Center Extreme Scale
CS Research

Center for Compressible Multiphase Turbulence
University of Florida

Sanjay Ranka
Herman Lam
Outline

- Parallelization and UQ of Rocfun and CMT-Nek beyond a million cores
  - Parallel performance and load balancing
  - Single processor (hybrid) performance
  - Energy management and thermal Issues

- Exascale Emulation with Novo-G
  - Exascale studies using fabrics of Behavioral Emulation Objects
  - Multiscale approach to study archs, apps, and systems
  - Multiobjective studies (performance, environment, dependability)
Communication Mapping: Types of Interactions

Across cells and elements

Levels of AMR

Across immersed interfaces

Lagrangian/Eulerian
Load Balancing: Types of Adaptivity

- Extreme event UQ-driven
- Computational steering
- Preferential particle clustering
- Lagrangian remap
- Adaptive mesh refinement
- Computational power focusing
Single Processor Performance (Hybrid Multicores)

Multiple Cores (e.g. Xeon Phi)

- **Common**
  - Multiple flows of Control
  - Multiple Local Memories

- **Differences**
  - Synchronization
  - Communication

GPU Cores

- **Common**
  - Single/Multiple flows of Control
  - Multiple Local Memories

- **Differences**
  - Amount of Local memory
  - Communication

*2012 AMD A-SERIES OVERVIEW*

- CPU (*Piledriver* Cores)
  - Core 1
  - Core 2
  - Core 3
  - Core 4
  - L2 Cache

- Discrete Class Graphics
  - AMD Radeon™ HD 7000

- “Trinity” Details
  - 32nm
  - ~1.3B transistors
  - 246mm²
  - Up to 4 x86 Cores

*FM2 Socket*
Proposed Research: Hybrid Multicores

- Code Generation for hybrid cores
  - Support for multiple types of cores
  - Support for Vectorization
  - Auto-tuning
- Load Balancing
  - Non-uniform decomposition
- Local Data Movement
  - Movement between different cores and memories
- Multiple levels of Hierarchy
  - Small Caches, Large Caches, Level 2 Caches ...
Develop Static and Dynamic Algorithms for Rockfun and CMT-Nek that can exploit

- Reconfiguration of
  - Processor (DVS)
  - Caches (DCR)
  - Buses
  - Memory
- Support Multi-objective optimization
  - Energy
  - Performance
- Support Multiple Constraints
  - Thermal Issues
Exploiting Reconfiguration for a Single Processor

Dynamic Cache Reconfiguration (DCR)
Multilevel Caches
Cache Reconfiguration

Dynamic Voltage Scaling (DVS)
Slack Reclamation
Energy and Thermal Constrained Scheduling

DVS + DCR System Wide Energy Optimization
Performance and Energy BEOs

BEOs
- Algorithmic Parameters
- Architectural Parameters

For Energy and Performance

NGEE
Emulate Processor, Network, Energy Requirements for Exascale Computation

Rocfun and CMT-Nek
Outline

- Parallelization and UQ of Rocfun and CMT-Nek beyond a million cores
  - Parallel performance and load balancing
  - Single processor (hybrid) performance
  - Energy management and thermal Issues

- Exascale Emulation with Novo-G
  - Exascale studies using fabrics of Behavioral Emulation Objects
  - Multiscale approach to study archs, apps, and systems
  - Multiobjective studies (performance, environment, dependability)
Why Study Exascale Systems?

- Wide variety of major research challenges
  - Design-space exploration and optimization of parallel applications and architectures at Exascale

Exascale Applications

Exascale Architectures

Unique challenges for partitioning, mapping, and optimization at unprecedented scale
How to Study Exascale Systems?

