A Case for Random Shortcut Topologies for HPC Interconnects

M. Koibuchi (National Institute of Informatics, JP)

H. Matsutani, H. Amano (Keio University, JP)

D. F. Hsu (Fordham University)

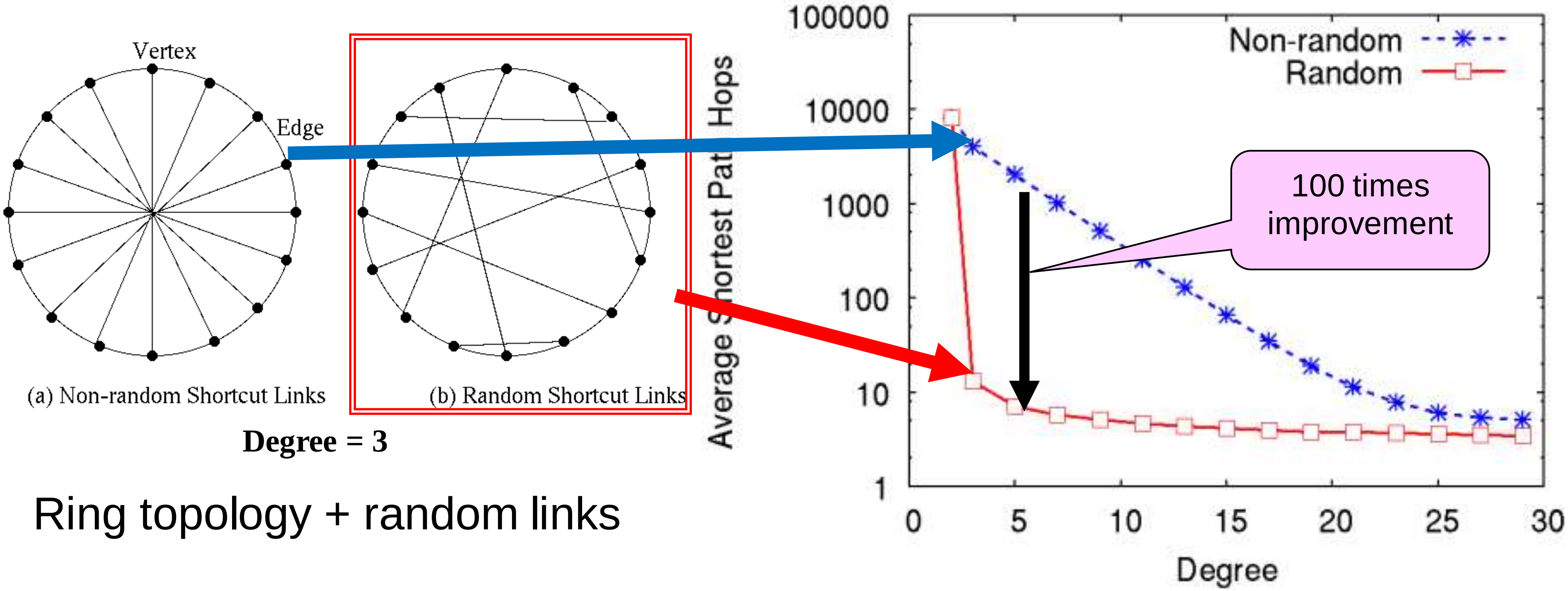
H. Casanova (University of Hawaii at Manoa)

Highlight

Objective：Make a Low-latency topology of HPC NWs

- Switch delay dominates NW delay（>100ns/hop）
  - Decreasing average path hops, and diameter

