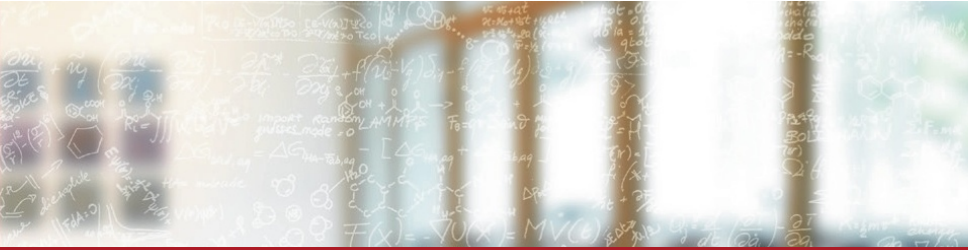
CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

ETH zürich



A PCIe Congestion-Aware Performance Model for Densely Populated Accelerator Servers

Maxime Martinasso[*], Grzegorz Kwasniewski[†], Sadaf R. Alam[*], Thomas C. Schulthess[*‡§], Torsten Hoefler[†]

[*] Swiss National Supercomputing Centre, ETH Zurich, 6900 Lugano, Switzerland
[†] Department of Computer Science, ETH Zurich, Universitätstr. 6, 8092 Zurich, Switzerland
[‡] Institute for Theoretical Physics, ETH Zurich, 8093 Zurich, Switzerland
[§] Computer Science and Mathematics Division, Oak Ridge National Laboratory, USA

Why more densely populated accelerator servers?

- accelerators are faster and more energy-efficient than CPU
- densely populated accelerator servers are high performance nodes
- reduce space occupancy of the data center

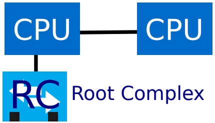
Cray CS Storm – new MeteoSwiss supercomputer

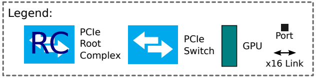
- 2 cabinets
- 12 hybrid computing nodes per cabinet
- 2 Intel Haswell 12-core CPUs per node
- 8 NVIDIA Tesla K80 GPU accelerators per node
- 2 GPU processors per accelerator
- 192 GPU processors in total
- 360 GPU teraflops in total
- **Production system**
- GPUs connected by PCI-Express

PCI EXPRESS

- generation 3, 16 GB/s using x16 wide lane
- dual simplex (a pair of unidirectional links)
- exchange buffer availability between pair of ports of a link
- tree-based topology

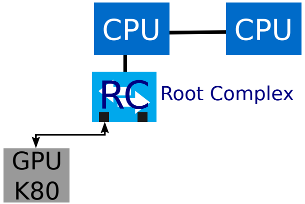
Building a densely populated accelerator servers with PCIe:

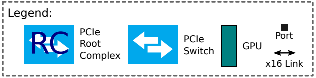




- generation 3, 16 GB/s using x16 wide lane
- dual simplex (a pair of unidirectional links)
- exchange buffer availability between pair of ports of a link
- tree-based topology

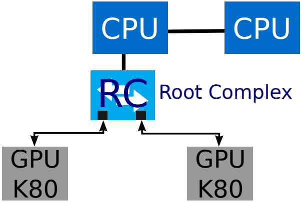
Building a densely populated accelerator servers with PCIe:

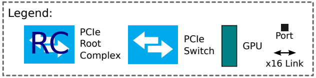




- generation 3, 16 GB/s using x16 wide lane
- dual simplex (a pair of unidirectional links)
- exchange buffer availability between pair of ports of a link
- tree-based topology

Building a densely populated accelerator servers with PCIe:

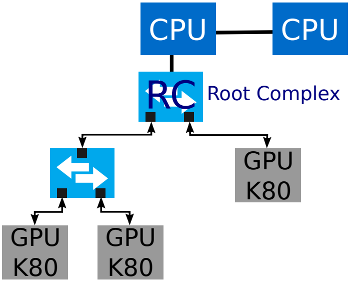


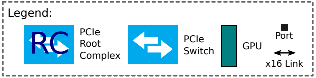


PCI EXPRESS®

- generation 3, 16 GB/s using x16 wide lane
- dual simplex (a pair of unidirectional links)
- exchange buffer availability between pair of ports of a link
- tree-based topology

