Analytical Performance Modeling of Hierarchical Interconnect Fabrics

Nikita Nikitin, Javier de San Pedro, Josep Carmona and Jordi Cortadella

Universitat Politècnica de Catalunya

Supported by Intel Corporation

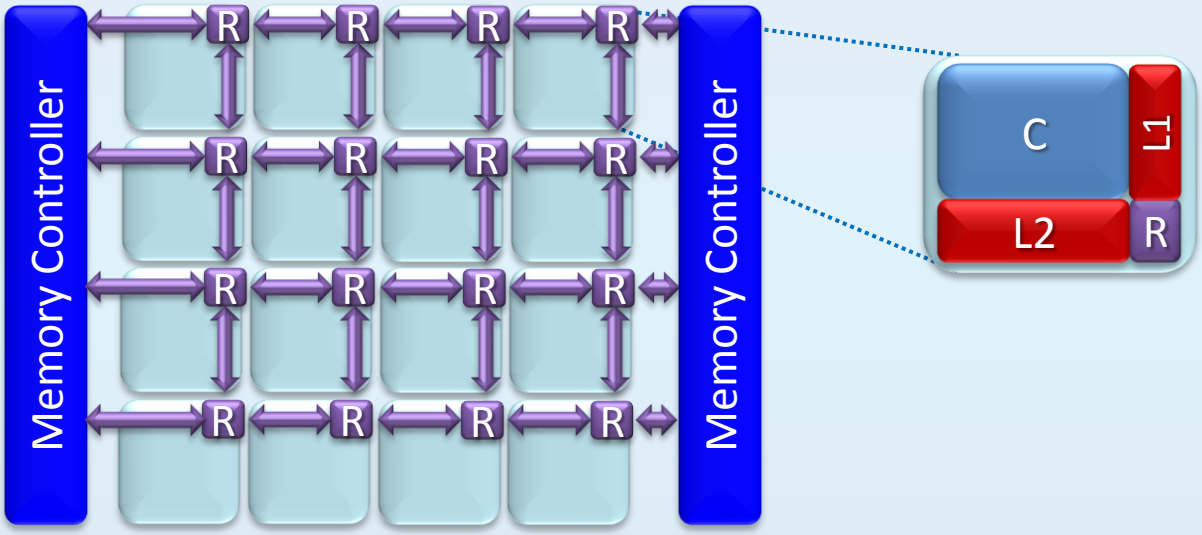
International Symposium on Networks-on-Chip (NOCS) 2012, Copenhagen, Denmark

Outline

- **Introduction**
  - **Hierarchical Chip Multiprocessors (CMPs)**
  - **Performance modeling for CMPs**
  - **The cyclic dependency between latency and traffic**
- Analytical performance modeling
  - Modeling traffic
  - Modeling latency
  - Methods to resolve the dependency
- Results and conclusions

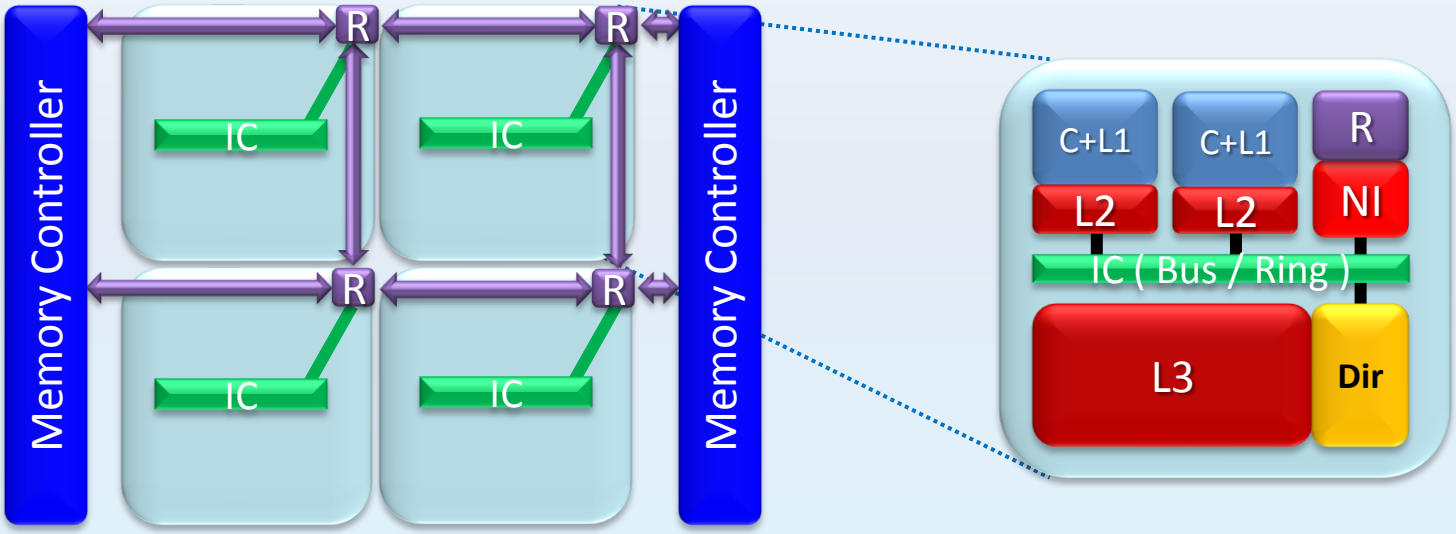
The trends in CMP design

- Hundreds of computing units per chip
  - Smaller, simpler, more power-efficient cores
- Advanced memory management
  - Larger on-chip cache
  - Increasing interconnect (IC) bandwidth
- Tiled architecture



Hierarchical interconnects

- Exploit locality of memory references*

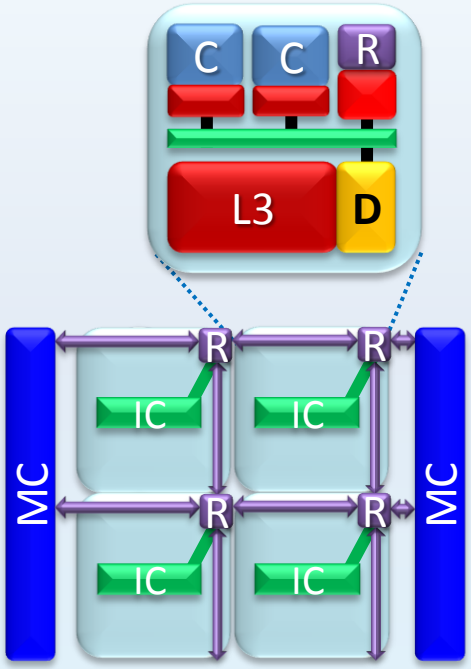


Tiled CMP with hierarchical interconnect

** "Design and Evaluation of a Hierarchical On-Chip Interconnect for Next-Generation CMPs", R.Das et al., HPCA, 2009*

Design of CMP architecture

- Goal: efficient use of chip resources
  - Maximize performance
  - Fit area/power/thermal budget