Idea：Classical topology with random shortcut links



(a) Non-random Shortcut Links

(b) Random Shortcut Links

Degree = 3

100 times improvement

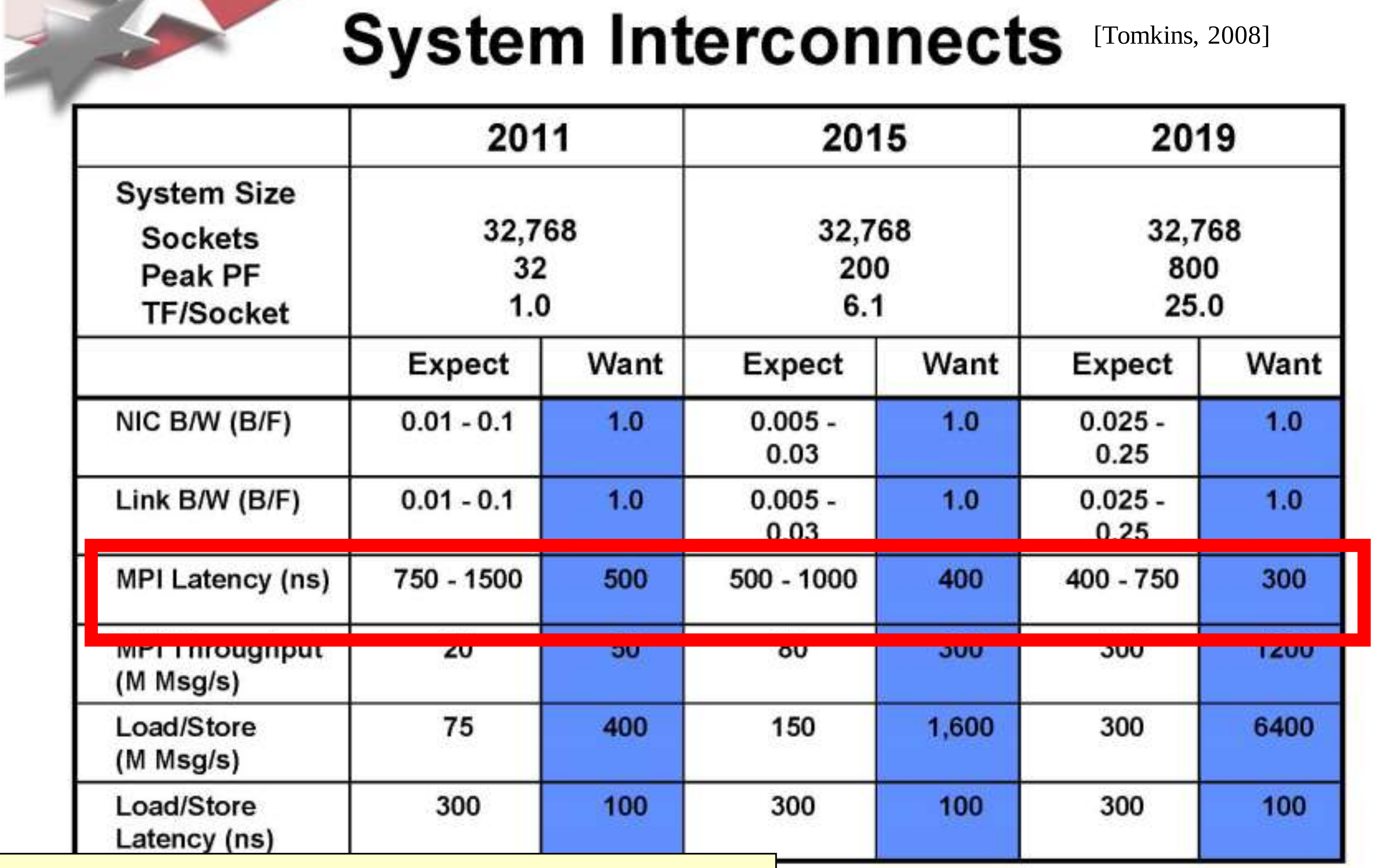
Ring topology + random links

32k-node network

Outline

- Motivation
- Graph analysis of random shortcut topology
- Simulation evaluation of random shortcut topology
- Discussion of limitations
- Conclusions

Motivation to Reduce Hop Counts



System Interconnects

[Tomkins, 2008]

	2011		2015		2019	
System Size Sockets Peak PF TF/Socket	32,768 32 1.0		32,768 200 6.1		32,768 800 25.0	
	Expect	Want	Expect	Want	Expect	Want
NIC B/W (B/F)	0.01 - 0.1	1.0	0.005 - 0.03	1.0	0.025 - 0.25	1.0
Link B/W (B/F)	0.01 - 0.1	1.0	0.005 - 0.03	1.0	0.025 - 0.25	1.0
MPI Latency (ns)	750 - 1500	500	500 - 1000	400	400 - 750	300
MPI Throughput (M Msg/s)	20	50	80	300	300	1200
Load/Store (M Msg/s)	75	400	150	1,600	300	6400
Load/Store Latency (ns)	300	100	300	100	300	100

1 us latency across system [Henmmert, 2008]

Switch delay: >100 ns, Link delay : 5ns/m

Existing HPC topology

Company	System [Network] Name	Max. number of nodes [x # CPUs]	Basic network topology	Injection [Recept'n] node BW in MBytes/s	# of data bits per link per direction	Raw network link BW per direction in Mbytes/sec	Raw network bisection BW (bidir) in Gbytes/s
Intel	ASCI Red Paragon	4,510 [x 2]	**2-D mesh 64 x 64**	400 [400]	16 bits	400	51.2
IBM	ASCI White SP Power3 [Colony]	512 [x 16]	**BMIN w/8-port bidirect. switches (fat-tree or Omega)**	500 [500]	8 bits (+1 bit of control)	500	256
Intel	Thunter Itanium2 Tiger4	1,024 [x 4]	**fat tree w/8-port bidirectional**	928 [928]	8 bits (+2 control for	1,333	1,365
			with express links				
IBM	ASC Purple pSeries 575 [Federation]	>1,280 [x 8]	**BMIN w/8-port bidirect. switches (fat-tree or Omega)**	2,000 [2,000]	8 bits (+2 bits of control)	2,000	2,560
IBM	Blue Gene/L eServer Sol. [Torus Net]	65,536 [x 2]	**3-D torus 32 x 32 x 64**	612,5 [1,050]	1 bit (bit serial)	175	358.4

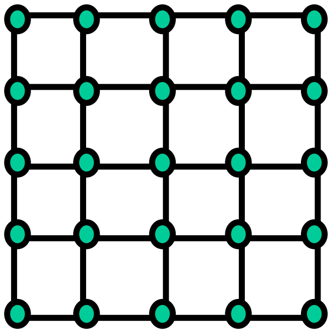
Mesh, torus …

Are such non-random topologies latency-sensitive?

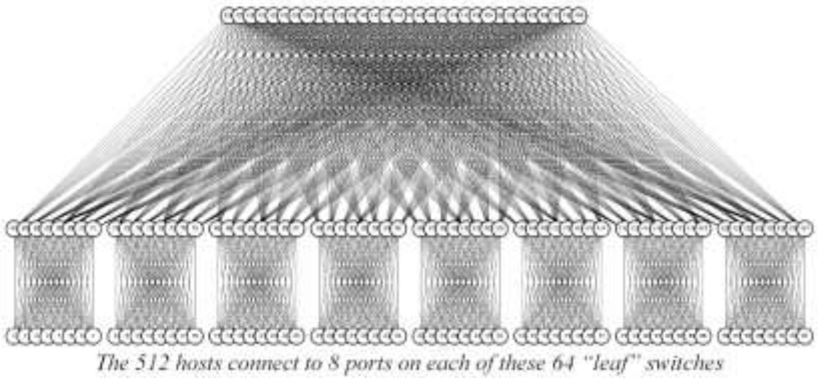
Timothy Pinkston, and Jose Duato, ***Computer Architecture: A Quantitative Approach4th Edition, Appendix E, 2006***

