Modeling and Simulation of Many Core Architectures

Sudhakar Yalamanchili

Center for Experimental Research in Computer Systems
School of Electrical and Computer Engineering
Georgia Institute of Technology

Motivation

"If you can not measure it, you can not improve it."

Lord Kelvin

"Nothing can be more fatal to progress than a too confident reliance on mathematical symbols; for the student is only too apt to take the easier course, and consider the formula not the fact as the physical reality."

Lord Kelvin

Many Core Simulation Group@GT

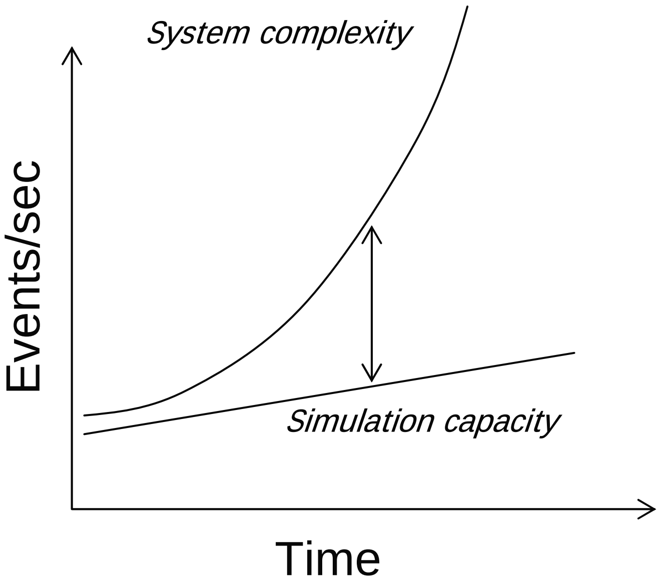
- Faculty
  - Tom Conte (SCS)
  - S. Mukhopadhyay (ECE)
  - George Riley (ECE)
  - S. Yalamanchili (ECE)

- Graduate Students
  - Paul Bryan
  - Gregory Diamos
  - Brian Hayes
  - Chad Kersey
  - Minki Lee
  - Elizabeth Lynch
  - Nikil Sathe

Joint Effort between CERCS and new Center for Manycore Computing

The State of the Practice

- System complexity is outpacing simulation capacity
  - Cannot perform analysis at scale
- Islands of simulators and simulation systems
  - Customized interactions
  - Little leverage of individual investments
- The problem will get worse faster



Simulation Wall!

Prioritized Major Challenges*

1. Cost of building a validated useful simulator
    - Composable
    - New methodologies for building simulators
2. Accuracy
    - Need for calibrated models
    - Methodologies for constructing calibrated models
3. Performance
    - Parallelism, multiscale, and hardware acceleration
4. Power and thermal models
5. Ease of use: Productivity and Management Tools
    - Visualization, deployment, debugging, etc.
    - Documentation & deployability

*From Outbrief: *Performance Prediction and Simulation for Exascale Interconnection Networks,* Interconnect Workshop, DoE Institute for Advanced Architectures, July 2008

Some things to Keep in Mind

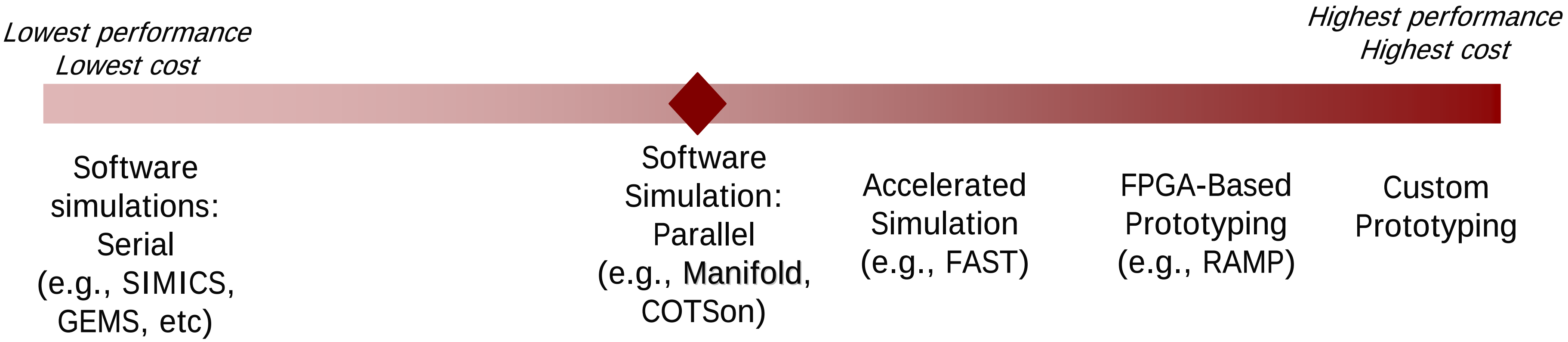
- Research
  - Requirements are changing, unknown, or speculative
  - Modeling what does not exist at the Exascale
  - Confidence levels

- Education
  - Lack of discipline-oriented courses
  - Need more rigor in education for architecture/system modeling and simulation
  - Knowledge of the third kind

Key Challenges

- Managing complexity (items 3 and 5)
  - Parallelism, sampling, acceleration
- Multi-model simulations (items 2 and 4)
  - Power and thermal challenges
  - Feedback between thermal and discrete event simulation
- Productivity (items 1 and 5)
  - Cost of simulator construction
  - Ease of use

Spectrum of Solutions



Lowest performance
Lowest cost

Highest performance
Highest cost

Software simulations: Serial (e.g., SIMICS, GEMS, etc)

Software Simulation: Parallel (e.g., Manifold, COTSon)

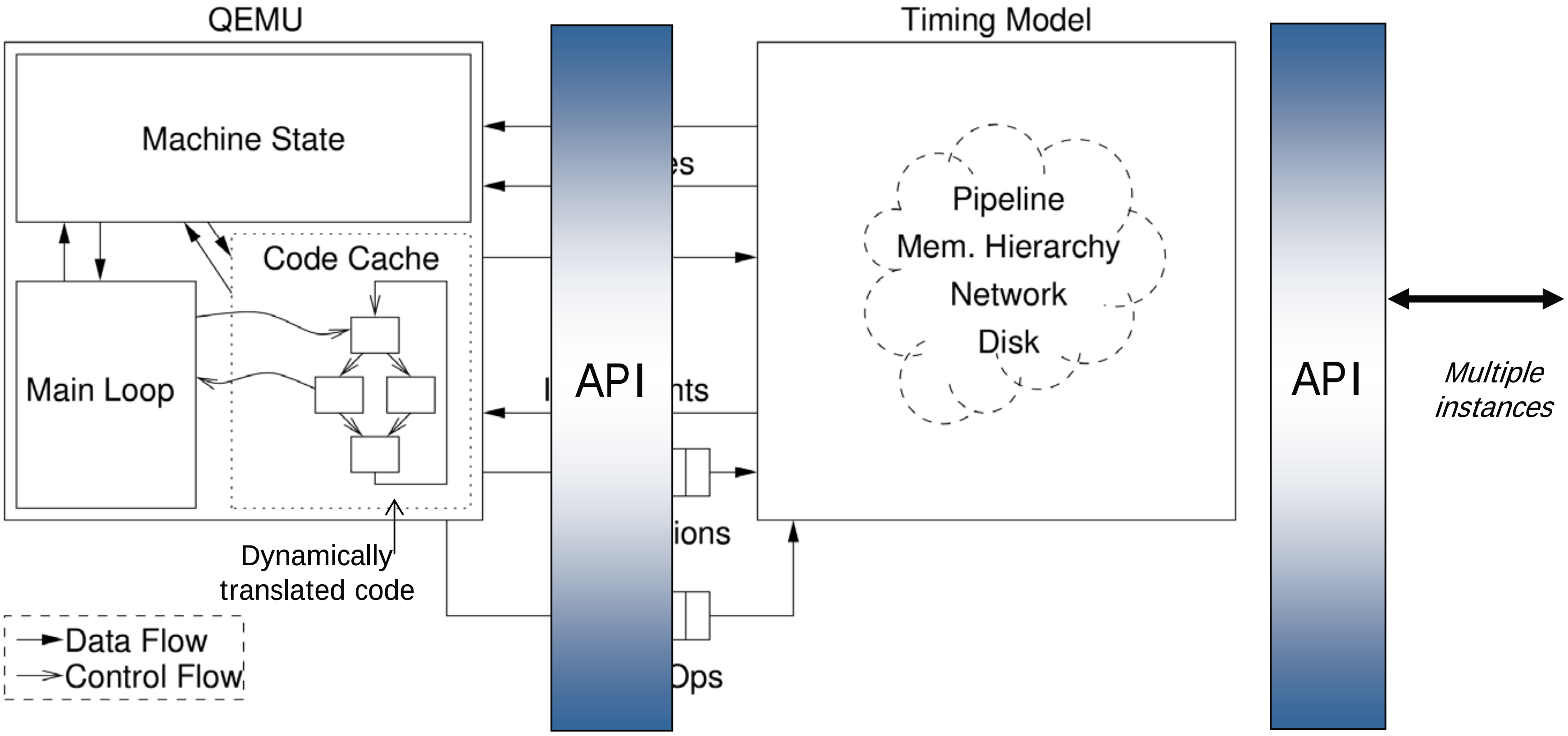
Accelerated Simulation (e.g., FAST)