- Multidimensional exploration space

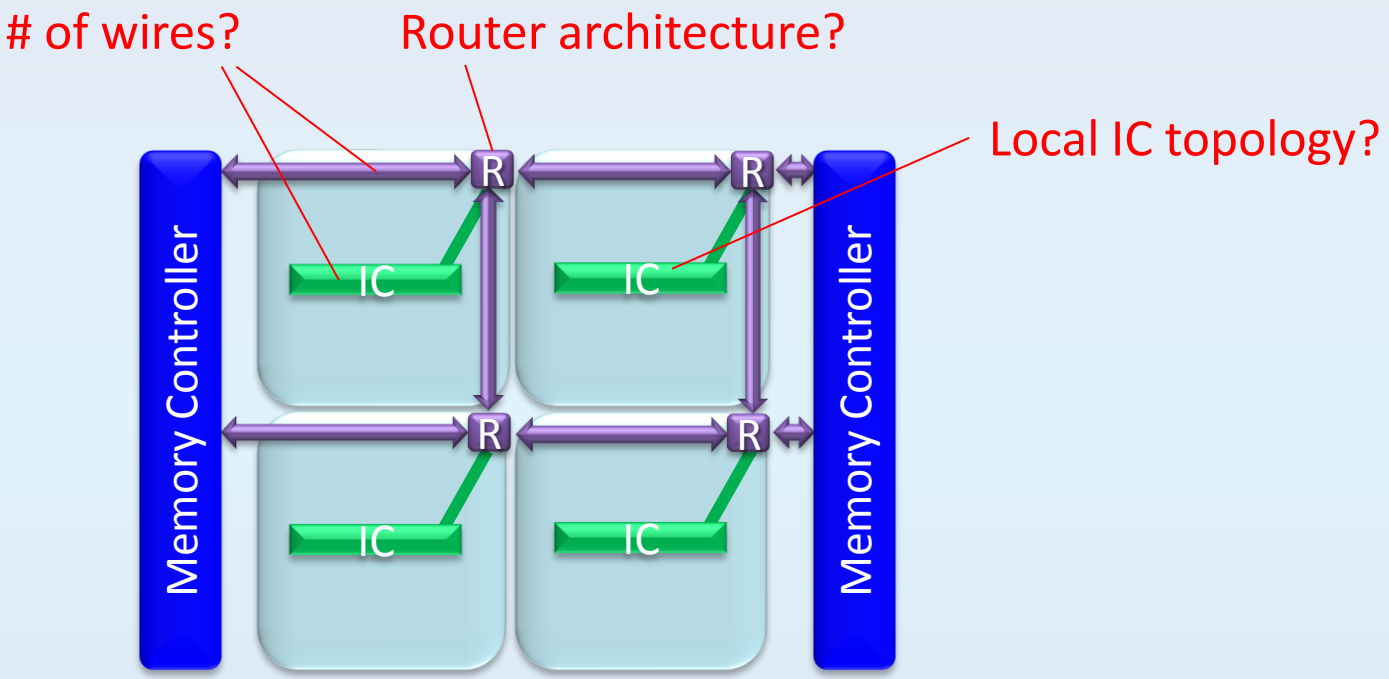
  (#cores / cache size /

  memory hierarchy / IC topologies /…)

- Means: automated design space exploration
  - Analytical performance models are essential

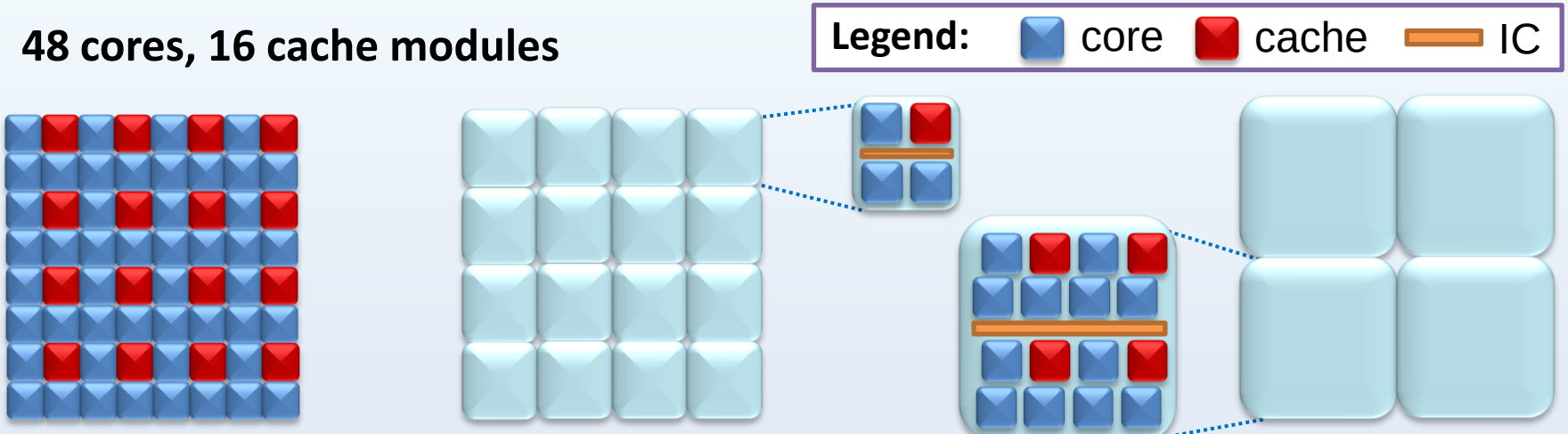
Contention modeling

- Contention impacts CMP performance
- Crucial evaluating hierarchical interconnects
  - Is the required bandwidth sustainable?