Topology Design

- Latency sensitive, less than 3KB packets [Hemmet,2007]
    - Reduce diameter and avg. shortest path hops
        - Switch delay >> link delay
- Enabling high-radix switches
    - Dozens of ports per switch
- Enabling user-defined routing paths
    - By updating routing tables (e,g, InfiniBand, Ethernet)



Low-radix Network


The 512 hosts connect to 8 ports on each of these 64 "leaf" switches

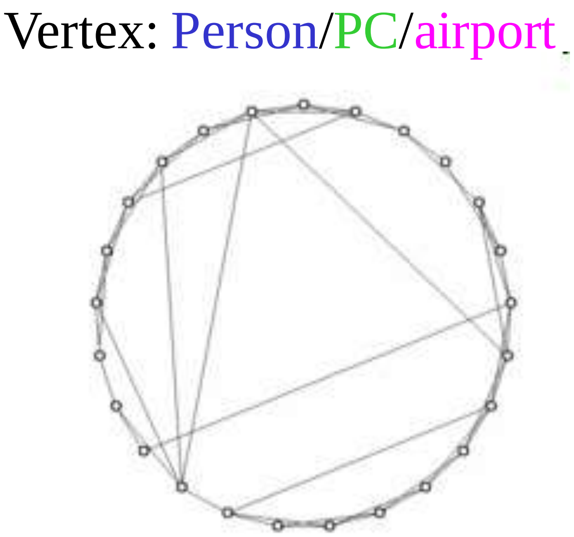
Myricom

High-radix Network

Outline

- Motivation
- Graph analysis of random shortcut topology
- Simulation evaluation of random shortcut topology
- Discussion of limitations
- Conclusions

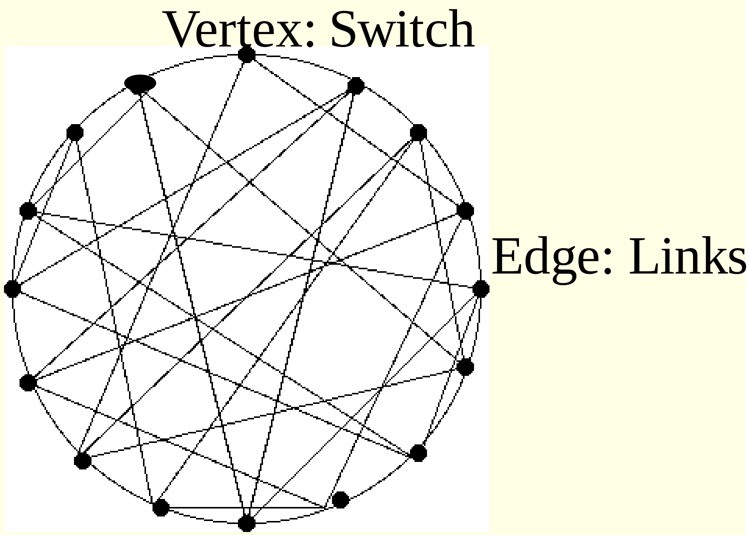
Randomness Makes Graph Shorter [6]



WS Model[Watts98]

Small-world phenomenon

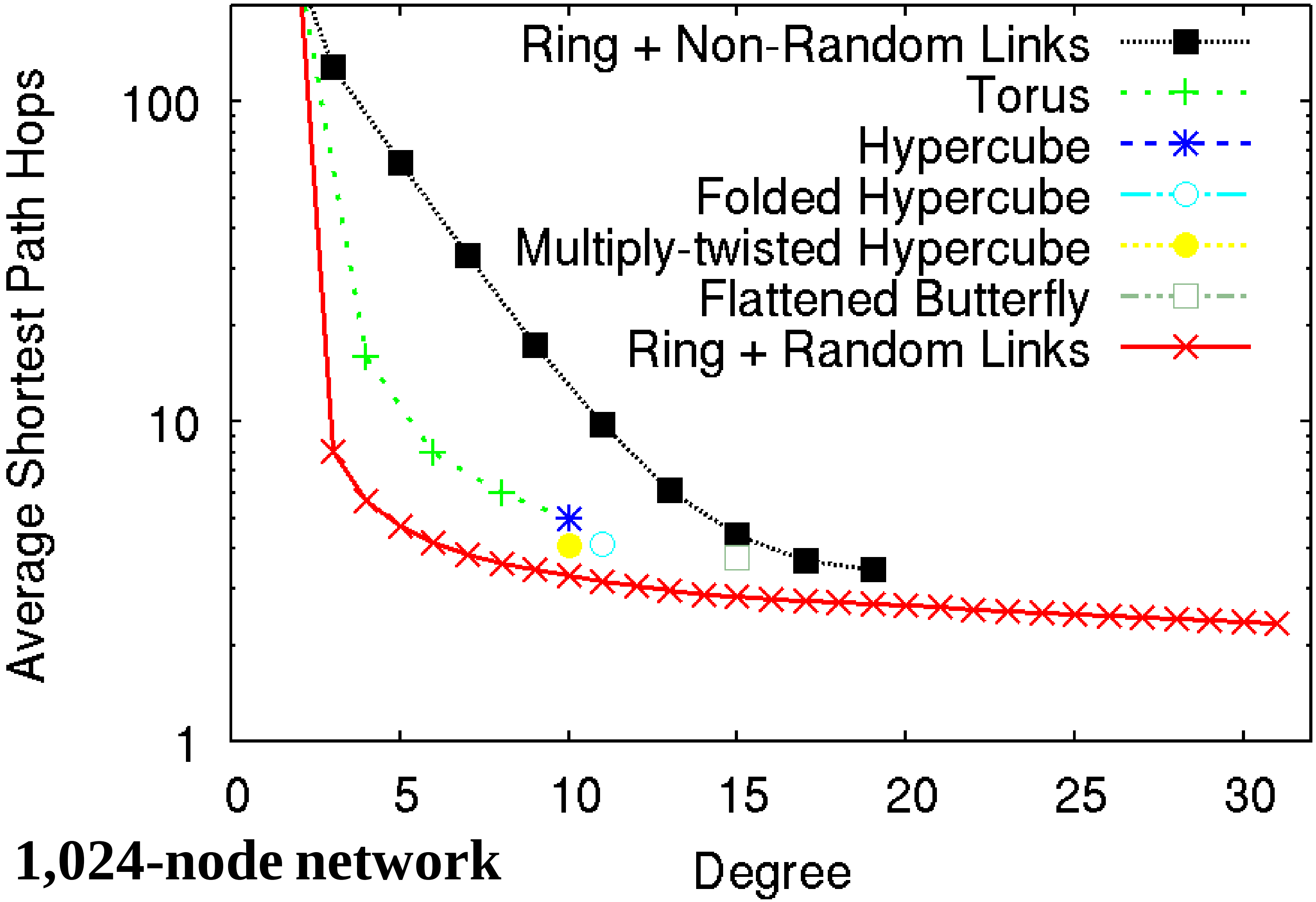
- Social network
- P2P network
- Airport distribution