FPGA-Based Prototyping (e.g., RAMP)

Custom Prototyping

- Simple Premise: Use parallel machines to simulate/emulate parallel machines
- Leverage mature point tools via standardized API for common services
  - Event management, time management, synchronization
- Cull the design space prior to committing to hardware prototyping or hardware acceleration strategies

Managing Complexity

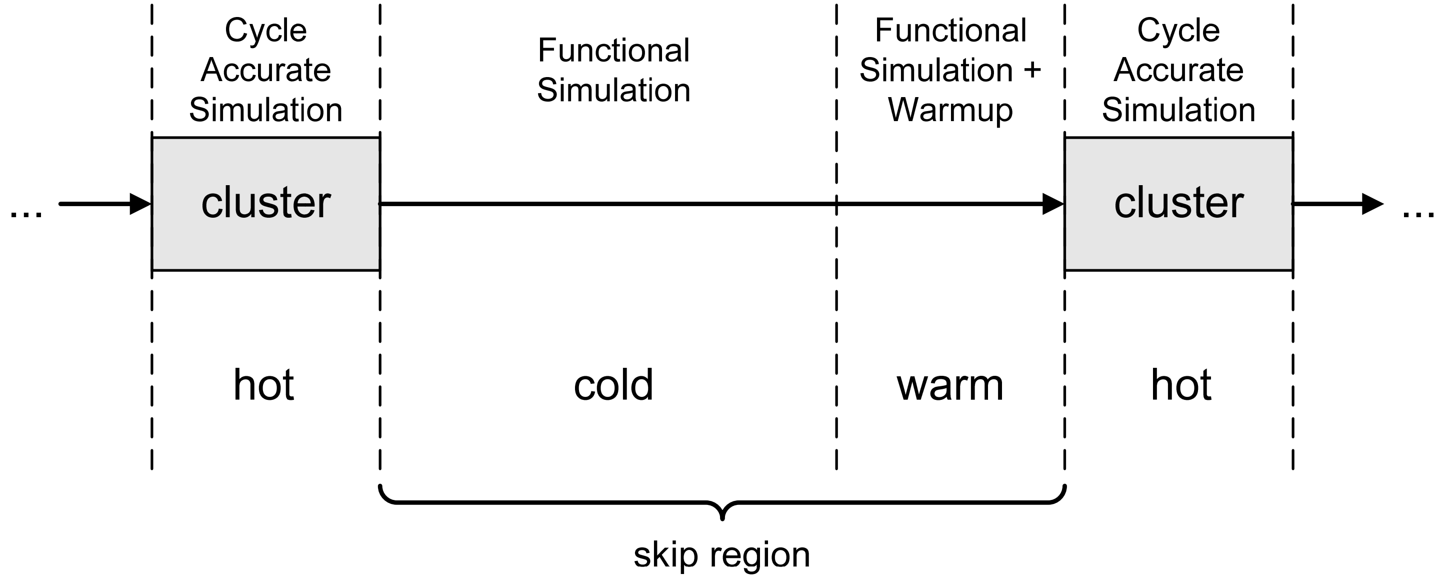


- Coupling timing models and functional models
- Timing models can be 4-5 orders of magnitude slower than real time

Solution Techniques

- Statistical Techniques
- Parallel Simulation
- Acceleration
  - FPGAs and more recently GPUs
- Regression and analytic models for design space exploration
  - For example, work of Lee& Brooks@Harvard

Cluster Sampling for Processor Simulation



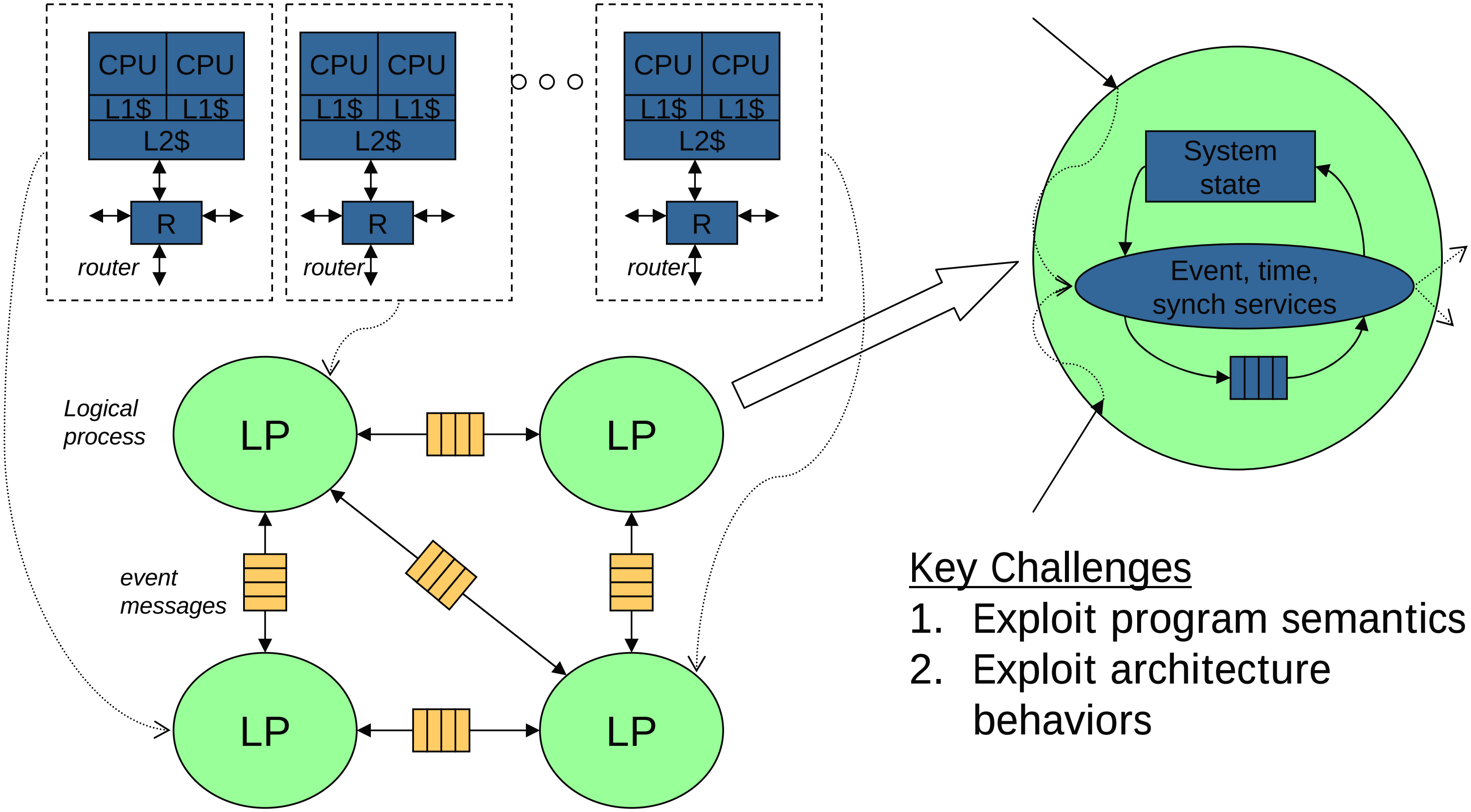
- Hot: simulator state is known
  - Measurements are accurate
- Warm: transition between cold to warm
  - Used to get the simulator to a known state
- Cold: simulator state is unknown
  - Measurements would not be accurate