Motivational example

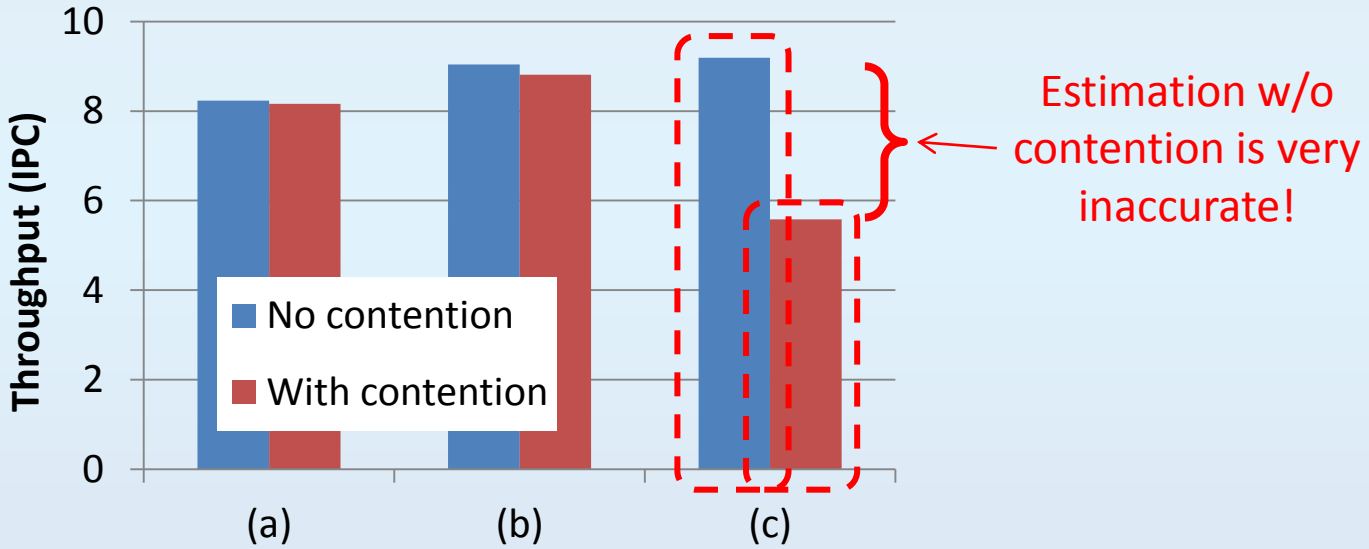
48 cores, 16 cache modules



(a) 8x8 mesh

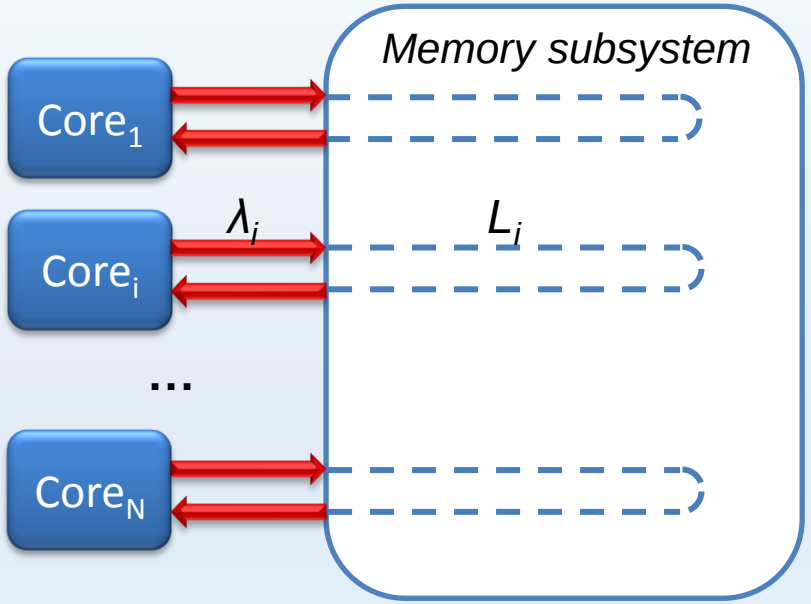
(b) 4x4 mesh with bus clusters

(c) 2x2 mesh with bus clusters



Analytical modeling of CMP performance

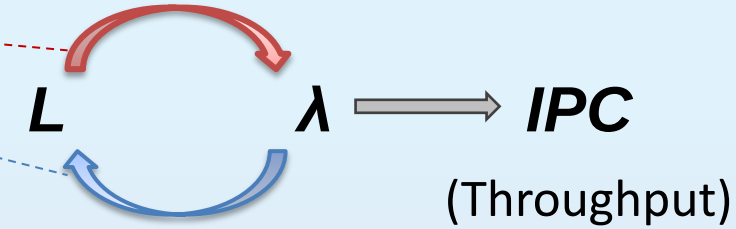
- Analytical models for ICs:
  - *Latency* ***L*** as a function of *traffic* ***λ***
  - λ defined by the workload



Emphasis: λ depends on L!

$\lambda\uparrow \rightarrow L\uparrow \rightarrow \lambda\downarrow \rightarrow L\downarrow \rightarrow \lambda\uparrow \rightarrow \bullet\bullet\bullet$

- This work: resolve the cyclic dependency of traffic and latency
  - Formulate λ as a function of L
  - Add existing model for $L(\lambda)$
  - Resolve the system efficiently



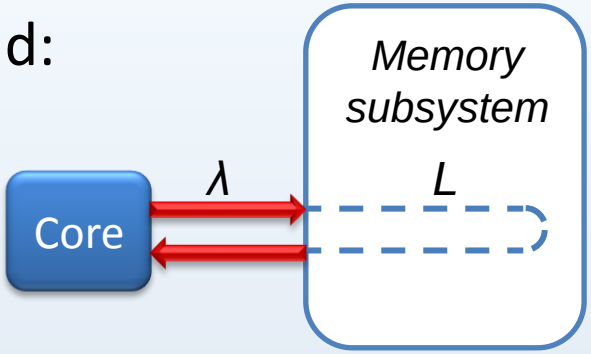
Outline

- Introduction
  - Hierarchical Chip Multiprocessors (CMPs)
  - Performance modeling for CMPs
  - The cyclic dependency between latency and traffic
- **Analytical performance modeling**
  - **Modeling traffic**
  - **Modeling latency**
  - **Methods to resolve the dependency**
- Results and conclusions

Modeling memory traffic

Parameters of core executing some workload:

1. CPI_0 - *ideal* Cycles Per Instruction
2. MPI - # Memory references Per Instruction



Real performance of in-order core:

$$CPI = CPI_0 + \underbrace{AccessRate \cdot AccessCost}_{\text{Memory access penalty}} = CPI_0 + MPI \cdot \underbrace{L}_{\text{Average latency of memory access}}$$

Traffic to memory (probability of a memory reference per cycle):

$$\lambda = MPI \cdot IPC = MPI \cdot \frac{1}{CPI} = \frac{MPI}{CPI_0 + MPI \cdot L}$$

Modeling average memory latency

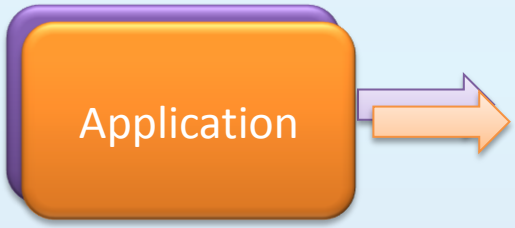
- Average latency of memory requests for a core:

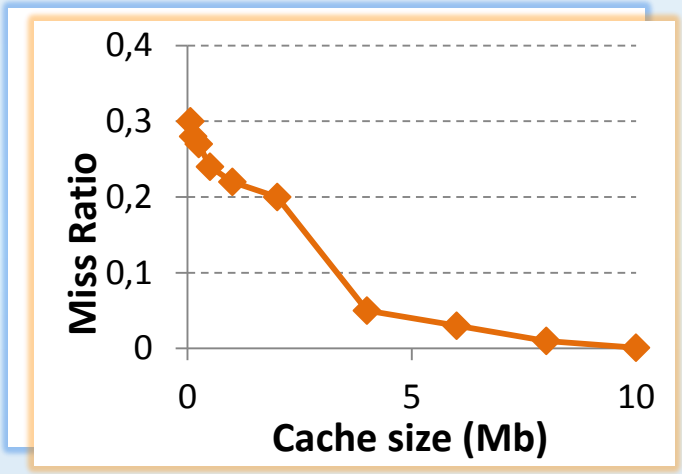
$$L = L_{hop\text{-}count} + L_{contention}$$

$$L_{hop\text{-}count} = p_{L1} \cdot L_{L1} + p_{L2} \cdot L_{L2} + p_{L3} \cdot L_{L3} + p_{MC} \cdot L_{MC}$$

Latencies are calculated using
- Cache latencies
- Interconnect topology
- Routing algorithm (XY)

