Optimized Core-links for Low-latency NoCs

Ryuta Kawano[†], Seiichi Tade[†], Ikki Fujiwara[††],
Hiroki Matsutani[†], Hideharu Amano[†], Michihiro Koibuchi[††]

[†] Keio University

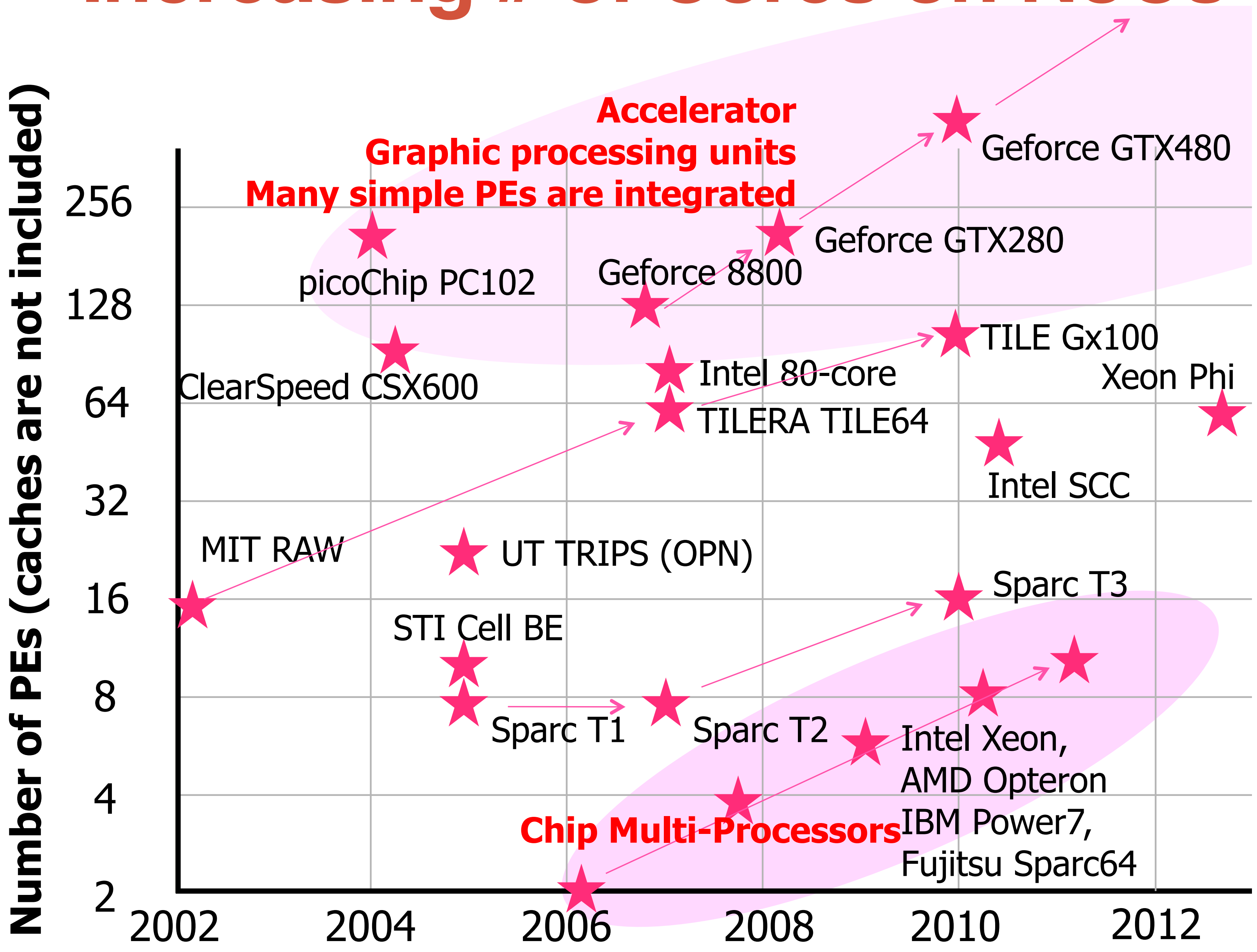
[††] National Institute of Informatics

blackbus@am.ics.keio.ac.jp

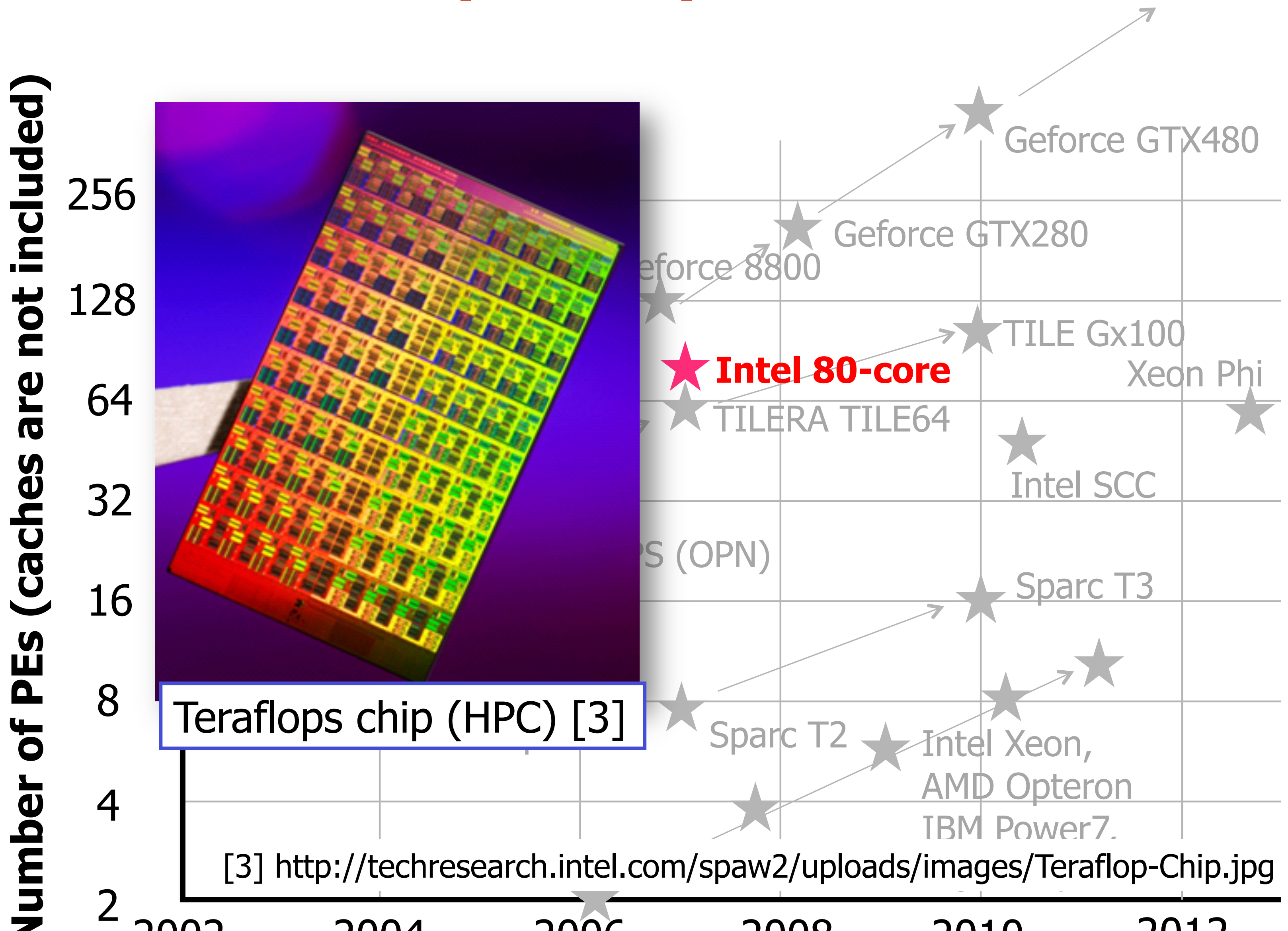
Contents

- Conventional NoCs
- Small-world Networks
  - Difficulty in applying on Chips
- How do we reduce path hops of NoCs?
  - Adding multiple links between a core and routers
  - Optimization method for picking core-links
- Evaluations
  - Zero-load latencies
  - Costs
  - Full-system simulation
- Conclusions