Its use for HPC interconnects

- Relatively high radix
- More uniformity of each switch degree
- Considering rack layout

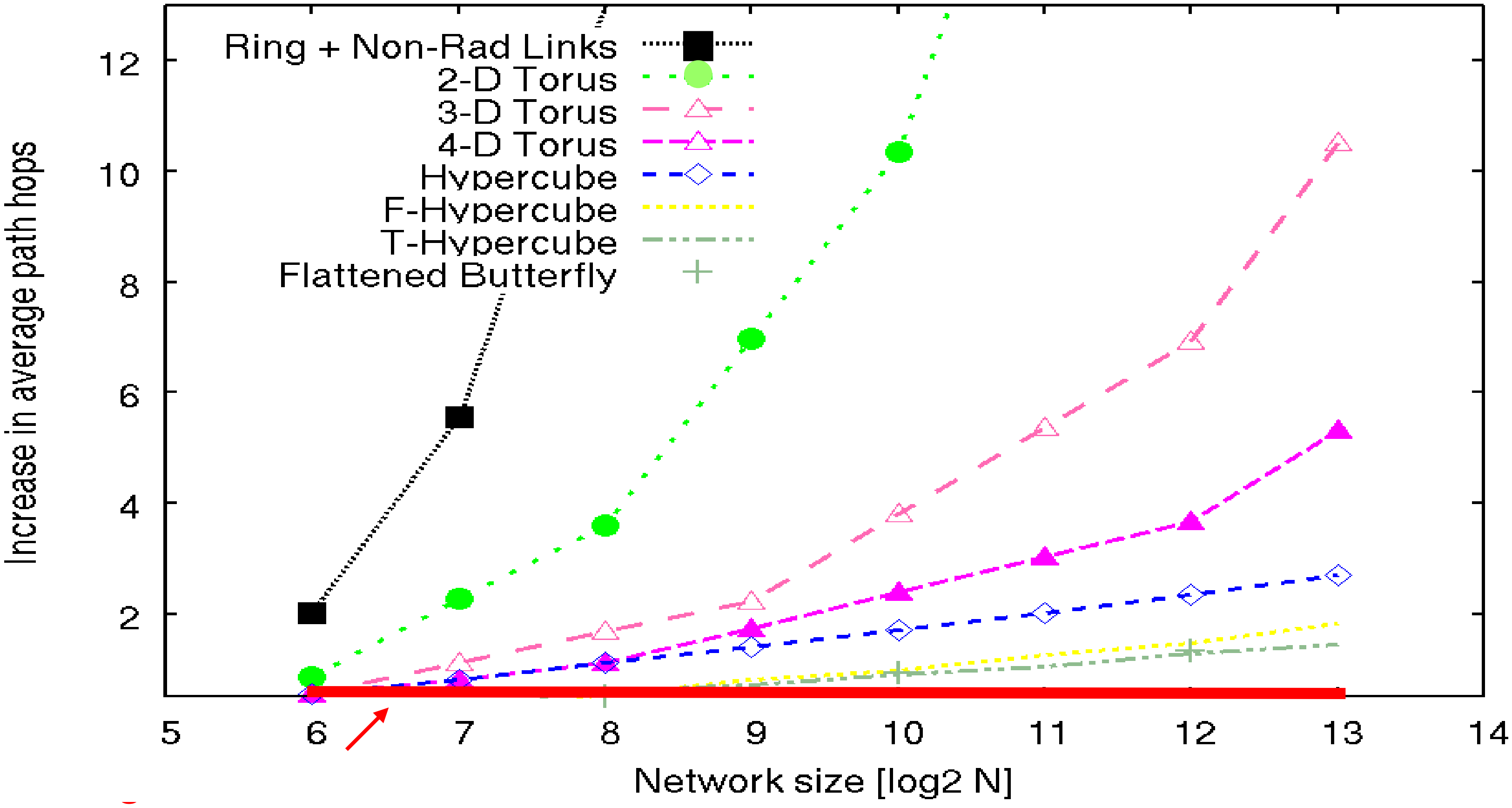
Average Shortest Path Hops



1,024-node network

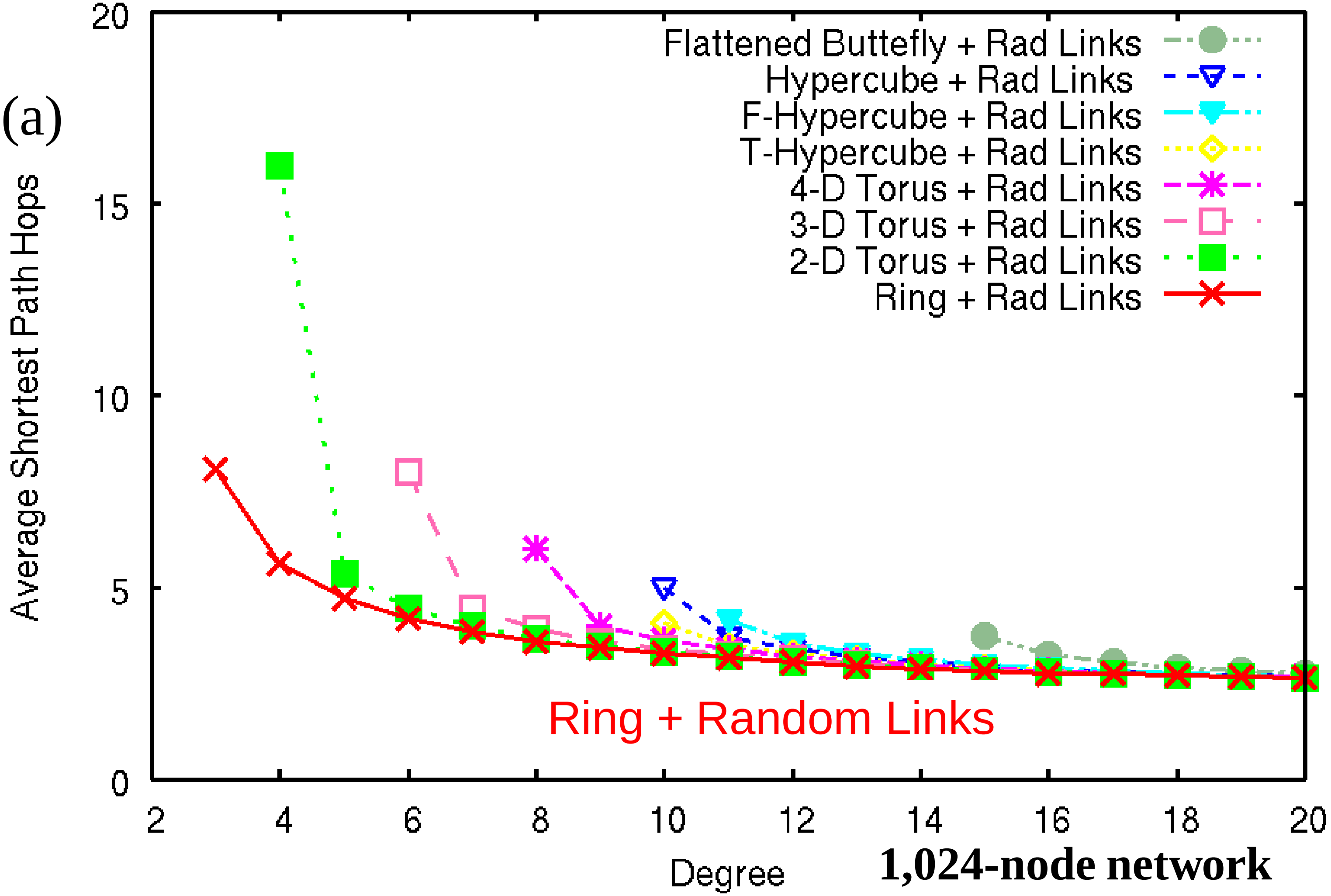
Random links provide better average path hops
- also better diameter

Topology Scalability