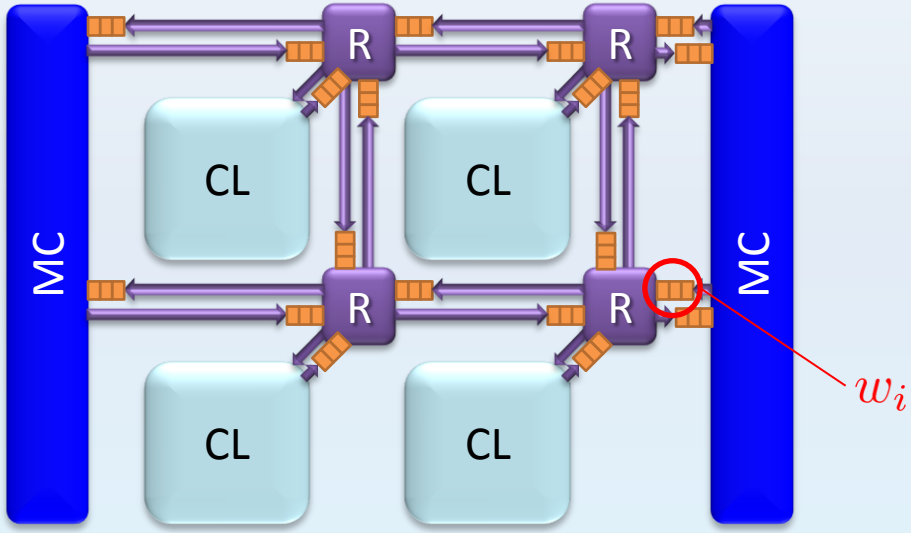
Probabilities are calculated using
- Miss ratio dependency on cache size



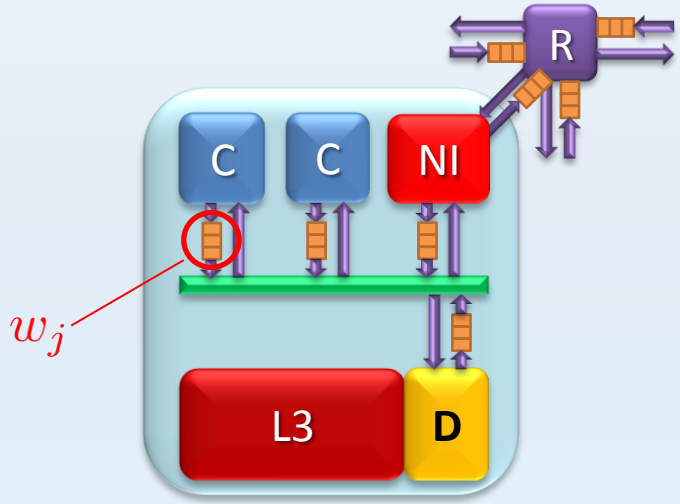


Modeling contention latency

"An Analytical Approach for Network-on-Chip Performance Analysis", Ogras et al., TCAD, 2010 (Best Paper Award)



Mesh NoC



Bus-based cluster

Delays in queues are defined by extending M/G/1 queuing model:

$$\bar{w}_q = W(\lambda_1, .., \lambda_N) \quad \Rightarrow \quad L_c = L(\lambda_1, .., \lambda_N)$$

The cyclic dependency of *L* and *λ*

System of non-linear equations

$$\forall c = 1, .., N: \quad \begin{cases} L_c = L(\lambda_1, .., \lambda_N) \\ \lambda_c = \lambda(L_c) \end{cases}$$

Analytical model for latency

$$\dots$$

$$\tau_j = TN_j + T \sum_{k=1, k \neq j}^{P} c_{jk} N_k + R$$

$$f_{ij} = \frac{\lambda_{ij}}{\sum_{k=1}^{P} \lambda_{ik}} \qquad c_{ij} = \sum_{k=1}^{P} f_{ik} f_{jk}$$

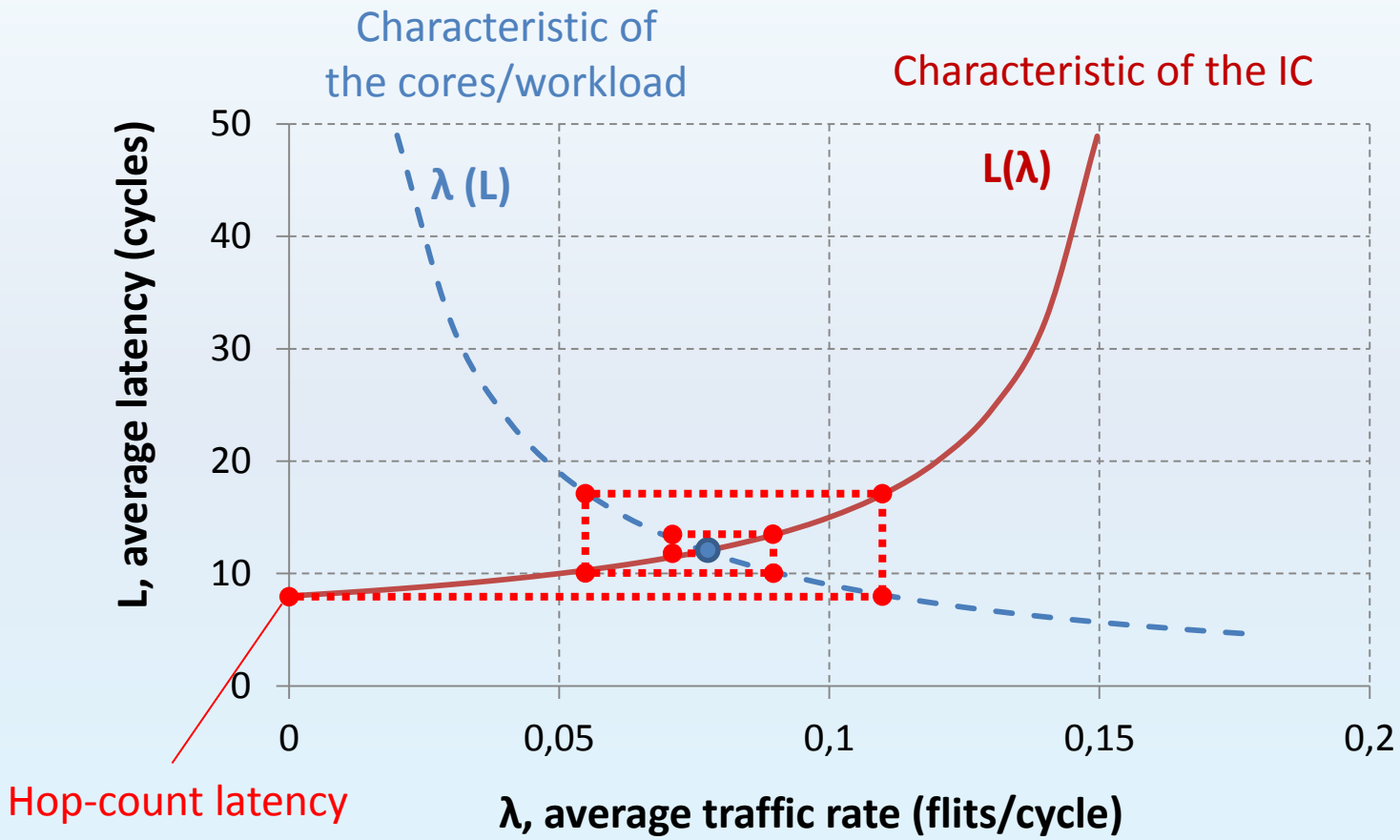
$$\dots$$

$$\lambda = \frac{MPI}{CPI_0 + MPI \cdot L}$$

- Solve using numerical methods
- General methods are very slow
  - 10x10 mesh (**10K vars./eqns.**) – MATLAB timeout after few hours
- Proposed methods:
  - Fixed-point iteration
  - Bisection search for λ

Any "black-box" model for *L(λ)*!

