T. Hoefler

The three L's in modern high-performance networking: low latency, low cost, low processing load

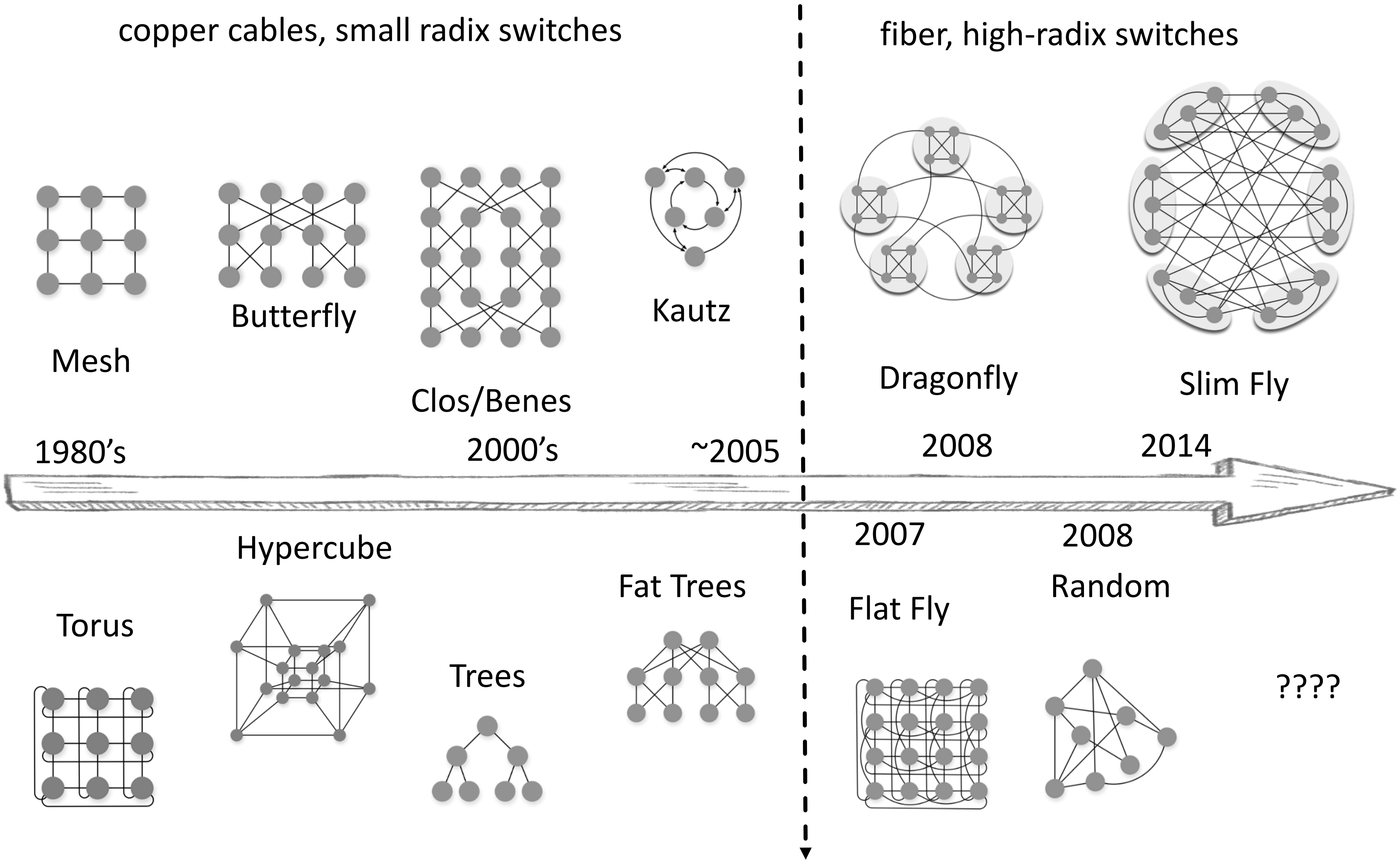
with support of M. Besta, S. Di Girolamo, K. Taranov @ SPCL -- R. Grant, R. Birghtwell @ Sandia Natl. Labs
keynote at HiPINEB @ HPCA, Vienna, Austria in February 2017

Low Latency

Low Cost

Low Processing Load

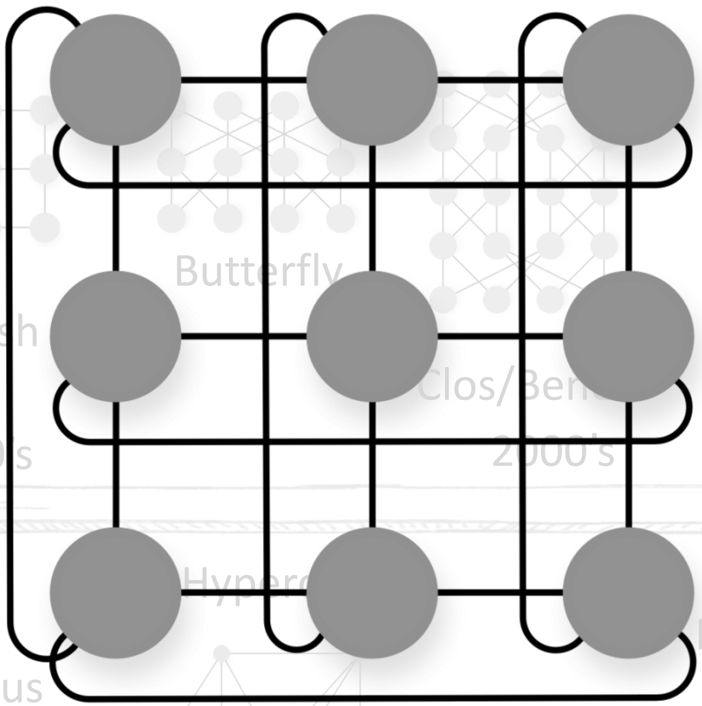
A BRIEF HISTORY OF NETWORK TOPOLOGIES

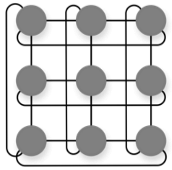


A BRIEF HISTORY OF NETWORK TOPOLOGIES

copper cables, small radix switches

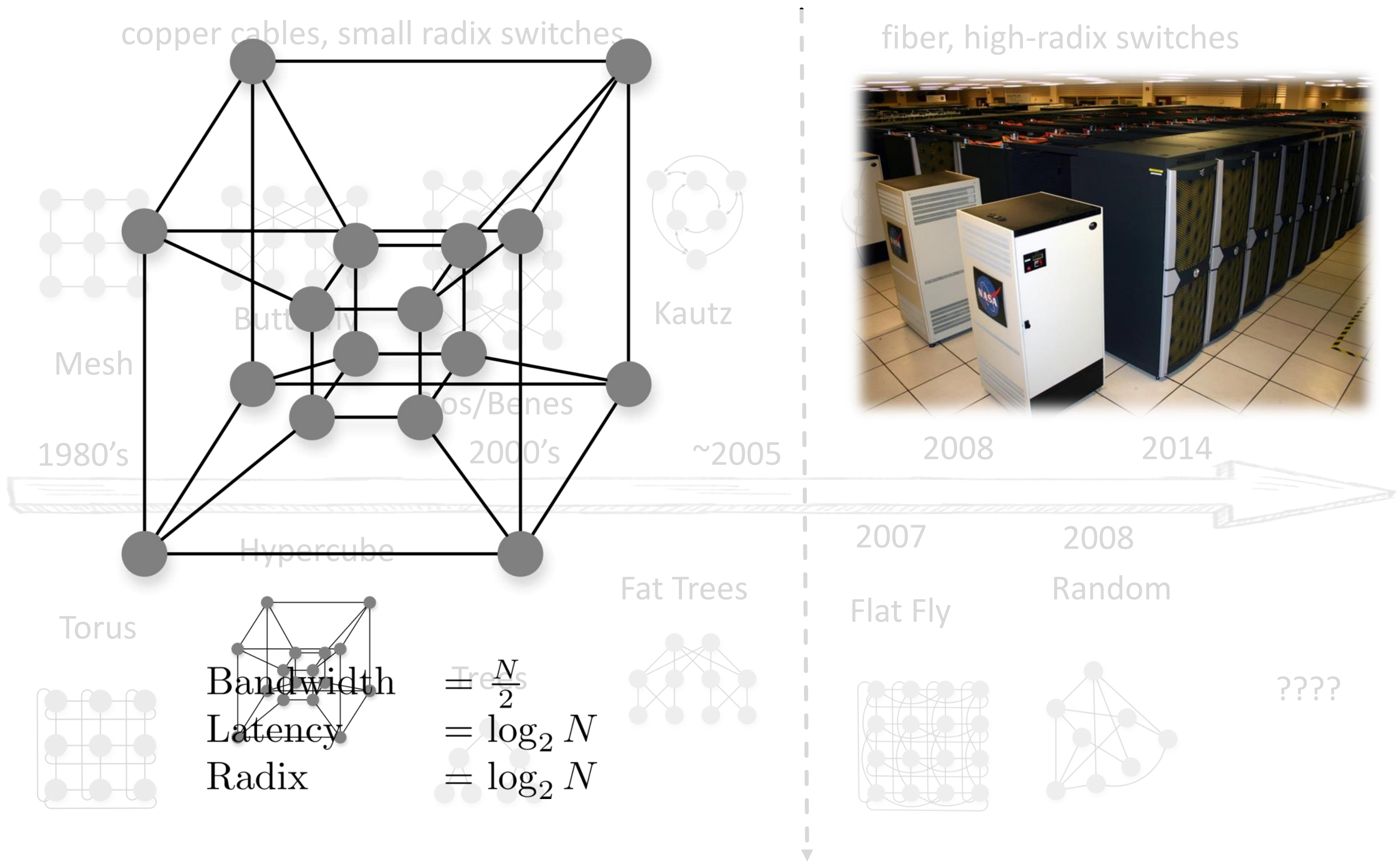
fiber, high-radix switches





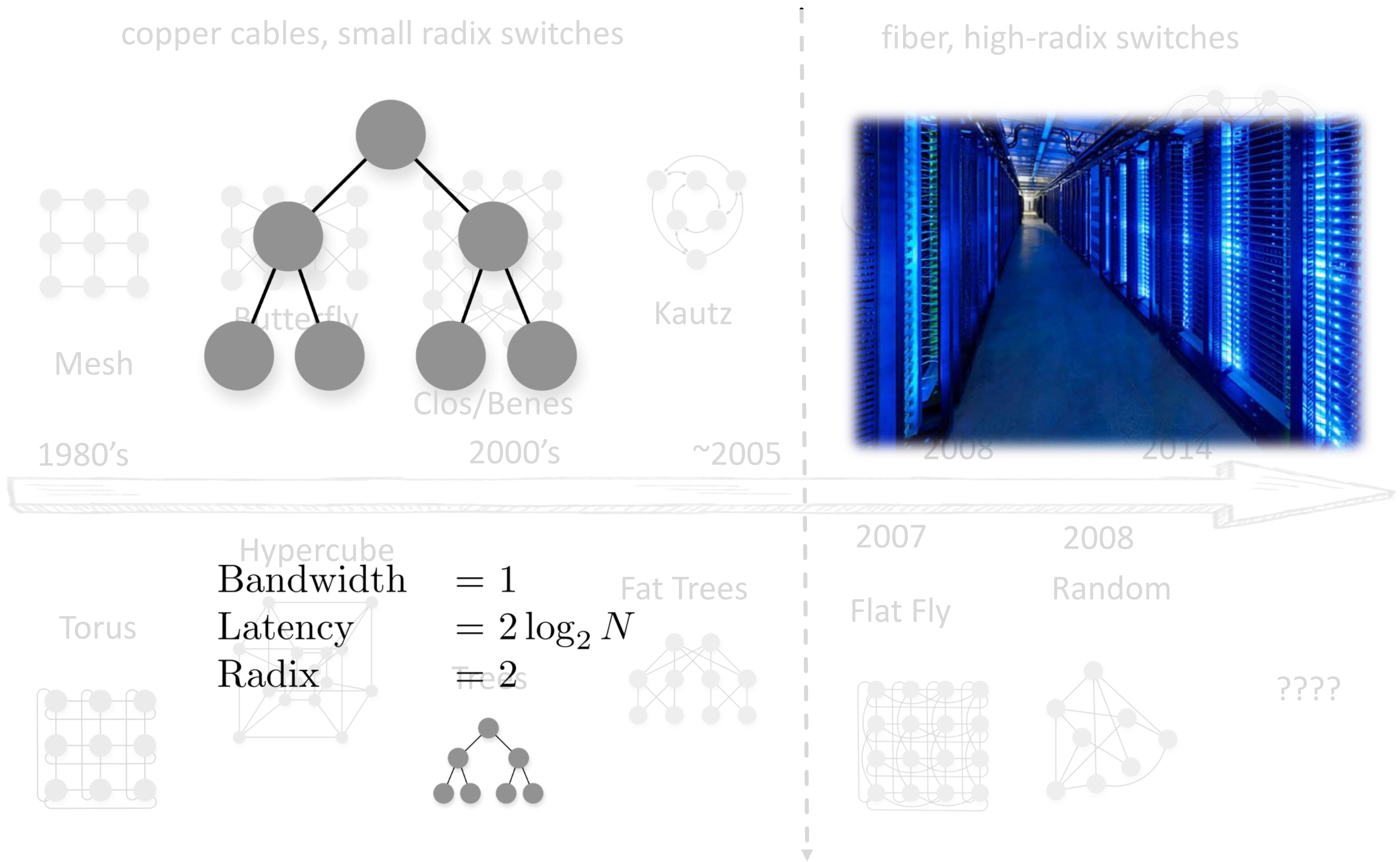
$$\text{Bandwidth} = 2\sqrt[d]{N^{d-1}}$$
$$\text{Latency} = \frac{d}{2}\sqrt[d]{N}$$
$$\text{Radix} = 2d$$

A BRIEF HISTORY OF NETWORK TOPOLOGIES



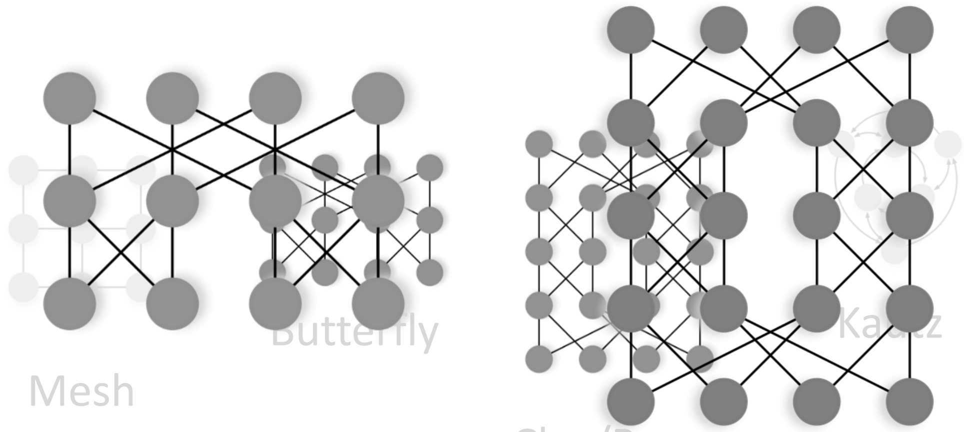
$$\begin{aligned} \text{Bandwidth} &= \frac{N}{2} \\ \text{Latency} &= \log_2 N \\ \text{Radix} &= \log_2 N \end{aligned}$$

A BRIEF HISTORY OF NETWORK TOPOLOGIES



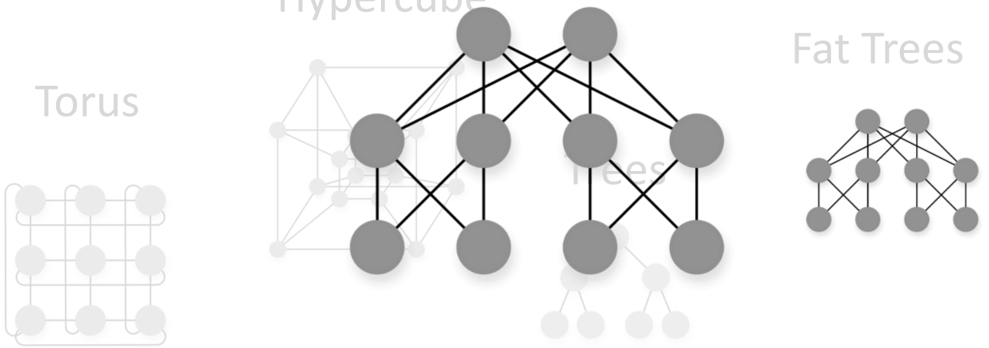
A BRIEF HISTORY OF NETWORK TOPOLOGIES

copper cables, small radix switches



$$\text{Bandwidth} = \frac{N}{2}$$
$$\text{Latency} = 2\log_2 N$$
$$\text{Radix} = 4$$

2000's ~2005 2008 2014



A BRIEF HISTORY OF NETWORK TOPOLOGIES

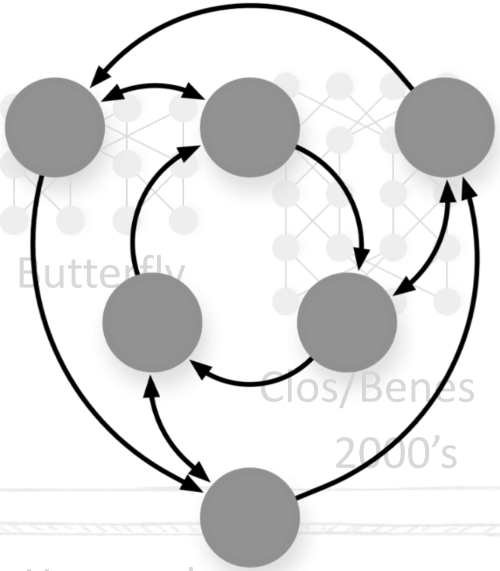
copper cables, small radix switches

fiber, high-radix switches

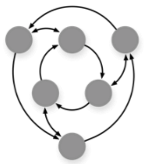
Mesh

Butterfly

Clos/Benes



Kautz



Dragonfly

Slim Fly

1980's

2000's

~2005

Hypercube

Fat Trees

Torus

Trees

Flat

????

$$
\begin{aligned}
\text{Bandwidth} &= \rightarrow \frac{N}{4} \\
\text{Latency} &= \log_k N \\
\text{Radix} &= k
\end{aligned}
$$

A BRIEF HISTORY OF NETWORK TOPOLOGIES

copper cables, small radix switches

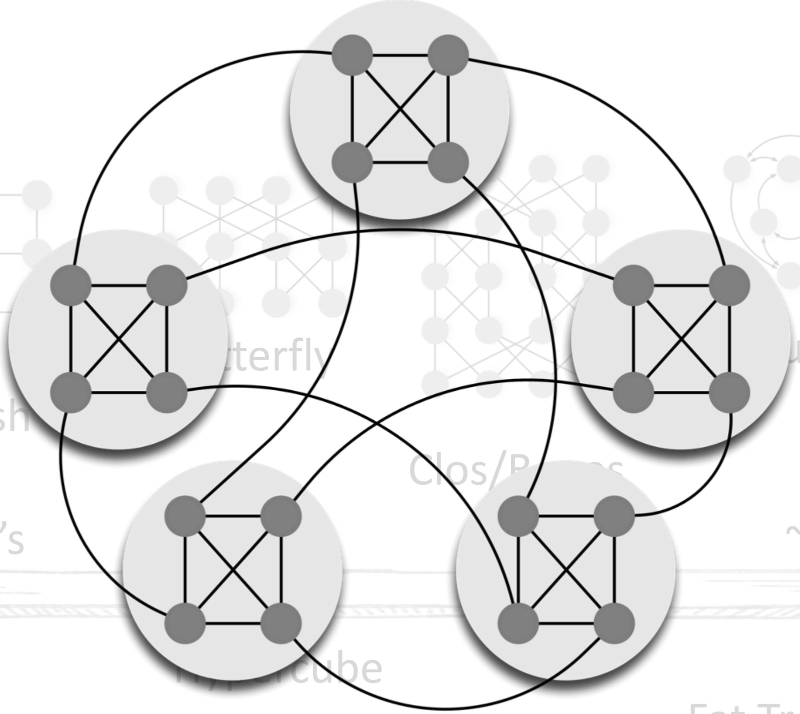
fiber, high-radix switches

Mesh

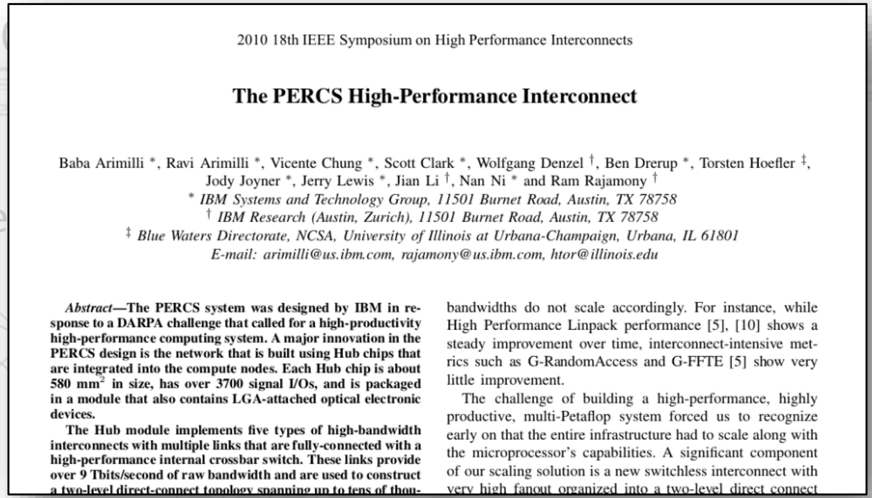
1980's

Torus

Fat Tree



$$\text{Bandwidth} \approx \frac{N}{4}$$
$$\text{Latency} = 3 - 5$$
$$\text{Radix} = 48 - 64$$



2010 18th IEEE Symposium on High Performance Interconnects

The PERCS High-Performance Interconnect

Baba Arimilli *, Ravi Arimilli *, Vicente Chung *, Scott Clark *, Wolfgang Denzel †, Ben Drerup *, Torsten Hoefler ‡, Jody Joyner *, Jerry Lewis *, Jian Li †, Nan Ni * and Ram Rajamony †
* *IBM Systems and Technology Group, 11501 Burnet Road, Austin, TX 78758*
† *IBM Research (Austin, Zurich), 11501 Burnet Road, Austin, TX 78758*
‡ *Blue Waters Directorate, NCSA, University of Illinois at Urbana-Champaign, Urbana, IL 61801*
E-mail: arimilli@us.ibm.com, rajamony@us.ibm.com, htor@illinois.edu

***Abstract*—The PERCS system was designed by IBM in response to a DARPA challenge that called for a high-productivity high-performance computing system. A major innovation in the PERCS design is the network that is built using Hub chips that are integrated into the compute nodes. Each Hub chip is about 580 mm^2 in size, has over 3700 signal I/Os, and is packaged in a module that also contains LGA-attached optical electronic devices.**

The Hub module implements five types of high-bandwidth interconnects with multiple links that are fully-connected with a high-performance internal crossbar switch. These links provide over 9 Tbits/second of raw bandwidth and are used to construct a two-level direct-connect topology spanning up to tens of thou-

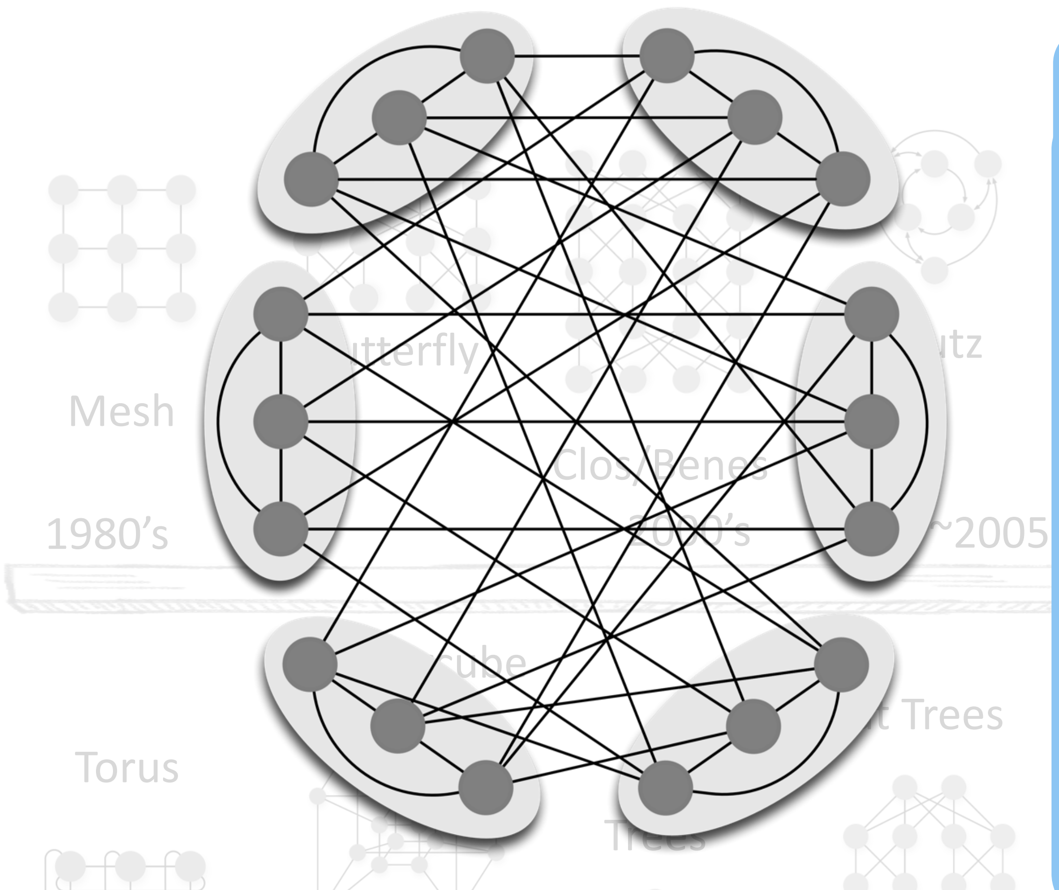
bandwidths do not scale accordingly. For instance, while High Performance Linpack performance [5], [10] shows a steady improvement over time, interconnect-intensive metrics such as G-RandomAccess and G-FFTE [5] show very little improvement.

