Integrated Modeling Challenges in Extreme-Scale Computing

Pradip Bose
IBM T. J. Watson Research Center
pbose@us.ibm.com

ISPASS-2011 Keynote; April 12, 2011
Outline of Talk

**Introduction**
- Setting the context: a view of future extreme-scale computing
- What is the primary “wall”: power or reliability?
- Why is pre-silicon modeling a grand challenge in itself?

**Integrated Modeling**
- Power/Temperature, Performance, Reliability
- Levels of Abstraction in Integrated Modeling
  - Relative versus absolute accuracy issues
  - Multi-core power and reliability-aware definition; dynamic management
    - Selected examples to illustrate the modeling complexities

**Concluding Remarks**
What is Extreme Scale Computing?

**Petascale and Exascale Systems**

- Exa- refers to $10^{18}$, which is 1000x Peta-
  - Exascale refers to a system that can handle a million trillion operations per second
- Various government agencies have identified exascale as a critical need in the 2018-2020 timeframe
- In *scientific* communities, the important operation is one floating point operation or calculation
  - Exascale in this context refers to $10^{18}$ flops
  - IBM Roadrunner system: peak of 1 petaflops in 2008
    - Top-ranked system in “Top500” list back in 2008/2009
  - IBM’s Blue Gene product family: L, P, Q systems have consistently been dominant players in the “Top500” and “Green500” lists.
  - So Exascale demands a ~1000x improvement in throughput in 10 years

Many Examples of BIG Applications that Need Extreme Scale Computing

- Whole Organ Simulation
- Smart Buildings
- Nuclear Energy
- Low Emission Engine Design
- Li/Air Batteries
- CO2 Sequestration
- Tumor Modeling

Pratt & Whitney on Intrepid
INCITE PI: Peter Bradley, Pratt & Whitney
- INCITE 2006-2007 technologies now being applied to next generation low emission engines.
- Important simulations can now be done 3X faster
- A key enabler for the depth of understanding and meeting emissions goals

P. Bose, ISPASS-2011 Keynote
The Power Wall → Transition to New Technology

**Bipolar to CMOS Transition**

- **Power**: 15X ↓
- **Density**: 50X ↑
- **Transistor Speed**: 3-4X ↓

**Traditional CMOS to 3D CMOS**

- **Power**: 10X ↓
- **Density**: 3-10X ↑
- **Transistor Speed**: 3X ↓

Opportunity for 3D Si
Power-Performance Wall →
Multi-Cores for the Processor Chip

The Cell Processor Chip

Heterogeneous multi-core chips

Homogeneous

POWER4: 2001
180 nm, Cu, SOI
2 cores / chip

POWER 4+:
130 nm

POWER5: 2004
130 nm, Cu, SOI
2 cores / chip
2 way SMT / core

POWER5+: 90nm

POWER7: 2010
45nm, Cu, SOI
8 cores/chip
4-way SMT/core

P. Bose, ISPASS-2011 Keynote
The Power Wall: A View of the Supercomputer Arena

Oxide thickness is near the limit in late CMOS design era

- Density improvements will continue but... power efficiency from technology will only improve very slowly.
- Historic trend of power efficiency improvement will slow

Nov 2009 Green 500 List:
If the world’s most power efficient supercomputer is extrapolated to a sustained Exaflop (by 2018), power would be ...
~2 GigaWatts

BG/P Compute Chip, 2007

- 4 PPC-440 cores, 850 MHz
- IBM 90nm CMOS ASIC
- 173 sq. mm.
- 208 million transistors
- 16 W

IBM has been a leader in large systems energy efficiency, but meeting the exascale goals is nothing short of a very grand challenge!

System-on-a-Chip (SoC)
General purpose commercial servers have been on a 2X performance every 2 years curve.

But special-purpose HPC supercomputers have been on a ~4X performance every 2 years curve.

Power-efficient accelerator sub-cores for special-purpose functions constitute the vision of workload-optimized hybrid systems of the future – esp. in emerging new application domains – Games market and the Cell multi-core heterogeneous chip was an early trend setter.

Nambiar et al., TPCTC 2010, LNCS 6417, 2011
P. Bose, ISPASS-2011 Keynote
A key principle in use in large-scale parallel HPC systems

Cost constraint for an exascale-regime system implies:

• manageable number of compute nodes $\rightarrow$ dozens of cores/chip

Also, cannot forget the serial (Amdahl) component of HPC codes!
Multi-dimensional tradeoff analysis and design space exploration across targeted workloads requires the support of careful, application-driven, dynamic management capability.