Challenge: Extensions to multithreaded/parallel codes

Courtesy: Paul Bryan & Tom Conte

Coarse Grain Parallel Simulation

Example Modeled System



Key Challenges

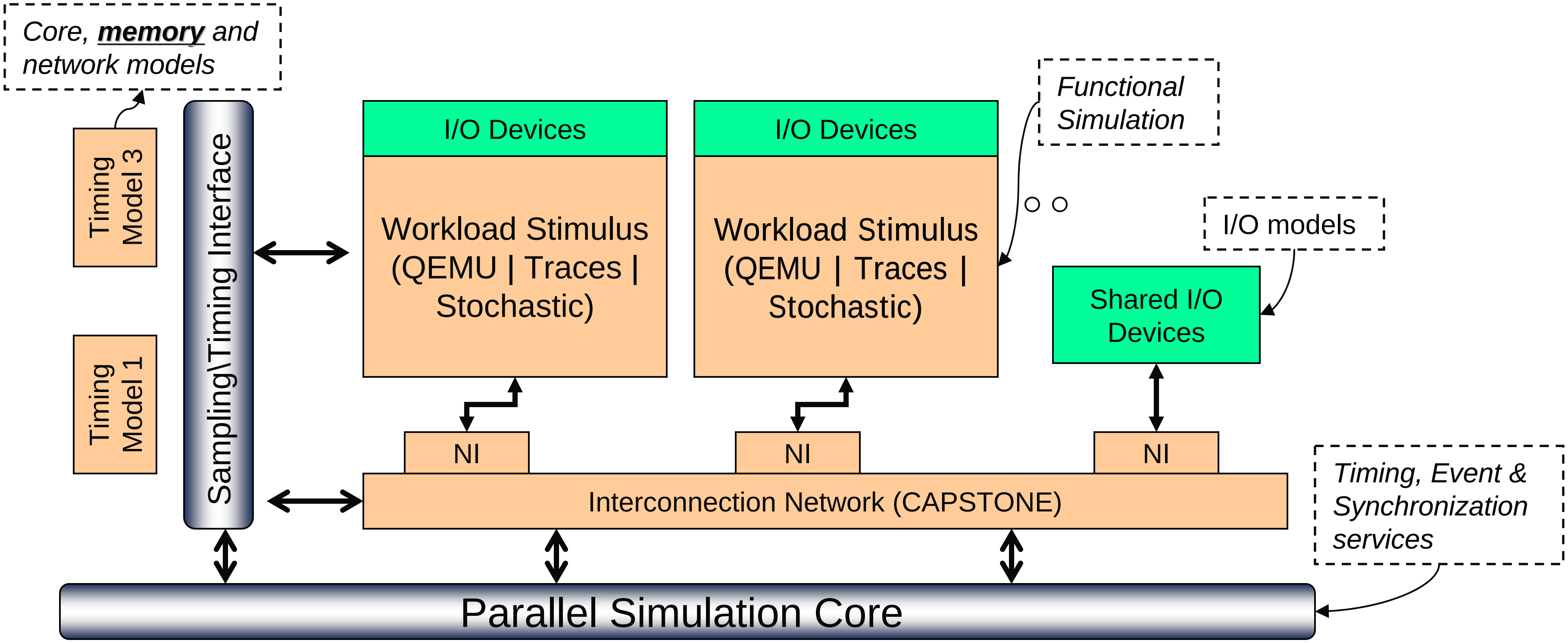
1. Exploit program semantics
2. Exploit architecture behaviors

1. D. Burger and D. Wood, “Accuracy vs. Performance in Parallel Simulation of Interconnection Networks”, *ICPP* 1995
2. A. Falcon P. Faraboschi D. Ortega, “An Adaptive Synchronization Technique for Parallel Simulation of Networked Clusters,” *ISPASS*, 2008
3. Parallel SST, SNL

Some Simple Goals

- Get to 2-3 orders of magnitude slower than real time for timing simulations
  - Use Petaflop machines to simulate Exascale Machines?
- Consider hardware support for global virtual time
  - Lynch & Riley (ongoing work)
  - Support in the NICs?
  - Hardware barrier synchronization support from the 90’s
  - Hardware, fine-grained all-to-all support

Manifold: Overview



Sponsor: Sandia National Laboratories

What Can Be Done?

- Provide an infrastructure to integrate mature point tools
  - Standardized API for time, event, synchronization, and management services
  - Support both time stepped and discrete event simulation
  - Central role for rigorous statistical methods
  - Near Term integration of QEMU with HP Labs COTSon
- Scale-up: Track Moore’s Law for Simulation Capacity?
  - Double simulation capacity every 12-18 months
  - High level composition of detailed models: on-chip and off-chip
  - Enable migration of models across hardware and software simulation platforms

Major Challenges

- Managing complexity (items 3 and 5)
    - Parallelism, sampling, acceleration
- Multi-model simulations (items 2 and 4)
    - Power and thermal challenges
    - Feedback between thermal and discrete event simulation
- Productivity (items 1 and 5)
    - Cost of simulator construction
    - Ease of use