The challenge of building a high-performance, highly productive, multi-Petaflop system forced us to recognize early on that the entire infrastructure had to scale along with the microprocessor's capabilities. A significant component of our scaling solution is a new switchless interconnect with very high fanout organized into a two-level direct connect

A BRIEF HISTORY OF NETWORK TOPOLOGIES

copper cables, small radix switches

fiber, high-radix switches



$$\text{Bandwidth} \approx \frac{N}{4}$$
$$\text{Latency} = 2 - 4$$
$$\text{Radix} = k$$

Key ideas:

"It's the diameter, stupid"

Lower diameter:
→ Less cables traversed
→ Less cables needed
→ Less routers needed

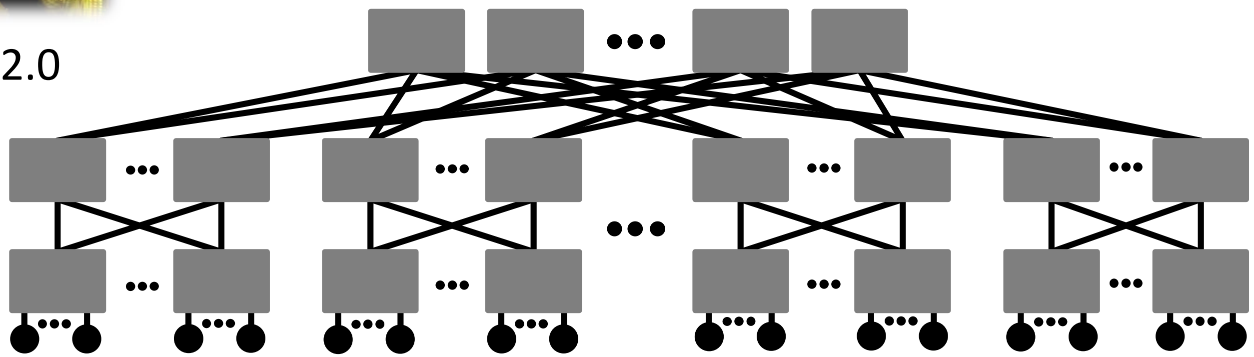
Cost and energy savings:
→ Up to 50% over Fat Tree
→ Up to 33% over Dragonfly

DESIGNING A LOW-DIAMETER NETWORK TOPOLOGY

EXAMPLE: FULL-BANDWIDTH FAT TREE VS HOFFMAN-SINGLETON GRAPH

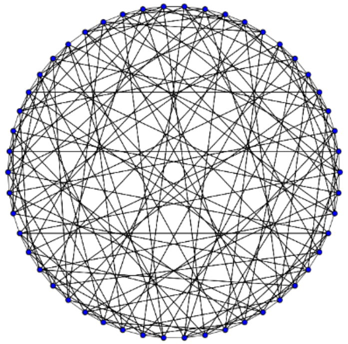
TSUBAME2.0

3-level fat tree:



diameter = 4

Slim Fly based on the Hoffman-Singleton Graph [1]:



diameter = 2
> ~50% fewer routers
> ~30% fewer cables

[1] Hoffman, Alan J.; Singleton, Robert R. (1960), *Moore graphs with diameter 2 and 3*, IBM Journal of Research and Development

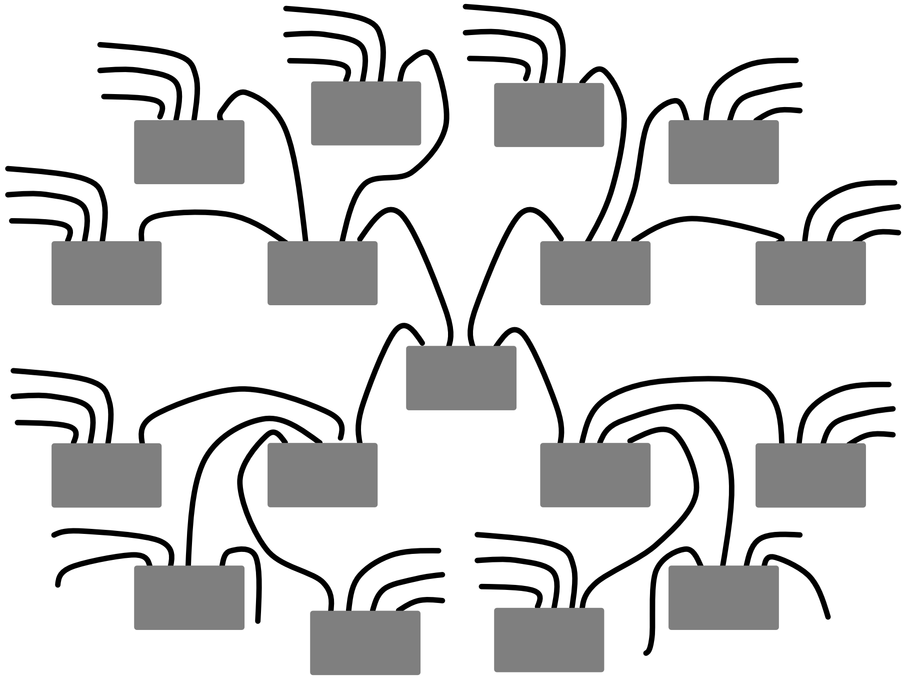
Limits on Low diameter Network Topologies

Key method

Optimize towards the Moore Bound [1]: the upper bound on the *number of vertices* in a graph with given *diameter D* and *radix k*.

$$MB(D,k) = 1 + k + k(k-1) + k(k-1)^2 + \cdots$$

$$MB(D,k) = 1 + k\sum_{i=0}^{D-1}(k-1)^i$$



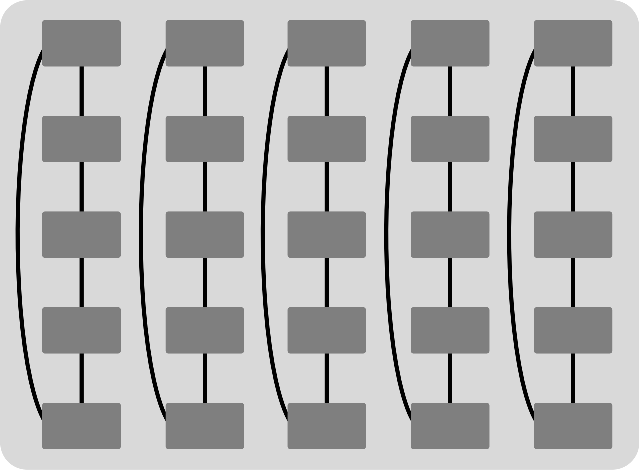
[1] M. Miller, J. Siráň. Moore graphs and beyond: A survey of the degree/diameter problem, Electronic Journal of Combinatorics, 2005.

The Slim Fly Principle – approaching the Moore bound!

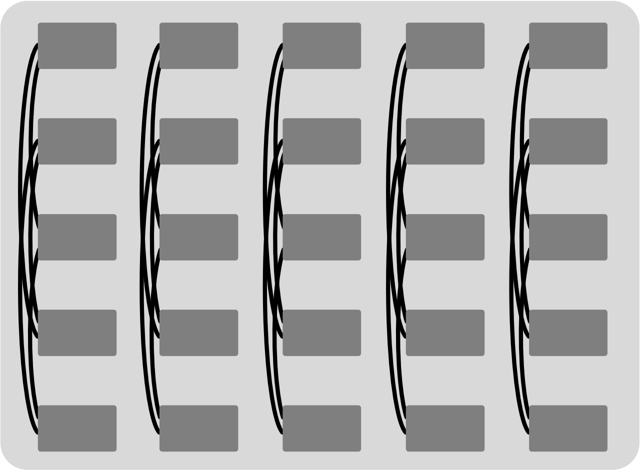
Connecting routers: Diameter 2

Example Slim Fly design for *diameter* = 2: *MMS graphs* [1]

A subgraph with
identical groups of routers



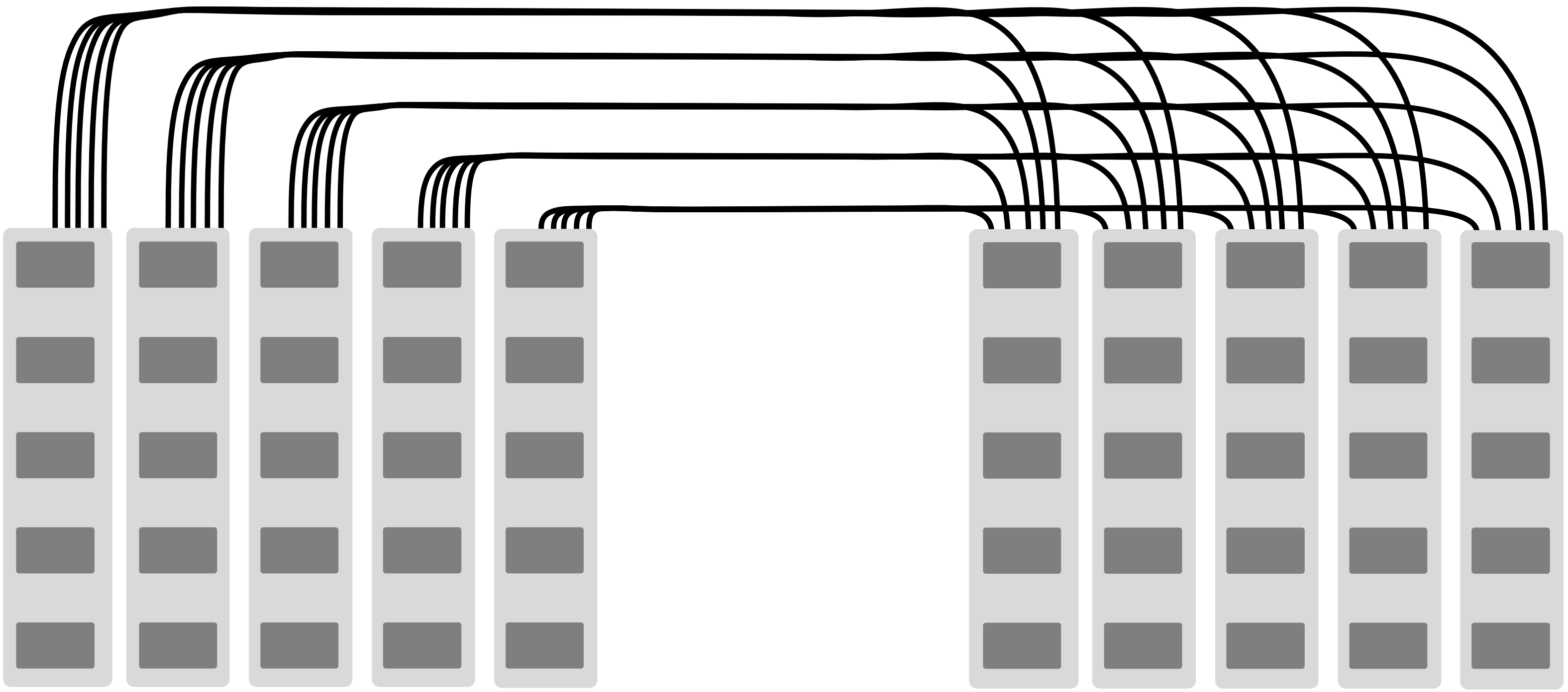
A subgraph with
identical groups of routers



[1] B. D. McKay, M. Miller, and J. Siráň. A note on large graphs of diameter two and given maximum degree. Journal of Combinatorial Theory, Series B, 74(1):110 – 118, 1998

The Slim Fly Principle – approaching the Moore bound!

Connecting routers: Diameter 2



Groups form a fully-connected bipartite graph

Low Latency

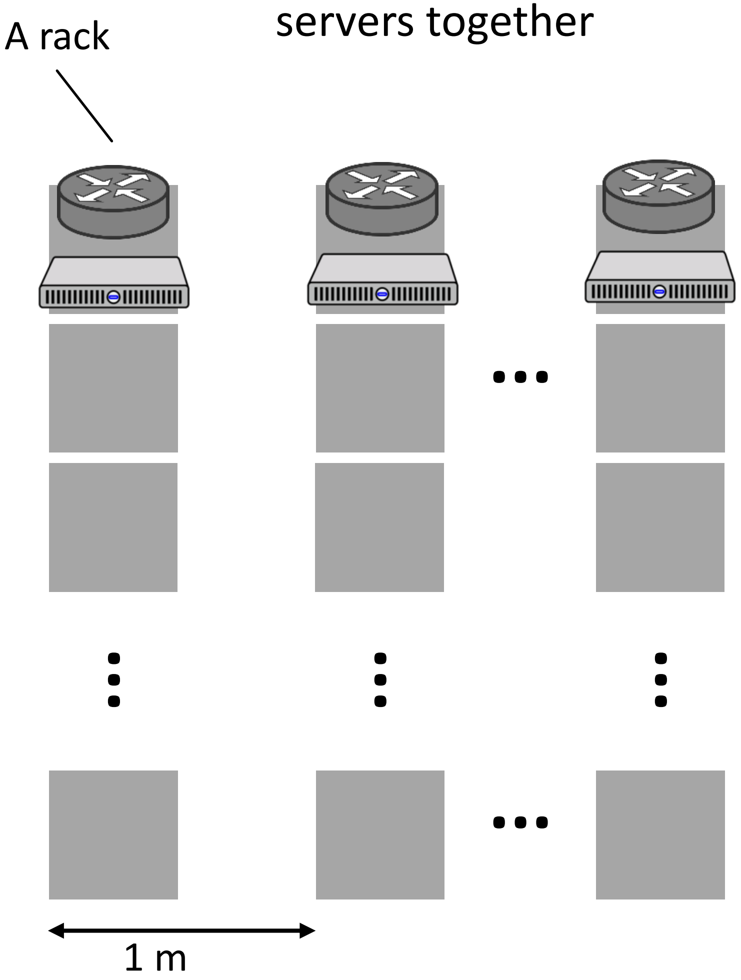
Low Cost

Low Processing Load

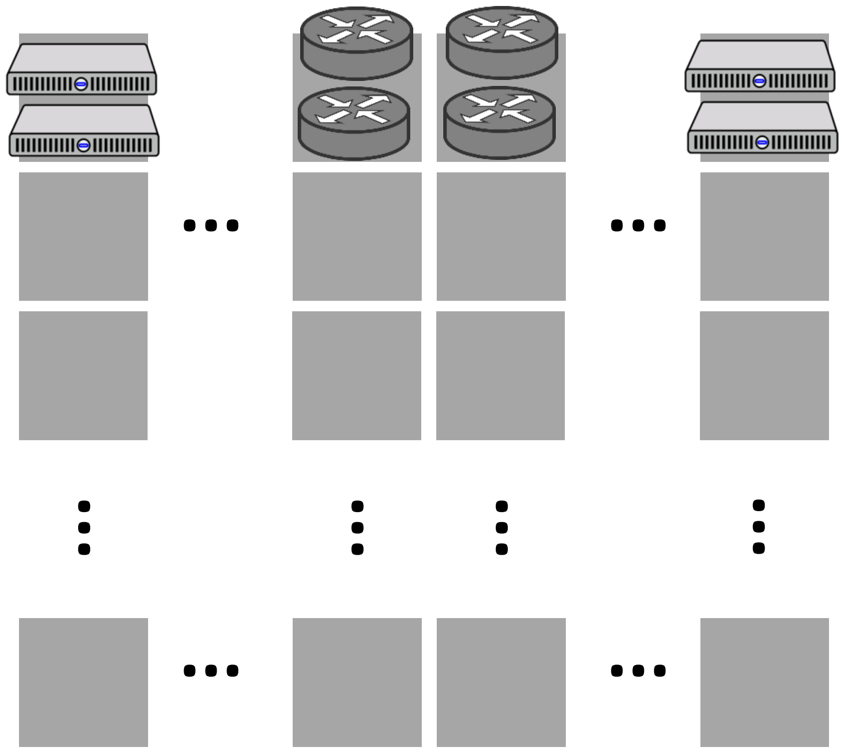
Slim Fly MMS - Cost Comparison

Cost Models: variants

Variant 1: Routers and servers together



Variant 2: Routers and servers separately



*Prices based on: COLFAX DIRECT HPC and Data Center Gear

COST COMPARISON
CABLE COST MODEL

- Cable cost as a function of distance
  - The functions obtained using linear regression*
  - Optical transceivers considered
  - Cables used:
    Mellanox IB FDR10 40Gb/s QSFP

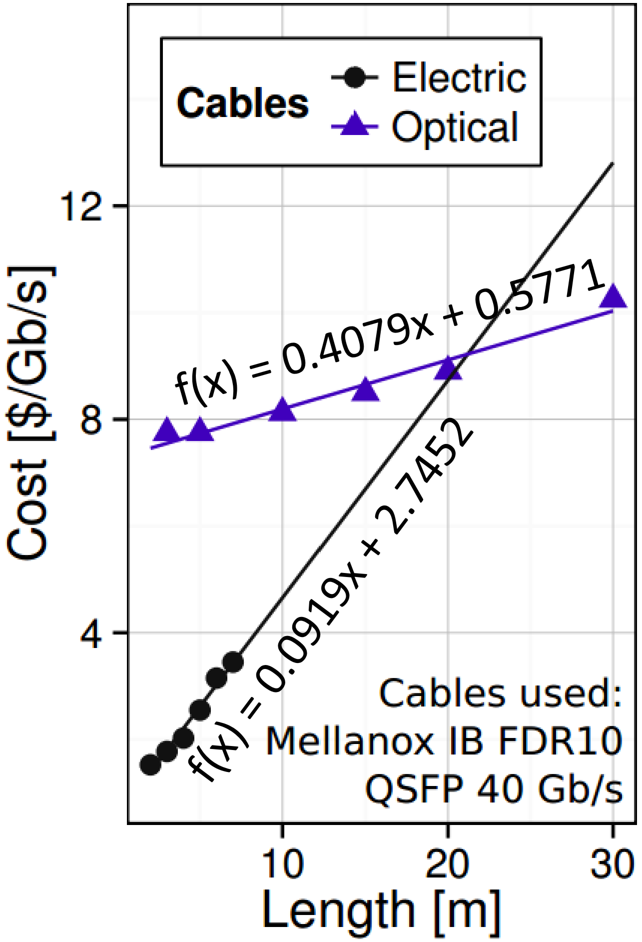
- Other used cables (studies in paper):

Mellanox IB QDR
56Gb/s QSFP

Mellanox Ethernet
40Gb/s QSFP

Mellanox Ethernet
10Gb/s SFP+

Elpeus Ethernet
10Gb/s SFP+



Cost Comparison

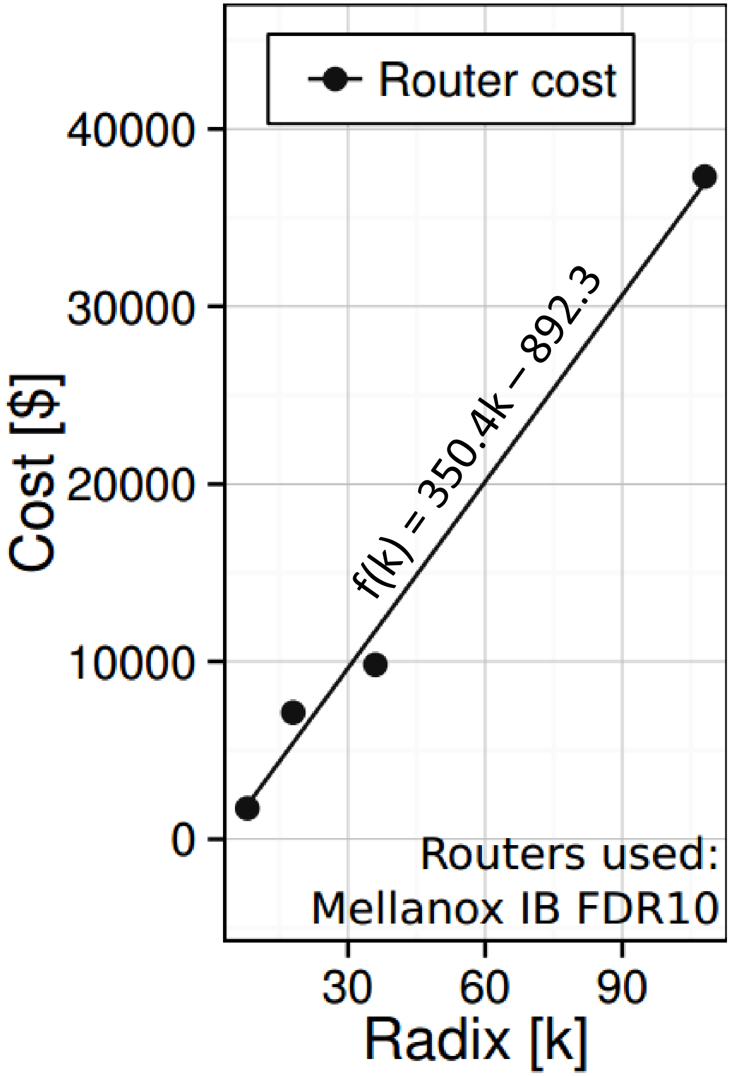
Router Cost Model

*Prices based on: COLFAX DIRECT HPC and Data Center Gear

- Router cost as a function of radix
  - The function obtained using linear regression*
  - Routers used:

Mellanox IB FDR10

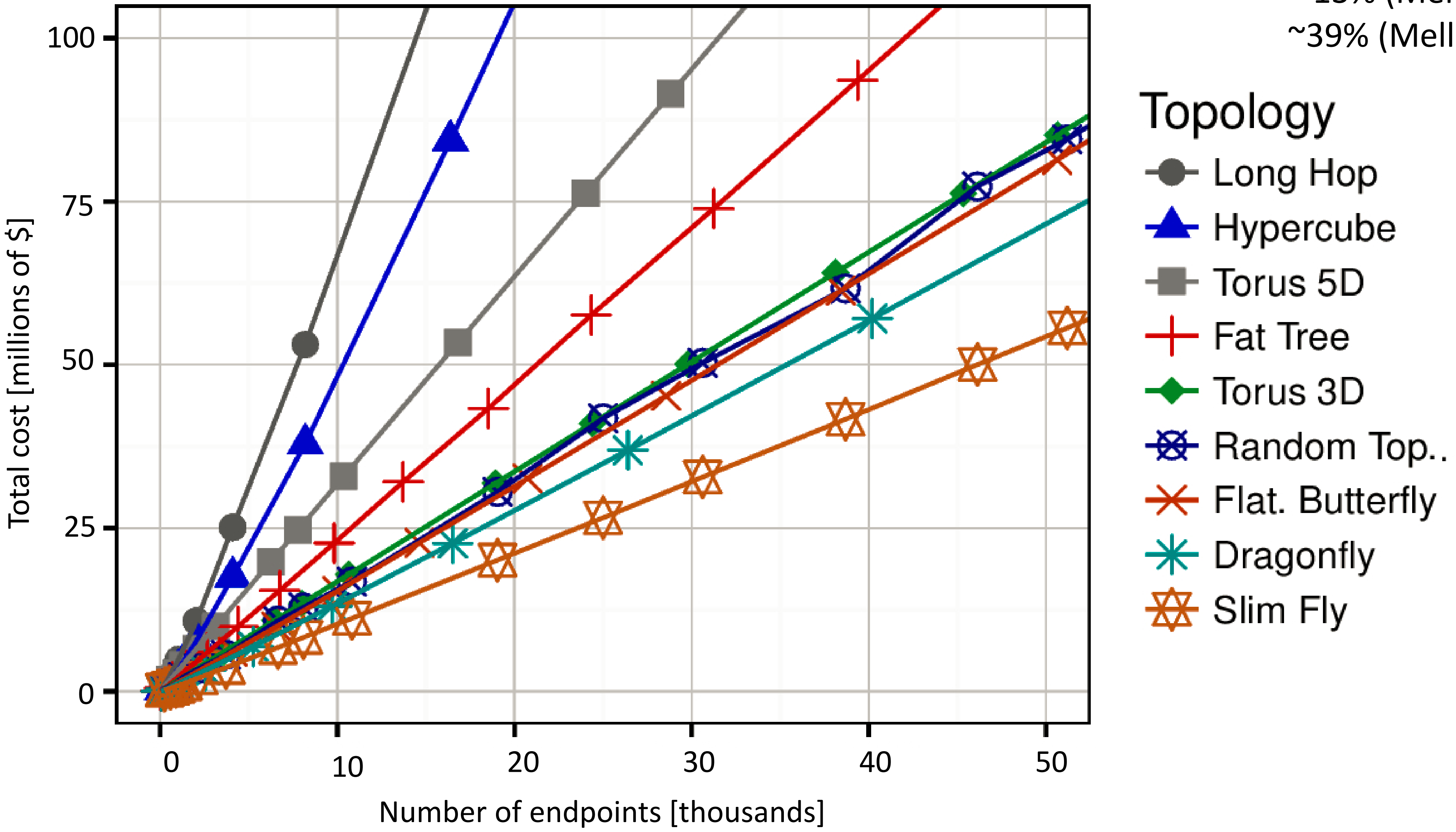
Mellanox Ethernet 10/40 Gb



Maciej Besta, TH: Slim Fly: A Cost Effective Low-Diameter Network Topology, Supercomputing 2014, SC14

Cost Comparison

Variant 1



Variant 2:

SF less expensive than Dragonfly by ~13% (Mellanox IB routers) up to ~39% (Mellanox Ethernet routers)

Maciej Besta, TH: Slim Fly: A Cost Effective Low-Diameter Network Topology, Supercomputing 2014, SC14

Cost Comparison

Detailed Case-Study

- A Slim Fly with;
  - N = 10,830
  - k = 43
  - N_r = 722

Cost & Power Comparison

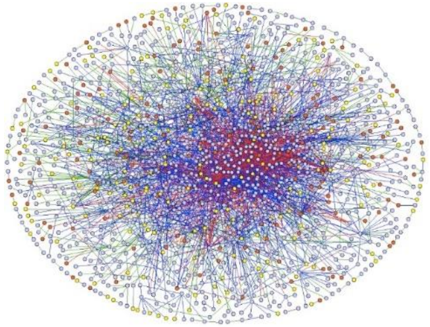
Detailed Case-Study: High-Radix Topologies

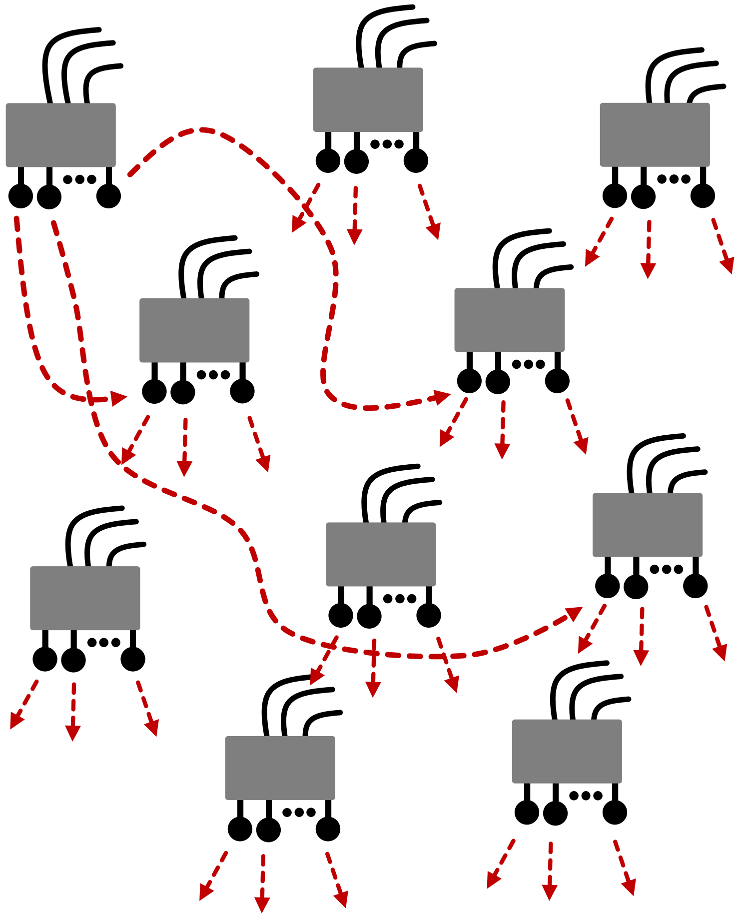
Topology	Fat tree	Random	Flat. Butterfly	Dragonfly	**Slim Fly**
Endpoints (N)	19,876	40,200	20,736	58,806	**10,830**
Routers (N_r)	2,311	4,020	1,728	5,346	**722**
Radix (k)	**43**	**43**	**43**	**43**	**43**
Electric cables	19,414	32,488	9,504	56,133	**6,669**
Fiber cables	40,215	33,842	20,736	29,524	**6,869**
Cost per node [$]	2,346	1,743	1,570	1,438	**1,033**
Power per node [W]	14.0	12.04	10.8	10.9	**8.02**