- *Power Shifting* across compute, communication and storage resources.
- Wear-leveling (proactive redundancy) to increase lifetime (MTBF): J. Shin et al. ISCA-2008.

Dynamic power-gating or DVFS features needed to implement power shifting or wear-leveling mechanism.
Application-Driven Dynamic Resource Management

- Multi-dimensional tradeoff analysis and design space exploration across targeted workloads requires the support of careful, application-driven, dynamic management capability
  - *Power Shifting* across compute, communication and storage resources
  - Wear-leveling (proactive redundancy) to increase lifetime (MTBF): J. Shin et al. ISCA08

*Dynamic power-gating or DVFS features needed to implement power shifting or wear-leveling mechanism*
Application-Driven Dynamic Resource Management

- Multi-dimensional tradeoff analysis and design space exploration across targeted workloads requires the support of careful, application-driven, dynamic management capability
  - *Power Shifting* across compute, communication and storage resources
  - Wear-leveling (proactive redundancy) to increase lifetime (MTBF): J. Shin et al. ISCA08

Dynamic power-gating or DVFS features needed to implement power shifting or wear-leveling mechanism
Reliability and Availability: The Other “Wall”

Massive numbers, advanced technologies, and quantity of data produce reliability issues in both hardware and software.

http://www.er.doe.gov/ASCR/ASCAC/Meetings/Aug06/Stevens.pdf

Key point: processors targeted for smaller-size systems are usually not suitable for building large-scale supercomputing systems

**Hardware Failures**
- One million compute nodes, each with a 10 year MTBF would constitute a system that that is likely to fail every 5 minutes

**Software Failures**
- Brute force techniques (checkpointing) may not be feasible due to disk bandwidth
- Time to checkpoint may dominate computation
- Need to look at reliability at the application level

e.g. SWAT project at UIUC (Sarita Adve’s group)

*ANL = Argonne National Lab*
In fact...reliability is (quite possibly) the primary wall!

If $R_N$ increases with $N$

$$R_N = \frac{MTTR}{MTTF} \text{ (recovery overhead)}$$

for a $N$-way system

A reliability-unaware extreme scale design may not even be able to complete a benchmark workload (e.g. Linpack), even with an unconstrained power budget because of too frequent errors and consequent rollbacks!

(See Meeta Gupta et al., MICRO-2009 for local vs. global recovery sensitivities at chip level)
Chip Level Reliability

- Chip-level functional robustness likely to decline in future
  - Increase in transient errors and hard faults
  - Maintaining historic levels of chip-level MTBF: cost-prohibitive
  - Burn-in difficulty, cost due to high power regime
  - Thermal hot spots are a new source of transient/hard failures

- System-level reliability targets: going to be hard to meet
  - Two “system” examples:
    - SoC with hundreds of core / non-core elements
    - Large HPC system with thousands or millions of processor cores/chips [extreme scale computing]

- Need new cost-effective solutions across the entire h/w-s/w system design stack to meet FIT targets at any given level of “system” abstraction
  - Design and analysis tools must evolve as well

Cost implication trend: not sustainable!
Chip/System Level Definition (Modeling) Approaches

A few specific examples
Towards an *integrated* modeling infrastructure


**Power Modeling Enhancements**

- Latch-counts + array power models
- Latch-counts + scaled CPAM based models + refined array power models
- Trace/exec driven simulation

**Package RLC models, Ldi/dt analysis**

**Temperature Modeling**

**Reliability Modeling**

**VALIDATION**

**Substrate Processor Simulator**

- System interconnect and tech. scaling parameters, models
- Uniprocessor CPI and Power sensitivities
- Multi-Core Power-Performance Modeling

**microarch design and definition**

- Toolset evolved: 2000-2008
- Not as integrated as one would like!
- Detailed and slow!
The Pre-Silicon Modeling Challenge in Extreme Scale Systems

- Why is this a grand challenge in itself?
  - Because the constraints are multi-dimensional, interdependent and extremely hard to meet at affordable cost. Example:
    - 20 MW system power
    - 1 exaflops sustained performance
    - MTBF of at least two weeks, preferably 1 month

- And, because cycle-accurate simulation speed is not scaling up
  - Host hardware (simulation platform) speed is not increasing
  - Number of cores and target MIPS is increasing exponentially
  - Cycle-accurate performance simulators are very hard to parallelize
Early Chip Planner Framework at IBM Watson

A step toward better integration of component models
Phased Power Modeling Methodology

- Concept → HLD → Implementation Phase

- Previous Generation Database
- Scaled Architecture Power Models
- MPwr
- SCHSim (circuit power)
- RTLSim (data switch factors)
- Unit Level Clock Gating Efficiency Estimate