Increasing # of Cores on NoCs



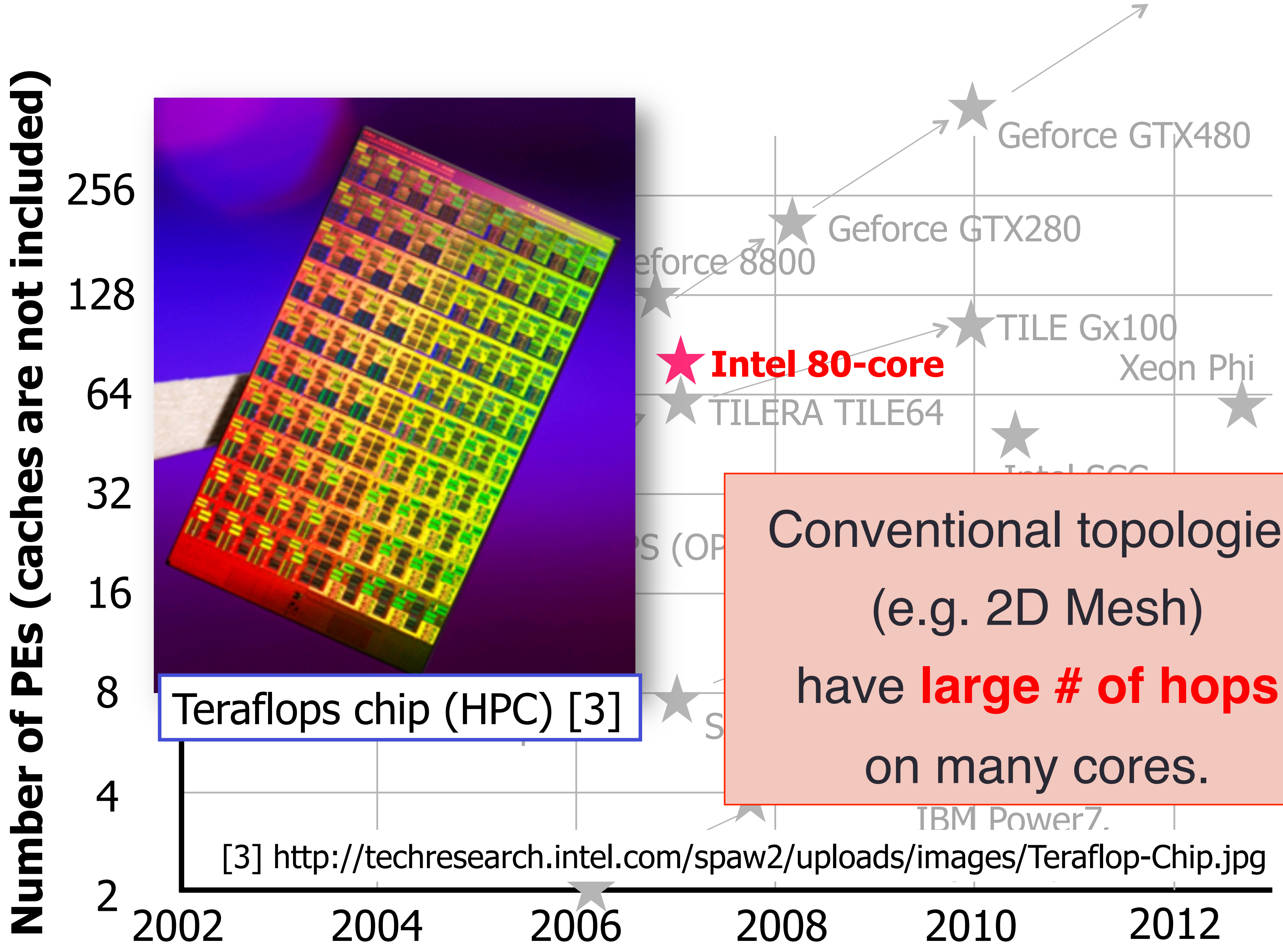
Intel Teraflops Chip: 80-tile 2D Mesh

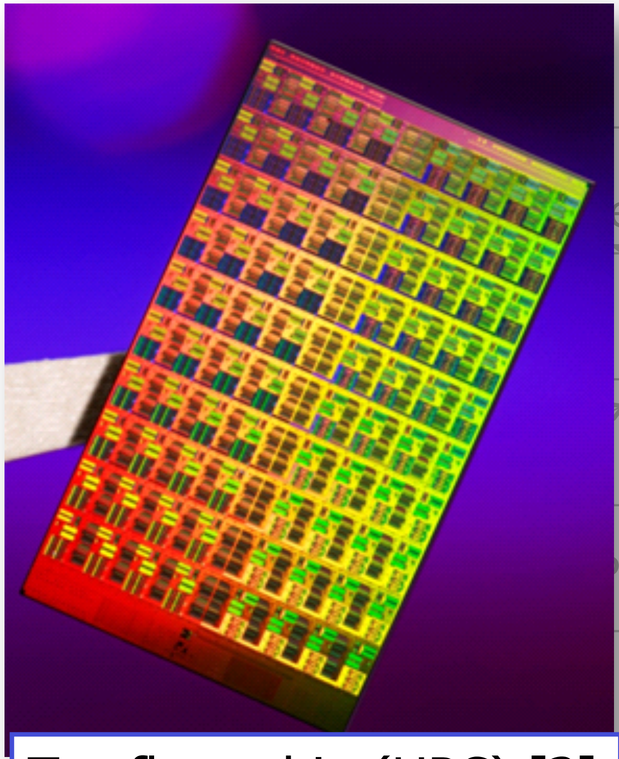


Teraflops chip (HPC) [3]

[3] http://techresearch.intel.com/spaw2/uploads/images/Teraflop-Chip.jpg

Intel Teraflops Chip: 80-tile 2D Mesh





Teraflops chip (HPC) [3]

Conventional topologies
(e.g. 2D Mesh)
have **large # of hops**
on many cores.

[3] http://techresearch.intel.com/spaw2/uploads/images/Teraflop-Chip.jpg