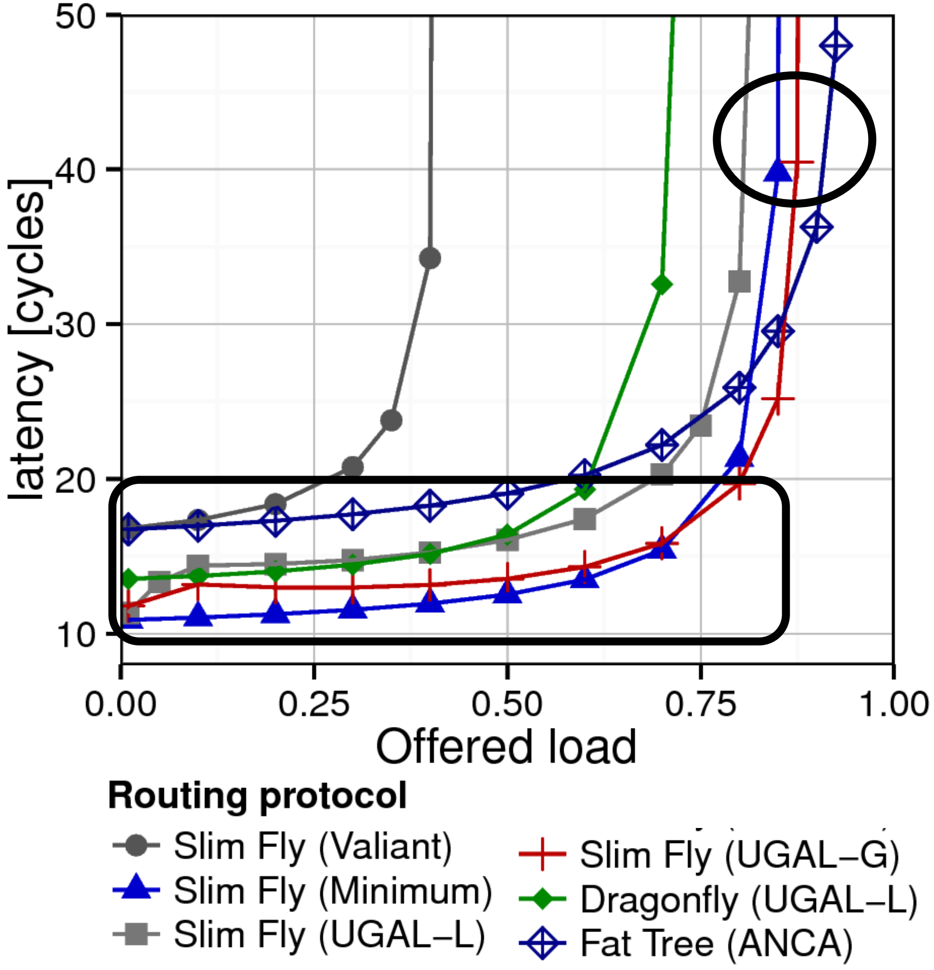
Topology	Fat tree	Random	Flat. Butterfly	Dragonfly	**Slim Fly**
Endpoints (N)	**10,718**	**9,702**	**10,000**	**9,702**	**10,830**
Routers (N_r)	1,531	1,386	1,000	1,386	**722**
Radix (k)	35	28	33	27	**43**
Electric cables	7,350	6,837	4,500	9,009	**6,669**
Fiber cables	24,806	7,716	10,000	4,900	**6,869**
Cost per node [$]	2,315	1,566	1,535	1,342	**1,033**
Power per node [W]	14.0	11.2	10.8	10.8	**8.02**

PERFORMANCE & ROUTING

RANDOM UNIFORM TRAFFIC





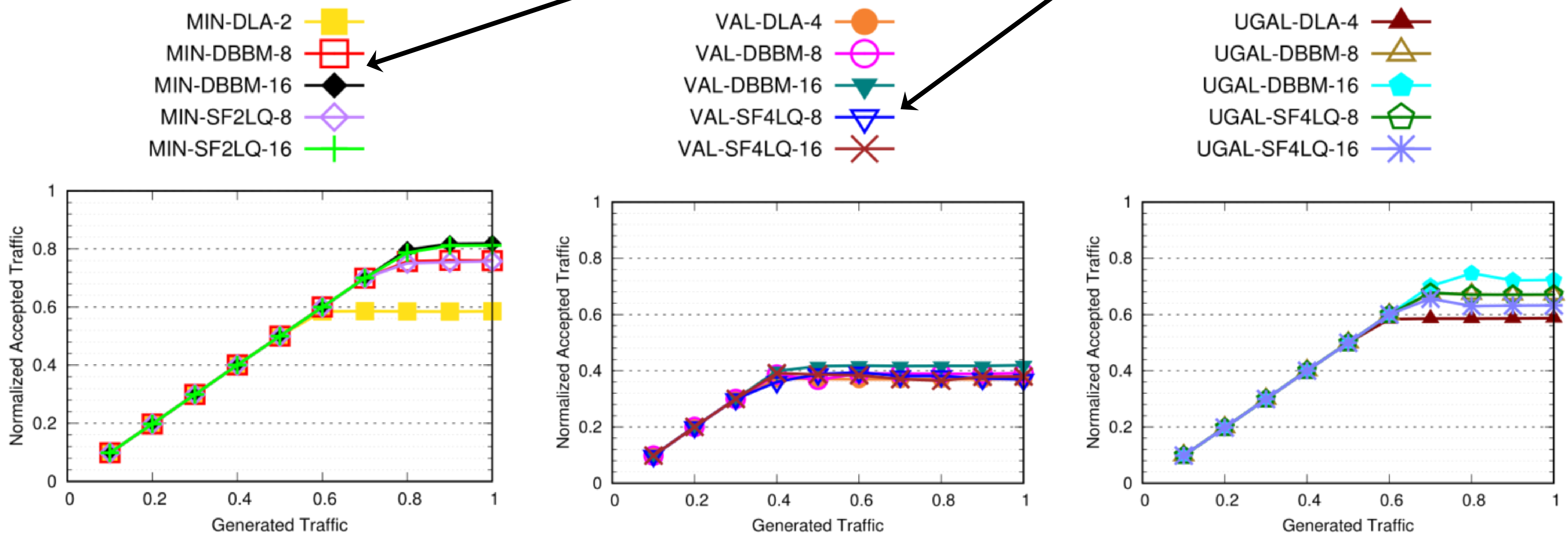


Tuning VNs and VCs to avoid HoL blocking – uniform traffic (no VOQ)

Simple 2VC deadlock avoidance

Destination-based buffer mgmt.

SF-optimized 4 VN and 8 VC



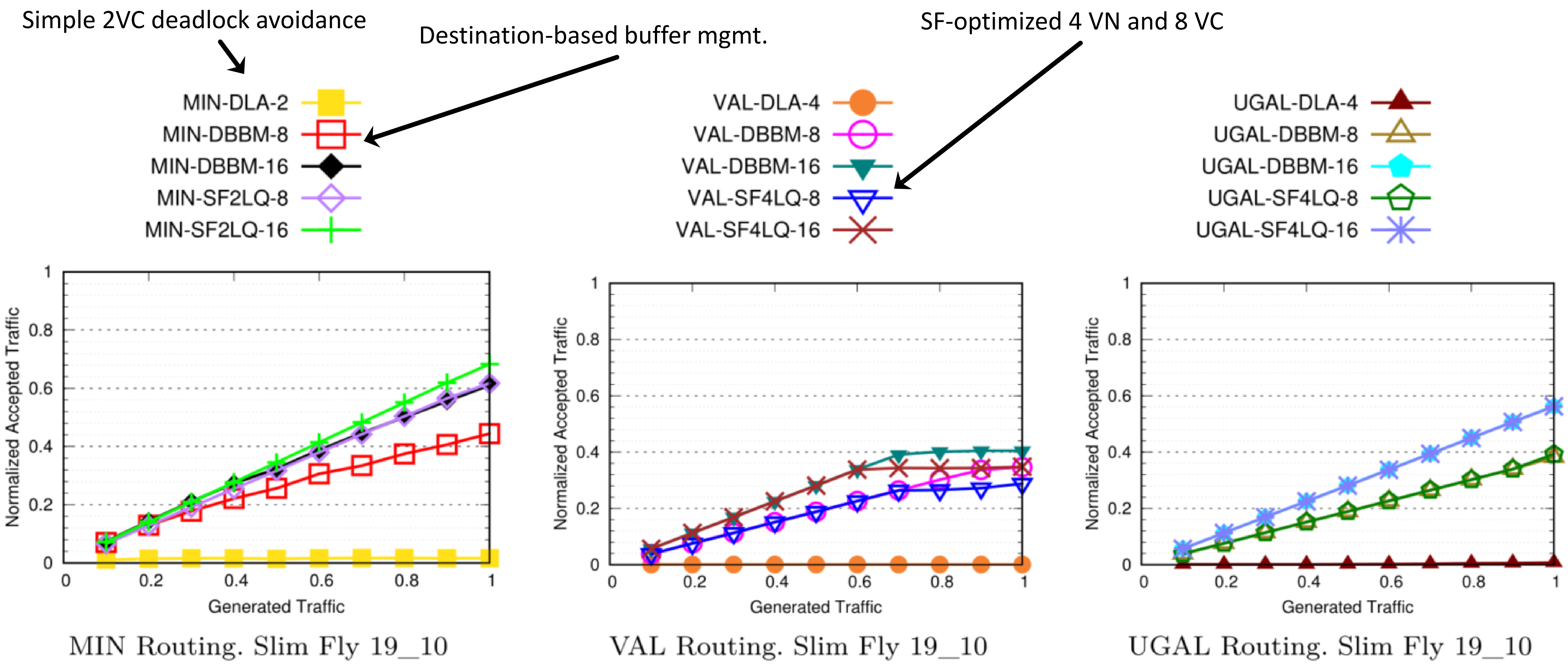
MIN Routing. Slim Fly 19_10

VAL Routing. Slim Fly 19_10

UGAL Routing. Slim Fly 19_10

Pedro Yébenes, Jesus Escudero-Sahuquillo, Pedro J. García, Francisco J. Quiles, TH: Head-of-Line Blocking Avoidance in Slim Fly networks using Deadlock-free Non-minimal and Adaptive Routing, CCPE, June 2016

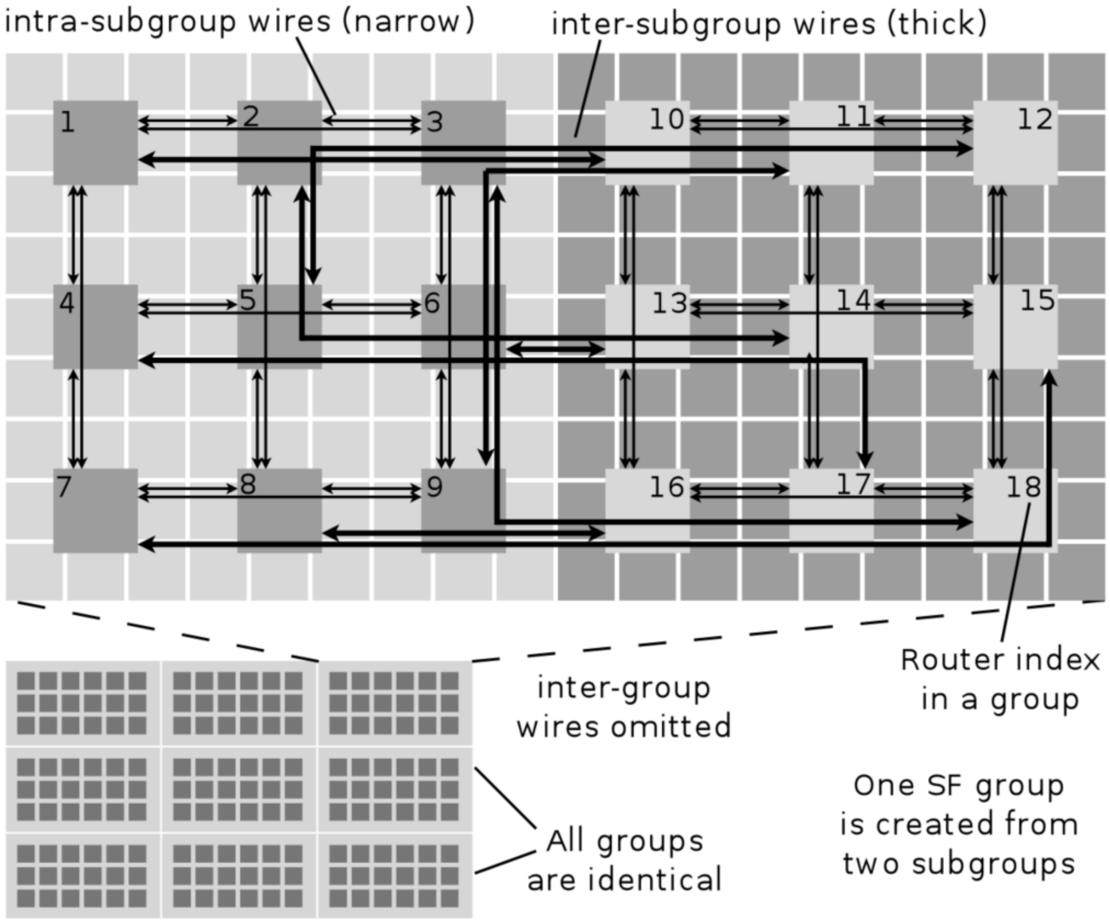
Tuning VNs and VCs to avoid HoL blocking – hotspot traffic (no VOQ)



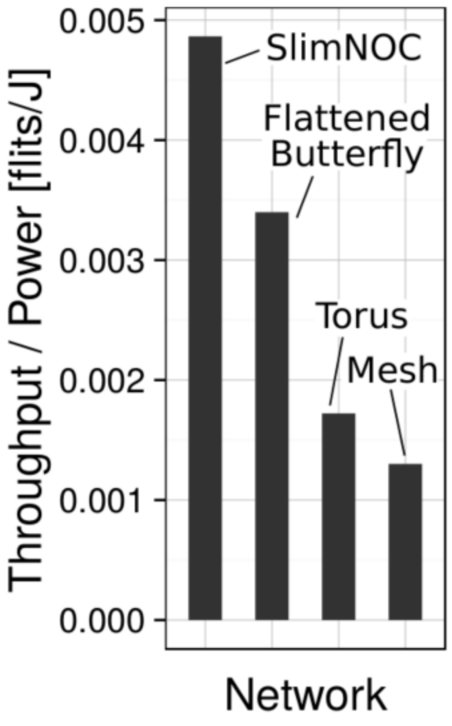
Pedro Yébenes, Jesus Escudero-Sahuquillo, Pedro J. García, Francisco J. Quiles, TH: Head-of-Line Blocking Avoidance in Slim Fly networks using Deadlock-free Non-minimal and Adaptive Routing, CCPE, June 2016

Slim NoC – Slim Fly topologies for on chip networks

- **New challenges – layout in the chip's metal layers**



1296 cores, 162 routers

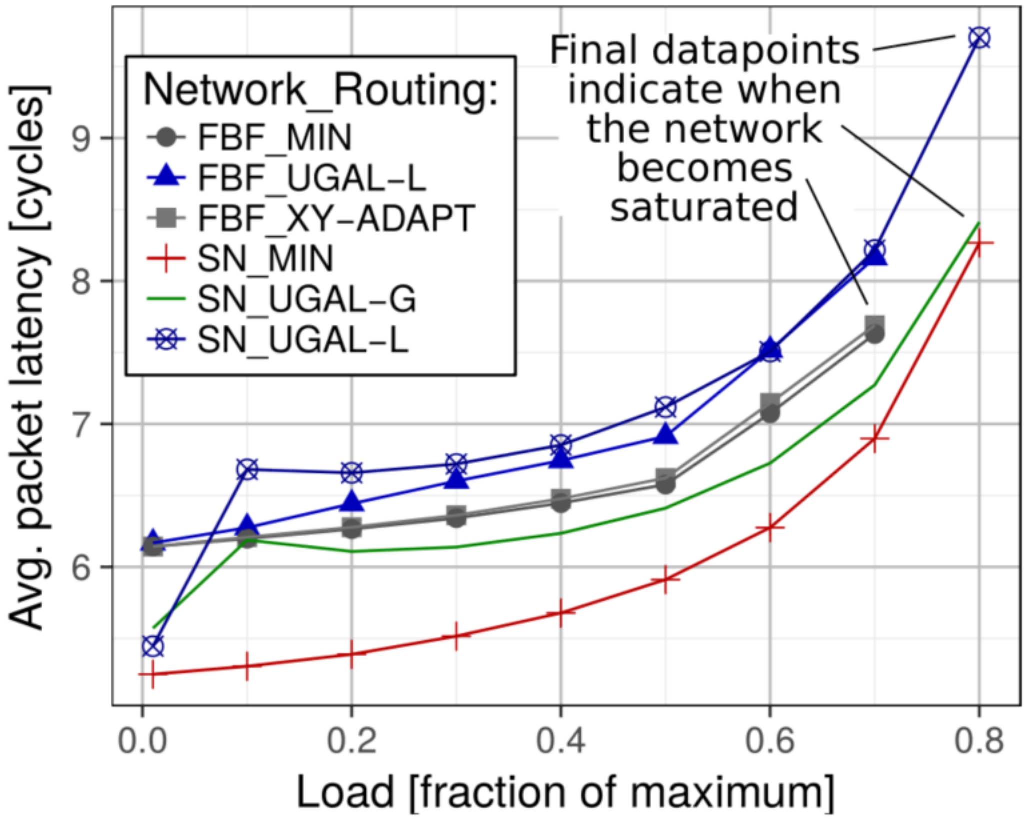


energy efficiency in 22 nm

Maciej Besta, Minhaj Hassan, Sudhakar Yalamanchili, Rachata Ausavarungnirun, Onur Mutlu, TH, Slim NoC: A Low-Diameter On-Chip Network Topology for High Energy-Efficiency and Scalability. ASPLOS'18

Slim NoC performance

uniform random load
Flattened Butterfly vs. Slim Fly



Energy-Delay Product
PARSEC/SPLASH

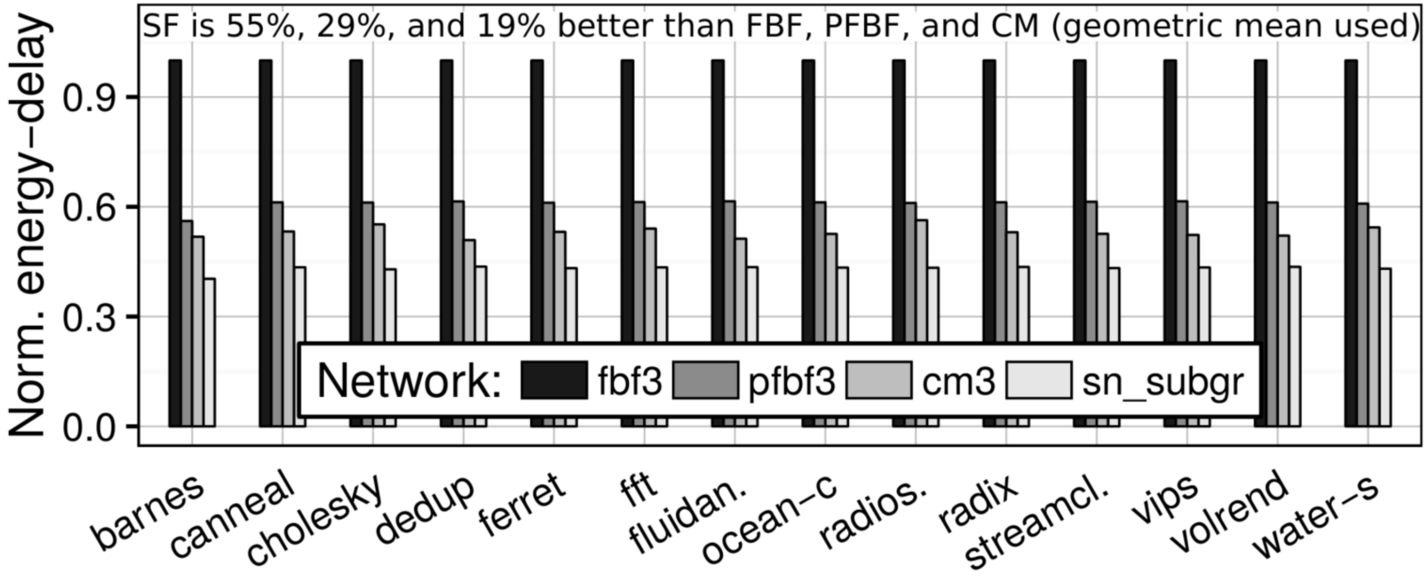


Figure Legend:
FBF/PFBF – Flattened Butterfly
CM – Concentrated Mesh
SN_SUBGR – Slim NOC

Maciej Besta, Minhaj Hassan, Sudhakar Yalamanchili, Rachata Ausavarungnirun, Onur Mutlu, TH, Slim NoC: A Low-Diameter On-Chip Network Topology for High Energy-Efficiency and Scalability. ASPLOS'18

Low Latency

Low Cost

Low Processing Load

The Development of High-Performance Networking Interfaces

Ethernet+TCP/IP — Scalable Coherent Interface — Fast Messages — Myrinet GM+MX — Quadrics QsNet — Virtual Interface Architecture — Cray Gemini — IB Verbs — OFED — Portals 4 — libfabric

Quadrics A Finmeccanica Company — InfiniBand Trade Association — OpenFabrics Alliance — portals

1980 — 1990 — 2000 — 2010 — 2020

sockets — coherent memory access — (active) message based — OS bypass — protocol offload — zero copy — remote direct memory access (RDMA) — triggered operations

InfiniBand Trade Association Launches the RoCE Initiative to Advance RDMA over Converged Ethernet Solutions

RoCE

RoCE delivers significant performance and efficiency gains to cloud, storage, virtualization and hyper-converged infrastructures

businessinsider.com

Microsoft to Drive RDMA Into Datacenters and Clouds

November 18, 2013 by Timothy Prickett Morgan

RDMA over Ethernet - the Rocky road to convergence

17 November 2015 | By Brandon Hoff

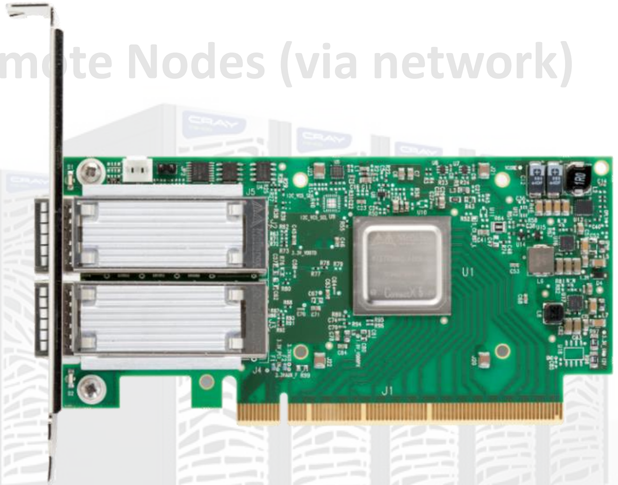
TOP500® SUPERCOMPUTER SITES

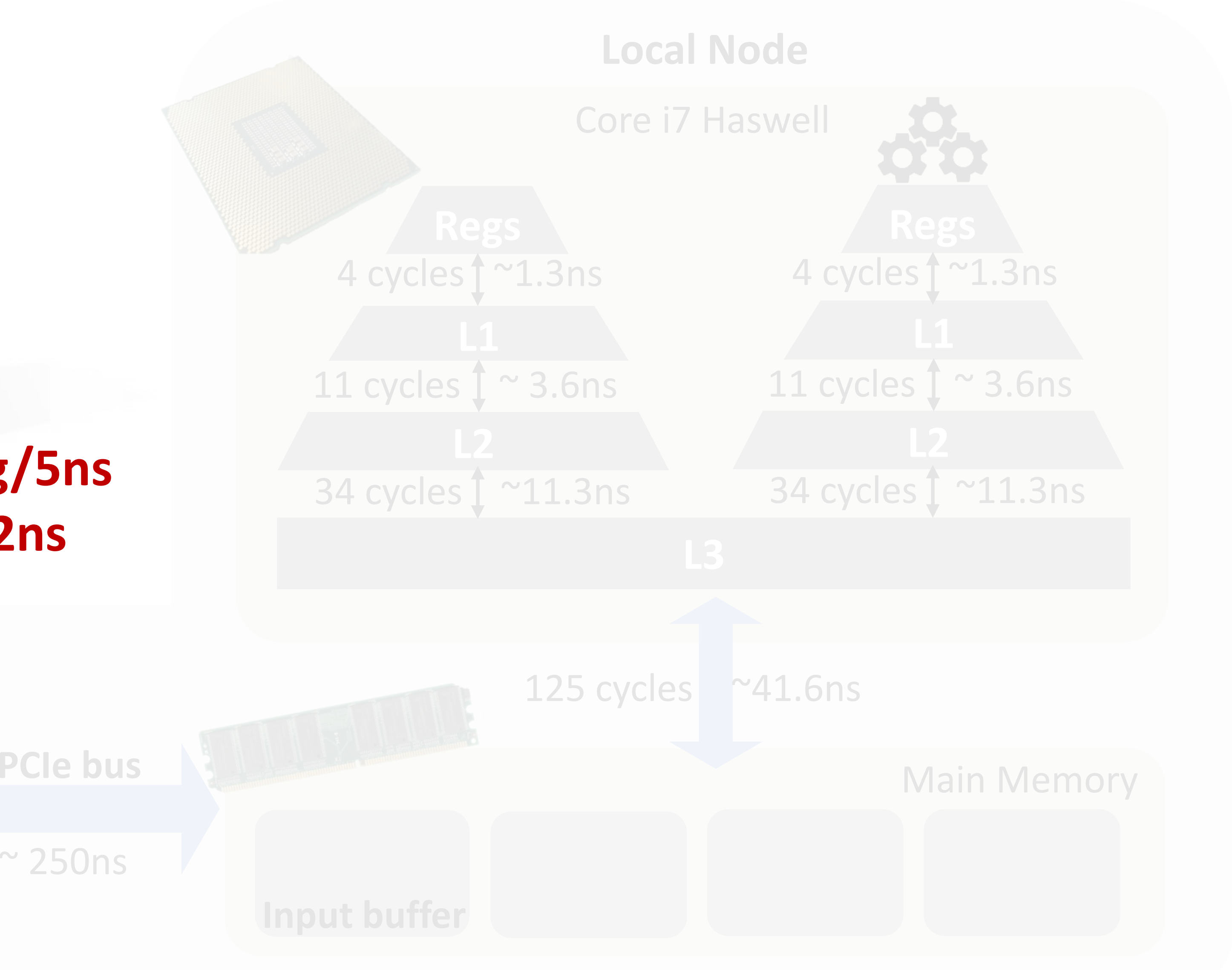
June 2017

95 / top-100 systems use RDMA

>285 / top-500 systems use RDMA

Data Processing in modern RDMA networks





Mellanox Connect-X5: 1 msg/5ns
Tomorrow (400G): 1 msg/1.2ns

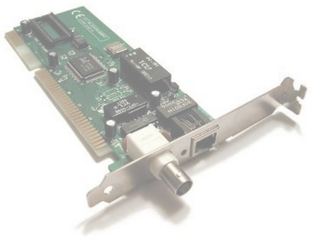
T. Hoefler, S. Di Girolamo, K. Taranov, R. E. Grant, R. Brightwell: sPIN: High-performance streaming Processing in the Network, ACM/IEEE Supercomputing 2018, SC18