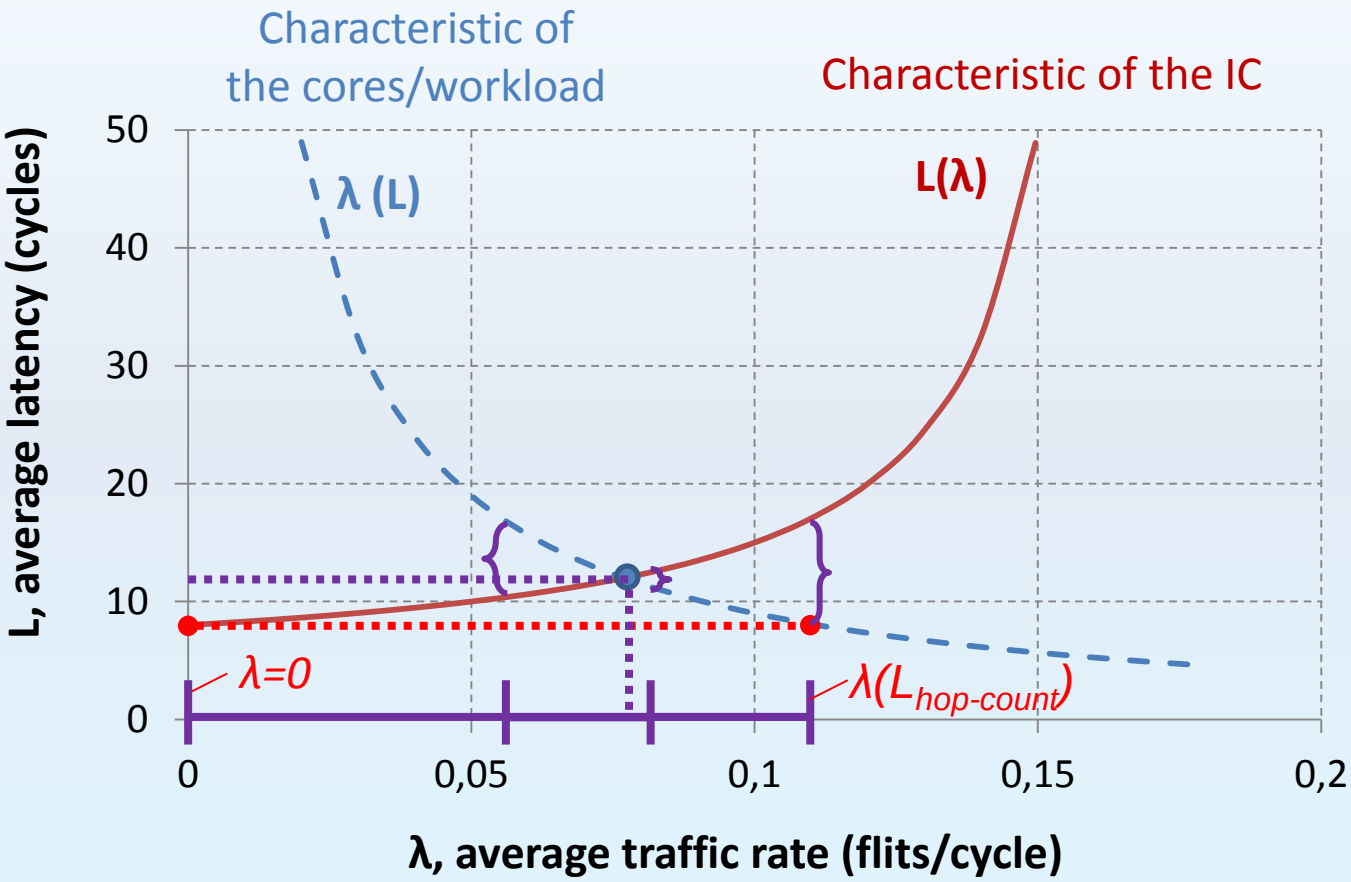
Fixed-point iteration



+ Fast (10x10 mesh in several ms)
+ Converges to the exact solution

– May not converge for high λ

Bisection search for λ



- Fast, as fixed-point
- Always converges to an approximate solution (good for homogeneous clusters)

Outline

- Introduction
  - Hierarchical Chip Multiprocessors (CMPs)
  - Performance modeling for CMPs
  - The cyclic dependency between latency and traffic
- Analytical performance modeling
  - Modeling traffic
  - Modeling latency
  - Methods to resolve the dependency
- **Results and conclusions**

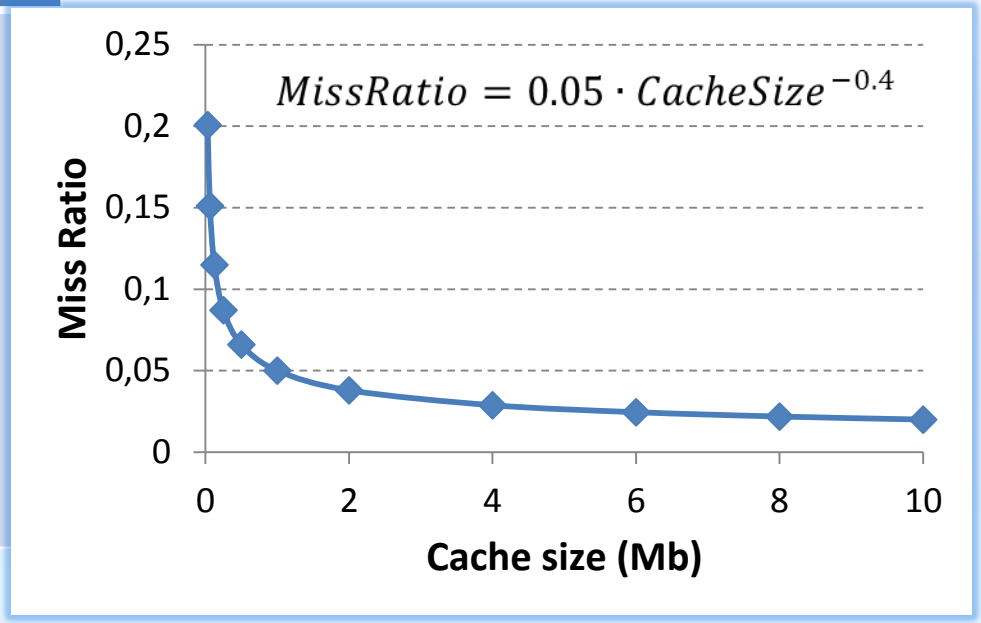
Performance of analytical methods

Test	Mesh	Cont. lat.	Num. of var./eqn.	Runtime (sec)		
				MATLAB	Fixed-Point	Bisection
T1	2 x 2	5%	236	0.023	0.001	0.001
T2	4 x 4	13%	1224	1.412	0.001	0.002
T3	6 x 6	8%	3108	30.831	0.002	0.003
T4	8 x 8	12%	6128	408.539	0.006	0.010
T5	10 x 10	23%	10260	Timeout (1hr)	0.010	0.012
T6	10 x 10	46%	10260	Timeout (1hr)	0.022	0.015
T7	10 x 10	55%	10260	Timeout (1hr)	NA	0.016

Case study: performance exploration

Parameter	Value
Chip area	350 mm^2
Core area	1.25 mm^2
Core IPC_0	2.0
MPI	0.5
L1 size	64, 128 Kb
L2 size	64 Kb to 3 Mb
Memory density	1 mm^2 / Mb
Mesh dimensions	2x2 to 16x16
MC latency	100 cycles