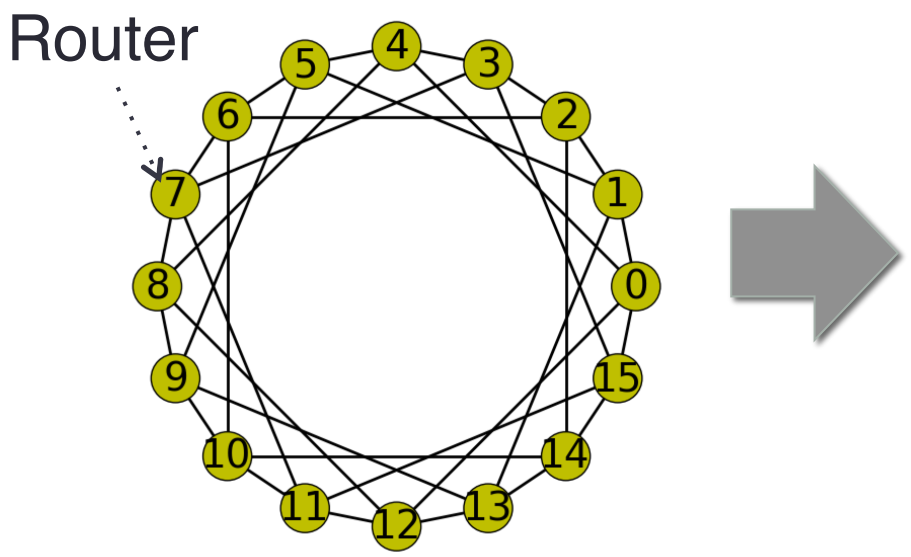
Contents

- Conventional NoCs
- <u>Small-world Networks</u>
  - Difficulty in applying on Chips
- How do we reduce path hops of NoCs?
  - Adding multiple links between a core and routers
  - Optimization method for picking core-links
- Evaluations
  - Zero-load latencies
  - Costs
  - Full-system simulation
- Conclusions

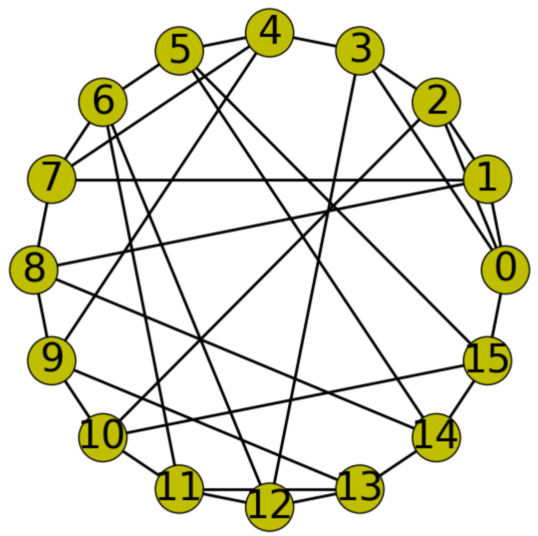
Small-world Topology for off-Chip Network

