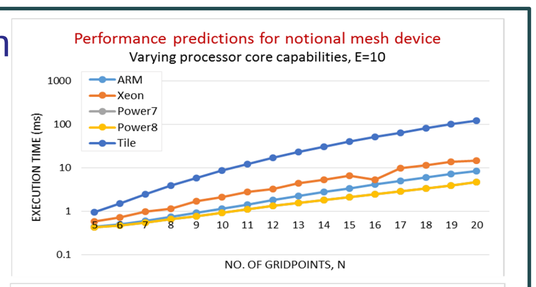
- Used in existing Petascale Nek5000 and under development CMT-Nek code

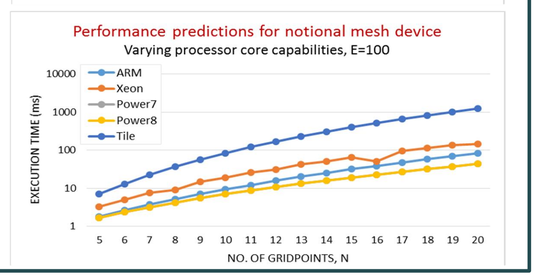
Performance Prediction

Architecture design space exploration with spectral element solver

- Mesh network with different processor cores
- Mesh networks with different latency







Conclusions & Future Work

Conclusions:

- Behavioral Emulation allows co-design from early stages of app development and machine design
- Reasonable simulation accuracy gives some confidence in use of BE for architecture design space exploration

Future Work:

- Extend BE framework for modeling nodes and systems
- Develop and evaluate methods for modeling communication behavior of an app (network congestion) at different scales
- Integrate with an existing scalable PDES (eg. SST) and explore knowledge-based optimizations to the simulation framework

IBM Power8 and APM X-Gene ARM64 calibration was performed on ASC testbeds at Sandia National Laboratories, Albuquerque, New Mexico. All other experiments were performed on CHREC, UF testbeds.

CHREC NSF Center for High-Performance Reconfigurable Computing

UF UNIVERSITY of FLORIDA