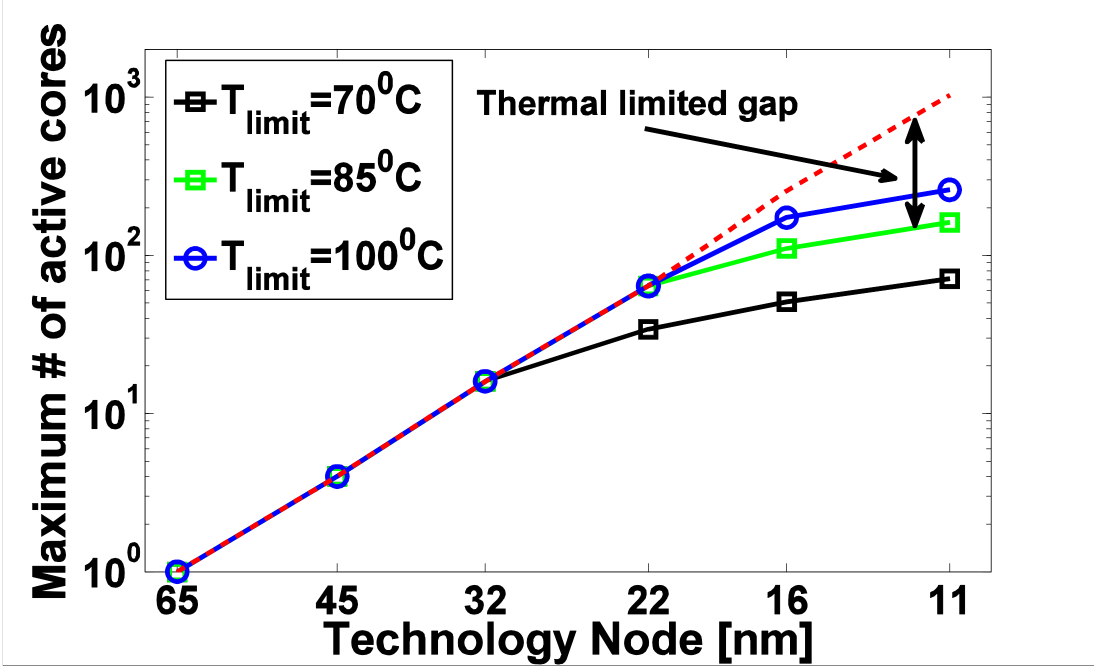
Power and Thermal Modeling

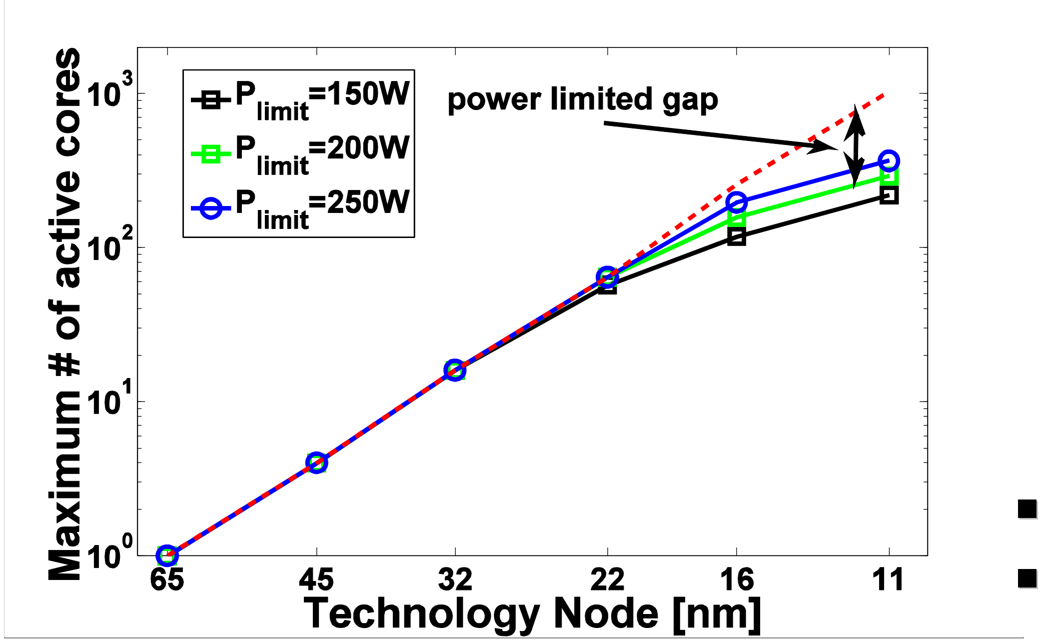
- Seek a fundamental understanding of energy and thermal challenges at Exascale

- Develop, model, and assess (new) architectural principles for energy management
  - Architectural techniques for energy management
  - Need to couple physics of heat management with detailed architecture simulation

Thermal and Power Scaling Limits

Temperature Limited Performance

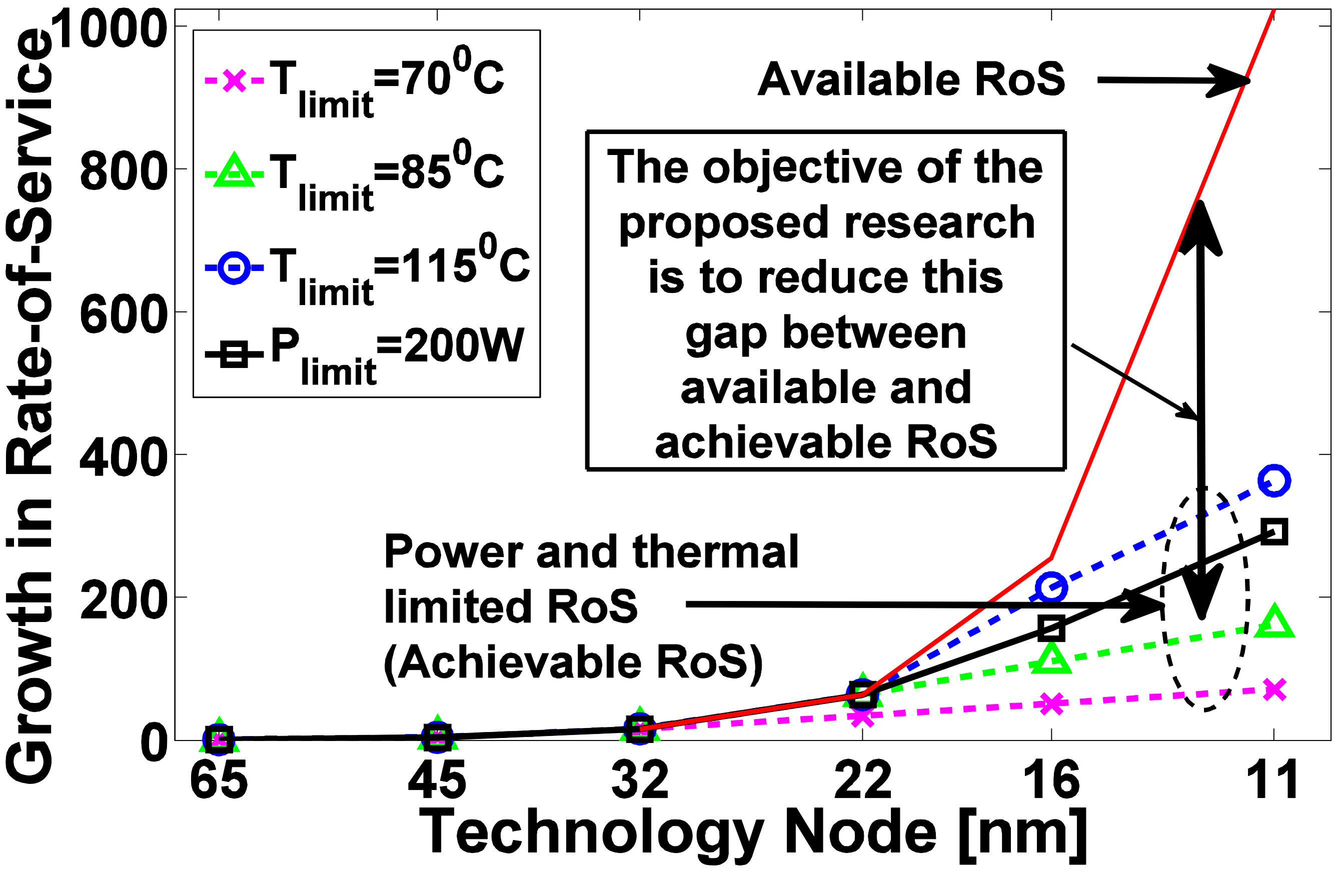




Power Limited Performance

- Hot Spot (UVa)
- IntSim (J. Meindl's group@GT)

Feasibility vs. Capacity Gap



The objective of the proposed research is to reduce this gap between available and achievable RoS

Power and thermal limited RoS (Achievable RoS)