Building a densely populated accelerator servers with PCIe:

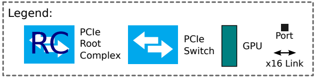


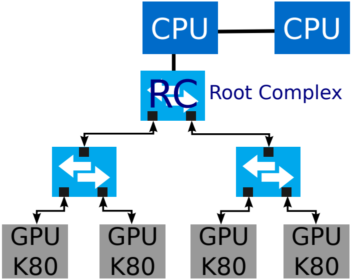


PCI EXPRESS

- generation 3, 16 GB/s using x16 wide lane
- dual simplex (a pair of unidirectional links)
- exchange buffer availability between pair of ports of a link
- tree-based topology

Building a densely populated accelerator servers with PCIe:

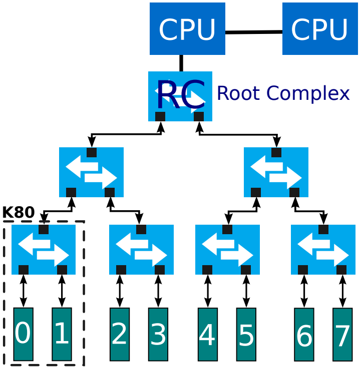


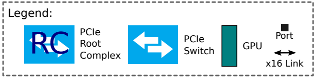


PCI EXPRESS®

- generation 3, 16 GB/s using x16 wide lane
- dual simplex (a pair of unidirectional links)
- exchange buffer availability between pair of ports of a link
- tree-based topology

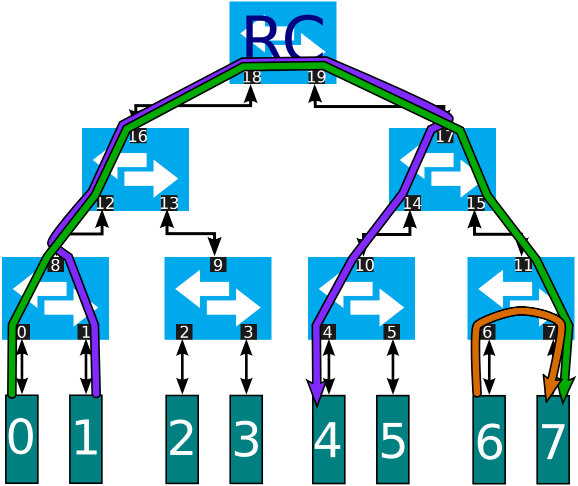
Building a densely populated accelerator servers with PCIe:





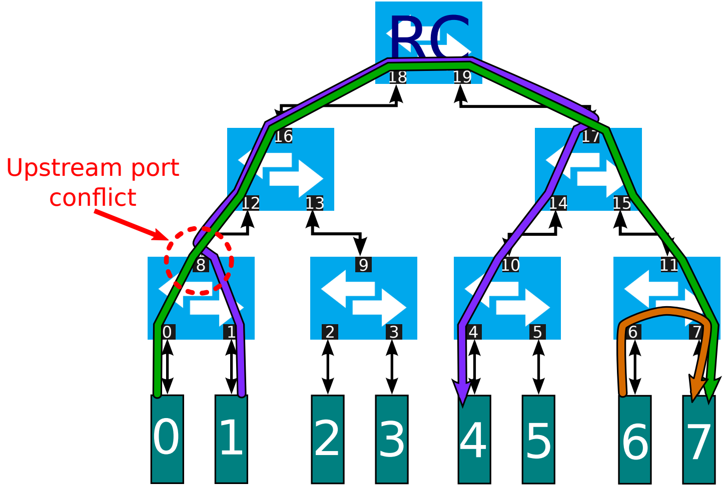
Communication conflicts

0→7, **1→4**, **6→7**



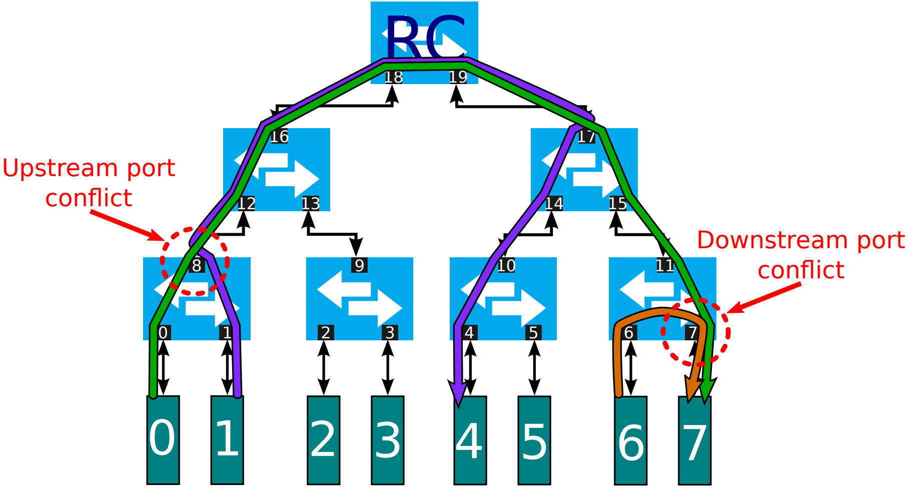
Communication conflicts

0→7, 1→4, 6→7



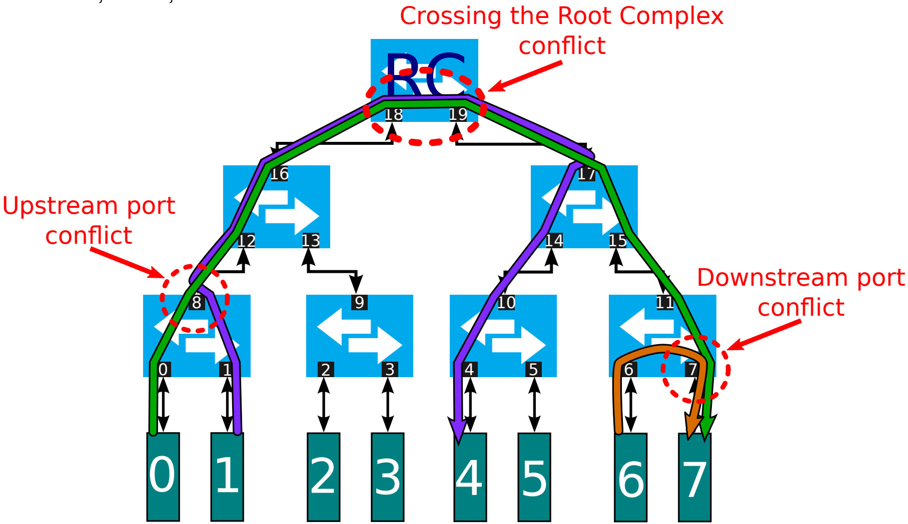
Communication conflicts

0→7, 1→4, 6→7



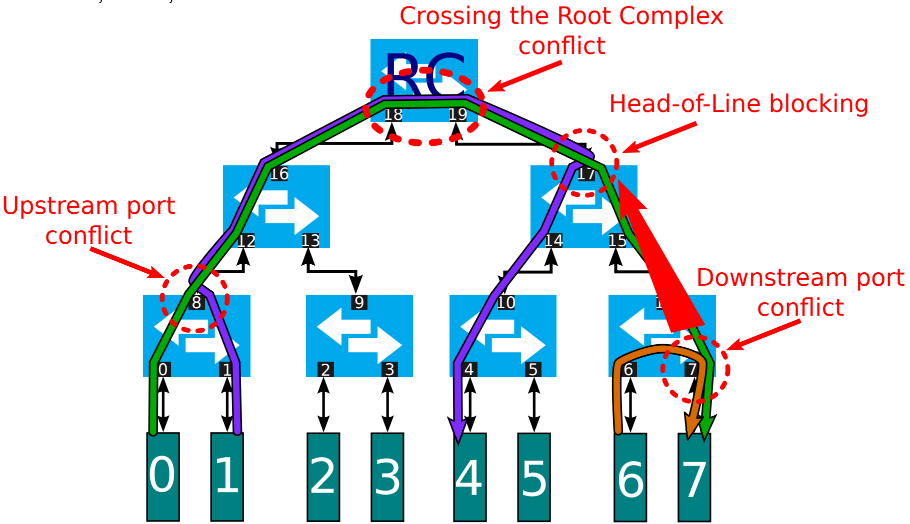
Communication conflicts

0→7, **1→4**, **6→7**

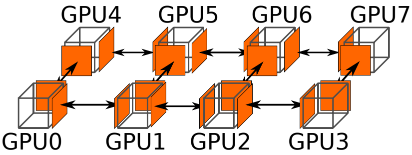


Communication conflicts

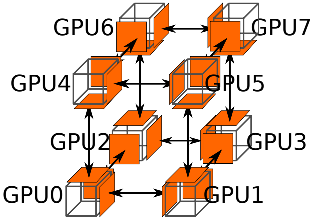
0→7, **1→4**, **6→7**



Motivation – COSMO halo exchange



2D domain decomposition

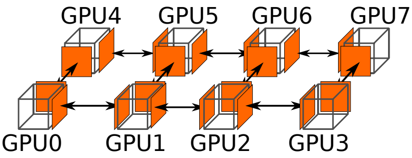


3D domain decomposition

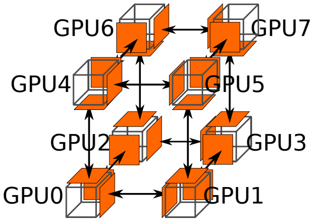