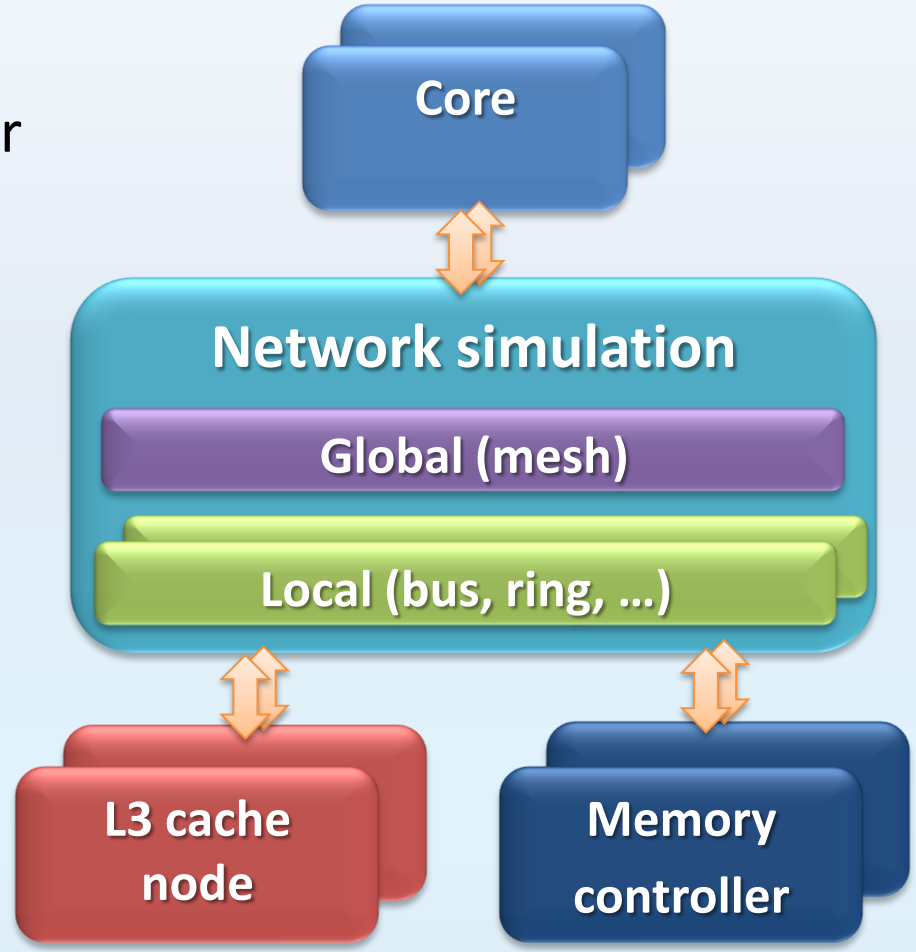
1062 configurations explored



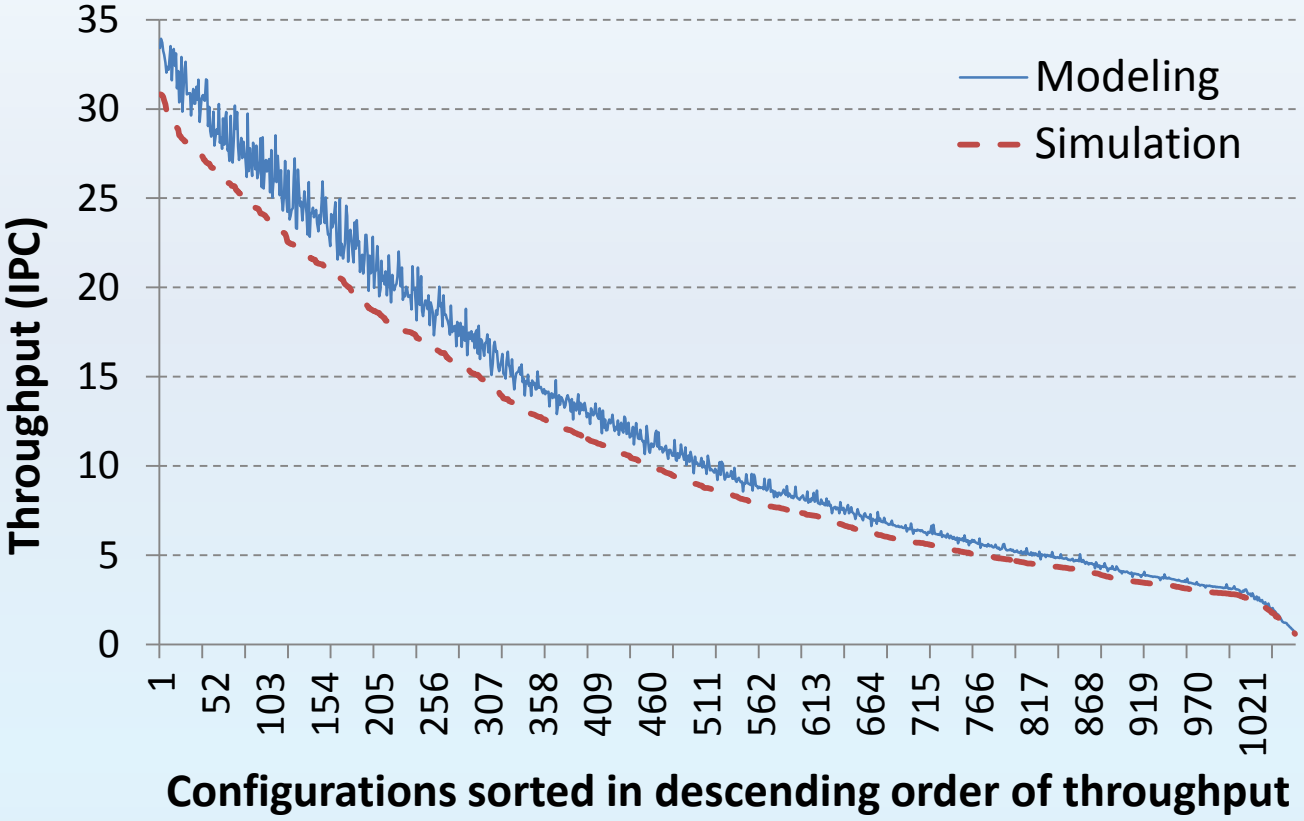
Cache Size	64K	128K	256K	512K	1M	2M	4M	8M
Area* (mm^2)	0.063	0.125	0.25	0.5	1.0	2.0	4.0	8.0
Latency (cycles)	2	3	4	5	6	7	8	9

Simulation environment

- Verify model by simulation
- Cycle-accurate NoC simulator
  - On top of BookSim 2.0
- Extensions
  - Hierarchical networks
  - Bus topologies
  - Probabilistic state-machines for cores and memories