Randomness is increasingly beneficial as network size increases

Choice of Baseline Topologies



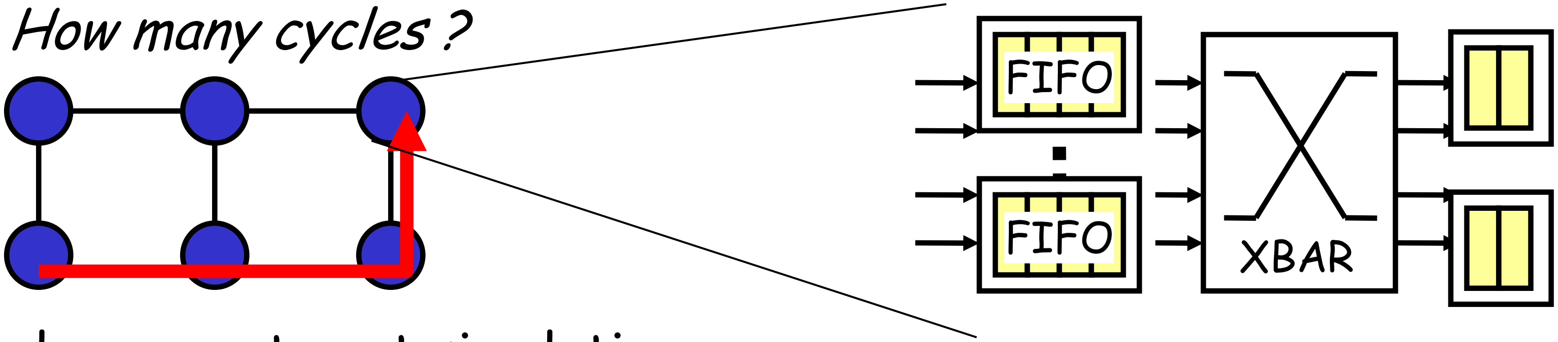
Ring is best due to a larger number of shortcuts

Outline

- Motivation
- Graph analysis of random shortcut topology
- Simulation evaluation of random shortcut topology
- Discussion of limitations
- Conclusions

Simulation Environment

How many cycles ?