Reduction of # of hops using ***small-world effects***

[Koibuchi et al. 2012]

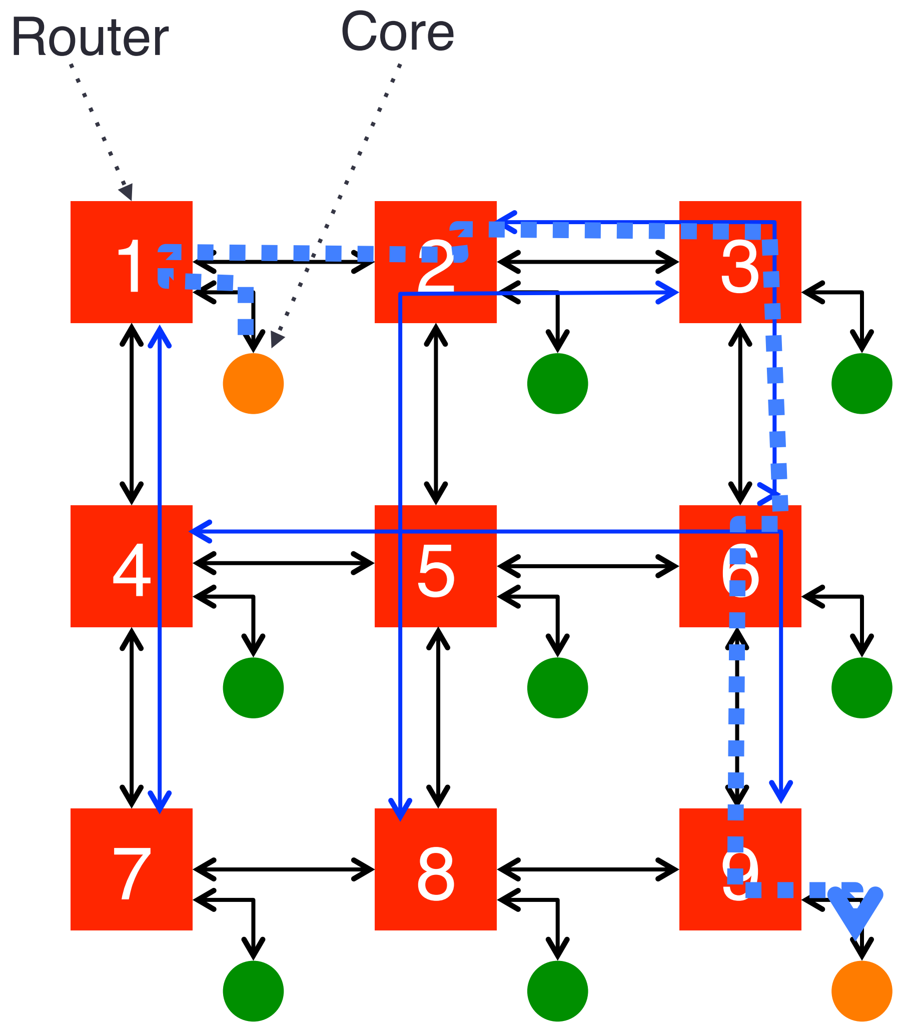


Ring + Non-Random Links



Ring + Random Links

Small-world Topology on Chips



2D Mesh

+ Inter-router additional links

[Ogras et al. 2006]

- Need to use **custom routing**
  - to reduce **path hops**
  - to avoid **deadlocks**

Contents

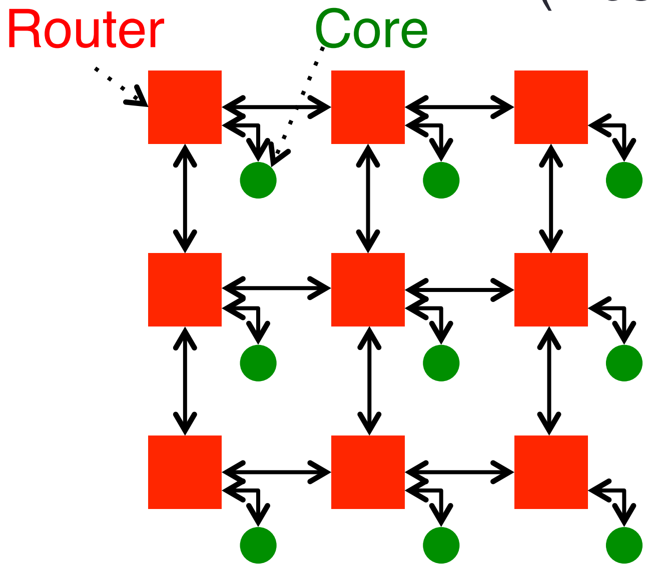
- Conventional NoCs
- Small-world Networks
  - Difficulty in applying on Chips
- <u>How do we reduce path hops of NoCs?</u>
  - Adding multiple links between a core and routers
  - Optimization method for picking core-links
- Evaluations
  - Zero-load latencies
  - Costs
  - Full-system simulation
- Conclusions