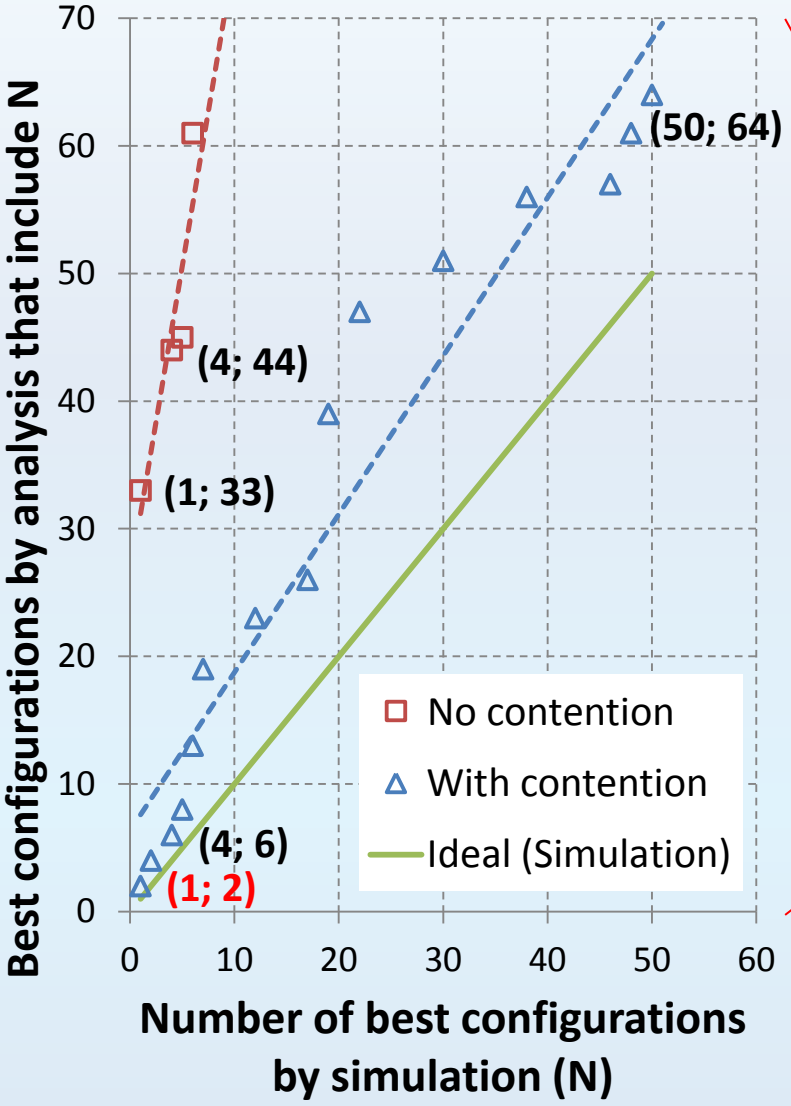


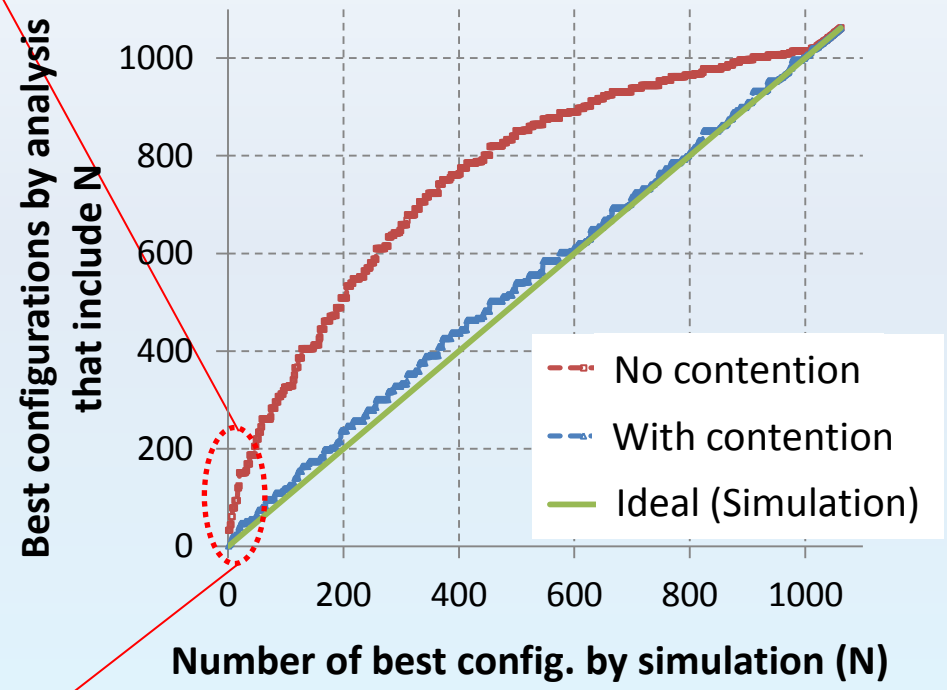
Faithfulness of the model



- Average difference in throughput is about 10%
- Corresponds to the error of the latency model

Best-throughput ordering





Simulation time:	5.5 hours
Modeling time:	16.8 sec (>1000x faster)

Conclusions

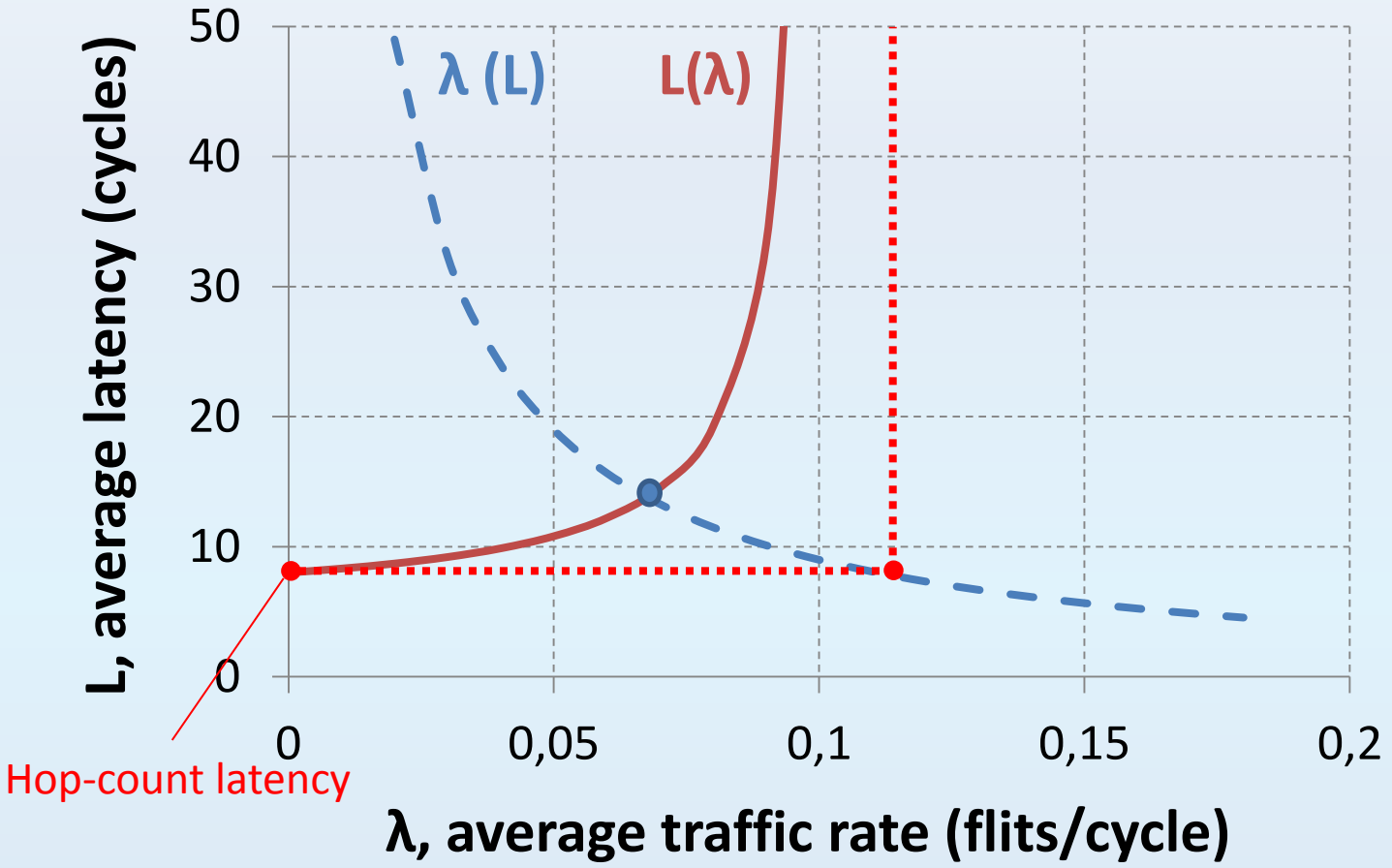
- Analytical modeling of contention in CMPs is essential
- *There exists cyclic dependency between latency and traffic of memory requests*
- This dependency can be efficiently resolved using numerical methods (fixed-point, bisection)
- Precision of the model is significantly improved