Mukhopadhyay and Yalamanchili

Scaling Principles

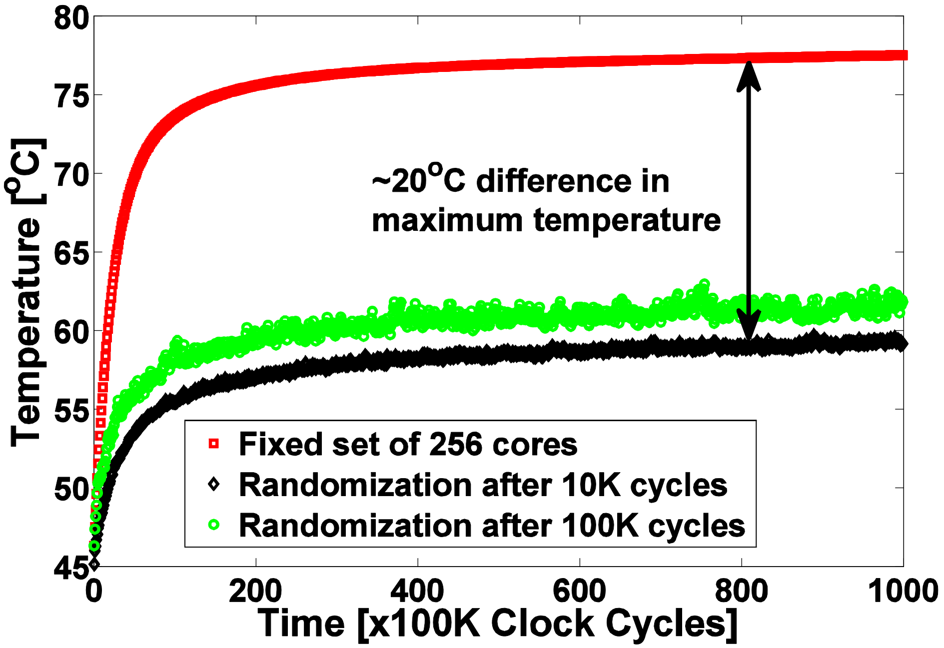
- Dynamic core scaling
  - Analog of traditional voltage-frequency scaling

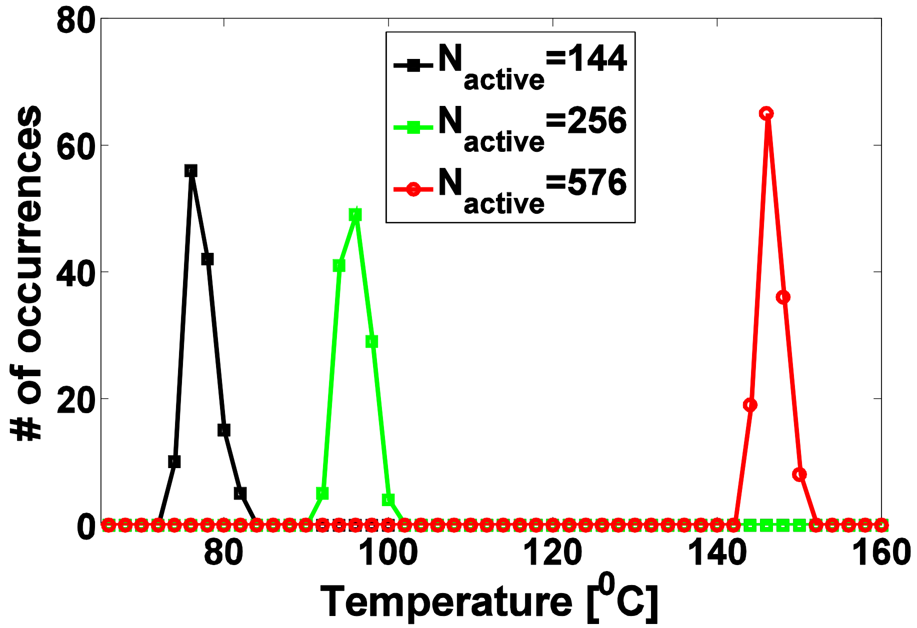
- Spatial scaling
  - Metrics for thermal proximity and thermal compactness for heat management
  - Exploit the physics!

- What do you manage architecturally?
  - Gradients vs. peak temperature

Effects of Spatiotemporal Scaling

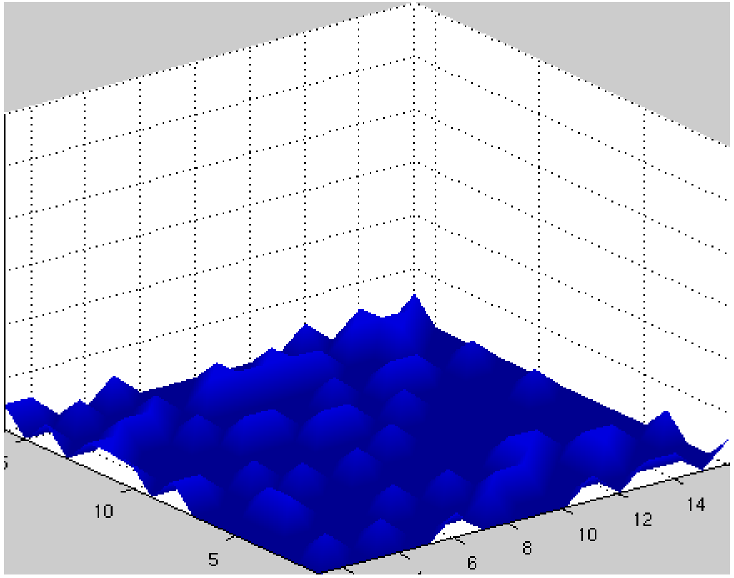
Randomized migration



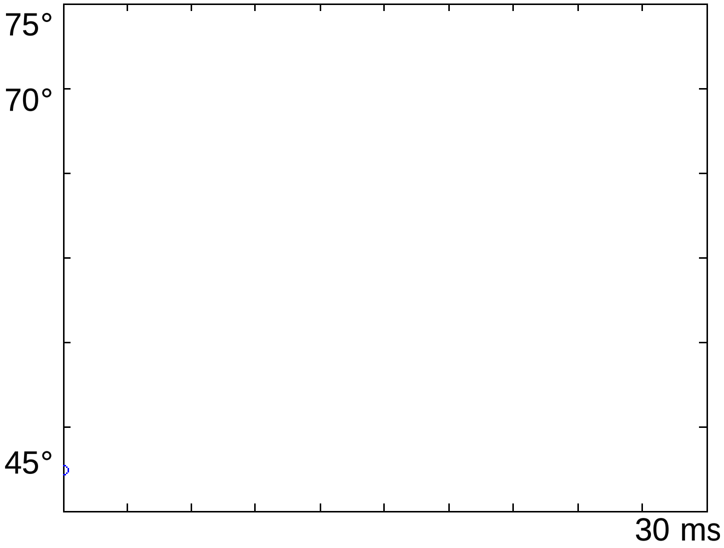


Impact of spatial scaling

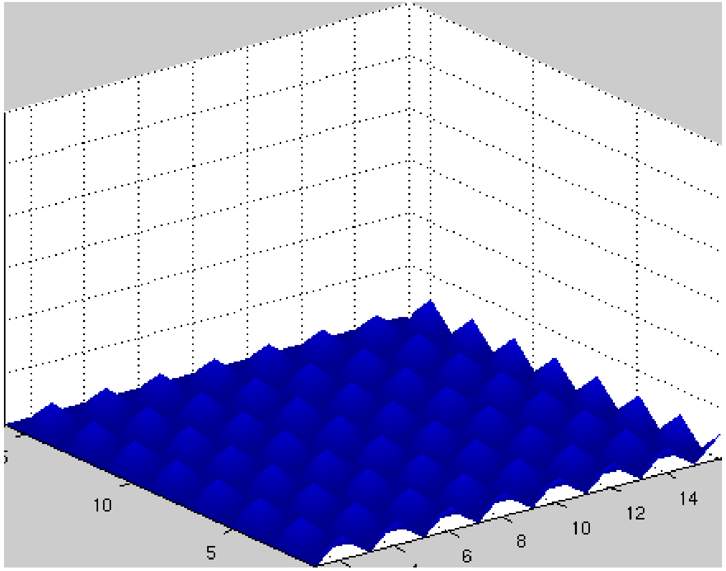
Random Pattern of On-Tiles



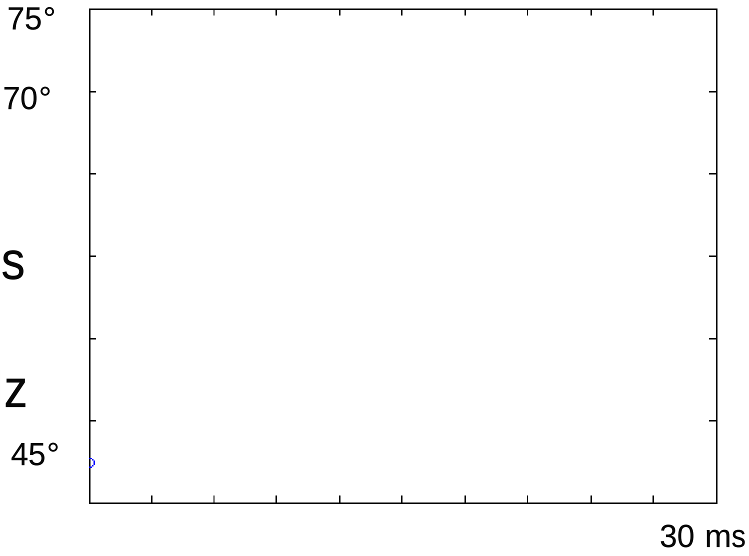
Spatial gradient:
10.5°@0.75mm
Temporal gradient:
2.5°@100Kcyles