Multiple Core-links to Reduce Path Hops

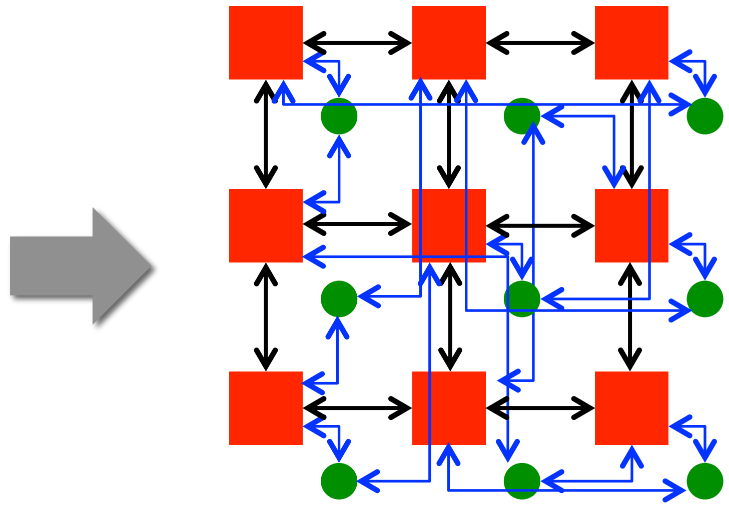
Idea: Router topology

+ **multiple links** between a core and multiple routers

(Mesh router topology)

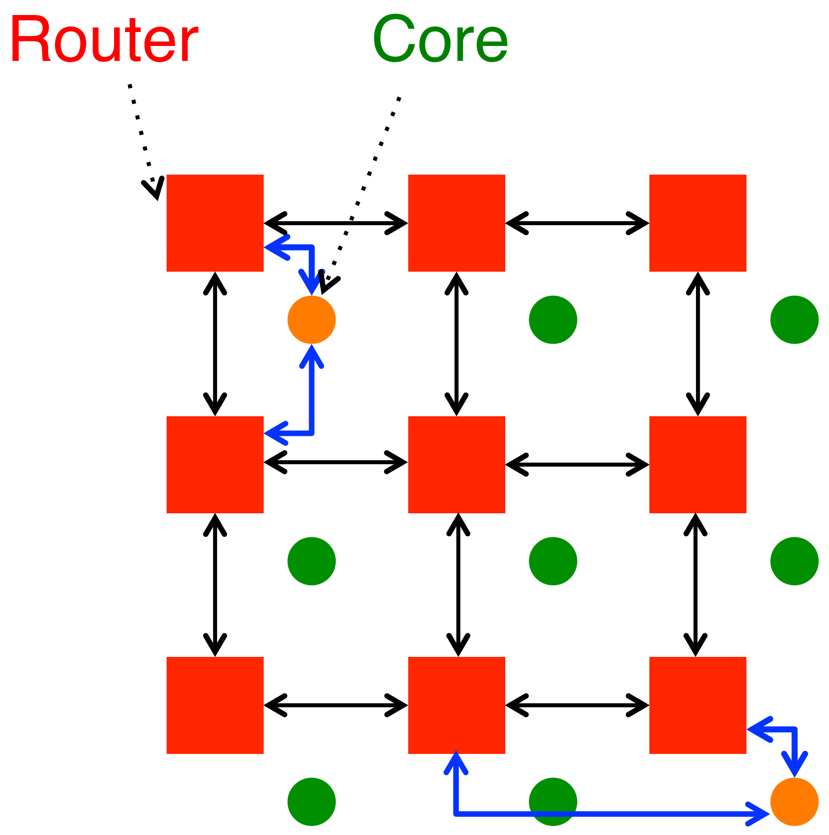


Conventional NoC



Multiple Core-link NoC

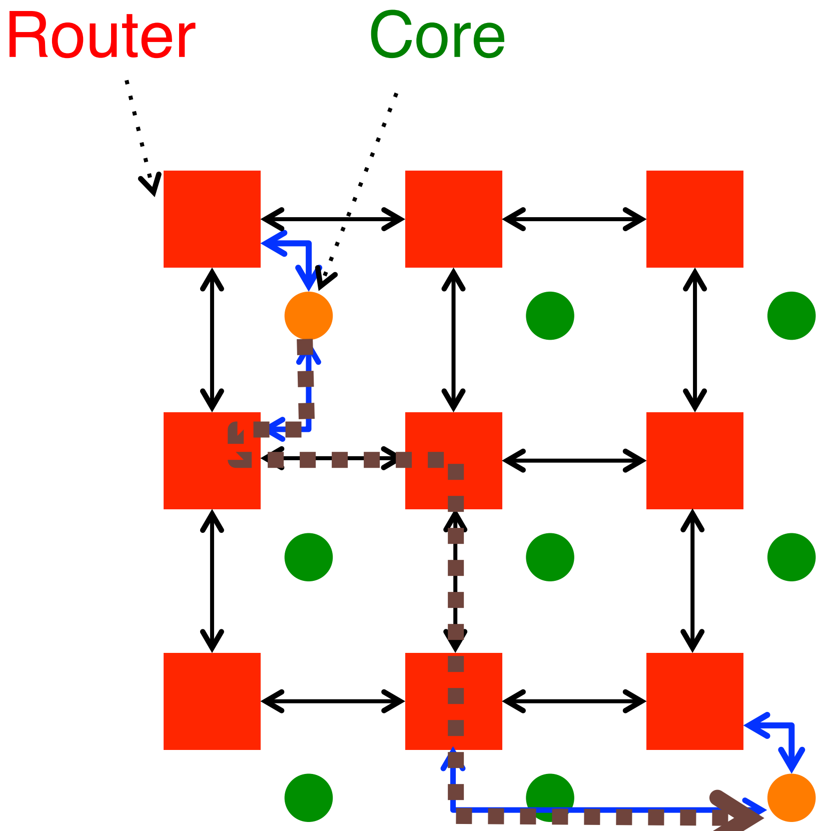
Our Idea: Reduction of First and Last 1-hop Latencies with Shortcut Core-links



Multiple Core-links

- Using shortest path between source and destination cores
  - Achieving lower hops by ***small-world effects***
  - Maintaining **regularities** of router topologies