- Current work: out-of-order cores, heterogeneity

Backup

Fixed-point convergence issues

Sufficient for convergence of $\bar{x}_{n+1} = F(\bar{x}_n)$: $\sum_i \left| \frac{\partial F}{\partial x_i} \right| < 1.$



Bisection search

Latency model

$$L_c = L(\lambda_1, .., \lambda_N)$$

Traffic model

$$\lambda_c = \frac{MPI}{CPI_0 + MPI \cdot L_c}$$

$$L_c^*(\lambda_c) = \frac{1}{\lambda_c} - \frac{1}{MPI \cdot IPC_0}$$

$$L(\bar{\lambda}) = \frac{1}{N}\sum_{c=1}^{N} L_c(\bar{\lambda}), \quad L^*(\bar{\lambda}) = \frac{1}{N}\sum_{c=1}^{N} L_c^*(\lambda_c)$$

$$F(\bar{\lambda}) = L(\bar{\lambda}) - L^*(\bar{\lambda})$$

Average latency calculation

- Average Memory Access Time (AMAT):

$$L_{hop\text{-}count} = p_{L1} \cdot L_{L1} + p_{L2} \cdot L_{L2} + p_{L3} \cdot L_{L3} + p_{MC} \cdot L_{MC}$$

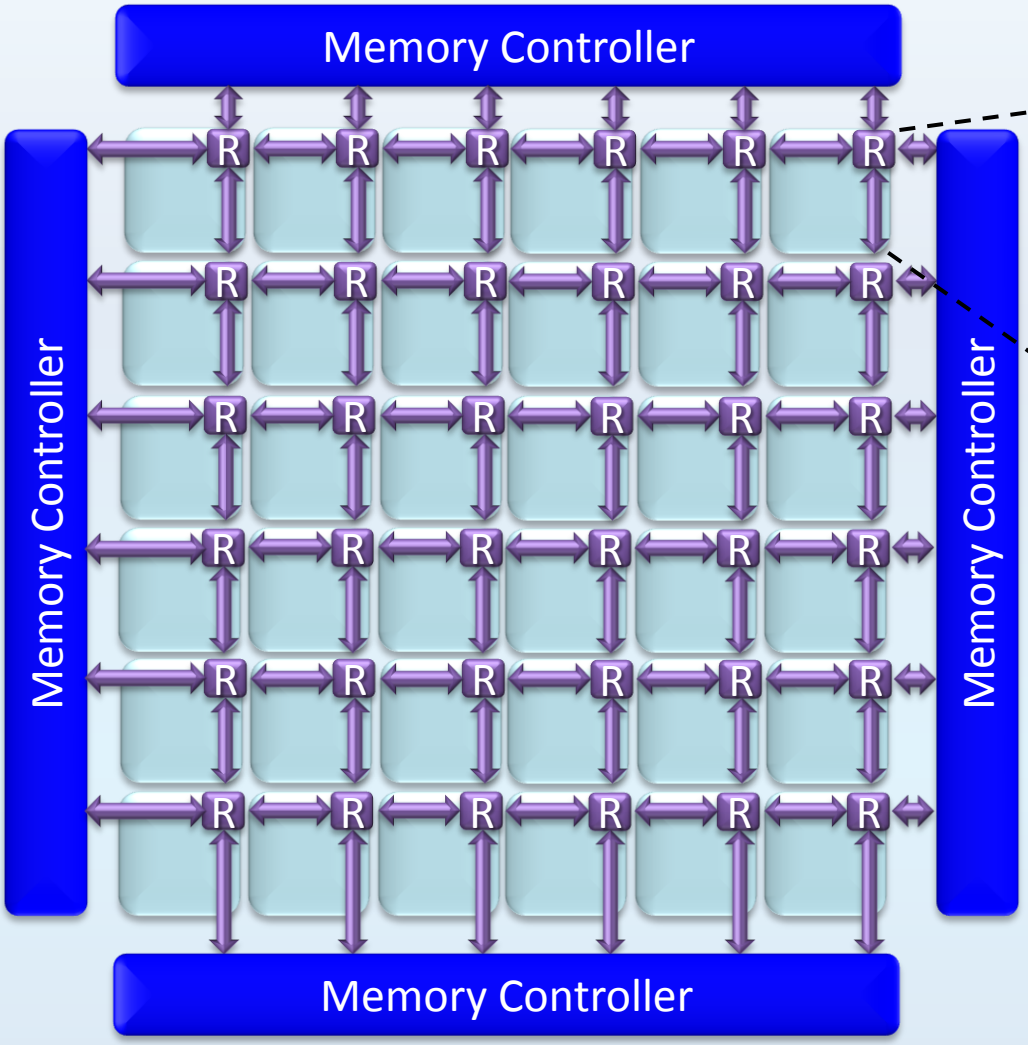
$$p_{L1} = Hit(L1)$$

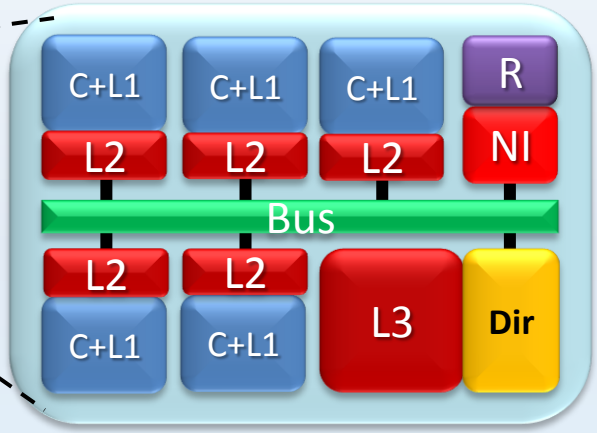
$$p_{L2} = Miss(L1) \cdot Hit(L2)$$

$$p_{L3} = Miss(L1) \cdot Miss(L2) \cdot Hit(L3)$$

$$p_{MC} = Miss(L1) \cdot Miss(L2) \cdot Miss(L3) \cdot Hit(MC)$$

Best configuration





- 6x6 mesh, 36 clusters, 5 cores/cluster
- total 180 cores with 64K L1, 256K L2
- 68Mb total shared L3

Throughput = 30.81 IPC

Runtime: Modeling vs Simulation