- How may we study Exascale w/o Exascale?
  - Analytical studies – *systems too complicated*
  - Software simulation – *simulations too slow at scale*
  - Behavioral emulation – *to be defined herein*
  - Functional emulation – *systems too massive and complex*
  - Prototype device – *future technology, does not exist*
  - Prototype system – *future technology, does not exist*

- Many pros and cons with various methods
  - We believe *behavioral emulation* is most promising in terms of balance (accuracy, timeliness, scale, versatility)
Approach: Behavioral Emulation with BEOs

- **Behavioral Emulation Objects (BEOs)**
  - Characterize and explore Exascale devices, nodes, & systems, represented by fabrics of interconnected **Architecture BEOs**
  - Architecture BEOs stimulated by corresponding set of **Application BEOs**

- **Multiscale, Multiobjective** (different domain, but same approach)
  - Hierarchical method based upon experimentation and exploration

<table>
<thead>
<tr>
<th>Macro Level</th>
<th>Meso Level</th>
<th>Micro Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMT Apps</td>
<td>System</td>
<td>Device</td>
</tr>
<tr>
<td>CMT Skeleton-apps</td>
<td>BEO fabrics</td>
<td>BEO fabrics</td>
</tr>
<tr>
<td>CMT Mini-apps</td>
<td>Node</td>
<td>Device</td>
</tr>
<tr>
<td>CMT Kernels</td>
<td>BEO fabrics</td>
<td>BEO fabrics</td>
</tr>
</tbody>
</table>

- **NGEE BEO Models**
  - Models abstracted from NGEE-Meso
  - Testbed experimentation in support
  - Notional **Exascale system** exploration

  - Models abstracted from NGEE-Micro
  - Testbed experimentation in support
  - Notional **Exascale node** exploration

  - Architectural studies
  - Testbed experimentation as foundation
  - Notional **Exascale device** exploration

---

**Notes**
- **Hierarchy**
  - Macro: CMT Apps, System, Device
  - Meso: CMT Skeleton-apps, Node
  - Micro: CMT Kernels

**Key**
- **CMT**: Computational Modeling Team
- **NGEE**: National Global Earth System Modeling Initiative
- **BEO**: Behavioral Emulation Object
Proposed Exascale Emulator

Pre-Emulation Platforms

- Many-core experimental testbeds
  - Examples: Intel, IBM, Nvidia, Others

Modeling, simulation, & estimation tools

- Examples: VisualSim, SST Micro, Others

Emulation Platform

- BEOs representing Exascale devices, nodes, or systems mapped to emulation platform
- BEO method independent of emulation platform
  - Discrete-event simulation modeling tool (e.g., VisualSim)
  - Software on conventional (many-core) computer
  - Software-defined hardware on reconfigurable supercomputer (e.g., Novo-G)
Three-Phase Workflow of NGEE

1. BEO Development
   - Miniapps, Kernels, Applications
   - Modeling
   - Architectural Studies
   - Experimentation
   - Application BEO Design
   - Architecture BEO Design

2. BEO Calibration
   - Testbed Experimentation, and External Simulators
   - Analysis and Refinement
   - Architecture BEO Calibration
   - Microbenchmarking, Experimentation, and External Simulators

3. Architecture Exploration
   - Analysis and Refinement
   - To external systems
   - Exploratory Studies
   - Metrics and Statistics
   - Visualization Tool
   - VizWall
Example:

**Micro Behavioral Emulation of Notional Device**

1. **BEO Development**
   - Application BEO Design
   - Architecture BEO Design
   - CMT kernel: parallel matrix multiplication (MM)

2. **BEO Calibration**
   - Run MM on TILE-Gx when possible; Run models on simulator when necessary
   - Feedback & model adjustments if necessary
   - Calibrate based on app-specific & workload params.

3. **Architecture Exploration**
   - Map app BEOs to emul. platform
   - Map arch BEOs to emulation
   - V&V and feedback
   - Architecture exploration using emulation platform

- Tile BEO (core)
- Comm BEOs (3D torus, 6D hypercube, etc.)
- TILE-Gx BEO (36 cores w/2D mesh)
- # of Tiles – O(1000))
- Comm BEO (3D torus, 6D hypercube, etc.)
Toward Emulation of Exascale Systems

1. BEO Development
   - Models of CMT mini-apps & skeleton-apps
     - Models of existing Petascale nodes & systems
     - Models of notional Exascale nodes & systems (based on vendor roadmaps)

2. BEO Calibration
   - Feedback & model adjustments if necessary
     - Run partially on existing systems & simulators when possible