Cycle-accurate net simulation

Switch structure

Comm. Latency and Throughput

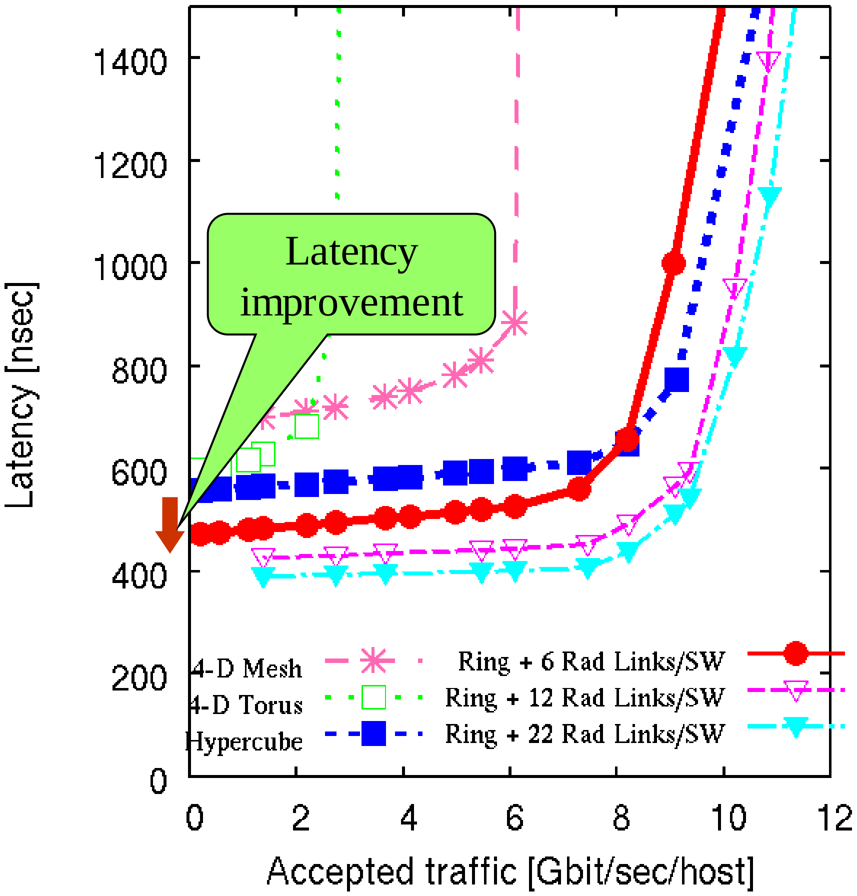
Table 1: Switch & network parameters

Packet length	33-flit (1-flit: 256 bit)
Switching technique	Virtual-cut through
Traffic Pattern	Uni, matrix-t, or bit rev
Number of VCs	2
Switch delay	> 100 ns
Link delay	20 ns

Topology & Routing

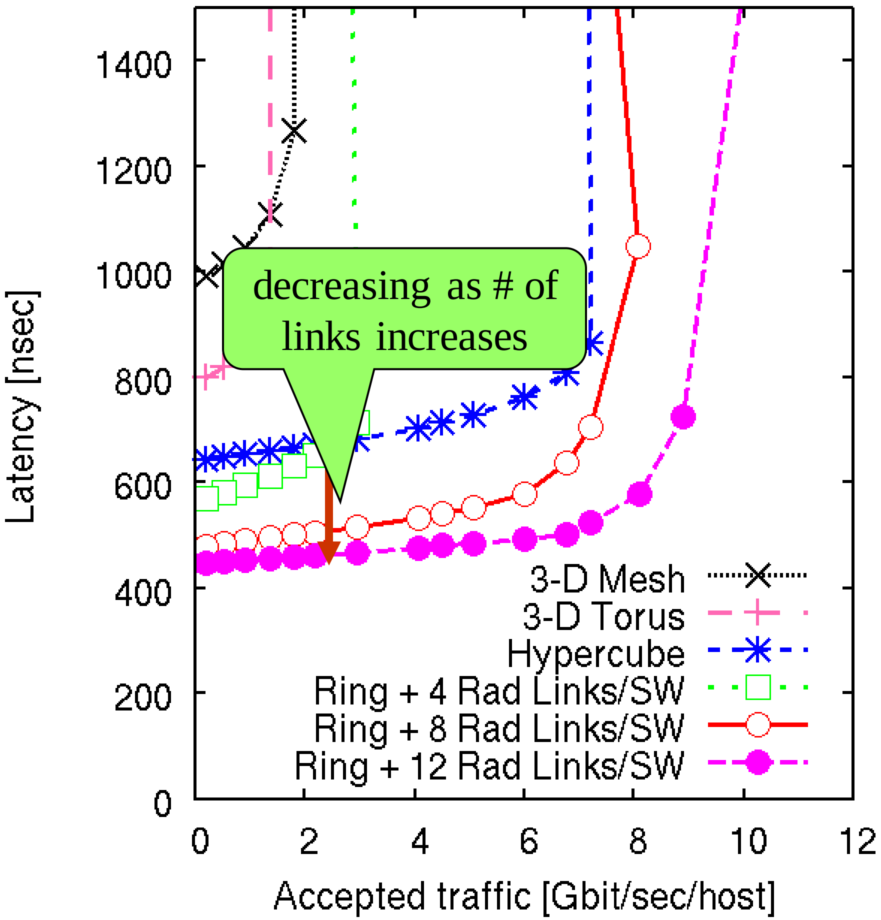
Mesh, Hypercube	Duato
Torus	DOR
Ring + Random	irregular