The future of High-Performance Networking Interfaces

sPIN
Streaming Processing In the Network

Scalable Coherent Interface   Myrinet GM+MX   Virtual Interface Architecture   OFED   libfabric

Ethernet+TCP/IP   Fast Messages   Quadrics QsNet   Cray Gemini   IB Verbs   Portals 4



Quadrics A Finmeccanica Company   INFINIBAND TRADE ASSOCIATION   OPENFABRICS ALLIANCE   portals

1980   1990   2000   2010   2020

sockets   (active) message based   protocol offload   remote direct memory access (RDMA)

coherent memory access   OS bypass   zero copy   triggered operations

fully programmable NIC acceleration

Established Principles for Compute Acceleration

Specialization   Programmability   Libraries

Ease-of-use   Portability   Efficiency

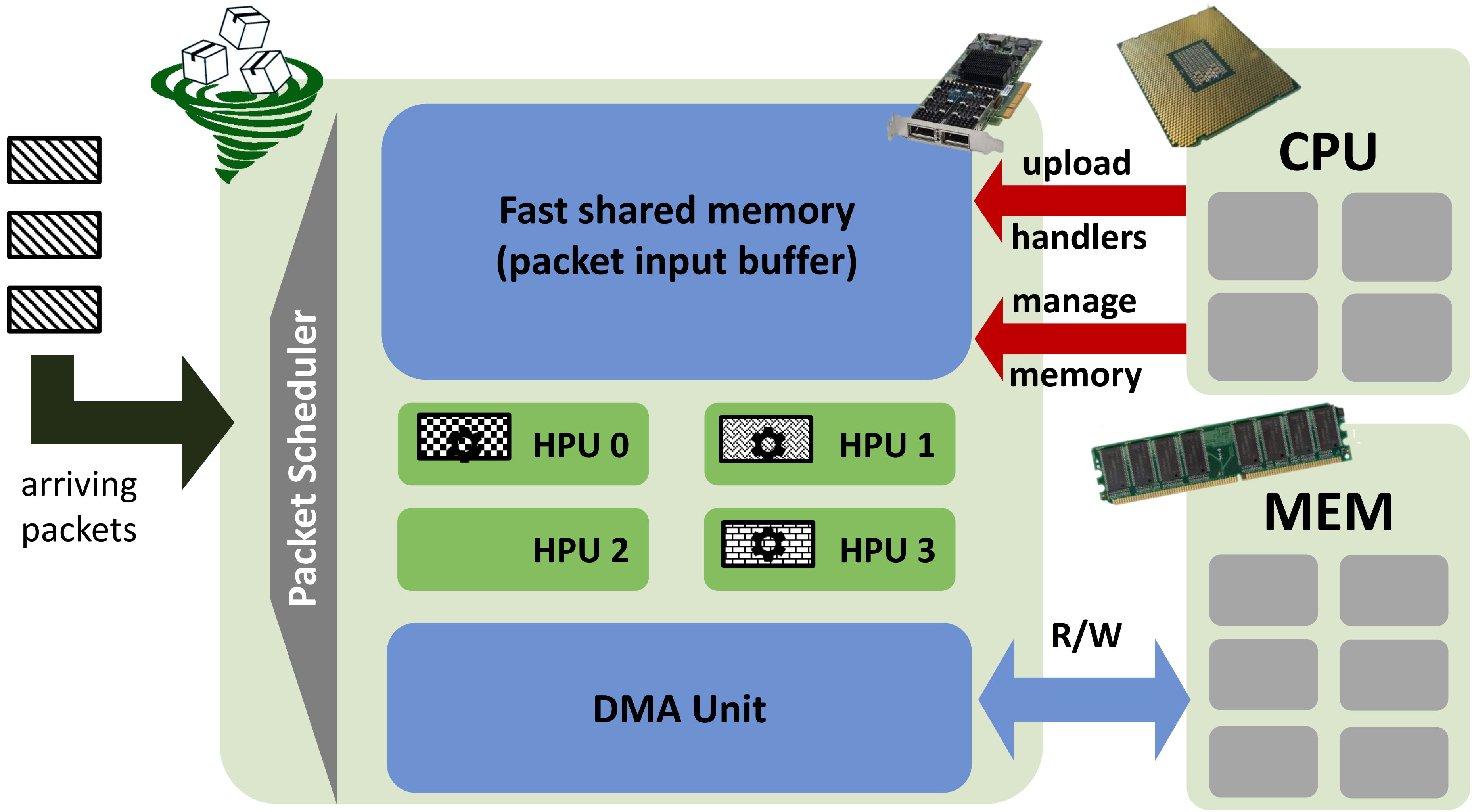
NVIDIA CUDA   OpenCL   OpenMP 4.0

TOP500 SUPERCOMPUTER SITES   June 2017

95 / top-100 systems use RDMA

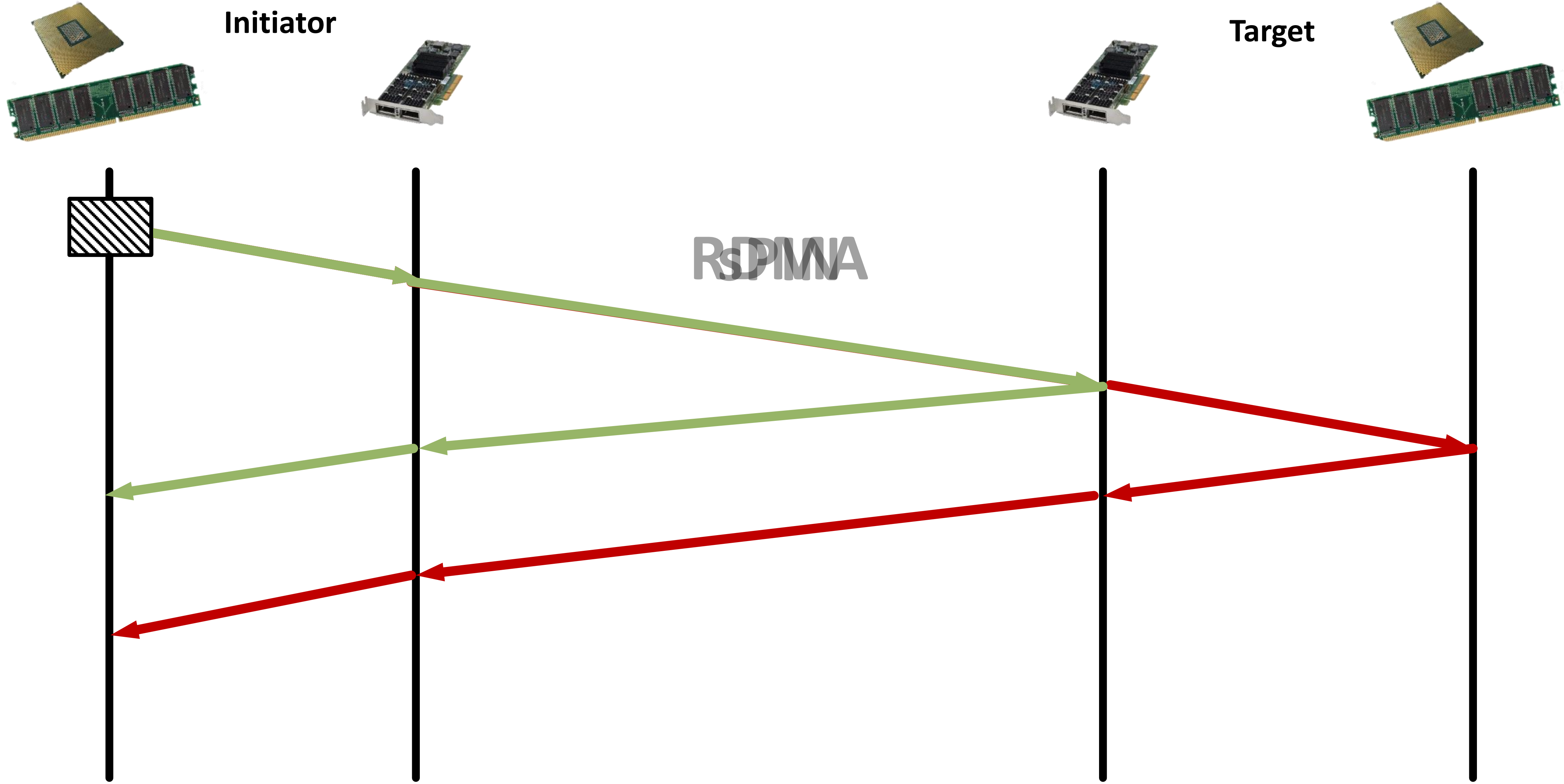
>285 / top-500 systems use RDMA

sPIN NIC - Abstract Machine Model



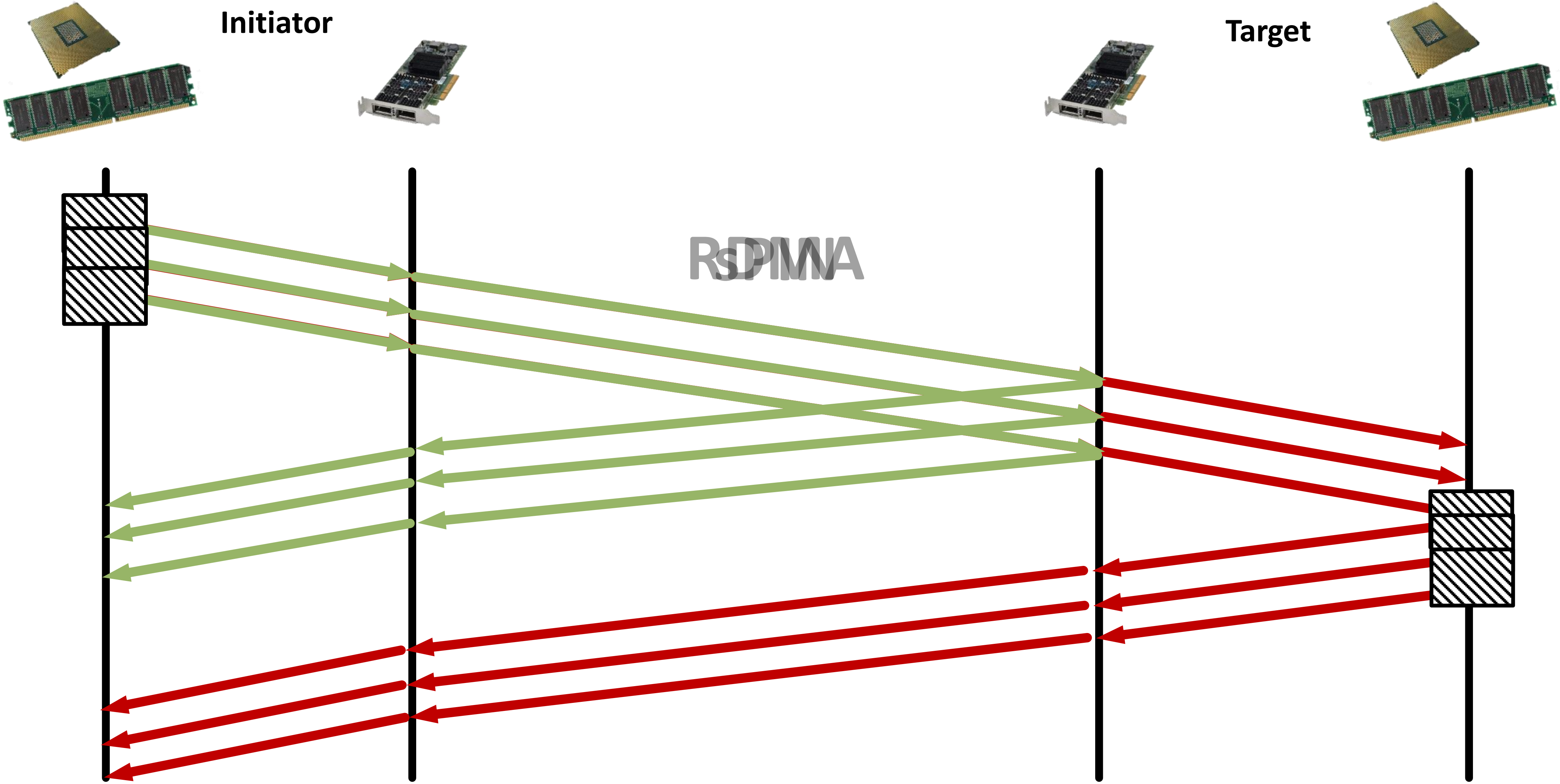
T. Hoefler, S. Di Girolamo, K. Taranov, R. E. Grant, R. Brightwell: sPIN: High-performance streaming Processing in the Network, ACM/IEEE Supercomputing 2018, SC18

RDMA vs. sPIN in action: Simple Ping Pong



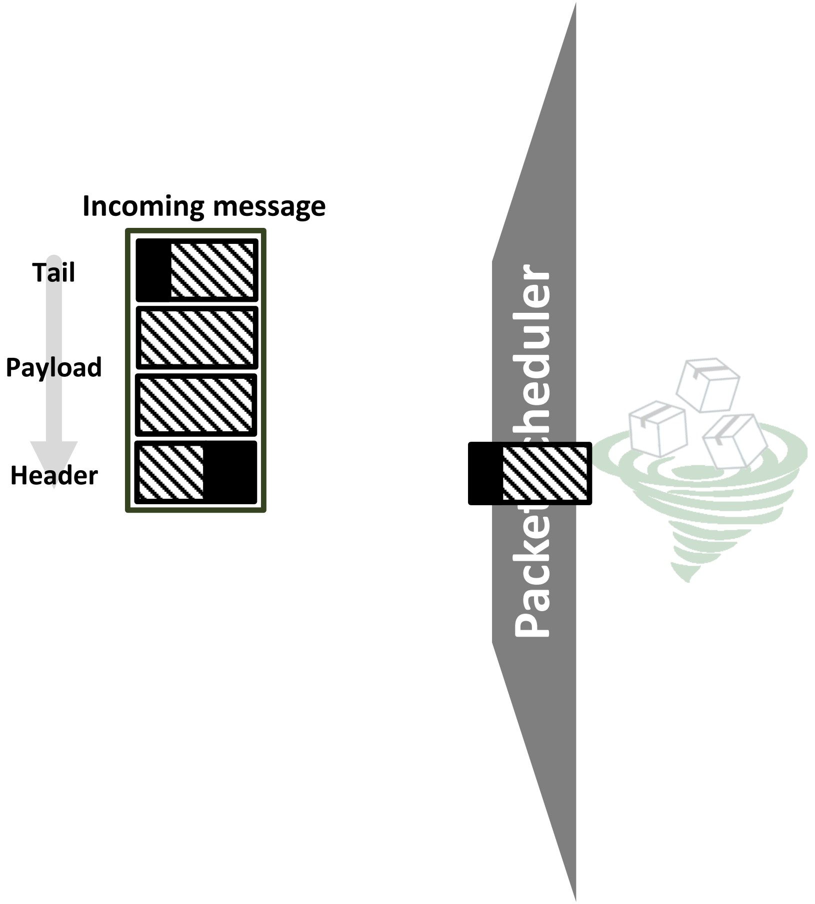
T. Hoefler, S. Di Girolamo, K. Taranov, R. E. Grant, R. Brightwell: sPIN: High-performance streaming Processing in the Network, ACM/IEEE Supercomputing 2018, SC18

RDMA vs. sPIN in action: Streaming Ping Pong



T. Hoefler, S. Di Girolamo, K. Taranov, R. E. Grant, R. Brightwell: sPIN: High-performance streaming Processing in the Network, ACM/IEEE Supercomputing 2018, SC18

sPIN – Programming Interface



Header handler

```
__handler int pp_header_handler(const ptl_header_t h, void *state) {
   pingpong_info_t *i = state;
   i->source = h.source_id;
   return PROCESS_DATA; // execute payload handler to put from device
}
```

Payload handler

```
__handler int pp_payload_handler(const ptl_payload_t p, void * state) {
   pingpong_info_t *i = state;
   PtlHandlerPutFromDevice(p.base, p.length, 1, 0, i->source, 10, 0, NULL, 0);
   return SUCCESS;
}
```

Completion handler

```
__handler int pp_completion_handler(int dropped_bytes,
                                     bool flow_control_triggered, void *state) {
   return SUCCESS;
}
```

```
connect(peer, /* … */, &pp_header_handler, &pp_payload_handler, &pp_completion_handler);
```

T. Hoefler, S. Di Girolamo, K. Taranov, R. E. Grant, R. Brightwell: sPIN: High-performance streaming Processing in the Network, ACM/IEEE Supercomputing 2018, SC18

Possible sPIN implementations

- **sPIN is a programming abstraction, similar to CUDA or OpenCL combined with OFED or Portals 4**
  - It enables a large variety of NIC implementations!
  - For example, massively multithreaded HPUs
    Including warp-like scheduling strategies
- **Main goal: sPIN must not obstruct line-rate**
  - Programmer must limit processing time per packet
    Little's Law: 500 instructions per handler, 2.5 GHz, IPC=1, 1 Tb/s → 25 kiB memory
  - Relies on fast shared memory (processing in packet buffers)
    Scratchpad or registers
  - Quick (single-cycle) handler invocation on packet arrival
    Pre-initialized memory & context
- **Can be implemented in most RDMA NICs with a firmware update**
  - Or in software in programmable (Smart) NICs

at 400G, process more than
833 million messages/s

Mellanox Technologies
Innova Flex
(Kintex FPGA)

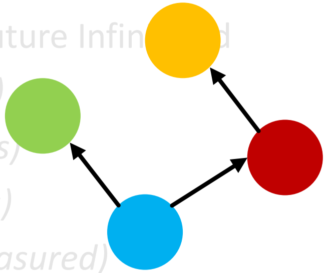
Microsoft
Catapult
(Virtex FPGA)

Broadcom
BCM58800 SoC
(Full Linux)

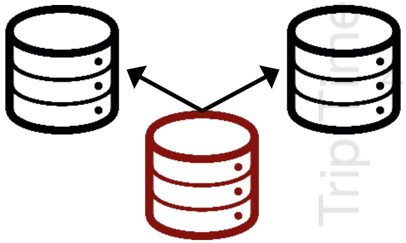
T. Hoefler, S. Di Girolamo, K. Taranov, R. E. Grant, R. Brightwell: sPIN: High-performance streaming Processing in the Network, ACM/IEEE Supercomputing 2018, SC18

Simulating a sPIN NIC – Ping Pong

- LogGOPSim v2 [1]: combine LogGOPSim (packet-level network) with gem5 (cycle accurate CPU simulation)
- Network (LogGOPSim):
  - Supports Portals 4 and MPI
  - Parametrized for future InfiniBand
    - *o=65ns (measured)*
    - *g=6.7ns (150 MM/s)*
    - *G=2.5ps (400 Gib/s)*
    - *Switch L=50ns (measured)*
    - *Wire L=33.4ns (10m cable)*
- NIC HPU
  - 2.5 GHz ARM Cortex A15 OOO
  - ARMv8-A 32 bit ISA
  - Single-cycle access SRAM (no DRAM)
  - Header matching m=30ns, per packet 2ns
    - *In parallel with g!*

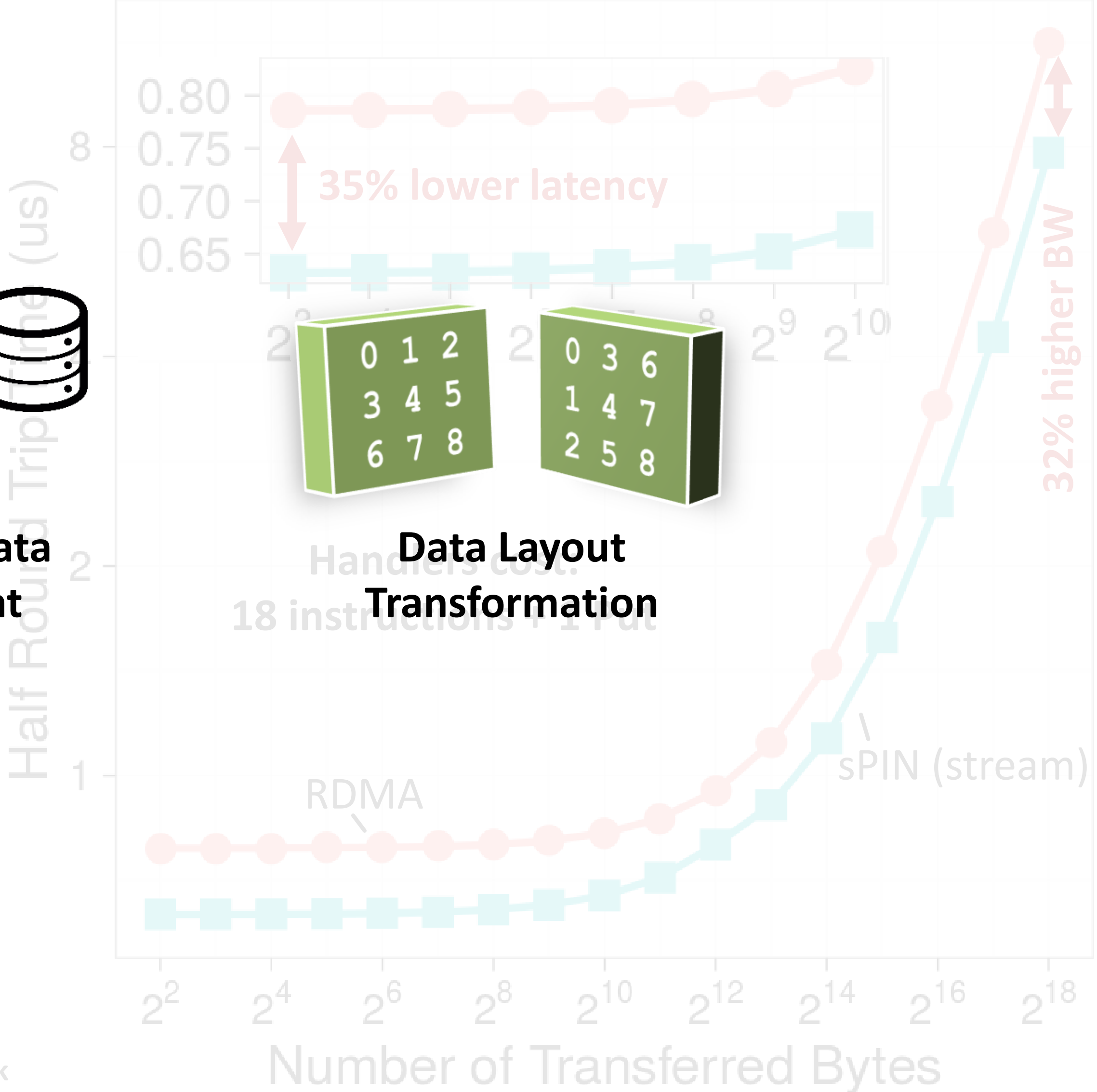


Network Group Communication



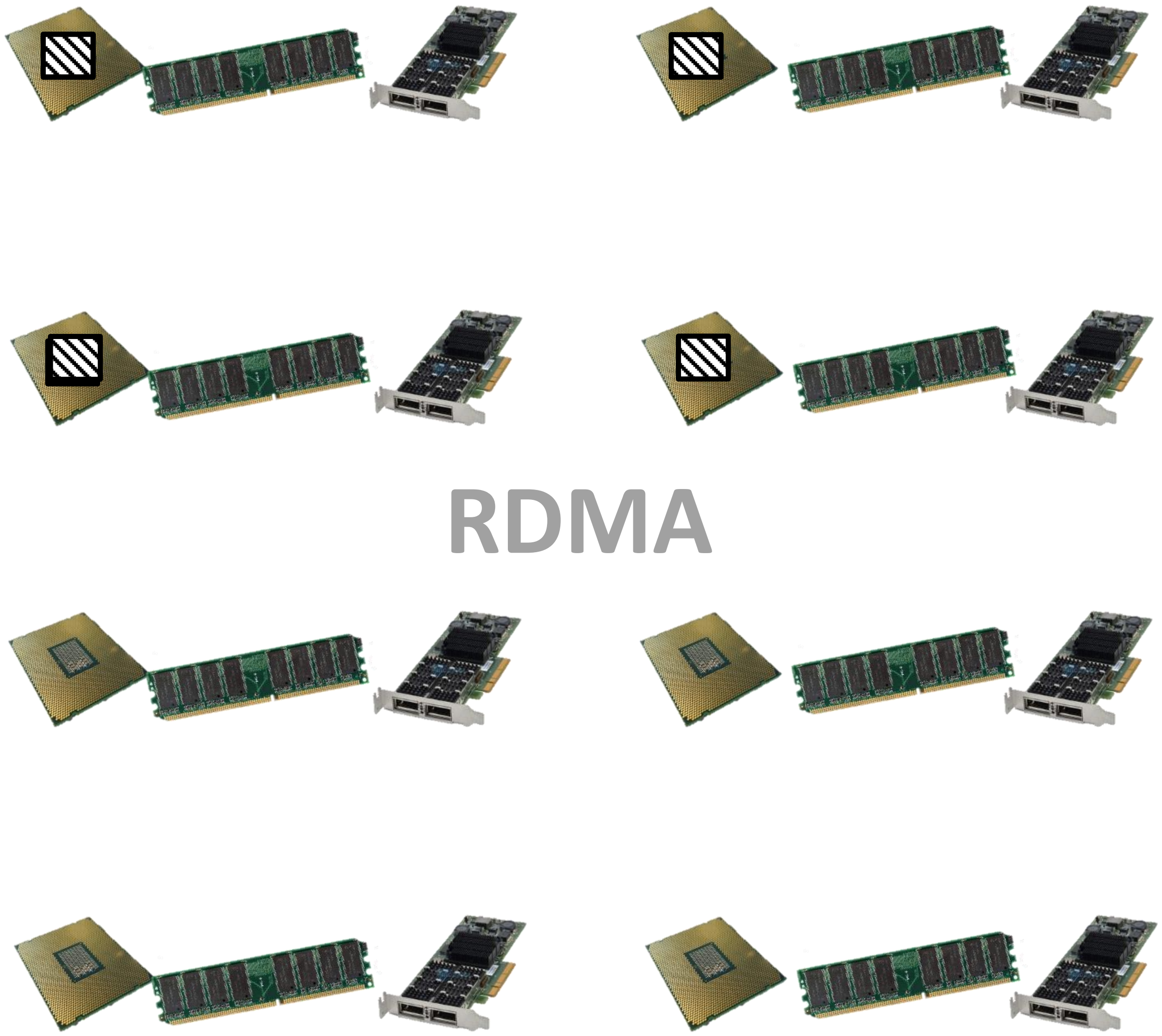
Distributed Data Management

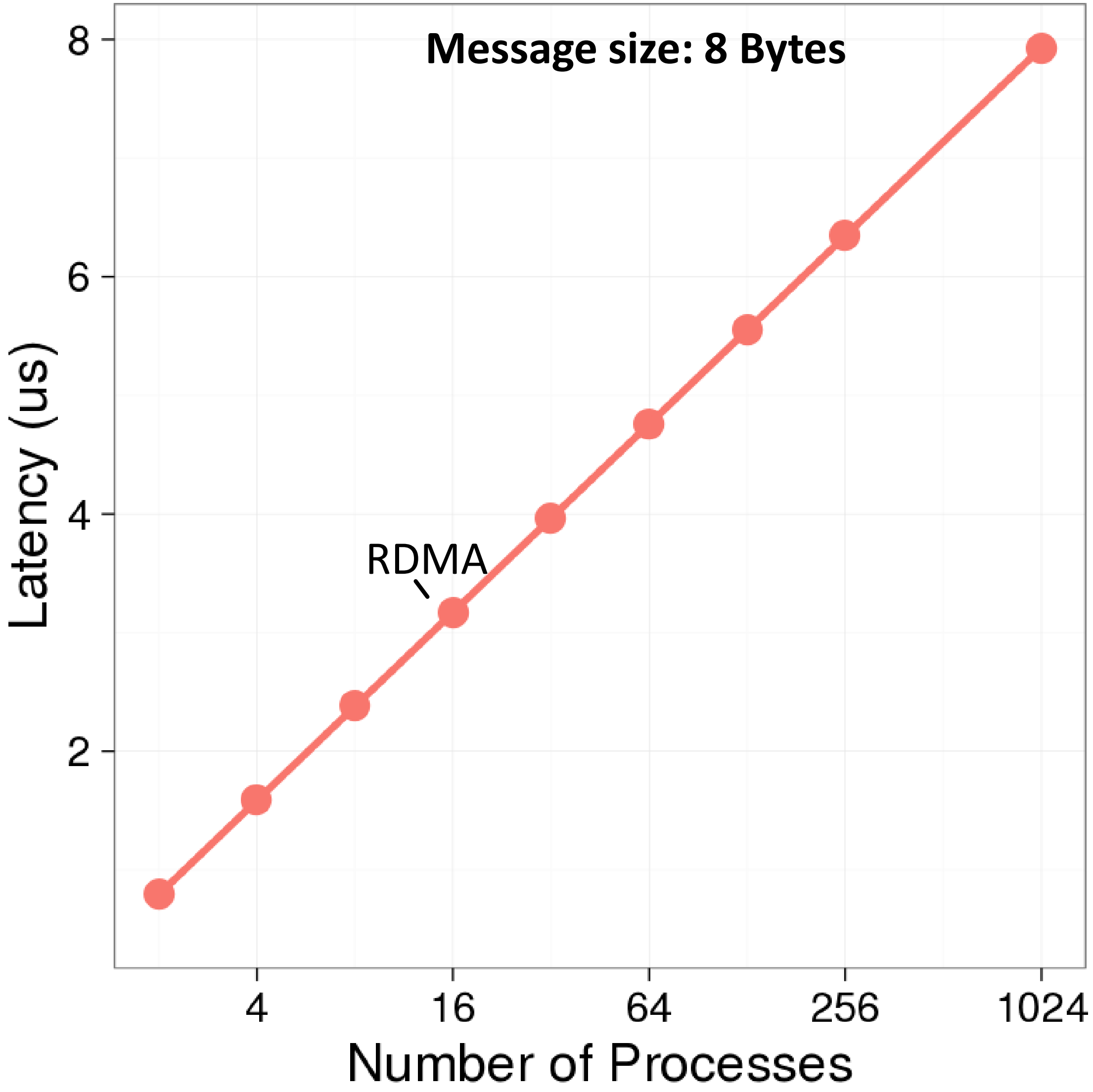
Data Layout Transformation



[1] S. Di Girolamo, K. Taranov, T. Schneider, E. Stalder, T. Hoefler, **LogGOPSim+gem5: Simulating Network Offload Engines Over Packet-Switched Networks**. Presented at ExaMPI'17