Our Idea: Reduction of First and Last 1-hop Latencies with Shortcut Core-links



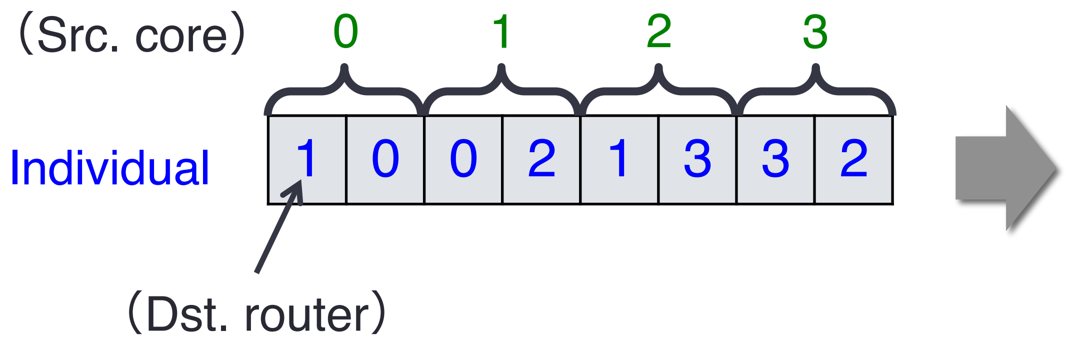
Multiple Core-links

- Using shortest path between source and destination cores
    - Achieving lower hops by ***small-world effects***
    - Maintaining **regularities** of router topologies

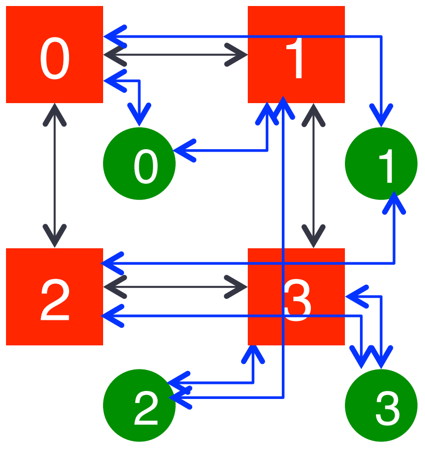
Optimization Method for Picking Core-links

- Problem: Lower operating frequency by **longer core-links**
- Solution: Optimization using **GA (Genetic Algorithm)**

Definition of Individual



Example of Individual



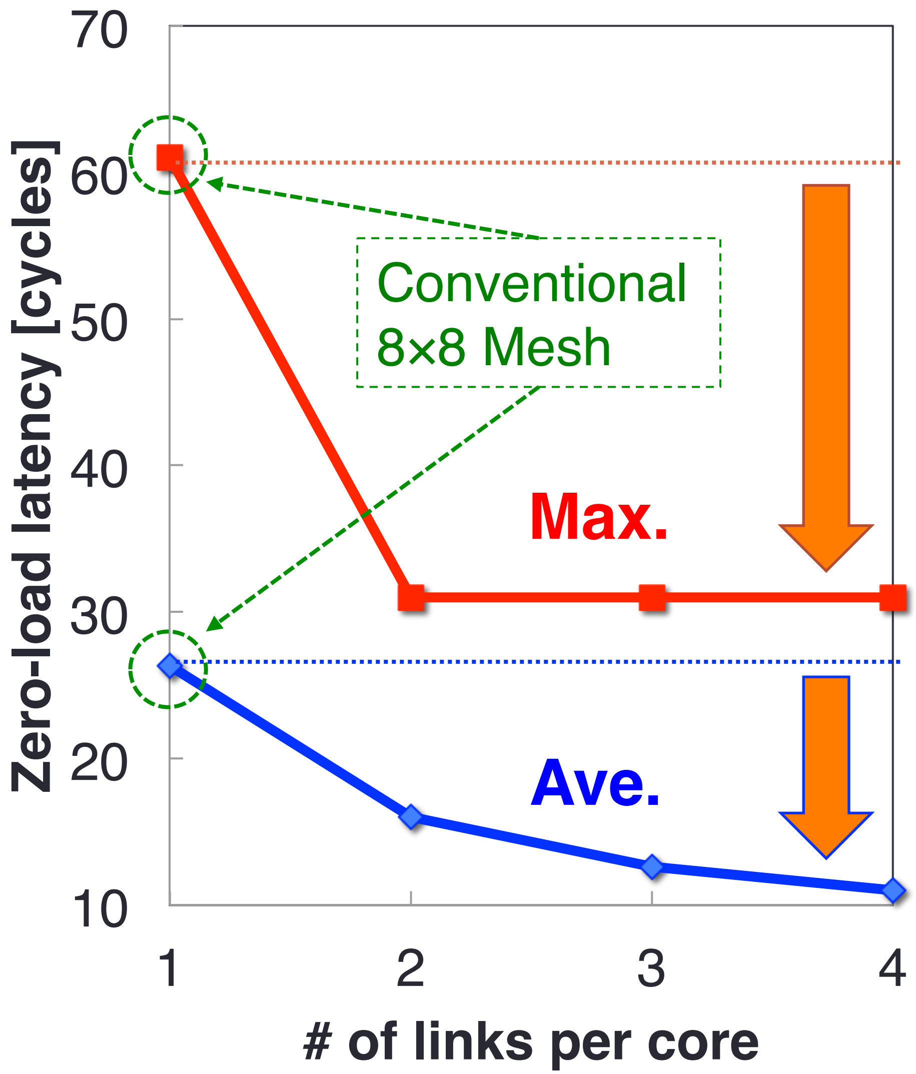
Corresponding Topology

Providing **best tradeoff** between **link length** and **# of hops**

Contents

- Conventional NoCs
- Small-world Networks
  - Difficulty in applying on Chips
- How do we reduce path hops of NoCs?
  - Adding multiple links between a core and routers
  - Optimization method for picking core-links
- <u>Evaluations</u>
  - Zero-load latencies
  - Costs
  - Full-system simulation
- Conclusions