- Benchmarks (e.g. SPEC)

- MSim performance model

- Gator Table
  - Gator (calc CGFs)
  - Event & Instr Freq

- Designer
  - Clocking Conditions (event expressions)

- Current Database
  - Performance Validation

- VHDL Contract

- Power Projection for Given Workload

2000+ pstats

H. Jacobson et al., HPCA-17, 2011
Power Model Requirements in the Many-Core System Era

- **Core-level abstraction is a must (for speed)**
  - Facilitates multi-core DPM algorithm studies
  - Also, fast power-perf tradeoff analyses for core
- **But… detailed reference model useful for macro-wise power budgeting and tracking**
  - Core power projection accuracy is important
- **Linear regression based abstraction is a very useful technique**
  - H. Jacobson et al., HPCA-17, 2011
    - See also: previous work: Powell et al. (HPCA 2010), Lee and Brooks (ASPLOS 2006)
- **POWER7 chip-specific model**
  - Detailed p7 reference power model
  - Formal attribute selection method
  - Support for microarchitecture scalability
Reference Power Model

- **p7 microprocessor chip**
  - High frequency aggressive superscalar out-of-order design
  - 32-thread, 8 core, 32kB I/D caches, 256kB L2 cache

- **p7 core reference power model**
  - Suitable for macro-level power analysis, tracking
  - 2300 µarch stats
  - 500 RTL macros
  - 2800 modeled clock/port/data gating domains

POWER7 (p7) Core + L2

H. Jacobson et al., HPCA-17, 2011
Power Model Abstraction

- Abstract model obtained through linear regression
  - 15,000+ sets of event stats obtained from simulation of Spec2k6, Commercial, Multimedia, and other workloads
  - Power calculated using reference model for each set of event stats
  - Linear regression performed to create abstract power model
    - \( \text{power} = C_0 + C_1*S_1 + \ldots + C_n*S_n \)
  - 10/90 coverage test used to validate the final power model

H. Jacobson et al., HPCA-17, 2011
A few attributes explain most of power variance
- First 8 principal component attributes explain 99% of variance
- Not necessarily the best for intuitive understanding by humans or ease of implementation

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Explained % of Variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>500</td>
<td>25</td>
</tr>
<tr>
<td>2000</td>
<td>90</td>
</tr>
<tr>
<td>2500</td>
<td>100</td>
</tr>
</tbody>
</table>

Explained % of Variance

H. Jacobson et al., HPCA-17, 2011
The Importance of Selecting the Right Attributes

- Single attribute (1)
  - IPC fitness corr. 0.905
  - Significant error spread

- Random attributes (8)
  - Best fit corr. 0.976
  - Worst fit corr. 0.109

- Domain experts (8)
  - Expert A fit corr. 0.968
  - Expert B fit corr. 0.971

**Conclusion**
- Need systematic approach to select high quality attributes
- See HPCA-17 paper for details
**Processor Core Power Proxy: A Hardware Feature in p7**

*IEEE Micro, 2011 (to appear)*  
*IBM J. R&D, vol. 55, no. 3, 2011*

**Goal:**
Estimate per-core chiplet power that we cannot directly measure

**Method:**
- For each functional unit, pick small subset of activities to infer power consumption (*e.g. cache & regfile reads & writes, execution pipeline issue)*
- Weight each activity to represent how much relative power it consumes
- Combine weighted Core, L2, and L3 activity, then add constant offset plus clock grid power to form:

\[
\text{Chiplet Active Power} = \sum (W_i \times A_i) + C + Kf
\]

**Result:**
- EnergyScale Firmware adjusts this value for effects of leakage, temperature, and voltage

---

*Hardware design was driven by power model abstraction research at IBM Watson (A. Buyuktosunoglu et al.)*

*Statements regarding EnergyScale features do not imply that IBM will introduce a system with this capability*
Power Proxy Measurements

- EnergyScale firmware budgets power across multiple processors and memory, used to:
  - Shift power to cores or other components (e.g. memory) that need it the most
    (Especially important to achieve higher overall performance under a power cap)
  - Enable Server Partition power accounting

Statements regarding EnergyScale features do not imply that IBM will introduce a system with this capability.
Pitfalls of Architectural Abstractions: An Example from Soft Error Rate (SER) Analysis

System SER = \sum [AVF(i) \times Raw\_SER(i)] \quad \text{AVF} + \text{SOFR abstraction}

Errors in AVF+SOFR-based estimation get very large, when number of modeled cores, C in the system becomes very large, or if the raw error rate of each of the N cores becomes very large.