Use Case 1: Broadcast acceleration



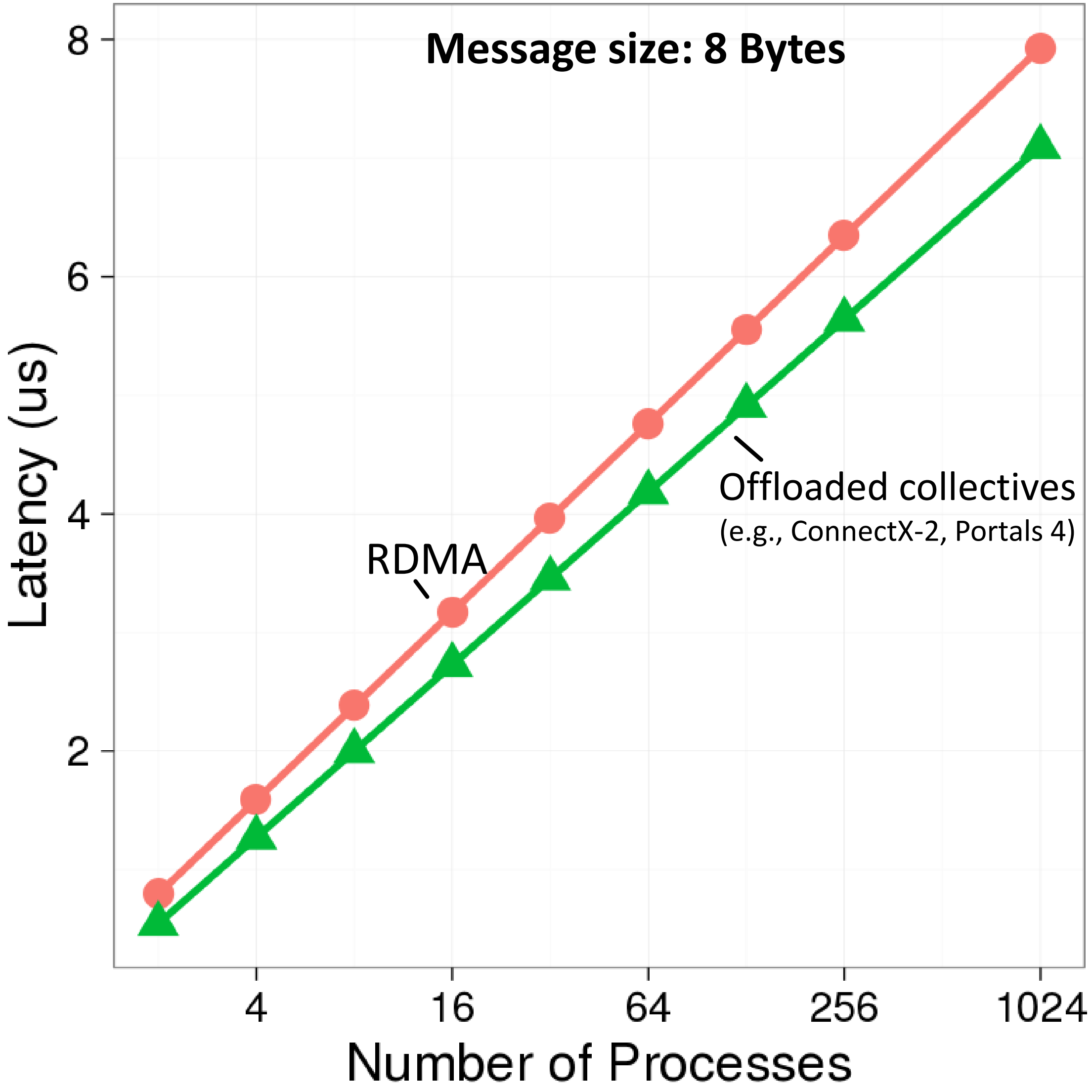


Liu, J., et al., High performance RDMA-based MPI implementation over InfiniBand. *International Journal of Parallel Programming* 2004

Use Case 1: Broadcast acceleration

Network Group Communication

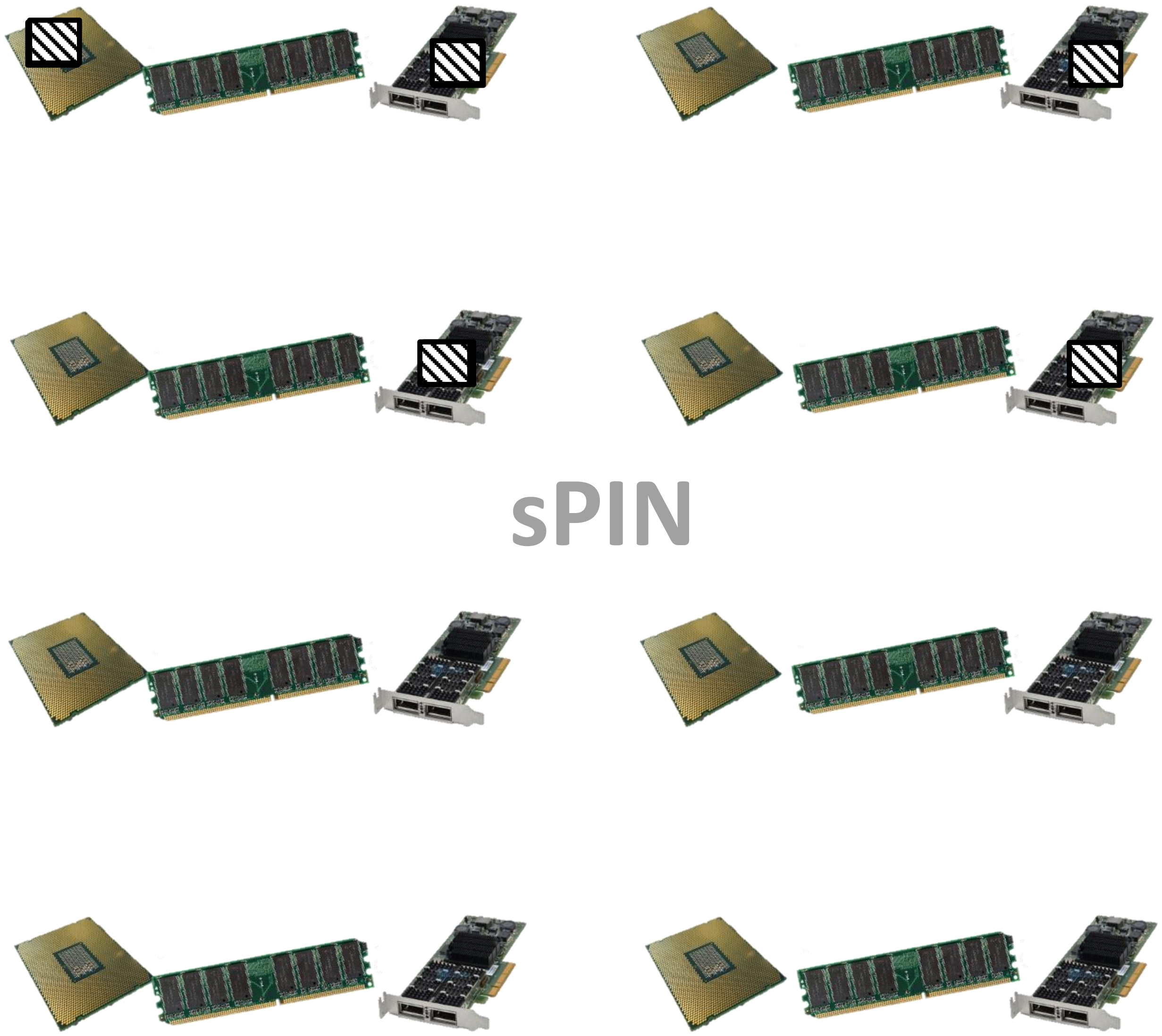
Portals 4

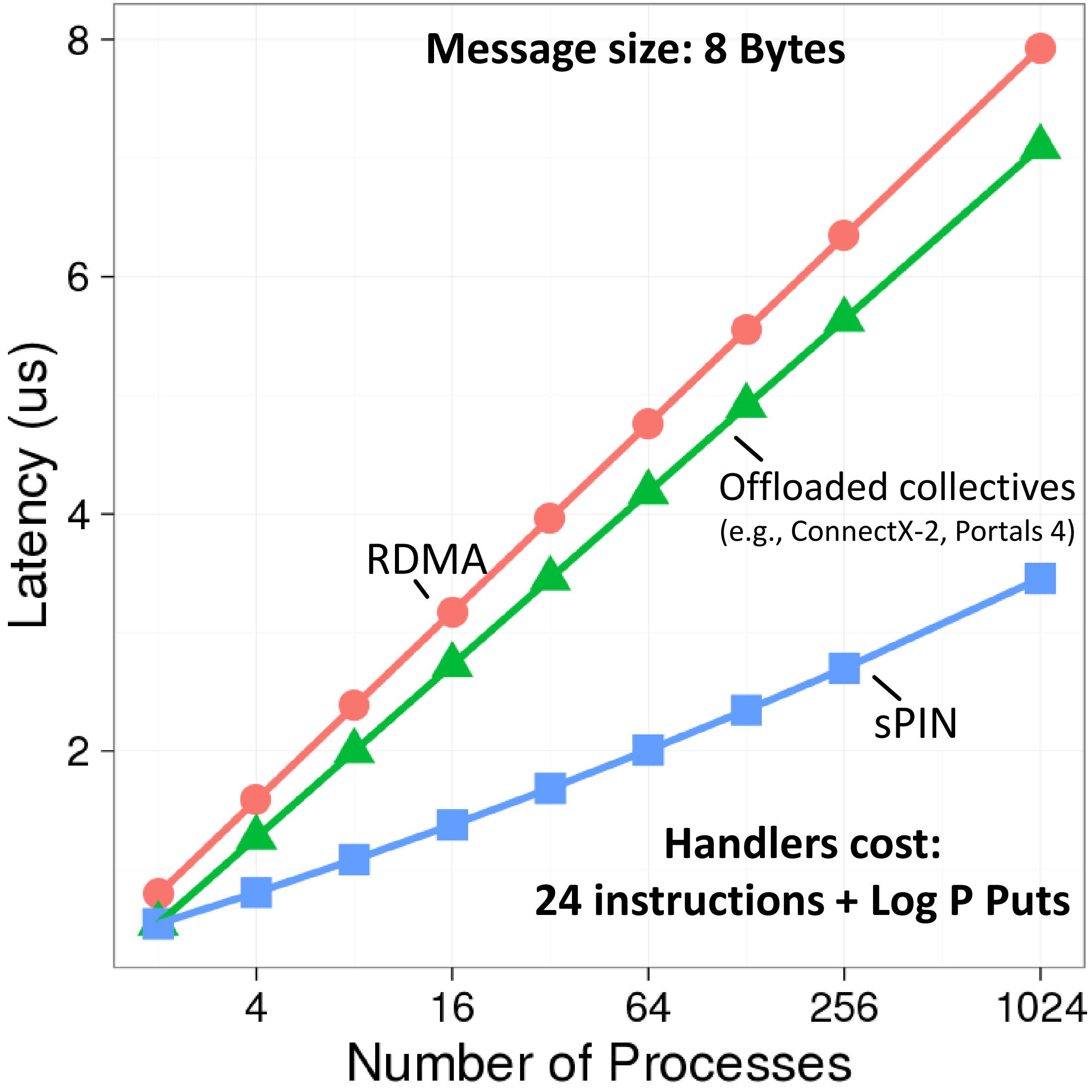


Underwood, K.D., et al., Enabling flexible collective communication offload with triggered operations. *HOTI'11*
Liu, J., et al., High performance RDMA-based MPI implementation over InfiniBand. *International Journal of Parallel Programming* 2004

Use Case 1: Broadcast acceleration

Network Group Communication

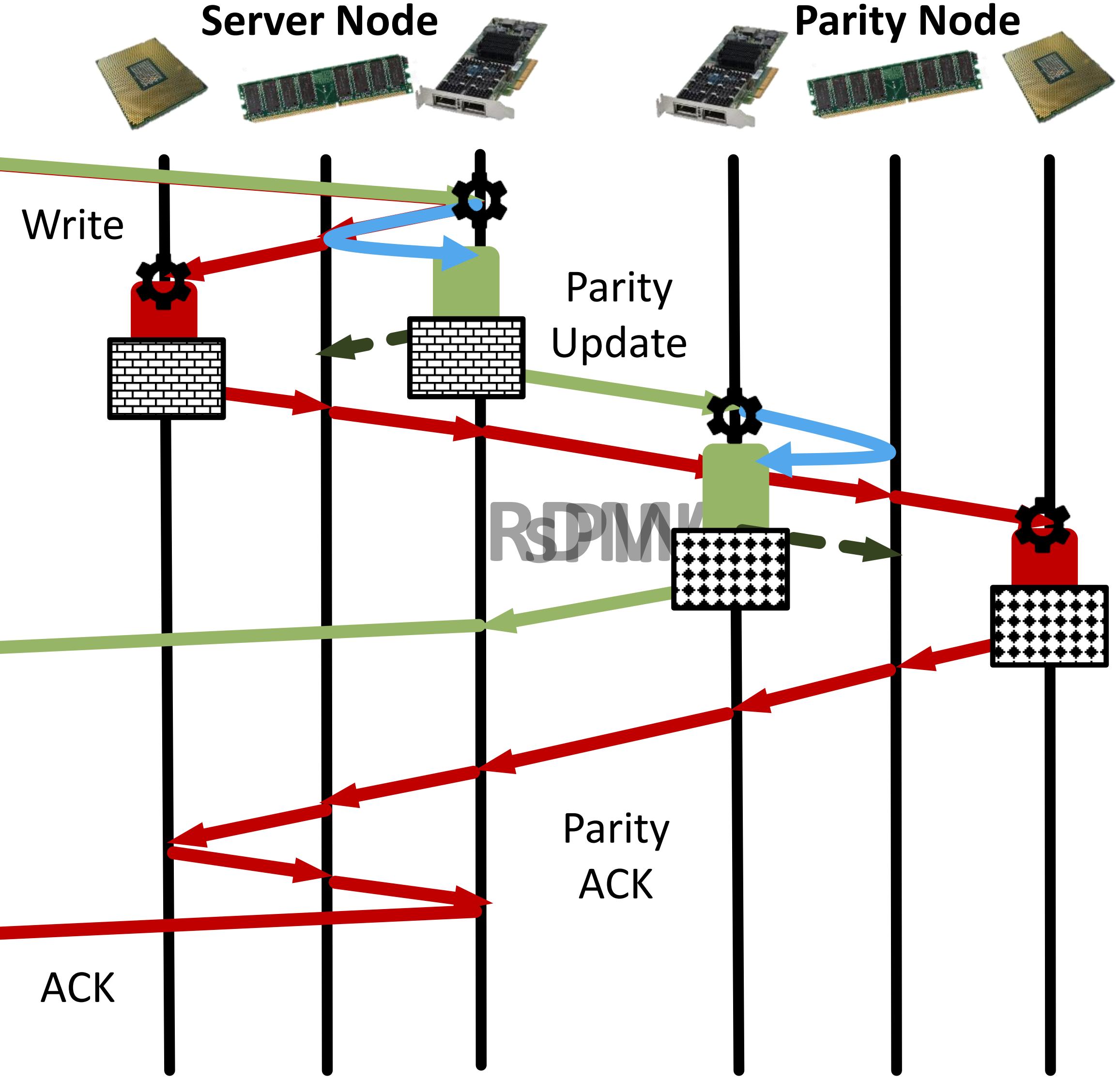


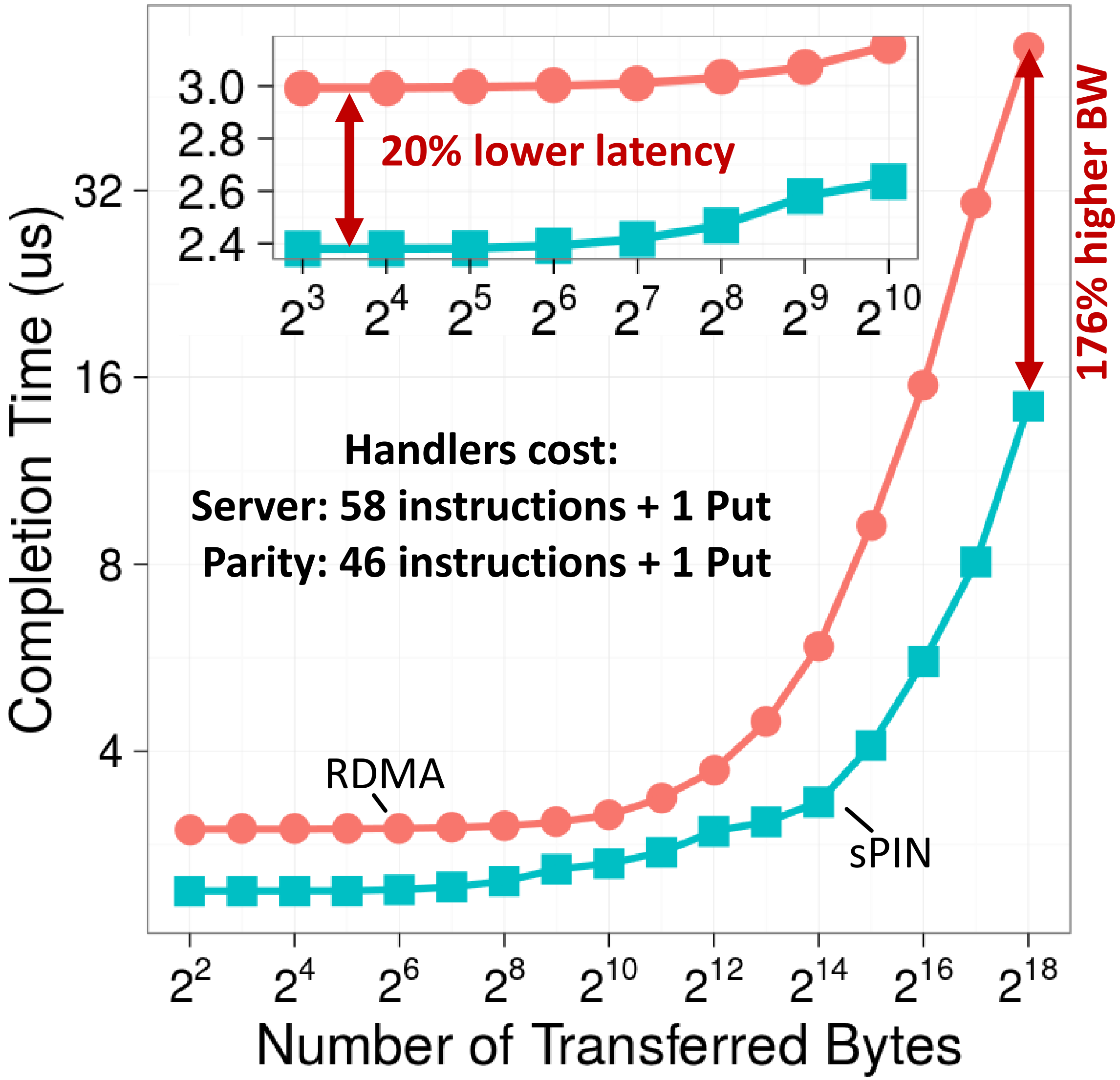


Underwood, K.D., et al., Enabling flexible collective communication offload with triggered operations. *HOTI'11*
Liu, J., et al., High performance RDMA-based MPI implementation over InfiniBand. *International Journal of Parallel Programming* 2004

Use Case 2: RAID acceleration

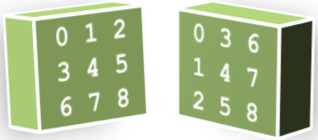
Distributed Data Management



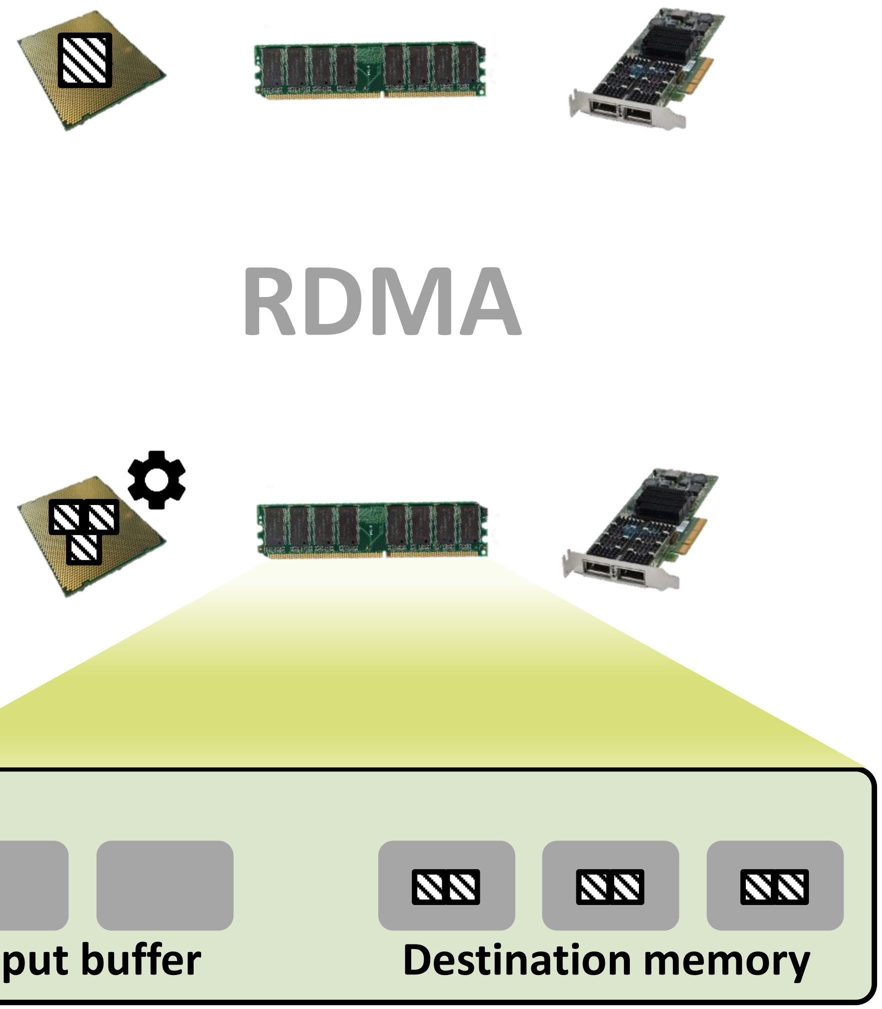


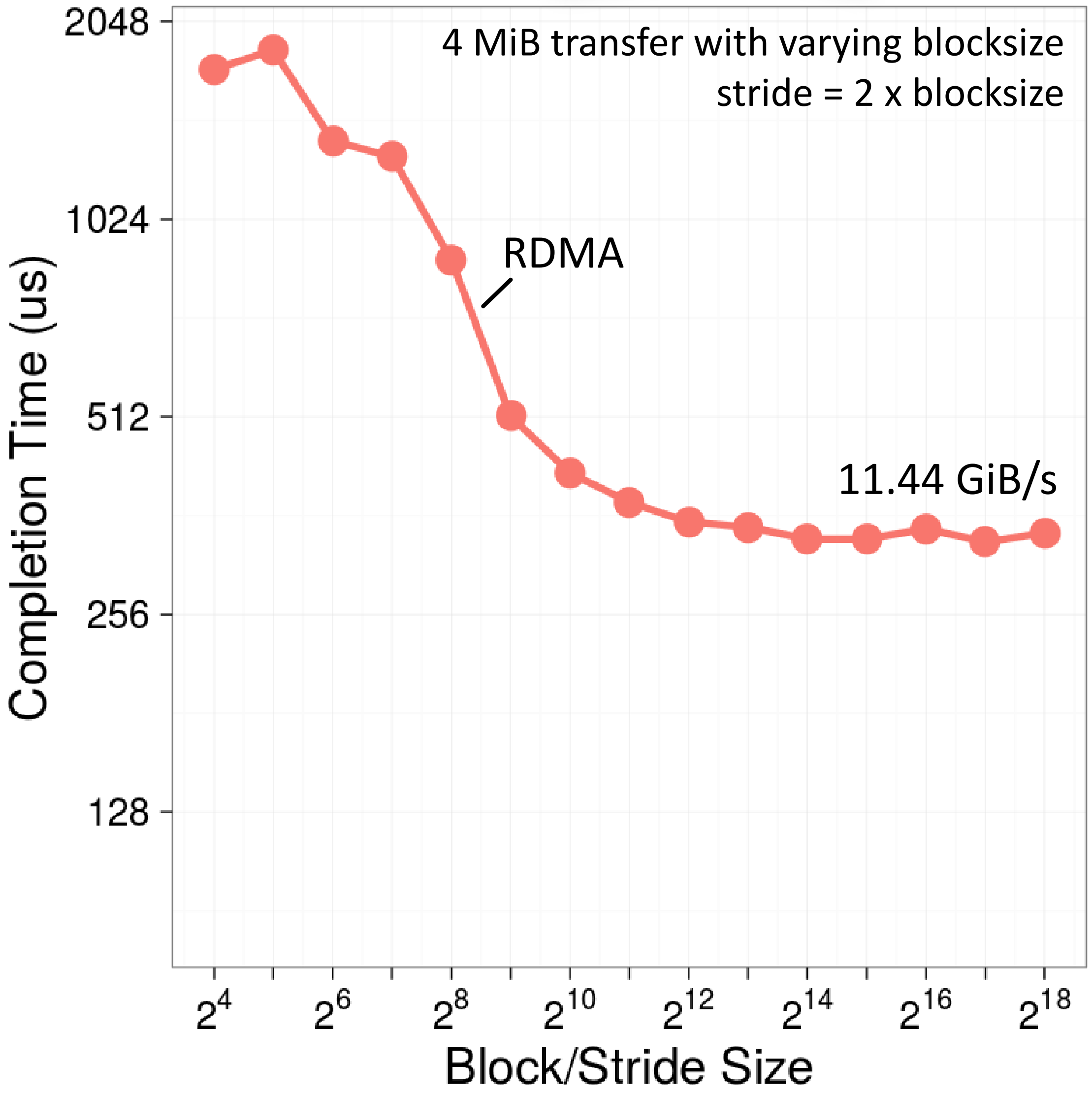
Shankar D. et al., High-performance and Resilient Key-Value Store with Online Erasure Coding for Big Data Workloads. ICDCS'17