Accepted Traffic vs Latency



Latency improvement

(a) 256 switch, bit-rev



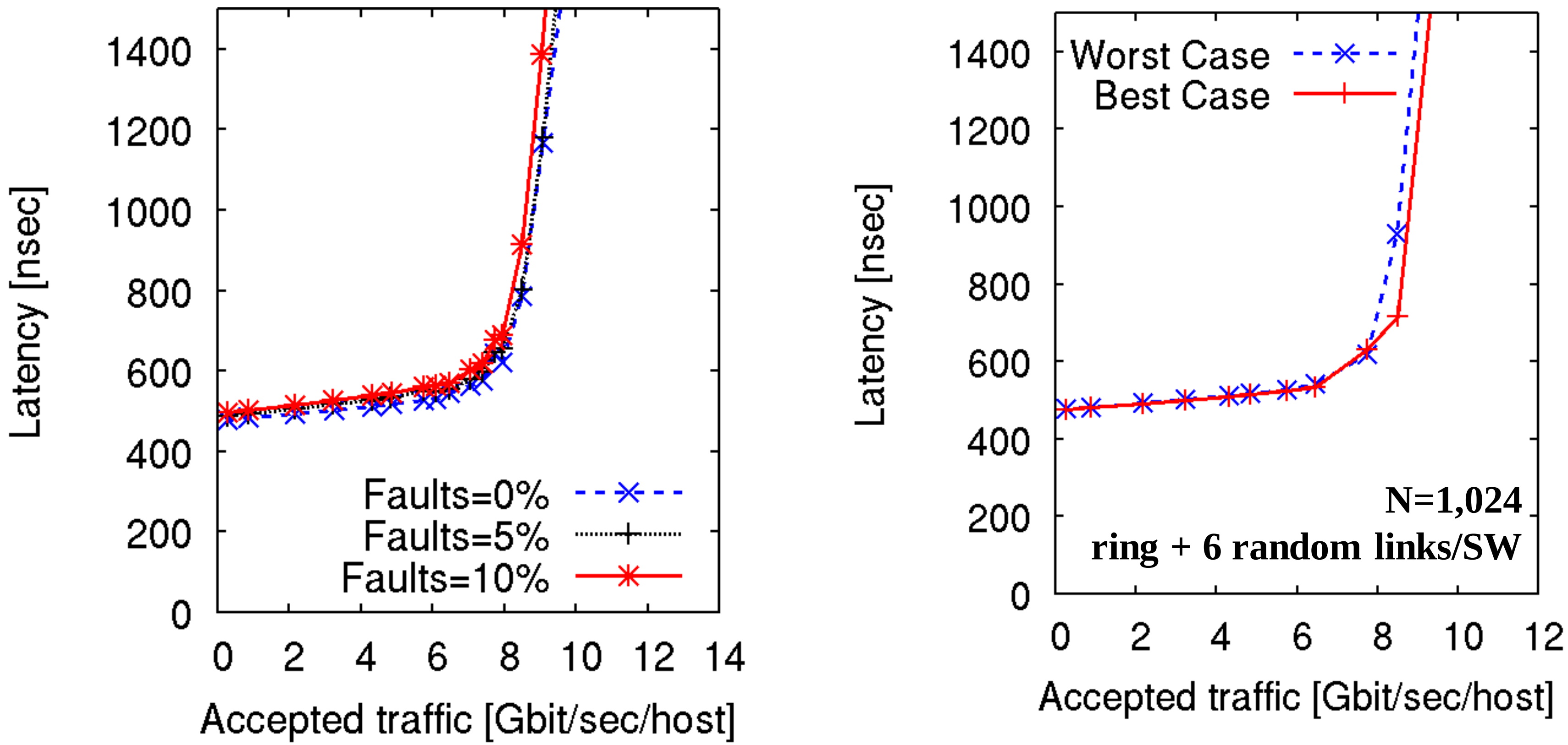
decreasing as # of links increases

(a) 512 switch, matrix-trans

（1）Random shortcuts improve latency by up to 18%

（2）As # of shortcuts increases, more beneficial

Performance Variability



(a) Fault Tolerance **(b) 20 different random instances**

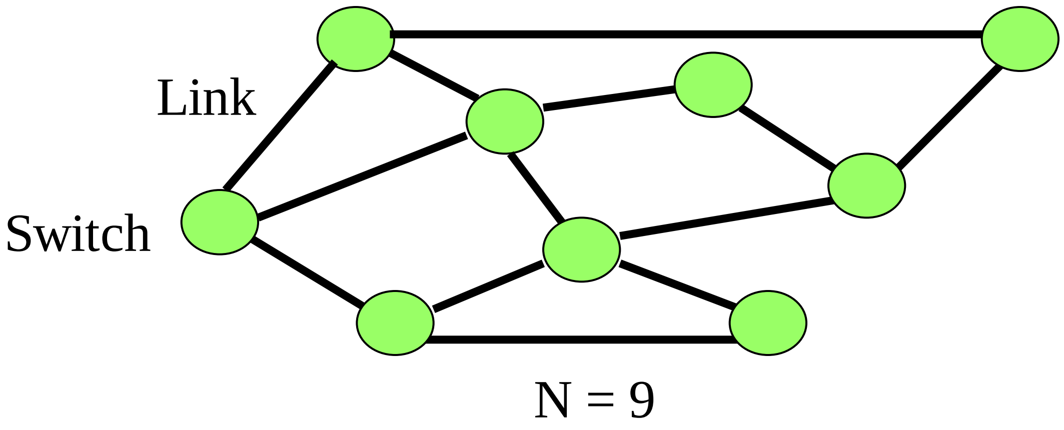
High-radix NW makes random topology robust to faulty links and variability of random generation

Outline

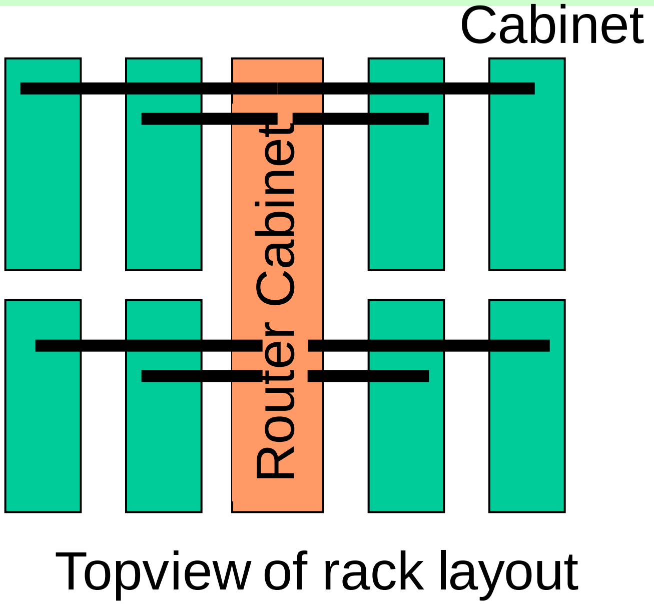
- Motivation
- Graph analysis of random shortcut topology
- Simulation evaluation of random shortcut topology
- Discussion of limitations
- Conclusions