Power Model Calibration/Validation Methodology

- **HotGen**
  - Parameter file
  - e.g. FXU utilization target: 30%

- **Microbenchmark**
  - Measurement

- **SIMP**: Actual chip with IR camera
  - H. Hamann et al. JSSC, Jan 07

- **Integrated Model (Power, Temp, Perf)**
  - Simulation
  - Compare

- **Calibrate**
  - Test case generation

**Zhigang Hu et al. 2005-06**
POWER5 Hotspot Patterns

- 50 different workloads for POWER5 imaged & analyzed
  - HotGen microbenchmark generator tool
- observed significant differences in circuit utilization

(H. Hamann et al., ISSCC-2006)
Optimal Pipeline Depth: TPCC Workload

Power-performance optimal

Performance optimal

Total FO4 Per Stage

bips

bips^3/W

Relative to Optimal FO4

0 0.2 0.4 0.6 0.8 1

V. Srinivasan et al., MICRO-2002
V. Zyuban et al., IEEETC, 8/2004

Note: Optimal point on x-axis is the important output of
such an analysis model; y-axis value absolute accuracy not very important!
CMP Space Exploration Results

The optimal core-count for a given core type

Analytical or hybrid models do quite well in such scenarios

Yingmin Li, Zhigang Hu et al., HPCA 2006
Chip-level Lifetime Reliability Analysis

◊ Floorplan

◊ Power

◊ Temperature

◊ FIT due to EM

◊ FIT due to NBTI

◊ FIT due to TDDB

Jeonghee Shin et al., DSN-2007, ISCA-2008

P. Bose, ISPASS-2011 Keynote
Power-Performance Tradeoffs (on-chip, global power management; DVFS): A Key Modeling Challenge!

- MaxBIPS within 1% of Oracle
- Verification complexity of multi-core power management algorithms – scalability – is a key issue [A. Lungu et al. MEMOCODE 2009]
Activity migration \textit{[temperature-aware task scheduling]} reduces maximum on-chip temperatures.

(a) DAXPY running on core 0

(b) DAXPY running on core 1

(c) DAXPY hopping every 7ms

Chip designs could leverage the lower temperatures for higher frequencies, lower-cost packaging or enhanced reliability.
Leveraging Spatial Heat Slack

Activity Migration reduces Hotspots

Summary: Core-hopping (4ms) reduces maximum on-chip temperature

Measurement-based analysis; very hard to project accurately via simulation

Maximum delta temperature

Workloads

<table>
<thead>
<tr>
<th>Workloads</th>
<th>Reduction in Temperatures (Celsius)</th>
</tr>
</thead>
<tbody>
<tr>
<td>daxpy</td>
<td>5.5</td>
</tr>
<tr>
<td>apsi</td>
<td>4.2</td>
</tr>
<tr>
<td>fma3d</td>
<td>3.3</td>
</tr>
<tr>
<td>lucas</td>
<td>4.9</td>
</tr>
<tr>
<td>swim</td>
<td>5.1</td>
</tr>
<tr>
<td>bzip2</td>
<td>2.2</td>
</tr>
<tr>
<td>twolf</td>
<td>2.3</td>
</tr>
<tr>
<td>vortex</td>
<td>2.0</td>
</tr>
<tr>
<td>vpr</td>
<td>3.5</td>
</tr>
</tbody>
</table>

% slow down

<table>
<thead>
<tr>
<th>Workloads</th>
<th>% slow down</th>
</tr>
</thead>
<tbody>
<tr>
<td>daxpy</td>
<td>0.1</td>
</tr>
<tr>
<td>apsi</td>
<td>-1.1</td>
</tr>
<tr>
<td>fma3d</td>
<td>-0.5</td>
</tr>
<tr>
<td>lucas</td>
<td>0.4</td>
</tr>
<tr>
<td>swim</td>
<td>1.0</td>
</tr>
<tr>
<td>bzip2</td>
<td>1.1</td>
</tr>
<tr>
<td>twolf</td>
<td>1.6</td>
</tr>
<tr>
<td>vortex</td>
<td>0.9</td>
</tr>
<tr>
<td>vpr</td>
<td>2.5</td>
</tr>
</tbody>
</table>

J. Choi, C-Y, Cher et al., ISLPED07
Power Gating as a Dynamic Management Knob

- Power Gating (PG) is becoming an essential actuation knob for dynamic power management
  - Header or footer transistor gates off power to the “macro” during idle durations
  - Applied at core-level (per-core PG) or within a core at the unit-level
  - PG is applicable to a broad range of compute nodes that exhibit variable idle times
    - Mobile, Desktop, Enterprise etc.