Use Case 3: MPI Datatypes acceleration



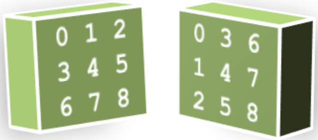
Data Layout Transformation



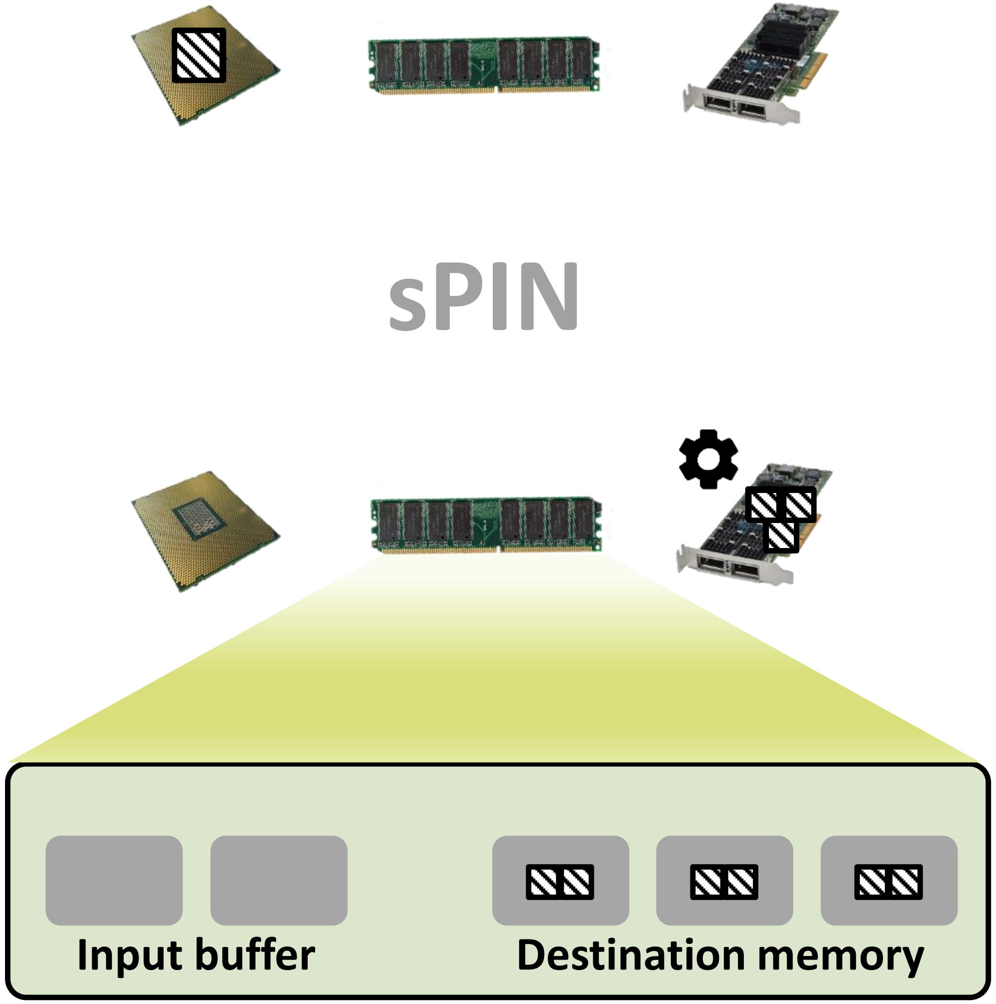


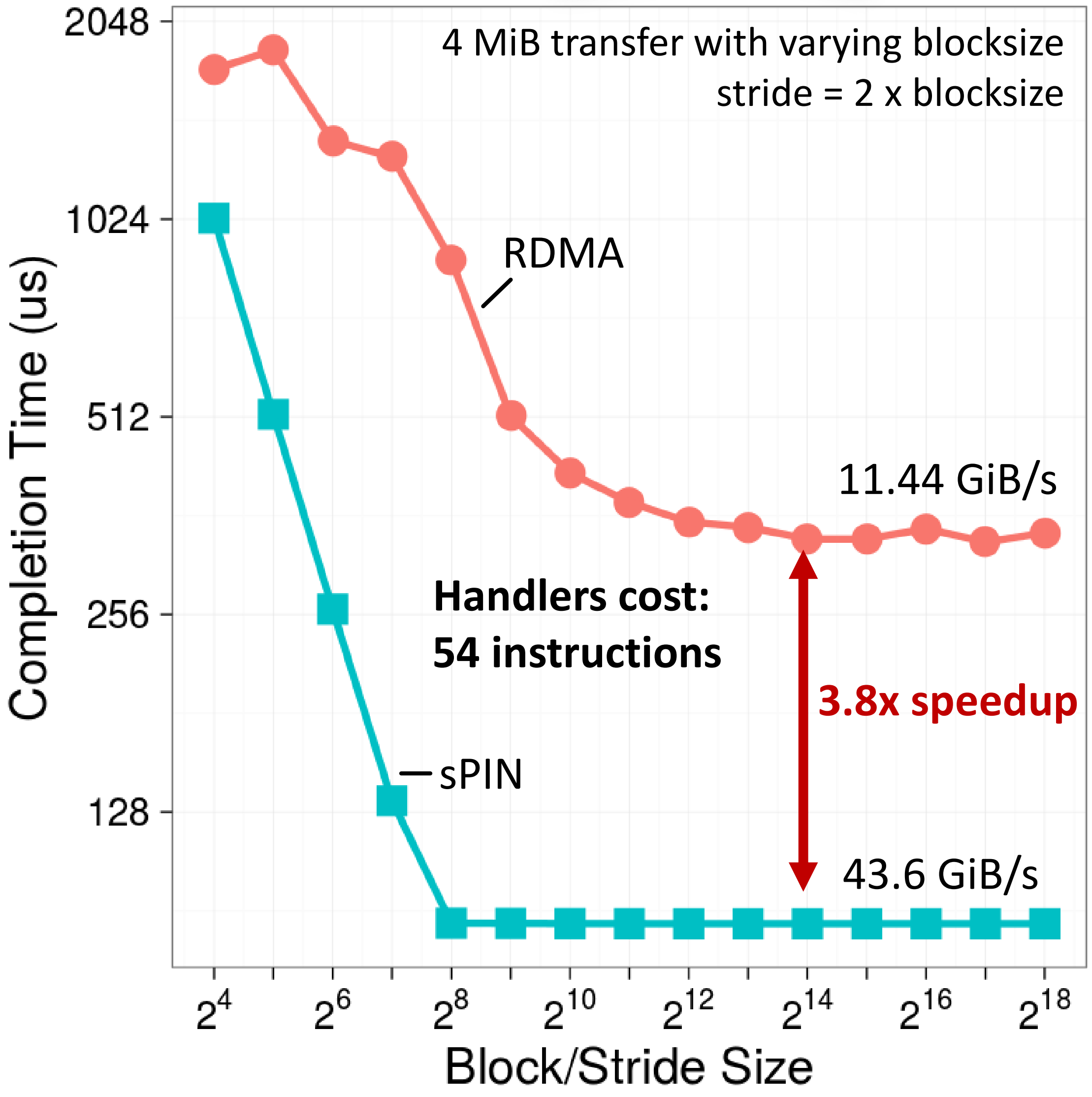
Gropp, W., et al., March. Improving the performance of MPI derived datatypes. *MPIDC'99*

Use Case 3: MPI Datatypes acceleration



Data Layout Transformation





Gropp, W., et al., March. Improving the performance of MPI derived datatypes. *MPIDC'99*

Further results and use-cases

Further results and use-cases

Use Case 4: MPI Rendezvous Protocol

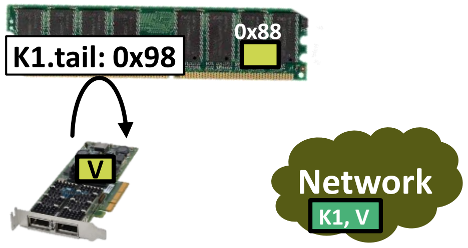
program	p	msgs	ovhd	ovhd	red
MILC	64	5.7M	5.5%	1.9%	65%
POP	64	772M	3.1%	2.4%	22%
coMD	72	5.3M	6.1%	2.4%	60%
coMD	360	28.1M	6.5%	2.8%	58%
Cloverleaf	72	2.7M	5.2%	2.4%	53%
Cloverleaf	360	15.3M	5.6%	3.2%	42%

Further results and use-cases

Use Case 4: MPI Rendezvous Protocol

program	p	msgs	ovhd	ovhd	red
MILC	64	5.7M	5.5%	1.9%	65%
POP	64	772M	3.1%	2.4%	22%
coMD	72	5.3M	6.1%	2.4%	60%
coMD	360	28.1M	6.5%	2.8%	58%
Cloverleaf	72	2.7M	5.2%	2.4%	53%
Cloverleaf	360	15.3M	5.6%	3.2%	42%

Use Case 5: Distributed KV Store



41% lower latency

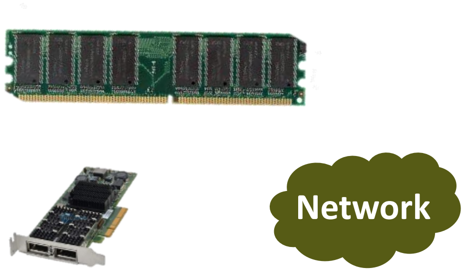
Kalia, A., et al., Using RDMA efficiently for key-value services. In *ACM SIGCOMM Computer Communication Review, 2014*

Further results and use-cases

Use Case 4: MPI Rendezvous Protocol

program	p	msgs	ovhd	ovhd	red
MILC	64	5.7M	5.5%	1.9%	65%
POP	64	772M	3.1%	2.4%	22%
coMD	72	5.3M	6.1%	2.4%	60%
coMD	360	28.1M	6.5%	2.8%	58%
Cloverleaf	72	2.7M	5.2%	2.4%	53%
Cloverleaf	360	15.3M	5.6%	3.2%	42%

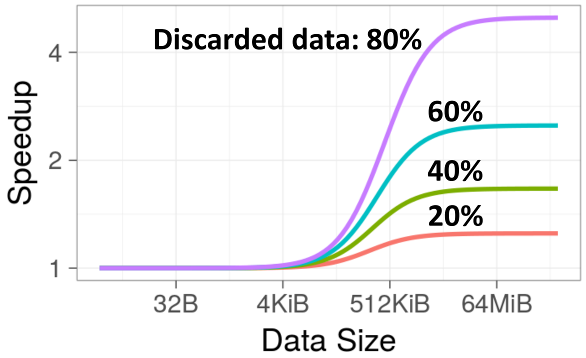
Use Case 5: Distributed KV Store



41% lower latency

Kalia, A., et al., Using RDMA efficiently for key-value services. In *ACM SIGCOMM Computer Communication Review, 2014*

Use Case 6: Conditional Read



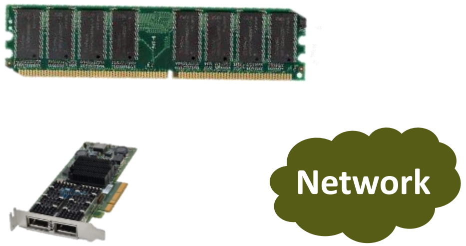
Barthels, C., et al., Designing Databases for Future High-Performance Networks. *IEEE Data Eng. Bulletin, 2017*

Further results and use-cases

Use Case 4: MPI Rendezvous Protocol

program	p	msgs	ovhd	ovhd	red
MILC	64	5.7M	5.5%	1.9%	65%
POP	64	772M	3.1%	2.4%	22%
coMD	72	5.3M	6.1%	2.4%	60%
coMD	360	28.1M	6.5%	2.8%	58%
Cloverleaf	72	2.7M	5.2%	2.4%	53%
Cloverleaf	360	15.3M	5.6%	3.2%	42%

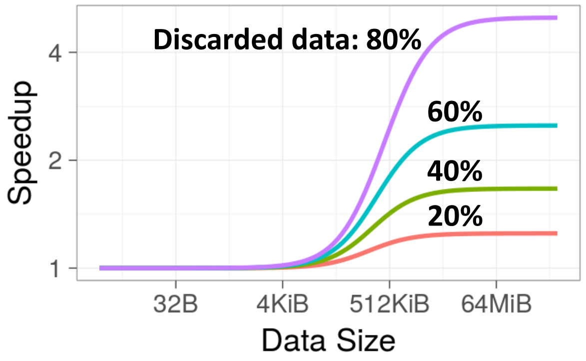
Use Case 5: Distributed KV Store



41% lower latency

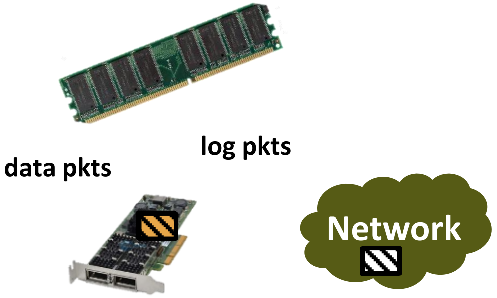
Kalia, A., et al., Using RDMA efficiently for key-value services. In *ACM SIGCOMM Computer Communication Review, 2014*

Use Case 6: Conditional Read



Barthels, C., et al., Designing Databases for Future High-Performance Networks. *IEEE Data Eng. Bulletin, 2017*

Use Case 7: Distributed Transactions



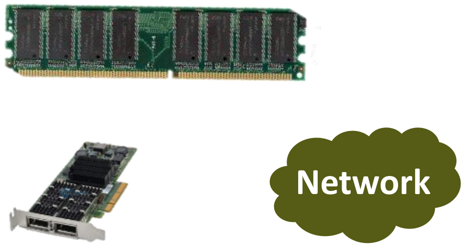
Dragojević, A, et al., No compromises: distributed transactions with consistency, availability, and performance. SOSP'15

Further results and use-cases

Use Case 4: MPI Rendezvous Protocol

program	p	msgs	ovhd	ovhd	red
MILC	64	5.7M	5.5%	1.9%	65%
POP	64	772M	3.1%	2.4%	22%
coMD	72	5.3M	6.1%	2.4%	60%
coMD	360	28.1M	6.5%	2.8%	58%
Cloverleaf	72	2.7M	5.2%	2.4%	53%
Cloverleaf	360	15.3M	5.6%	3.2%	42%

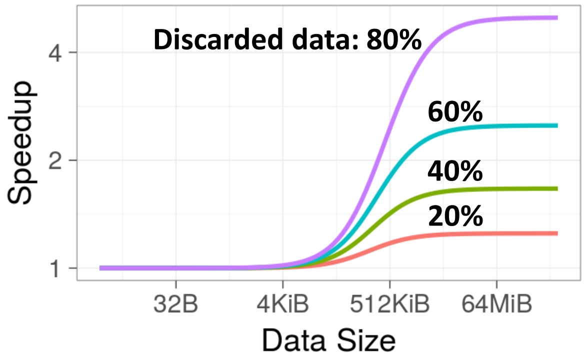
Use Case 5: Distributed KV Store



41% lower latency

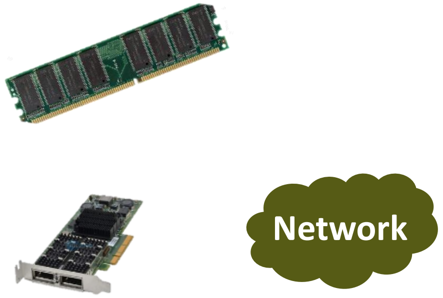
Kalia, A., et al., Using RDMA efficiently for key-value services. In *ACM SIGCOMM Computer Communication Review, 2014*

Use Case 6: Conditional Read



Barthels, C., et al., Designing Databases for Future High-Performance Networks. *IEEE Data Eng. Bulletin, 2017*

Use Case 7: Distributed Transactions



Dragojević, A, et al., No compromises: distributed transactions with consistency, availability, and performance. SOSP'15

Use Case 8: FT Broadcast

Bosilca, G., et al., Failure Detection and Propagation in HPC systems. SC'16

Further results and use-cases

Use Case 4: MPI Rendezvous Protocol

program	p	msgs	ovhd	ovhd	red
MILC	64	5.7M	5.5%	1.9%	65%
POP	64	772M	3.1%	2.4%	22%
coMD	72	5.3M	6.1%	2.4%	60%
coMD	360	28.1M	6.5%	2.8%	58%
Cloverleaf	72	2.7M	5.2%	2.4%	53%
Cloverleaf	360	15.3M	5.6%	3.2%	42%

Use Case 5: Distributed KV Store

41% lower latency

Kalia, A., et al., Usin... e services. In ACM SIGCOMM Com...

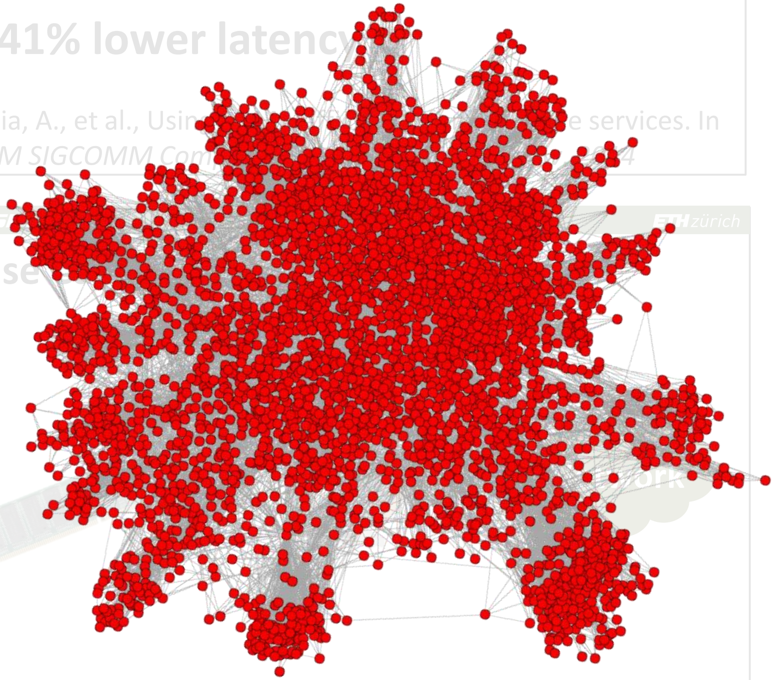
Use Case 6: Conditional Read

Discarded data: 80%
60%
40%
20%
Speedup
Data Size
32B 4KiB 512KiB 64MiB

Barthels, C., et al., Designing Databases for Future High-Performance Networks. *IEEE Data Eng. Bulletin, 2017*

The Next 700 sPIN use-cases

… just think about sPIN graph kernels ….



Use Case 7: Distributed Transactions

Network

Dragojević, A, et al., No compromises: distributed transactions with consistency, availability, and performance. SOSP'15

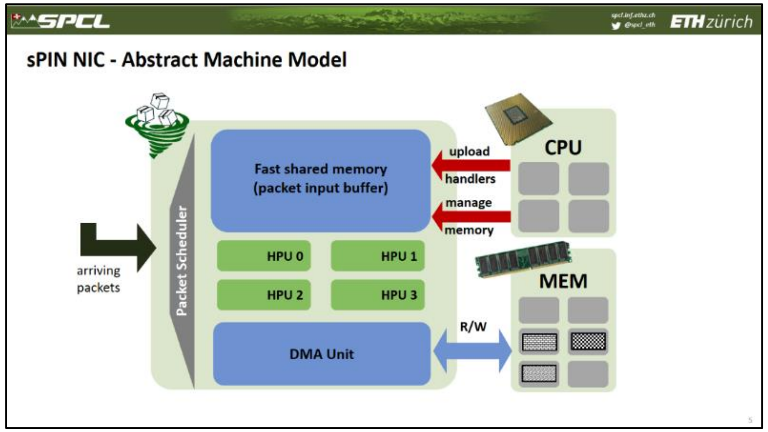
Bosilca, G., et al., Failure Detection and Propagation in HPC systems. SC'16

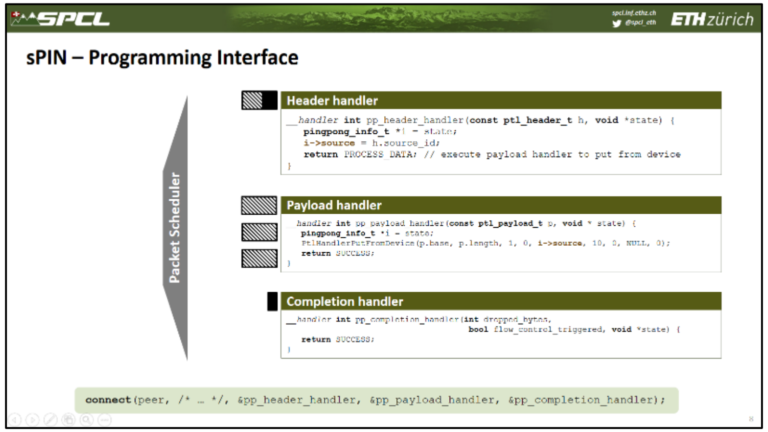
Use Case 9: Distributed Consensus

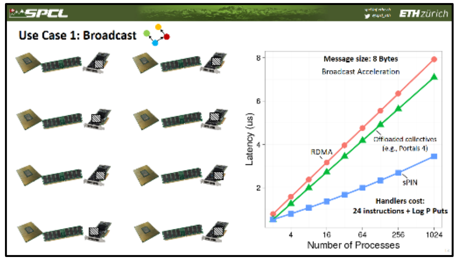
Consensus

István, Z., et al., Consensus in a Box: Inexpensive Coordination in Hardware. NSDI'16