<u>Which order of communications is the fastest?</u>

0→4 1→0 2→1 3→7 4→0 5→4 6→2 7→6	0→1 1→0 2→1 3→2 4→0 5→1 6→2 7→3	...
0→1 1→2 2→6 3→2 4→5 5→6 6→7 7→3	0→4 1→5 2→6 3→7 4→5 5→6 6→5 7→6	
1→5 2→3 5→1 6→5	1→2 2→3 5→4 6→7	

Motivation – COSMO halo exchange



2D domain decomposition



3D domain decomposition

<u>Which order of communications is the fastest?</u>

$0\to4$	$1\to0$	$2\to1$	$3\to7$	$4\to0$	$5\to4$	$6\to2$	$7\to6$
$0\to1$	$1\to2$	$2\to6$	$3\to2$	$4\to5$	$5\to6$	$6\to7$	$7\to3$
	$1\to5$	$2\to3$			$5\to1$	$6\to5$	

$0\to1$	$1\to0$	$2\to1$	$3\to2$	$4\to0$	$5\to1$	$6\to2$	$7\to3$
$0\to4$	$1\to5$	$2\to6$	$3\to7$	$4\to5$	$5\to6$	$6\to5$	$7\to6$
	$1\to2$	$2\to3$			$5\to4$	$6\to7$	