Cyclic Pattern of On-Tiles



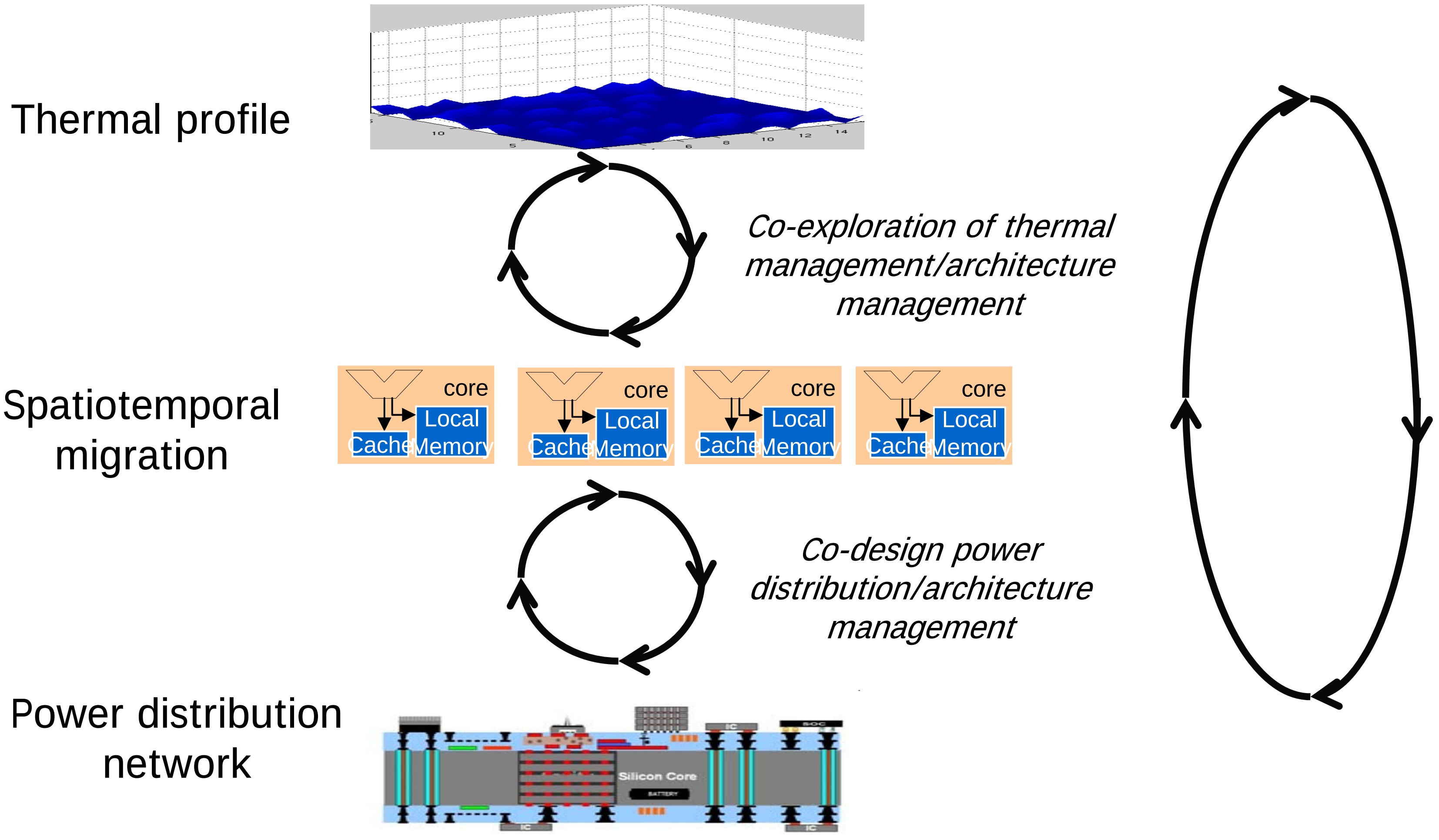
Spatial gradient:
2.5°@0.75mm
Temporal gradient:
1.99°@100Kcyles



- 64 on-tiles
- 256 total tiles
- 100K cycles interval@3GHz

Courtesy: Nikil Sathe

The Need for Feedback



Thermal profile

Co-exploration of thermal management/architecture management

Spatiotemporal migration

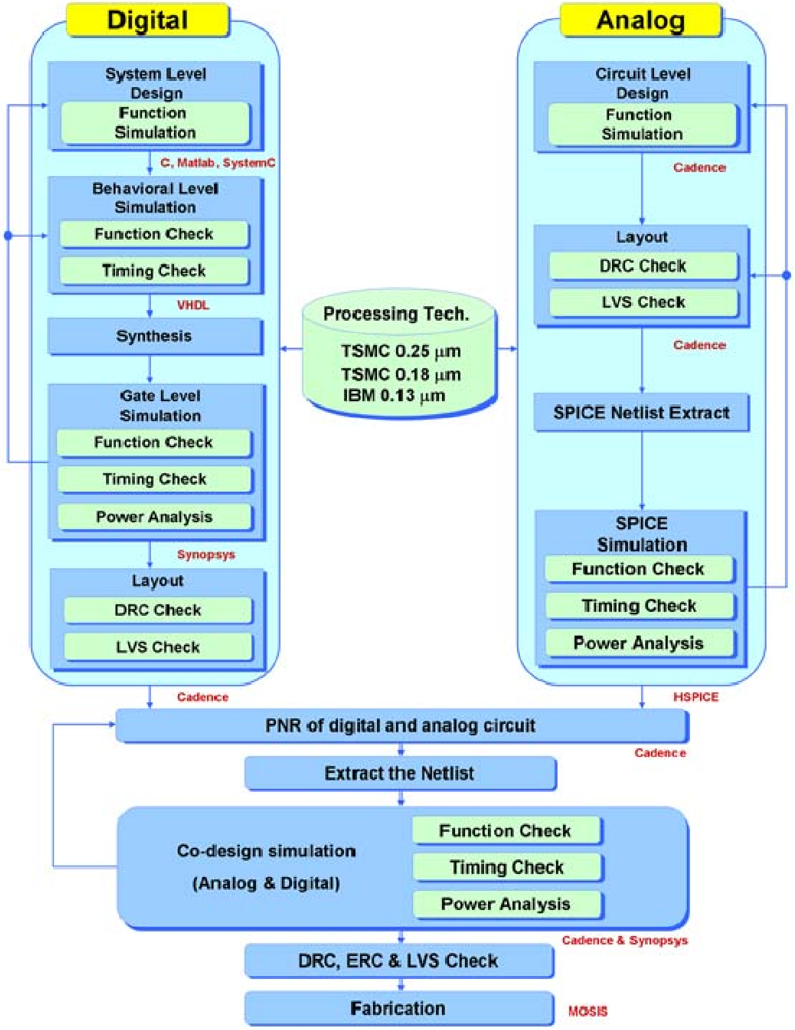
Co-design power distribution/architecture management