Zero-load Latency



- 8×8 Mesh router topology + optimized core-links
  - Max. core-link length: 4 tiles

Reduction of max. / ave. zero-load latencies by up to **49 % / 58 %**

Costs (8×8 Mesh)

- **Wire Density Overhead on each tile**
  - two links per core

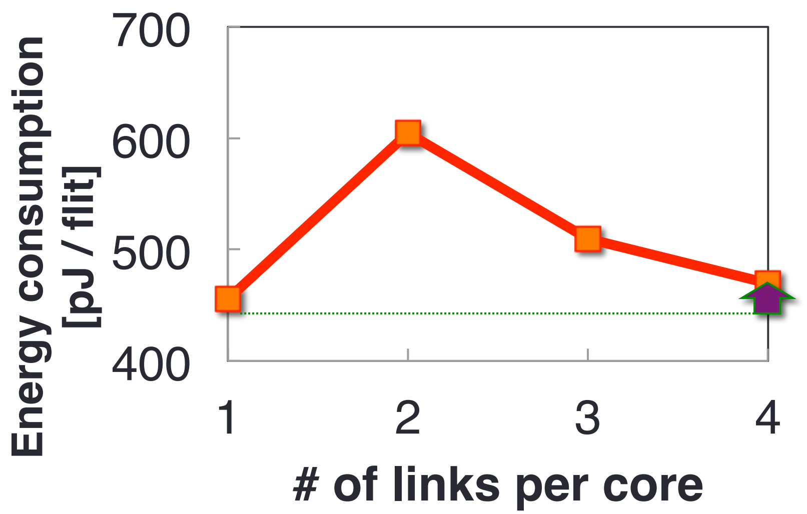
Max.	5.40 links
Ave.	3.06 links
SD	1.58 links

- **Router area (Fujitsu 65nm Process)**

1 link per core	7.71 mm^2
2 links per core	9.86 mm^2

  - Increase by **27.8 %**

- **Energy Consumption**
  - 1.2 V supply voltage
  - 65 nm CMOS Process
  - Wire capacitance load: 0.20 [pJ / mm] (from ITRS 2007)



  - Increase by **3.0 %** at minimum

Parameters of Full System Simulation

- GEM5 [Binkert et al. 2011] is used as full-system simulator
- Using 7 applications from the OpenMP implementation of NAS Parallel Benchmarks

Parameters of Router Network

Switching	Wormhole
Packet length	1- or 5-flit
Flit length	128-bit
# of VCs	3
Size of VC	4 flits
Router latency	3 [cycles]
Link latency	1 or 2 [cycles]
Router topology	8×8 Mesh
Max. link length	4 [tiles]

Routing

Inter-router	XY routing
Between router and core	Selecting shortest path

Parameters of Simulation

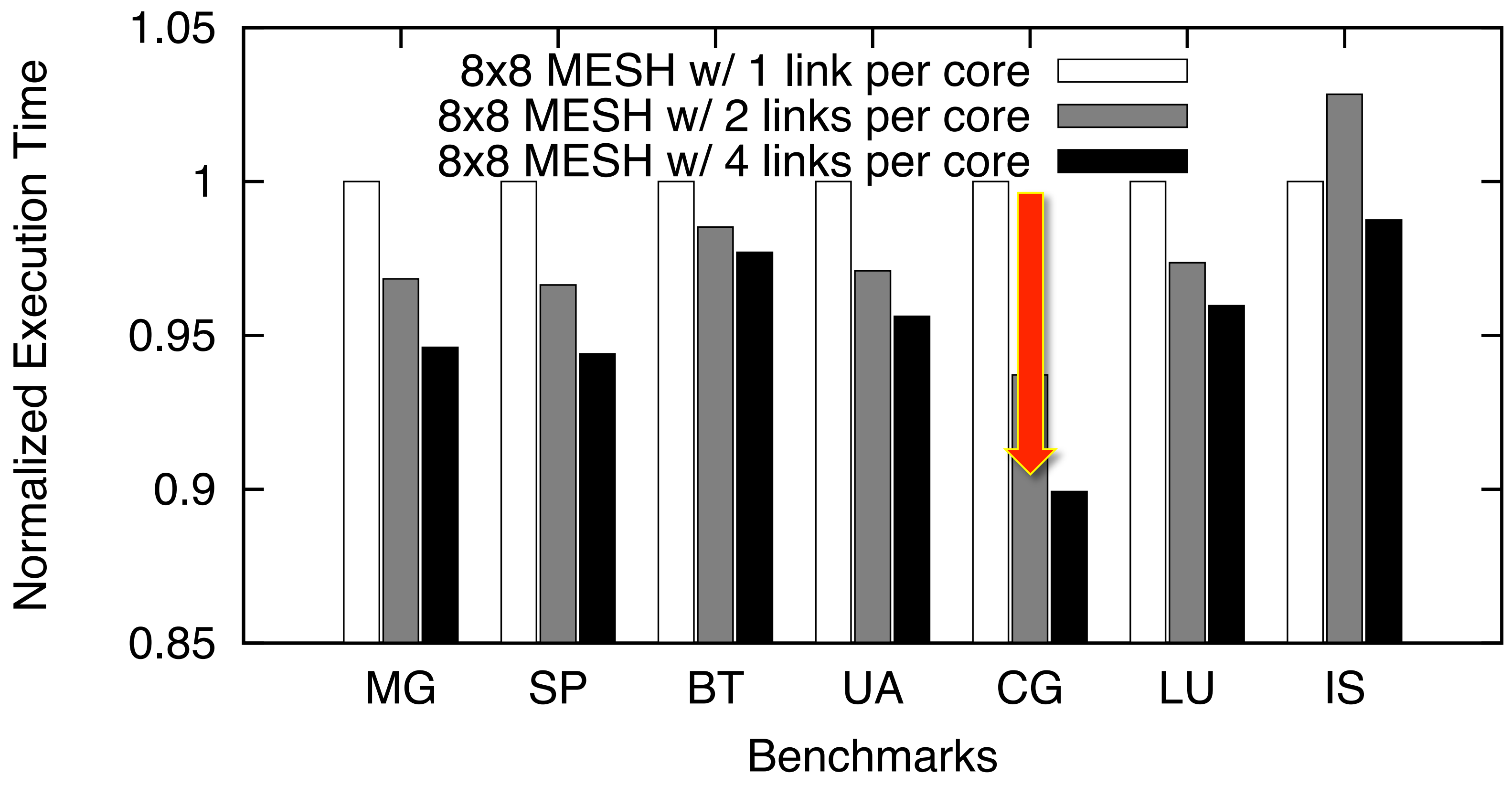
Processor	x86 (64-bit)
L1 cache size	32 KB (line: 64 B)
L1 cache latency	1 cycle
L2 cache size	256 KB (assoc: 8)
L2 cache latency	6 cycles
Memory size	2 GB
Memory latency	160 cycles