sPIN *Streaming Processing in the Network* for Network Acceleration

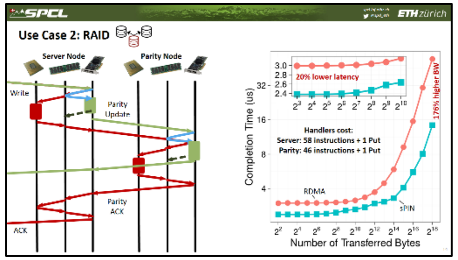


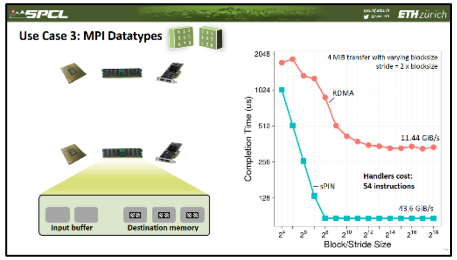


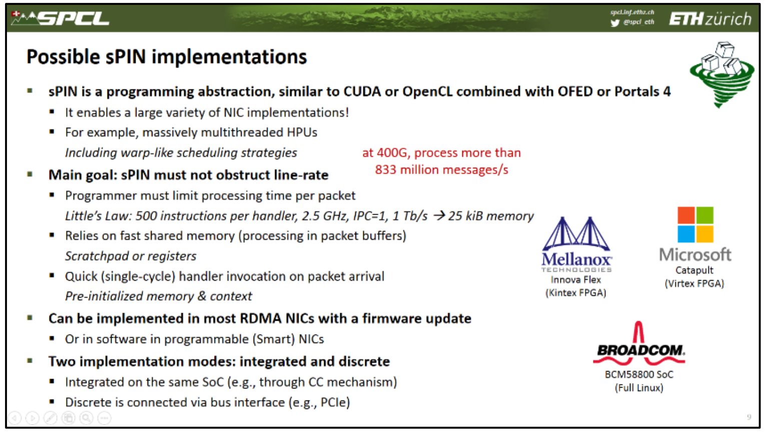


sPIN

beyond RDMA

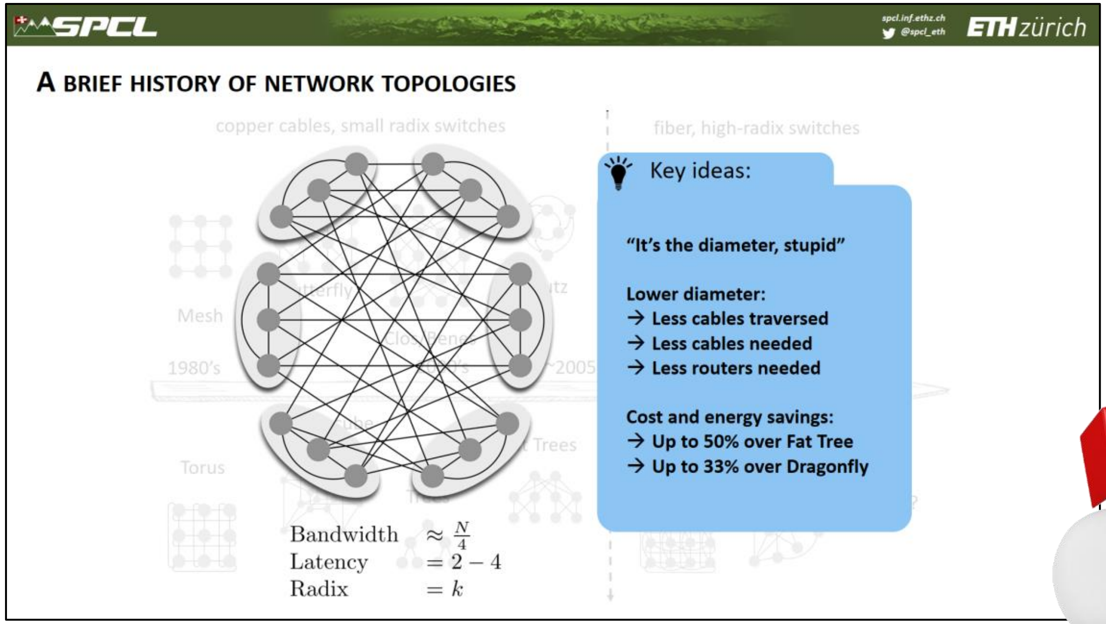


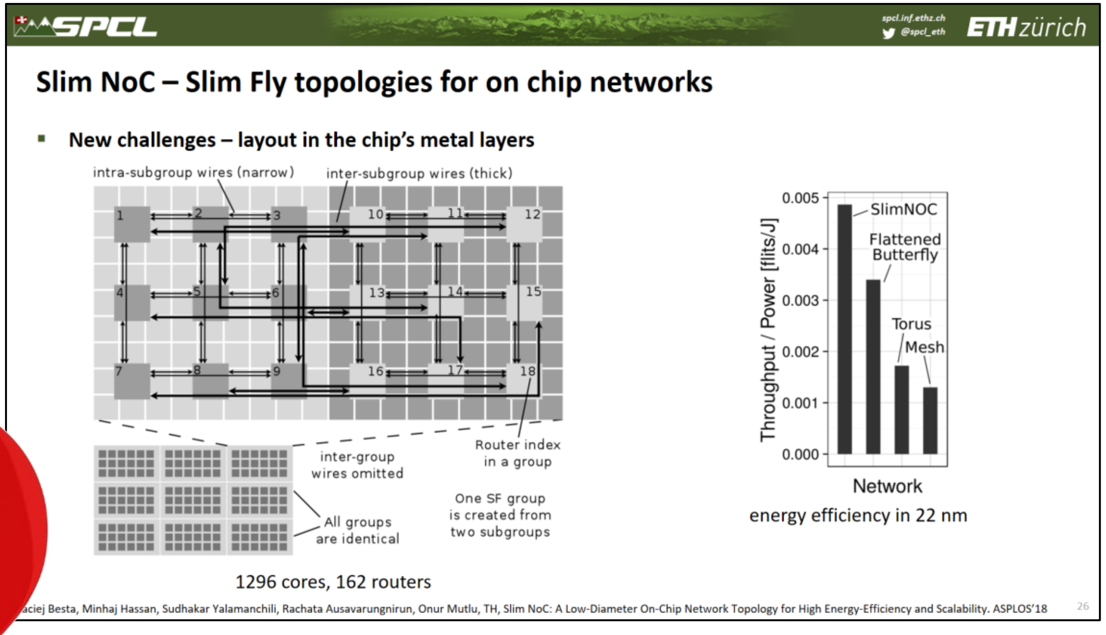


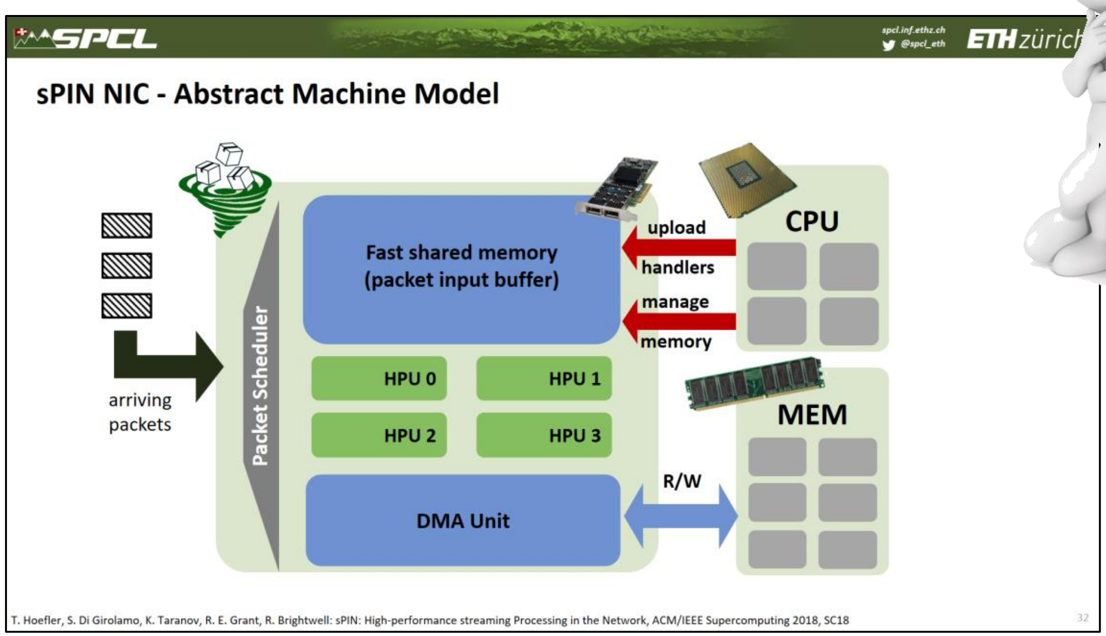


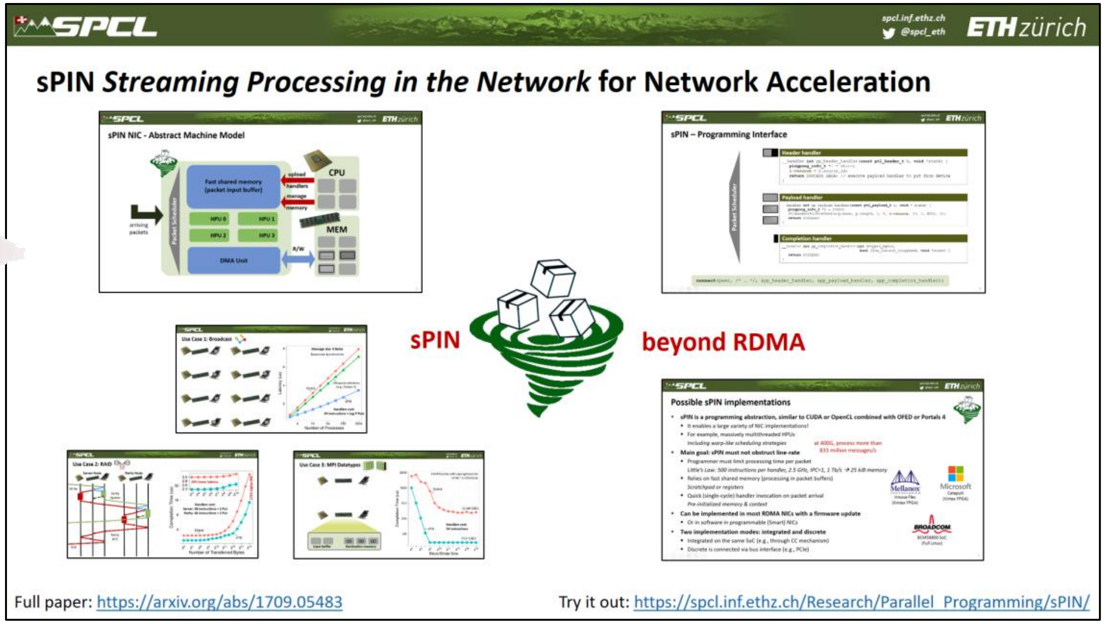
Full paper: https://arxiv.org/abs/1709.05483

Try it out: https://spcl.inf.ethz.ch/Research/Parallel_Programming/sPIN/









Try it: https://spcl.inf.ethz.ch/Research/Scalable_Networking/SlimFly/ (google "spcl slim fly")

https://spcl.inf.ethz.ch/Research/Parallel_Programming/sPIN/ (google "spcl spin")

Backup Slides

Slim Fly Backup

DESIGNING AN EFFICIENT NETWORK TOPOLOGY

CONNECTING ROUTERS: DIAMETER 2

1 *Select a prime power q*

$$q = 4w + \delta;$$

$w \in \mathbb{N} \quad \delta \in \{-1,0,1\},$

A Slim Fly based on q:

Number of routers: $2q^2$

Network radix: $(3q - \delta)/2$

2 *Construct a finite field* $\mathcal{F}_q$

Assuming q is prime:

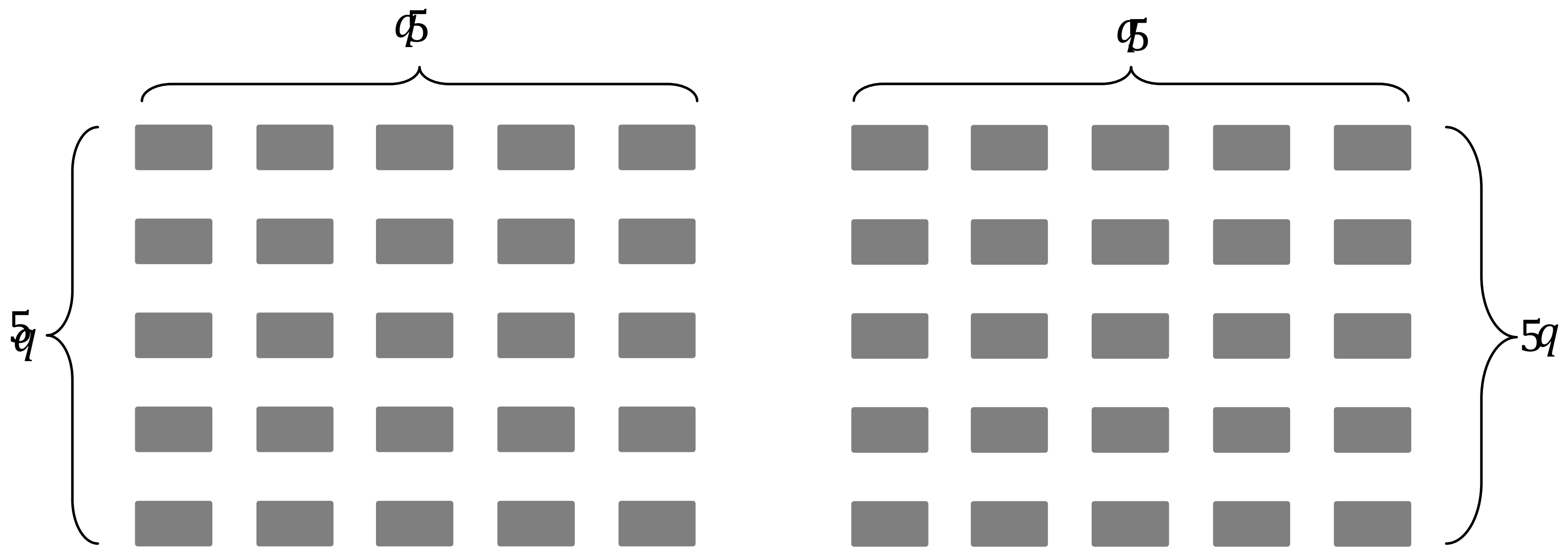
$\mathcal{F}_q = \mathbb{Z}/q\mathbb{Z} = \{0,1, \dots, q-1\}$

with modular arithmetic.

E *Example*: $q = 5$

50 routers
network radix: 7

$\mathcal{F}_5 = \{0,1,2,3,4\}$



Designing an Efficient Network Topology

Connecting routers: Diameter 2

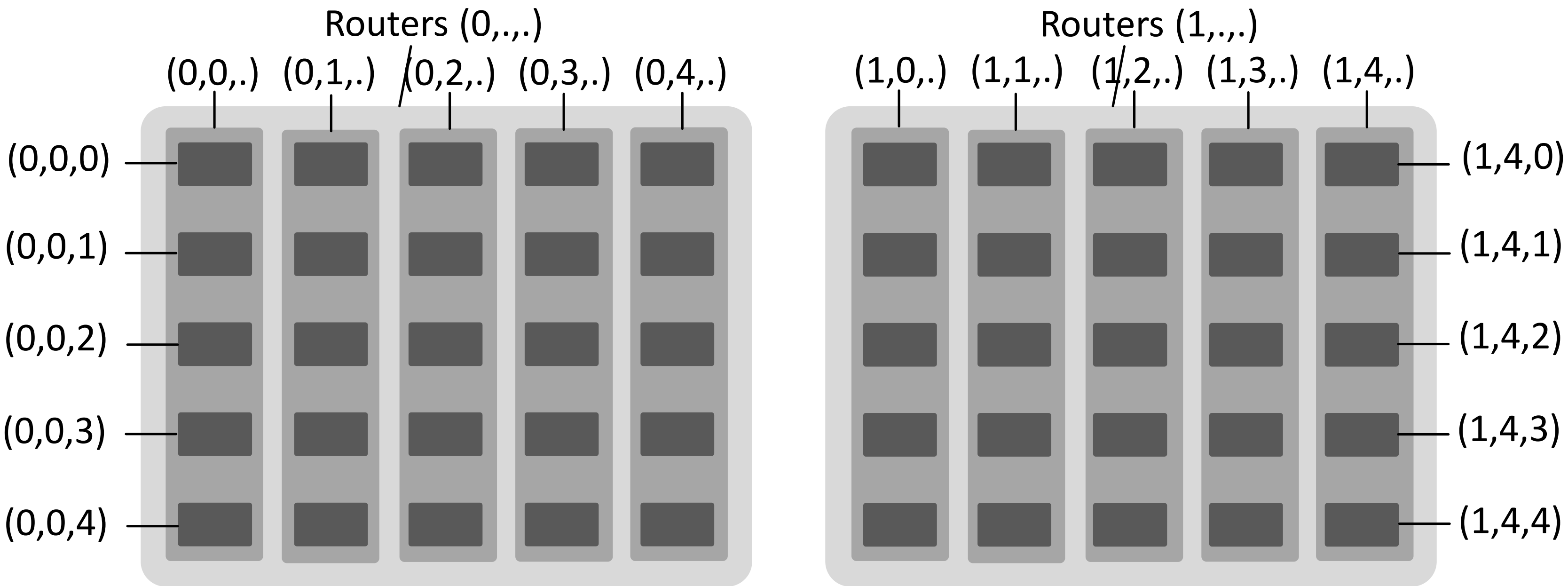
3 *Label the routers*

Set of routers:

$\{0,1\} \times \mathcal{F}_q \times \mathcal{F}_q$

E *Example:* $q = 5$

…



Designing An Efficient Network Topology

Connecting routers: Diameter 2

4 *Find primitive element* ξ

$\xi \in \mathcal{F}_q$ generates $\mathcal{F}_q$

All non-zero elements of $\mathcal{F}_q$ can be written as $\xi^i; \; i \in \mathbb{N}$

5 *Build Generator Sets*

$$X = \{1, \xi^2, \dots, \xi^{q-3}\}$$

$$X' = \{\xi, \xi^3, \dots, \xi^{q-2}\}$$

E *Example:* $q = 5$

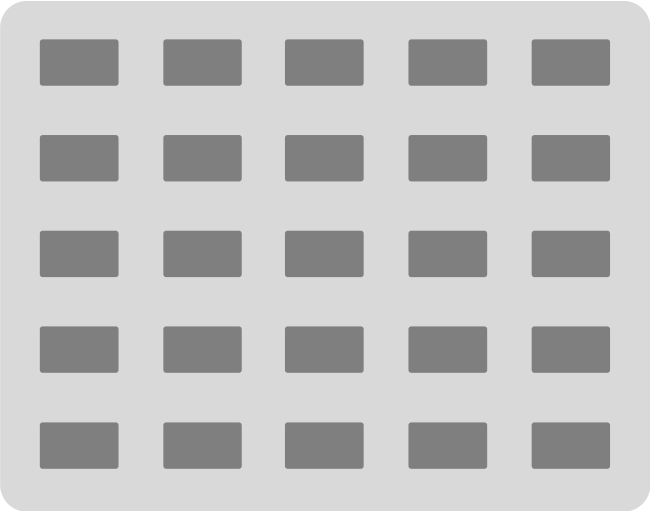
$$\mathcal{F}_5 = \{0,1,2,3,4\}$$

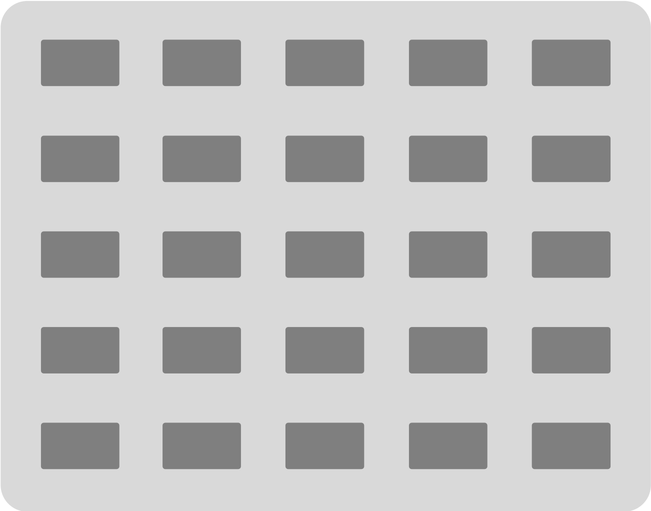
$$\xi = 2$$

$$1 = \xi^4 \, mod \, 5 = 2^4 \, mod \, 5 = 16 \, mod \, 5$$

$$X = \{1,4\}$$

$$X' = \{2,3\}$$





Designing an Efficient Network Topology

Connecting routers: Diameter 2

6 *Intra-group connections*

Two routers in one group are connected iff their "vertical Manhattan distance" is an element from:

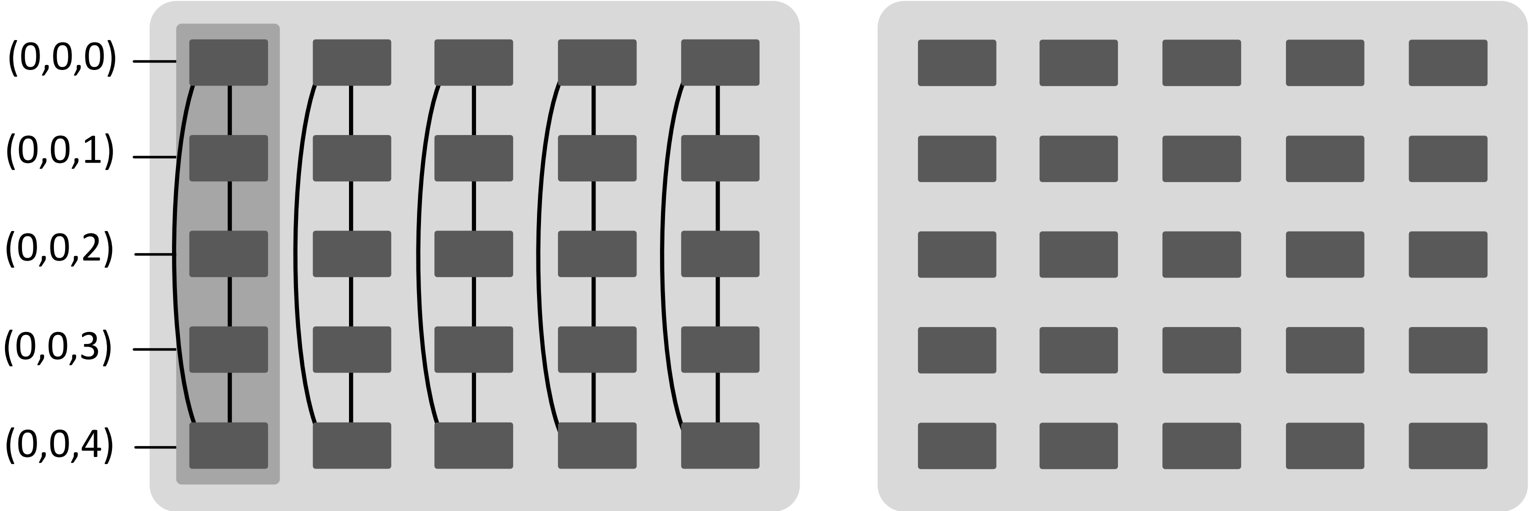
$$X = \{1, \xi^2, \ldots, \xi^{q-3}\}$$ (for subgraph 0)

$$X' = \{\xi, \xi^3, \ldots, \xi^{q-2}\}$$ (for subgraph 1)

E *Example:* $q = 5$

Take Routers $(0,0,.)$

$X = \{1, 4\}$



Designing An Efficient Network Topology

Connecting routers: Diameter 2

6 *Intra-group connections*

Two routers in one group are connected iff their "vertical Manhattan distance" is an element from:

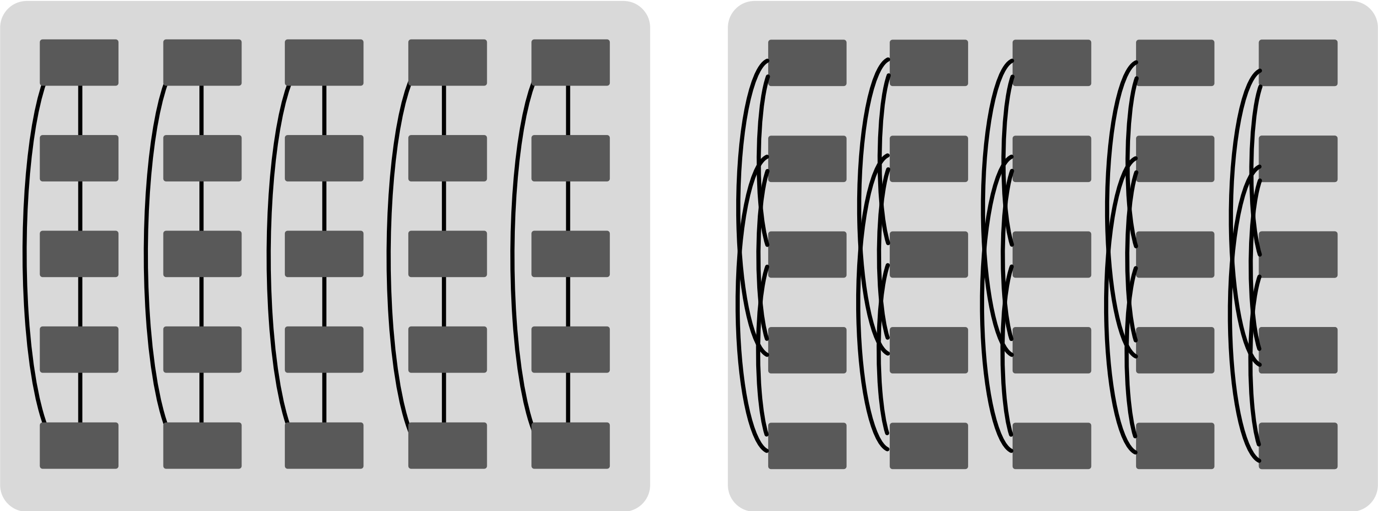
$X = \{1, \xi^2, \ldots, \xi^{q-3}\}$ (for subgraph 0)

$X' = \{\xi, \xi^3, \ldots, \xi^{q-2}\}$ (for subgraph 1)

E *Example:* $q = 5$

Take Routers $(1,4,.)$

$X' = \{2,3\}$



Designing An Efficient Network Topology

Connecting routers: Diameter 2

7 *Inter-group connections*

Router $(0, x, y) \leftrightarrow (1, m, c)$

iff $y = mx + c$

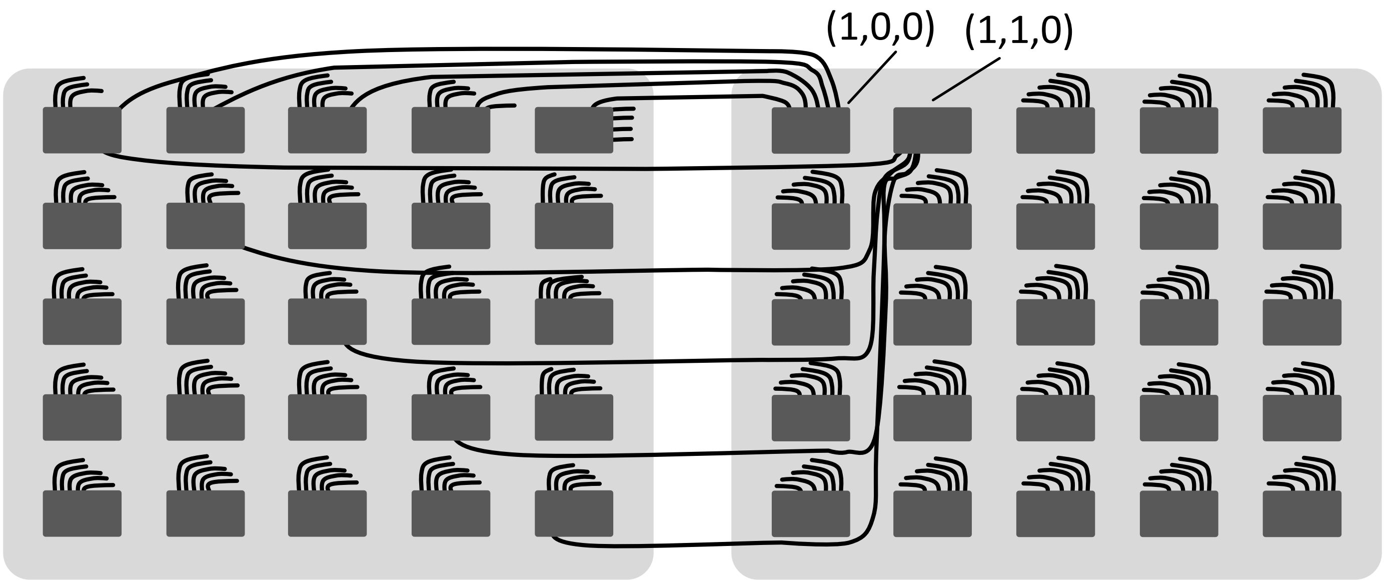
E *Example:* $q = 5$

Take Router $(1,0,0)$ $m = 0, c = 0$

$(1,0,0) \leftrightarrow (0, x, 0)$

Take Router $(1,1,0)$ $m = 1, c = 0$

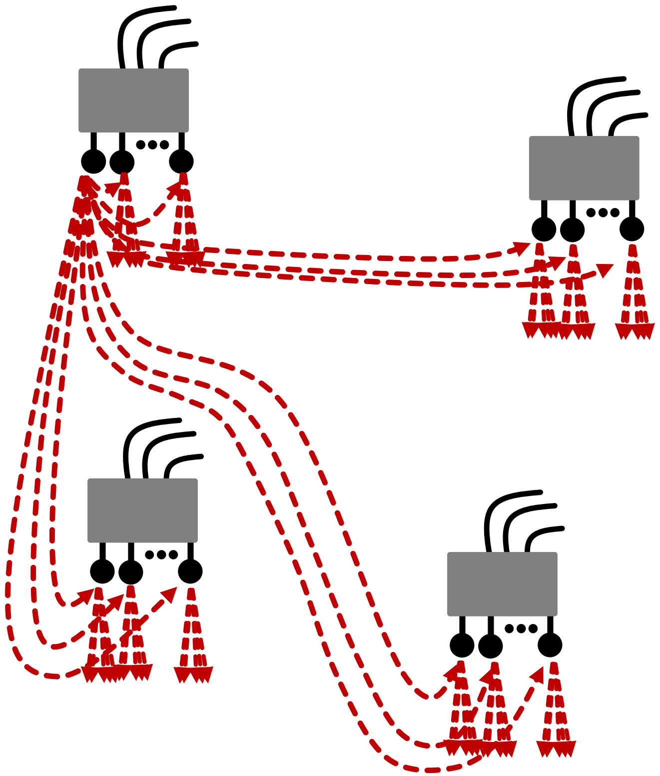
$(1,0,0) \leftrightarrow (0, x, x)$

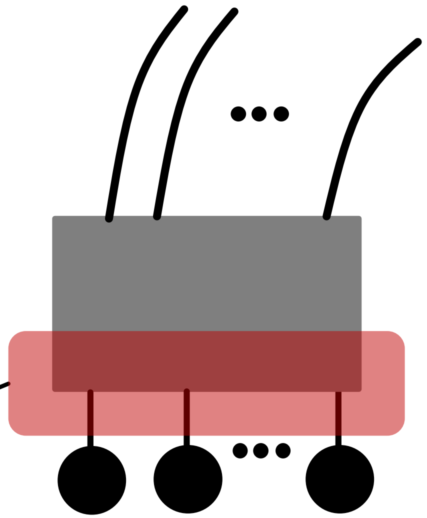


Designing An Efficient Network Topology

Attaching endpoints: Diameter 2

- How many endpoints do we attach to each router?
- As many to ensure *full global bandwidth:*
  - Global bandwidth: the theoretical cumulative throughput in all-to-all in a steady state





Number of ports to endpoints = ?