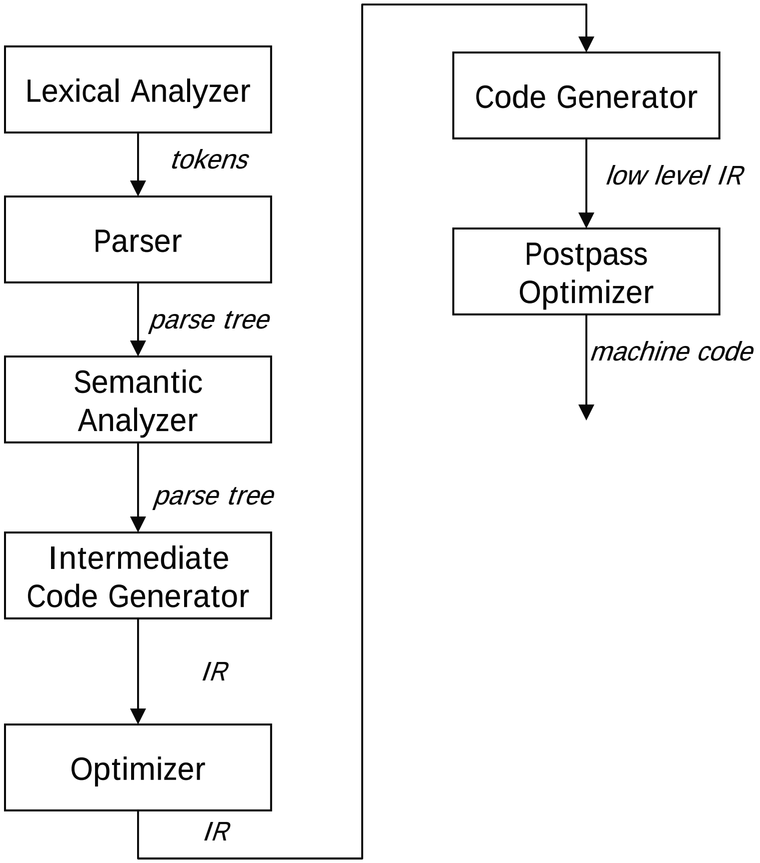
Power distribution network

Major Challenges

- Managing complexity (items 3 and 5)
  - Parallelism, sampling, acceleration
- Multi-model simulations (items 2 and 4)
  - Power and thermal challenges
  - Feedback between thermal and discrete even simulation
- Productivity (items 1 and 5)
  - Cost of simulator construction
  - Ease of use