Chip Configuration

# of CPUs / L1 caches	8
# of L2 caches	48
# of Directory controllers	8

Full-system Simulation Results

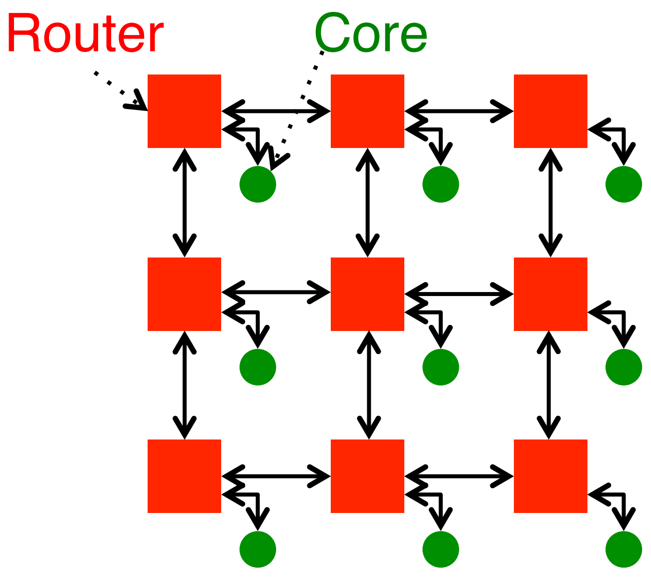
- 8×8 Mesh router topology
- Max. core-link length: 4 tiles



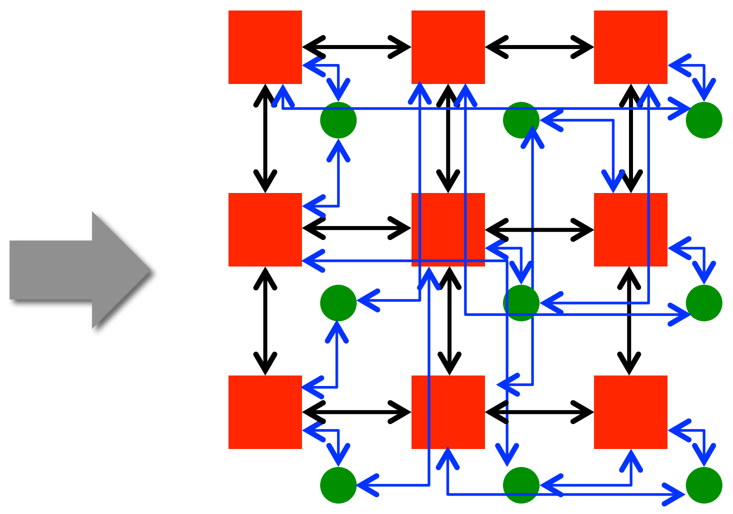
Reduction of application execution time by up to **10.1 %**

Conclusions

- Idea: **Multiple links** between a core and routers on NoCs
- Design : **GA optimization** for picking core-links
  - Reduction of max. / ave. zero-load latencies by up to **49 % / 58 %**
  - Reduction of application execution time by up to **10.1 %**



Conventional NoC



Our Optimized Core-link NoC

Contents

- Conventional NoCs
- Small-world Networks
  - Difficulty in applying on Chips
- How do we reduce path hops of NoCs?
  - Adding multiple links between a core and routers
  - Optimization method for picking core-links
- Evaluations
  - Zero-load latencies
  - Costs
  - Full-system simulation
- Conclusions

Backup Slides

Definition of Fitness Function *f_k*

- Length of the j-th core-link for the i-th core $(0 \le i < N, 0 \le j < x)$: $l_{i,j}$
- # of hops between the p-th and q-th core $(0 \le p < N, 0 \le q < N)$: $h_{p,q}$
- # of links longer than the given maximum link length : I
- Supplemental parameters : $\alpha = \beta = 1000$

$$f_k = \begin{cases} \alpha(\beta \cdot I + \sum_{i=0}^{N-1} \sum_{j=0}^{x-1} l_{i,j}) & (I > 0) \quad (1) \\ \max(h) + \mathrm{mean}(h) & (\text{otherwise}) \quad (2) \end{cases}$$

Reducing link length with function (1)

Reducing # of hops with function (2)

GA Parameters

- # of inds: 100, # of gens: 20000
- P. of crossover, mutation: 1 %, 20 %
- Tournament size: 3