Designing An Efficient Network Topology

Attaching endpoints: Diameter 2

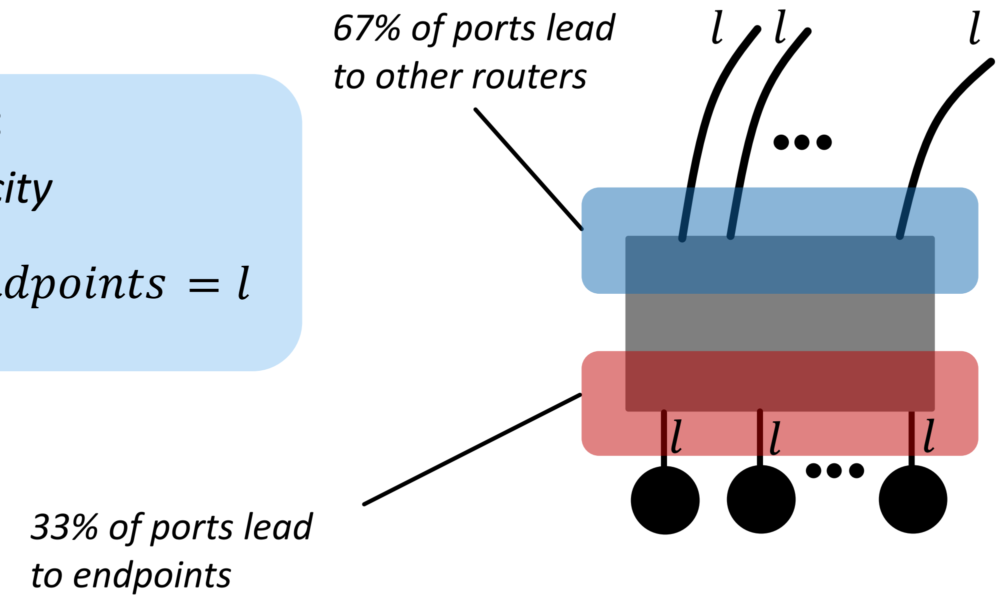
1 Get load *l* per router-router channel (average number of routes per channel)

$$l = \frac{total\ number\ of\ routes}{total\ number\ of\ channels}$$

2 Make the network balanced, i.e.,:

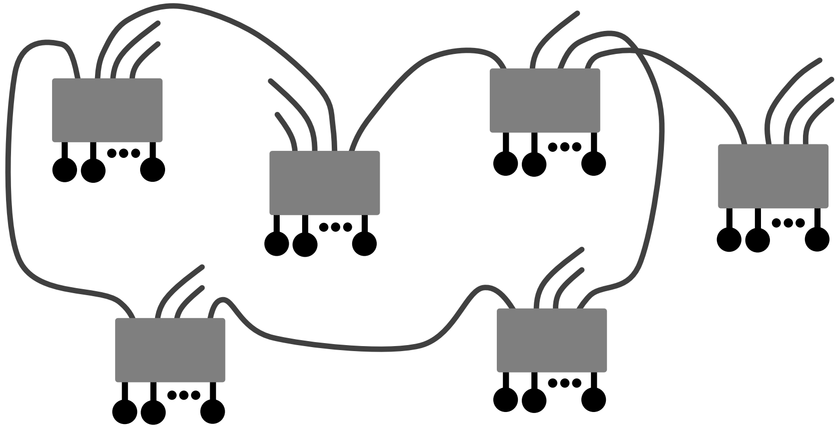
each endpoint can inject at full capacity

$$local\ uplink\ load = number\ of\ endpoints\ = l$$



Structure Analysis

Resiliency

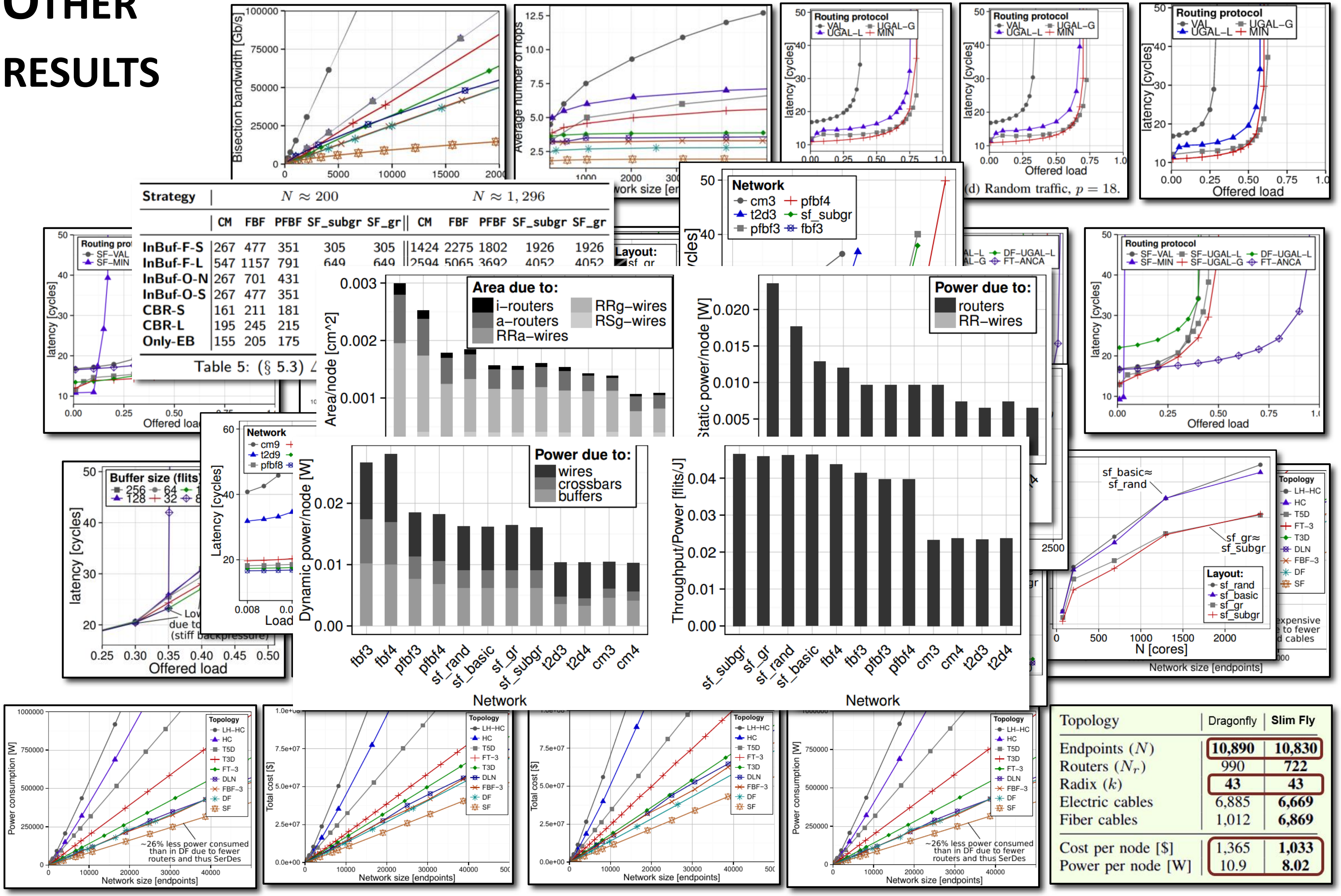


- Disconnection metrics
- Other studied metrics:
  - Average path length (increase by 2); SF is 10% more resilient than DF

Number of endpoints	Torus3D	Torus5D	Hypercube	Long Hop	Fat tree	Dragonfly	Flat. Butterfly	Random	**Slim Fly**
512	30%	-	40%	55%	35%	-	55%	60%	**60%**
1024	25%	40%	40%	55%	40%	50%	60%	-	**-**
2048	20%	-	40%	55%	40%	55%	65%	65%	**65%**
4096	15%	-	45%	55%	55%	60%	70%	70%	**70%**
8192	10%	35%	45%	55%	60%	65%	-	75%	**75%**

"-" means that a given topology does not have a variant of a given size

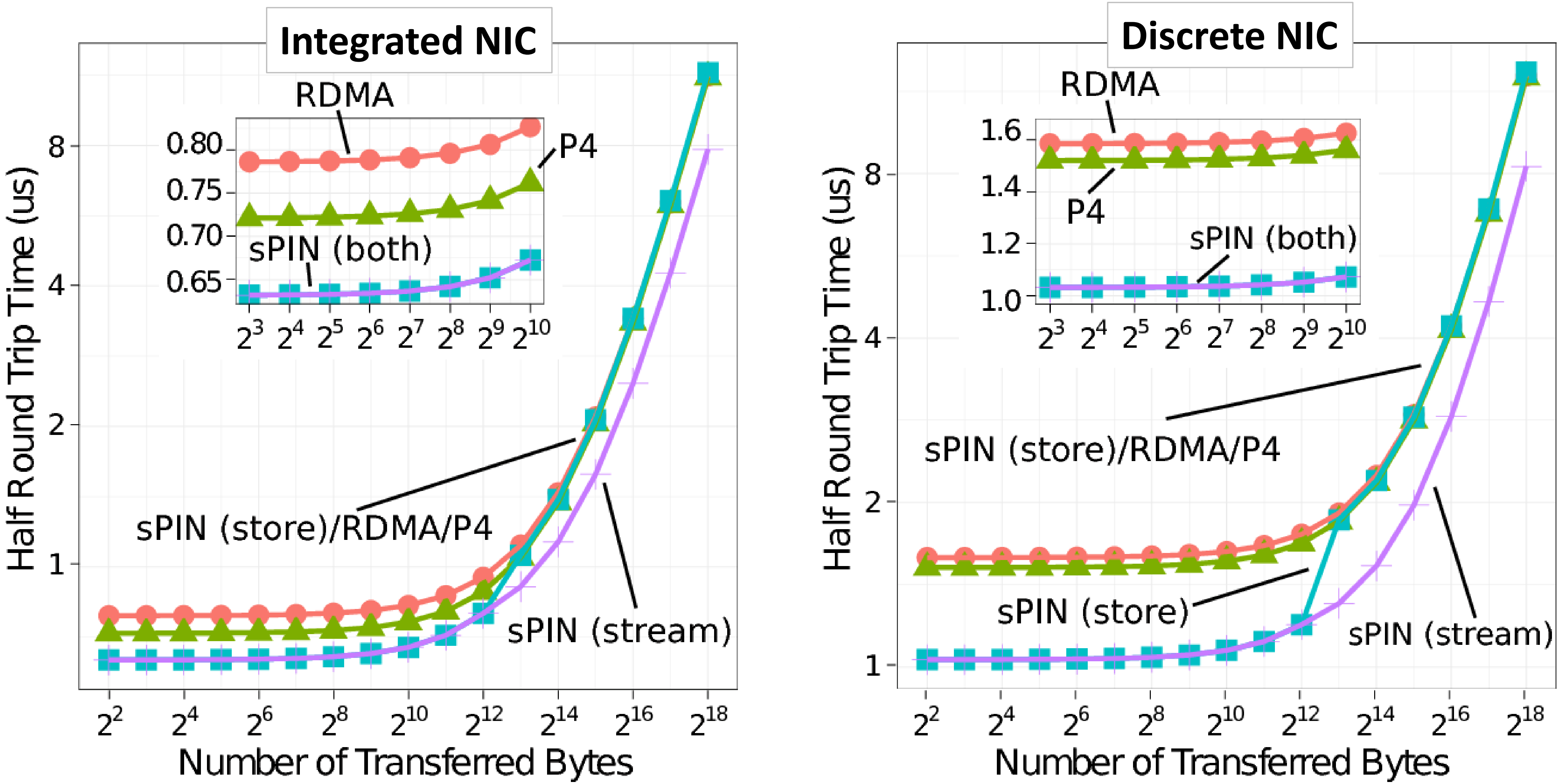
OTHER RESULTS



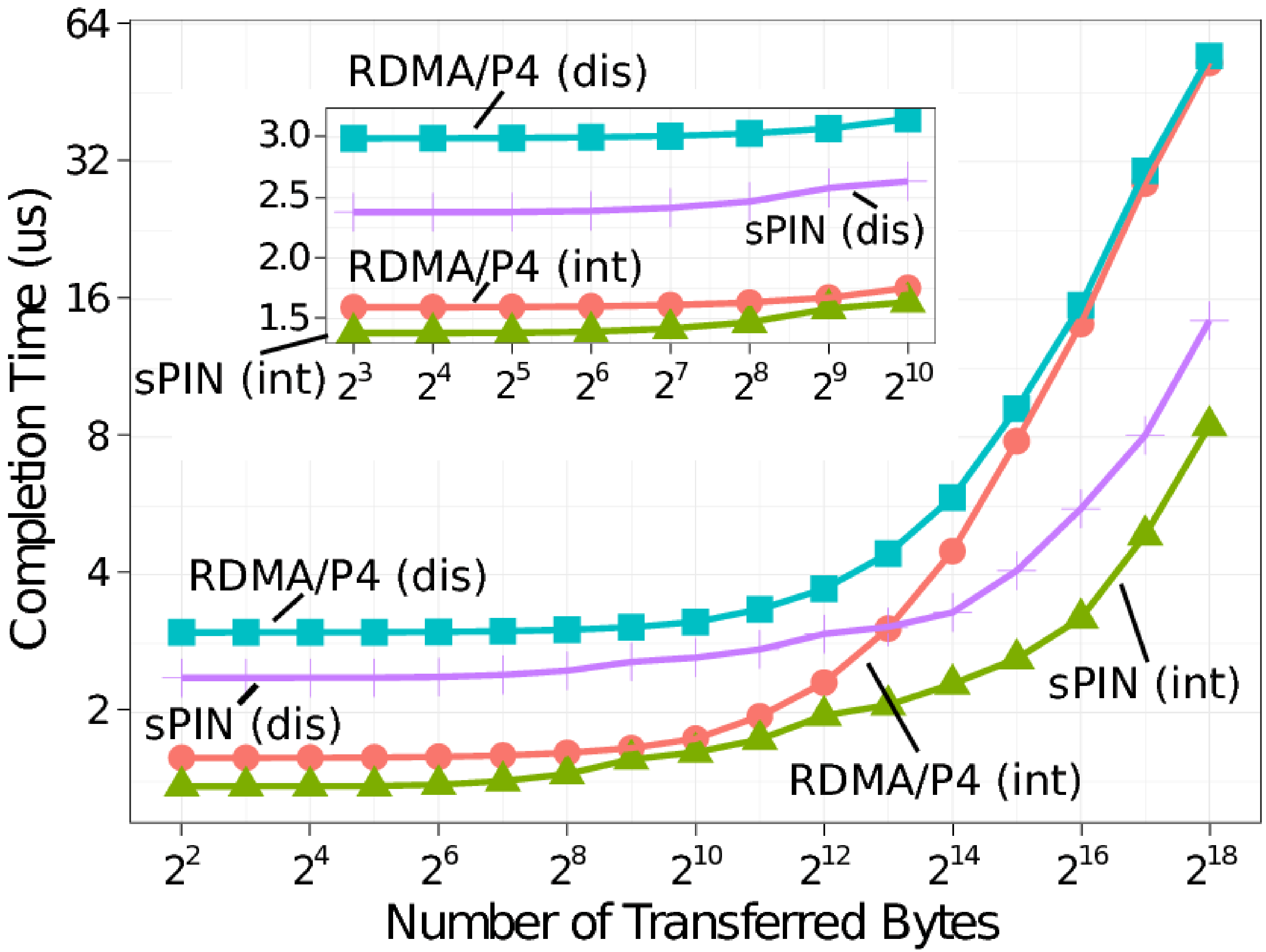
Topology	Dragonfly	Slim Fly
Endpoints (N)	10,890	**10,830**
Routers (N_r)	990	**722**
Radix (k)	43	**43**
Electric cables	6,885	**6,669**
Fiber cables	1,012	**6,869**
Cost per node [$]	1,365	**1,033**
Power per node [W]	10.9	**8.02**

sPIN backup

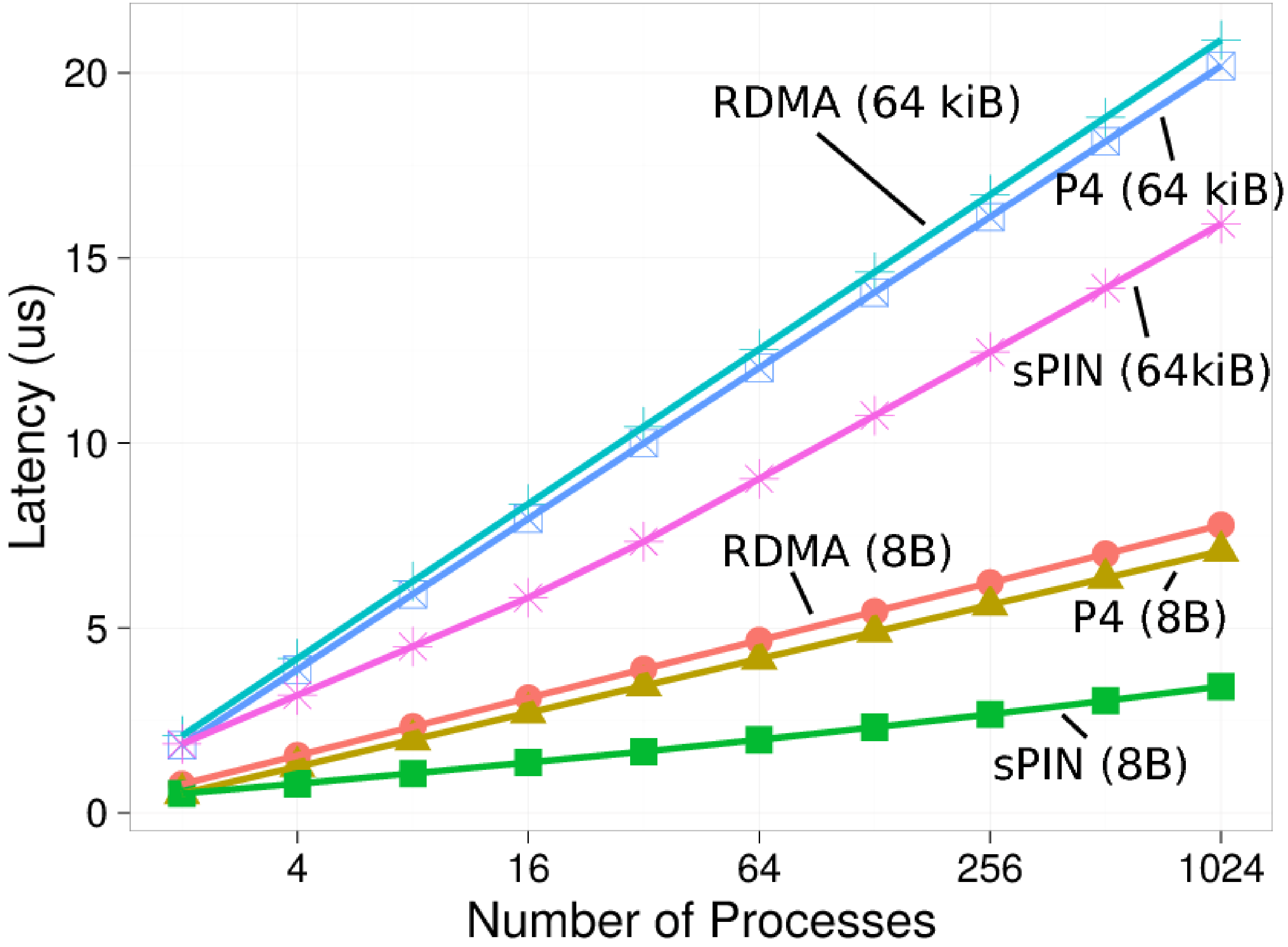
Ping-Pong results (integrated/discrete)



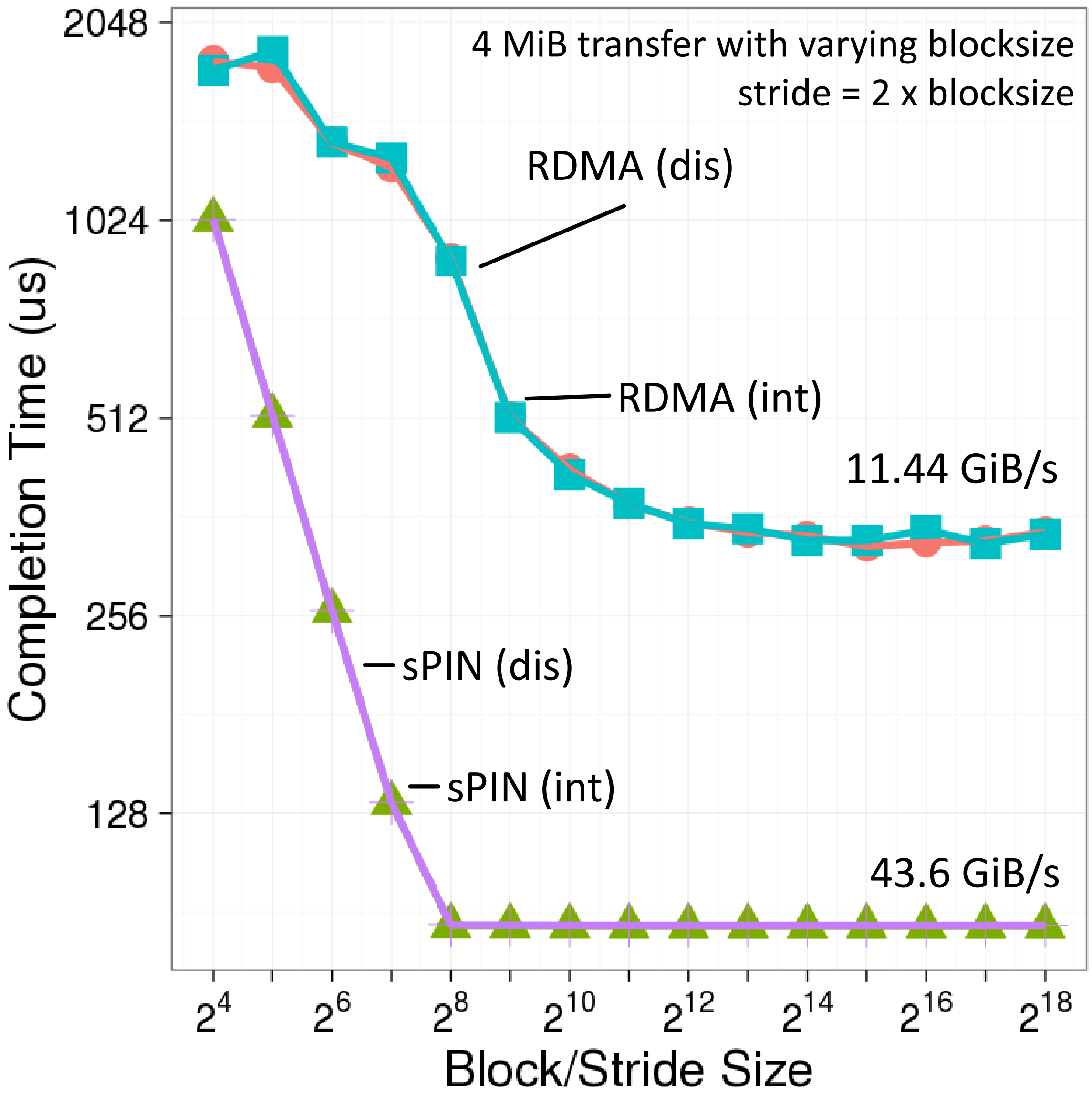
RAID acceleration (integrated/discrete)



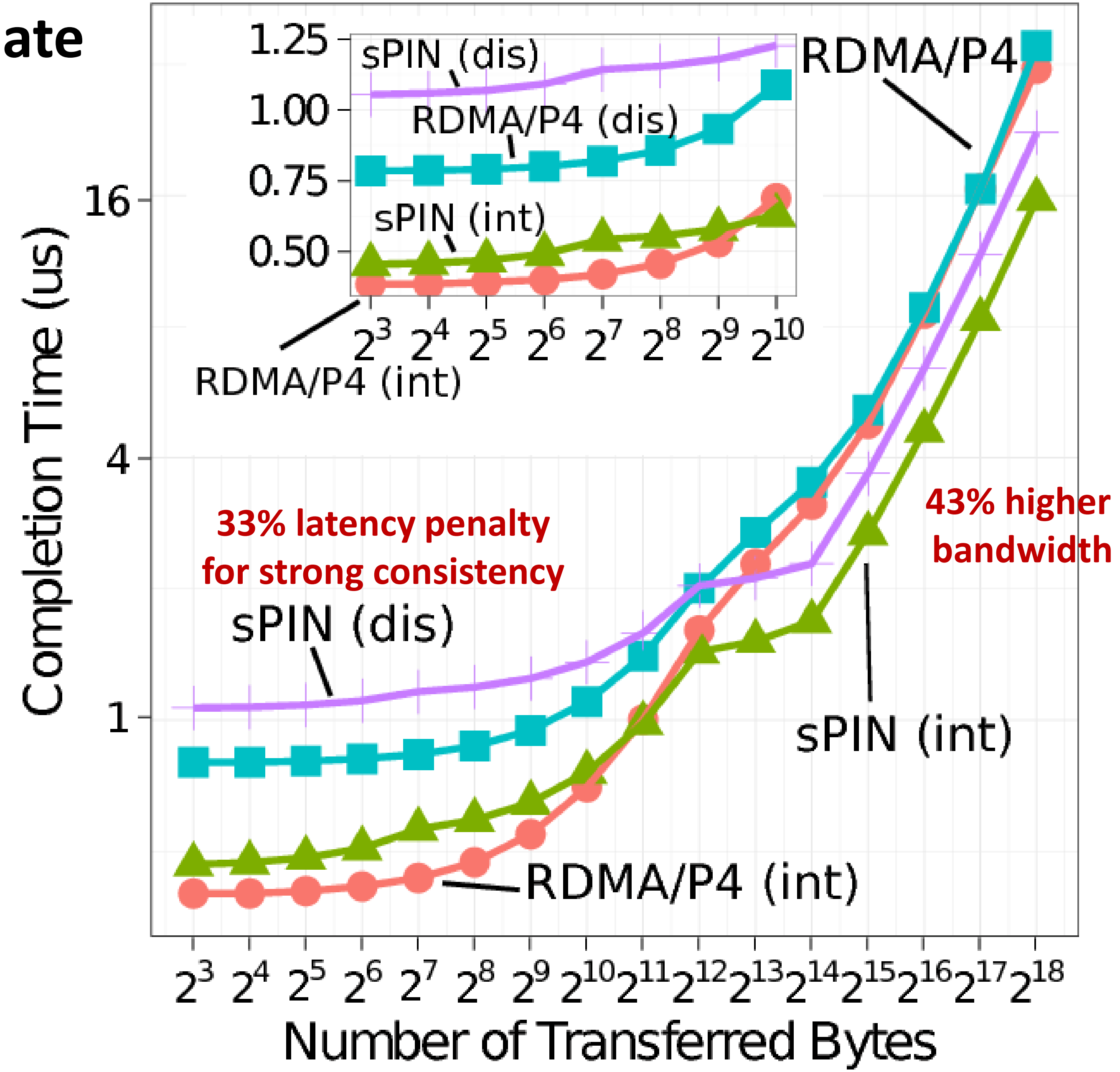
Broadcast acceleration for large messages



MPI Datatypes acceleration (integrated/discrete)



Remote Accumulate



HPUs needed depending on packet size and execution time per packet