A Thought: Learn from Design Flows





Lexical Analyzer

tokens

Parser

parse tree

Semantic Analyzer

parse tree

Intermediate Code Generator

IR

Optimizer

IR

Code Generator

low level IR

Postpass Optimizer

machine code

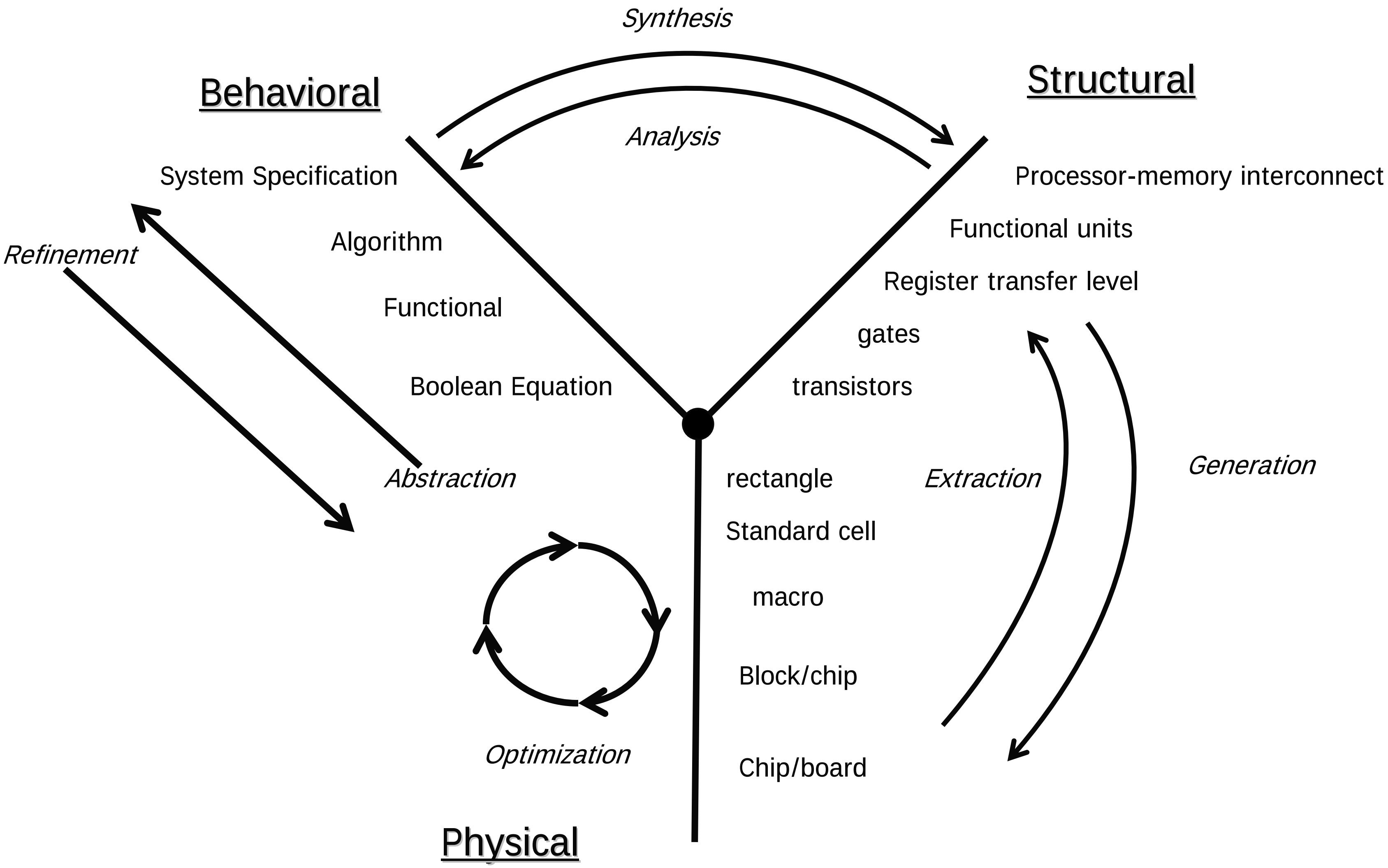
Implications

- Need a hierarchy of representations
  - Accompanied by successive refinement

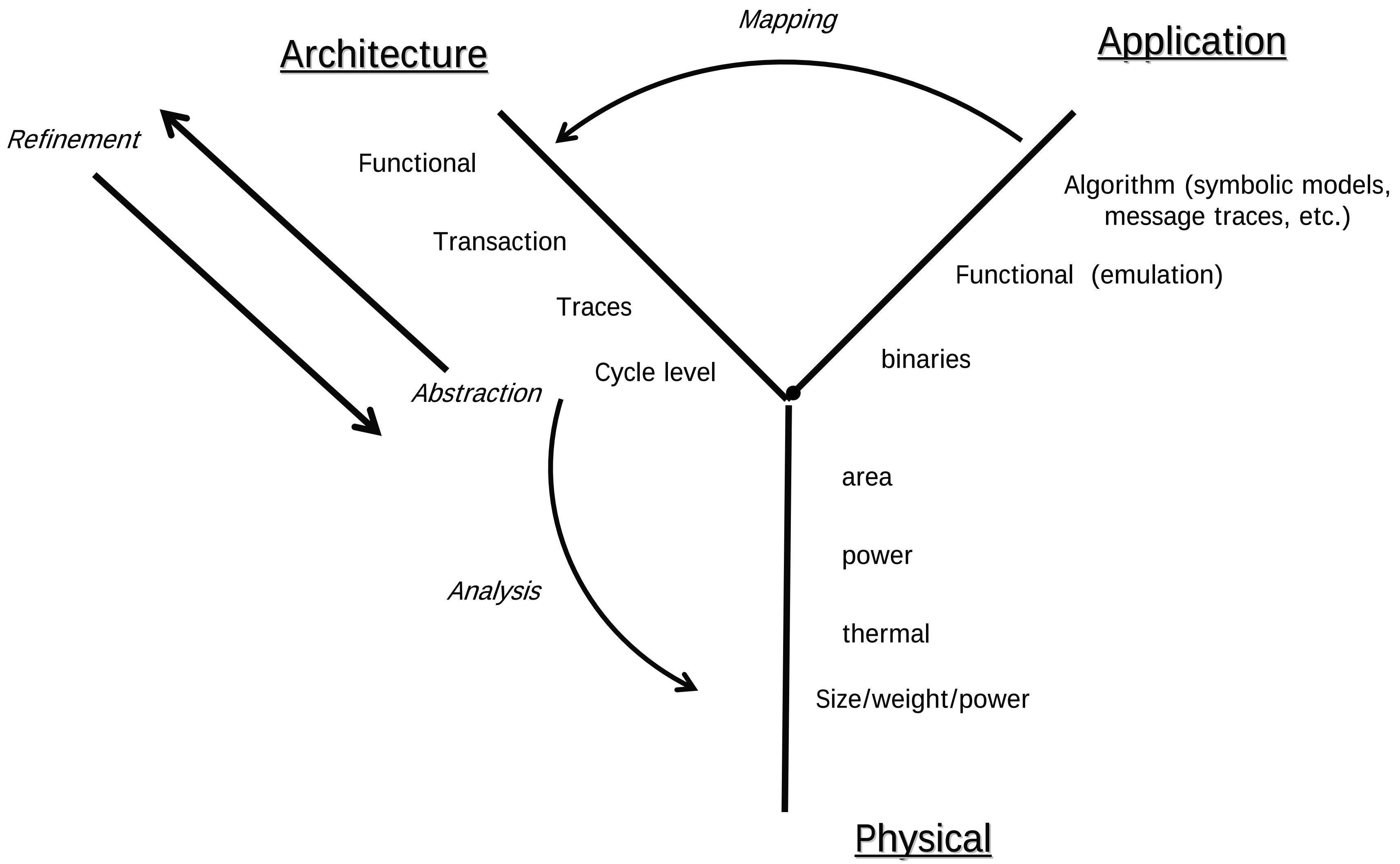
- Some example simulation flow steps
  - Parsing a system description language
  - Component partitioning & assignment,
  - Design rule (model) check

- Need Structure
  - Who is the customer for these tools?

Example Y-Chart Based Design (Gajski, Kuhn)



Example Y-Chart Based Design for Simulation



Mapping

Architecture

Application

Refinement

Functional

Transaction

Traces

Cycle level

Abstraction

Algorithm (symbolic models, message traces, etc.)

Functional (emulation)

binaries

area

power

thermal

Size/weight/power

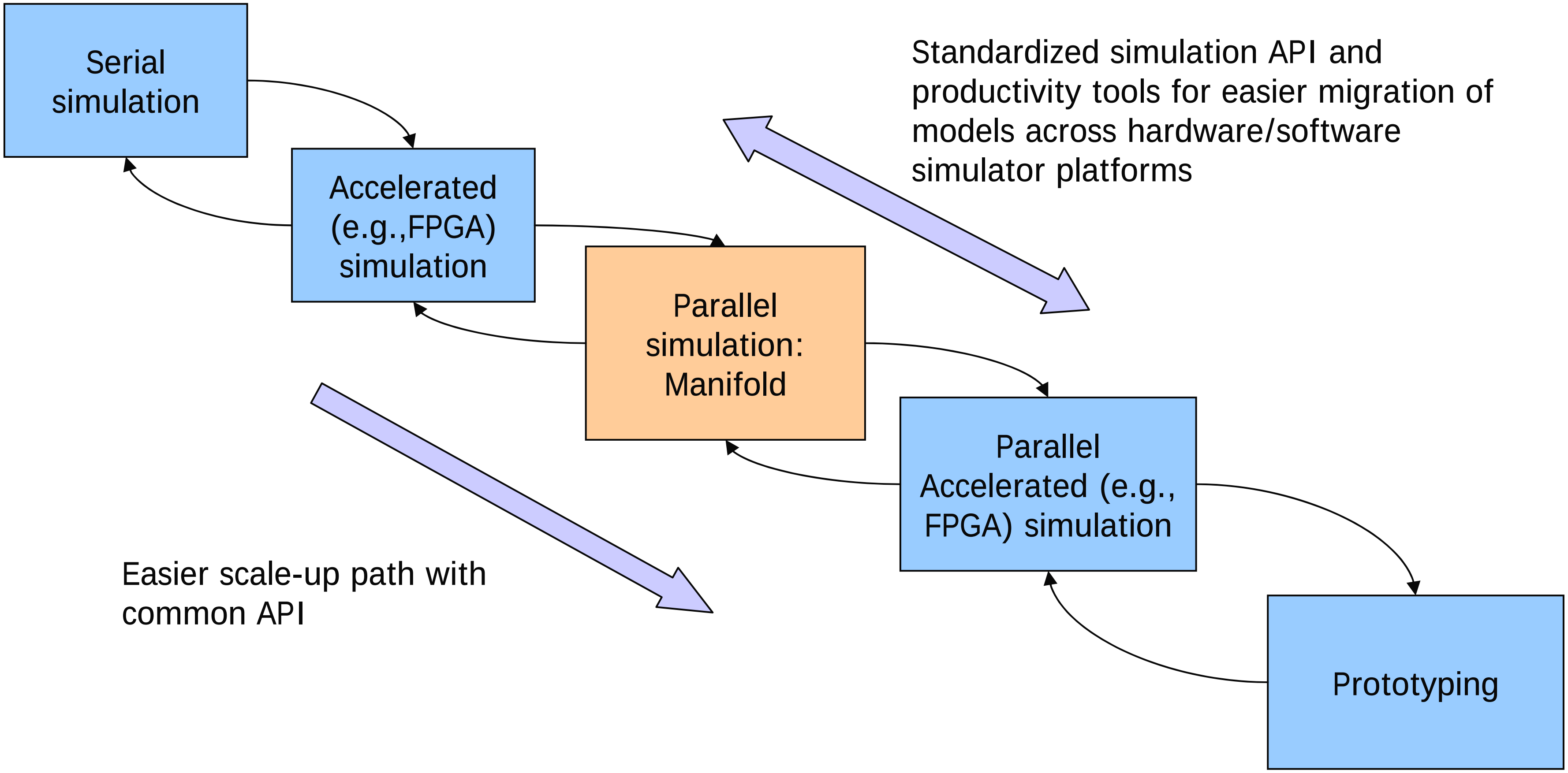
Analysis

Physical

What Can Be Done?

- Create a simulation flow for the construction of models
  - Spiral or waterfall model of construction
  - Use architecture/system description languages
- Are we doomed to build what we can predict?
- Reporting Methodology
  - Challenging!
  - Publication of individual software and a centralized managed code base to support reproducibility

An Ecosystem for Many Core Simulation



Serial simulation

Accelerated (e.g.,FPGA) simulation

Parallel simulation: Manifold

Parallel Accelerated (e.g., FPGA) simulation

Prototyping

Standardized simulation API and productivity tools for easier migration of models across hardware/software simulator platforms

Easier scale-up path with common API

From DoE IAA White Paper (SNL, ORNL, GTech, UMD, UT)

Summary

Coupling/Feedback Across Modeling Tools

Thank You

Productivity Tools

Complexity Management