3. Architecture Exploration
   - Map App BEOs to emul. platform
   - Map Arch BEOs to emulation

4. Calibration when possible

5. Application BEO Design

6. Application BEO Calibration

7. Notional Petascale and Exascale exploration studies

8. V&V and feedback

Visualization Tool: VizWall
Fundamental Design of a BEO

BEO is basic primitive in NGEE studies of Exascale systems

- **Performance factors** (execution time, speedup, latencies, throughputs, hotspots)
- **Environmental factors** (power, energy, cooling, temperature)
- **Dependability factors** (reliability, availability, redundancy, overhead)
BEO Emulation Plane

- Contains
  - Static behavioral model (developed in development phase)
  - Set of behavioral parameters (specified dynamically during calibration phase)

- Performs functions to
  - Mimic appropriate behavior of BEO
  - Interact with other BEOs via tokens to support emulation studies
BEO Management Plane

- Responsible for
  - Measuring, collecting, and/or calculating metrics and statistics to support architectural exploration in terms of performance, energy, temperature, reliability, and scalability
  - Interacting with other BEOs

- Separation of management from emulation plane
  - Minimize its interference with emulation functions
NGEE: Novo-G Exascale Emulator

- **Emulation Platform**
  - BEOs representing Exascale devices, nodes, or systems mapped to emulation platform
  - BEO method independent of emulation platform
    - Discrete-event simulation modeling tool (e.g., VisualSim)
    - Software on conventional (many-core) computer
    - Software-defined hardware on reconfigurable supercomputer (e.g., Novo-G)

  - **Commercially available, flexible, ease of use**
  - **For small-scale devices, nodes, and systems**
  - **Emulation platform to be developed in software**
  - **Higher performance than simulators, but not sufficient for Exascale**
  - **Even the proposed BEO approach to emulation is challenging for studying Exascale systems**
    - Exascale, multiscale, multiobjective
  - **High-risk, high-payoff, and only viable solution for full Exascale system emulation**
Novo-G Exascale Emulator (NGEE)

Pre-Emulation Platforms
Many-core experimental testbeds

Examples
- Intel
- IBM
- Nvidia
- Others

Modeling, simulation, & estimation tools

Examples
- VisualSim
- SST Micro
- Others

Novo-G Emulation Platform

3D Torus
Novo-G Supercomputer

- Developed and deployed at CHREC (2009-present, PIs: George and Lam)
  - Supports a broad range of apps, tools, and systems research tasks in CHREC and globally
- Most powerful reconfigurable computer in known world
  - For some apps and uses, could be fastest computer of any kind in world!
  - Yet, 1000s of times less cost, size, power, cooling than high-end conventional supercomputers
- 2012 Schwarzkopf Prize
  - CHREC and Novo-G recognized with 2012 Alexander Schwarzkopf Prize for Technology Innovation at NSF

Novo-G Annual Growth

- **2009**: 96 top-end Stratix-III FPGAs, each with 4.25GB SDRAM
- **2010**: 96 more Stratix-III FPGAs, each with 4.25GB SDRAM
- **2011**: 96 top-end Stratix-IV FPGAs, each with 8.50GB SDRAM
- **2012**: 96 more Stratix-IV FPGAs, each with 8.50GB SDRAM
- **2013**: 32 top-end Stratix-V FPGAs (4x4x2 torus or 5D h-cube)
SST Simulation Framework (c/o Sandia)

Integration with SST

- SST as pre-emulation tool
  - Support BEO development
  - BEO calibration
- SST as run-time tool
  - Dynamic support for emulation studies
Conclusions

- **Novel approach to study systems at unprecedented scale**
  - Behavioral emulation for Exascale studies with BEO fabrics
  - Ideal balance of accuracy, timeliness, scale, versatility

- **Multiscale approach to study archs, apps, and systems**
  - **Micro** = devices for Exascale
  - **Meso** = nodes for Exascale
  - **Macro** = systems for Exascale

- **Multiobjective studies**
  - **Performance** (runtime, speedup, latency, throughput, hotspots)
  - **Environment** (power, energy, cooling, temperature)
  - **Dependability** (reliability, availability, redundancy, o/h)
Do you have any questions?