- Our end target is efficiency at all levels: from chips, all the way through to the data center level
Methodology for Core-Level Power Gating Analysis

- Use bit-vector traces (utilization) from instrumented cycle-accurate perf. simulator
- Workloads: SPEC, other traces
- Implement trace driven simulator for power gating algorithms, obtain:
  - Leakage power savings estimate
  - Projected performance impact
    - Assume constant performance impact of 3 cycles on wake-up

A. Lungu et al., ISLPED-2009
Power Savings Potential for Power Gating of Functional Units

Power gate potential function of break-even point for **FXU0 and FXU1** units

- **FXU0, FP benchmarks**
- **FXU0, INT Benchmarks**
- **FXU1, FP Benchmarks**
- **FXU1, INT Benchmarks**

% Leakage Savings

- **Power Savings Potential for Power Gating of Functional Units**

Power gate potential function of break-even point for **LSU0 and LSU1** units

- **LSU0, FP benchmarks**
- **LSU0, INT Benchmarks**
- **LSU1, FP Benchmarks**
- **LSU1, INT Benchmarks**

% Leakage Savings

- **39.77**
- **46.85**
- **60.66**
- **65.26**

Lungu et al., ISLPED-2009

Large Potential for Power Gating!
Pitfalls of Current Power Gating Algorithms

- Idle interval prediction can be consistently wrong:
  - => power gating algorithm consistently wastes power instead of saving

- Possible scenarios in loops
  - Idle monitor failure
    - Idle detect 3, break-even 20
    - Average leakage power loss 100%
  - Utilization monitor failure
    - Utilization threshold 30%
    - Average leakage power loss 98.5%

A. Lungu et al., ISLPED-2009
Single Level Idle Detect Power Gating Algorithm

Power savings of idle counter solution function of idle_detect for FXU0 unit (FP benchmark)

Projected performance impact of idle counter solution (FP benchmark)

A. Lungu et al., ISLPED-2009
Two Level (Guarded) Power Gating Algorithms

- **Observations:**
  - Efficiency requirement of power saving schemes: **save power**
  - Single level idle prediction algorithms can behave incorrectly and **waste power**

- **Target:**
  - Improve quality of power gating schemes by reducing or eliminating their risk of wasting power

- **Idea:**
  - Add second level monitor to control enabling of power gating scheme
    - Improve efficiency of power wasting cases without degrading power saving of the common case

A. Lungu et al., ISLPED-2009
Power Gating in a Datacenter Setting

N. Madan et al., HPCA-17, 2011
Problems with Core-Level Power Gating

Cannot be aggressive with PG as penalties can be huge
Cannot be overly conservative as power saving potential is lost
Augmenting Core-Level Power Gating with Guarding

Resource Utilization, Idle & Burst Distribution

Power Gating Module

Guarded Gating Module

Perf Loss% #Wake-ups

#Cores ON/OFF Unit-level PG

(Dis/En)able Gating

Incoming Tasks

PWR ON/OFF

Core1

Core2

CoreN

NETWORK

End Users

N. Madan et al., HPCA-17, 2011

Bose, ISPASS-2011 Keynote
Proposed Guard Mechanism

- Monitor system response time
  - Response time can be very high when the system is overly utilized
- Monitor number of core wake-ups
  - Wake-up latency and switching power can be negligible too
- Only If **both** monitors show unacceptable behavior
  - Disable power manager
- Re-enable power manager after a programmable time period
- Alert the system manager

See N. Madan et al., HPCA-17, 2011 for Evaluation Results

More coverage at: Energy-Secure Architectures: Tutorial at ISCA-2011
Queuing Model Based Evaluation Framework

Statistical Model: Task Length, Inter-Arrival Time

USC Datacenter Trace: Task Length, Inter-Arrival Time

Power Gating Module (IdlePG, UtilPG)

Incoming Tasks

PWR ON/OFF

#Cores ON/OFF

Core1

Core2

CoreN

Tasks with Expired Time Slice

See N. Madan et al., HPCA-17, 2011 for evaluation results
Concluding Remarks

- **Power and Reliability Walls are Key Impediments to Realization of Extreme Scale Computing Targets of the Future**
  - Reliability may well be the more fundamental obstacle beyond a certain size of the system

- **Integrated Models (power/temperature, performance, reliability) are a Grand Challenge**
  - Analytical abstraction methods are essential for speed
  - Yet, accuracy requirements at core/chip and other component level are more stringent than ever because of the implications of the huge scale (system size)
Thank you!