Routing Scalability Issues

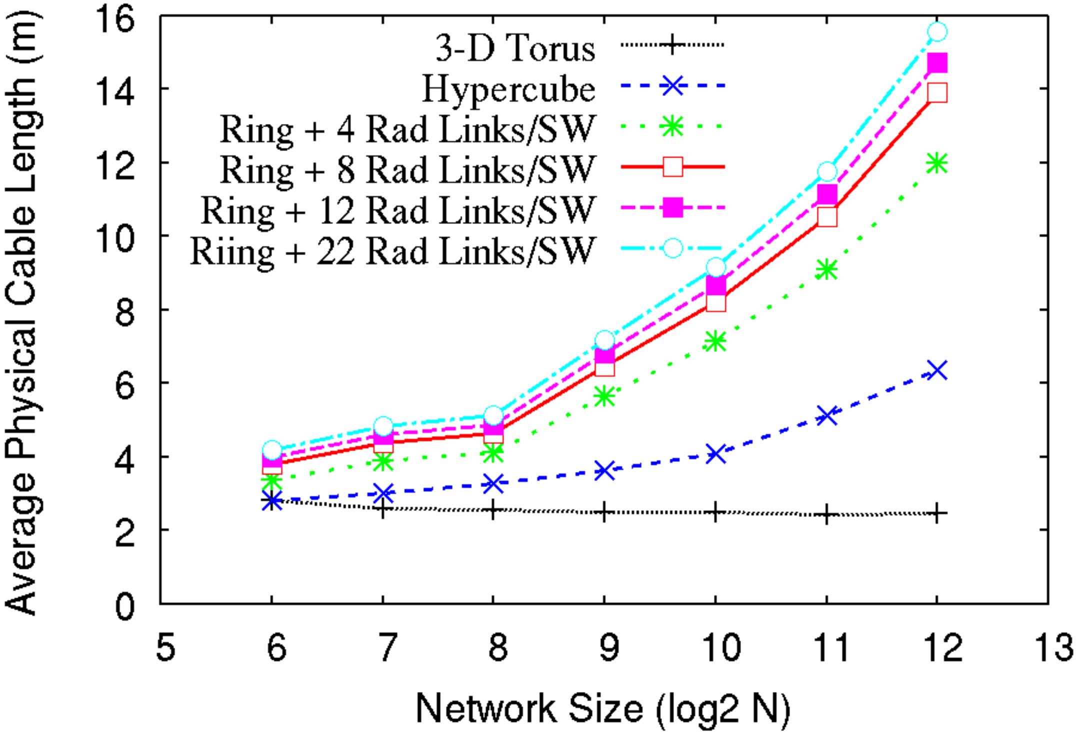
- Address and routing-table size at switch
  - InfiniBand LID: 48k
  - General issue regardless of topology
- Computational cost of path search
  - Topology-agnostic deadlock-free routing [Flich,TPDS2012]
    - $O(N^2)$ or higher
    - Only when initially deploying the system



Physical Cable Length



Topview of rack layout



InfiniBand Link length
passive copper 10m
active copper:40m
Optical:100m~

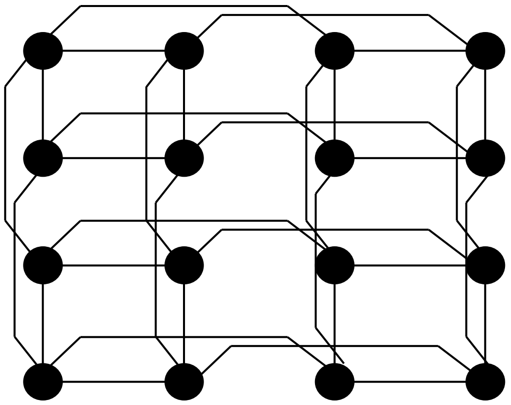
Random Top. can use the same media

Wiring cost does not increase much

Parameters (Cray BlackWindows)
128 nodes/cabinet
cabinet footprint : 0.57m x1.44m
2m cable overhead
75 nodes/m^2 density [Kim,ISCA07]

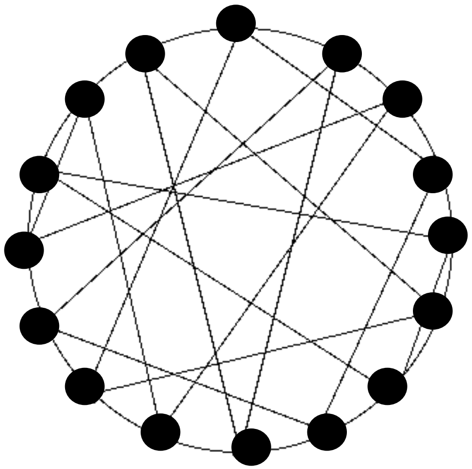
Conclusions

- Use of random shortcuts at HPC interconnects
  - Ring + random shortcuts is best
  - Advantage of high-radix networks
    - Little variability of sampling and performance
- Random shortcut topology imposes no constraints on the number of switches, and links



Hypercube
(Non-random topology)

Up to 18% lower latency



Random Shortcut Topology
(Ring + random shortcuts)