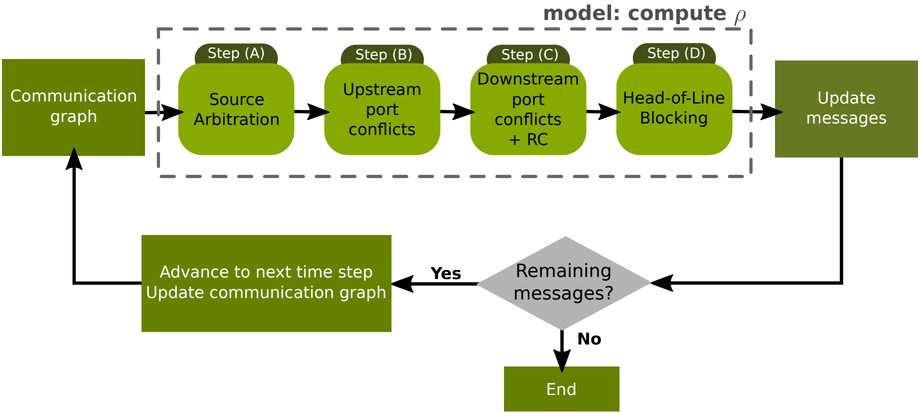
...

2D domain decomposition example: $20,376$ possibilities

3D domain decomposition has more than 1.6 Million possibilities

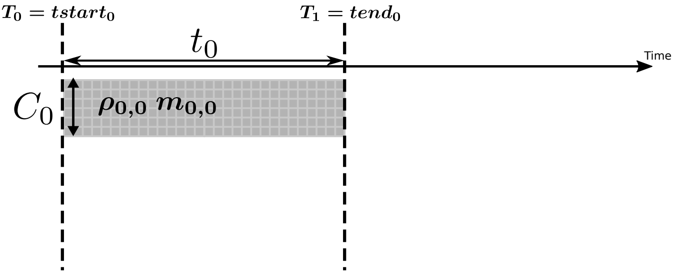
PCIe performance model

We want to identify the congestion factors $\rho \in [0, 1]$ which limit the available bandwidth per communication at each communication phase.



Communication phase – update messages

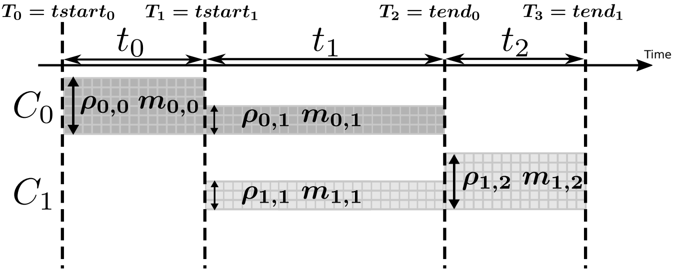
Elapsed time L_c, message size M_c, set of communication phases S_c:



$$L_{C_0} = t_0 = \frac{1}{B} \cdot \frac{m_{0,0}}{\rho_{0,0}} \text{ and } M_{C_0} = m_{0,0}$$

Communication phase – update messages

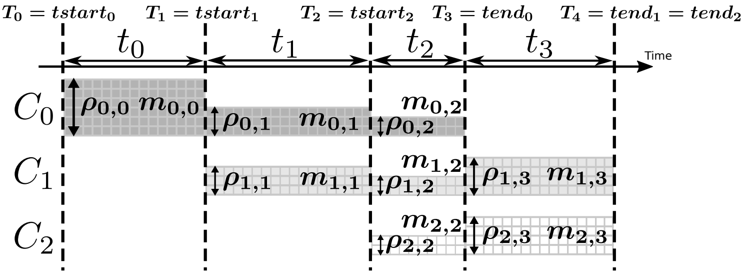
Elapsed time L_c, message size M_c, set of communication phases S_c:



$$L_{C_0} = t_0 + t_1 = \frac{1}{B} \cdot \left(\frac{m_{0,0}}{\rho_{0,0}} + \frac{m_{0,1}}{\rho_{0,1}}\right) \text{ and } M_{C_0} = m_{0,0} + m_{0,1}$$

Communication phase – update messages

Elapsed time L_c, message size M_c, set of communication phases S_c:



$$L_c = \sum_{i \in S_c} t_i = \frac{1}{B} \sum_{i \in S_c} \frac{m_{c,i}}{\rho_{c,i}} \text{ and } M_c = \sum_{i \in S_c} m_{c,i}$$

(1) start time and M_c are known

(2) $\rho_{c,i}$ are given by the model

with (1) and (2) L_c are computable

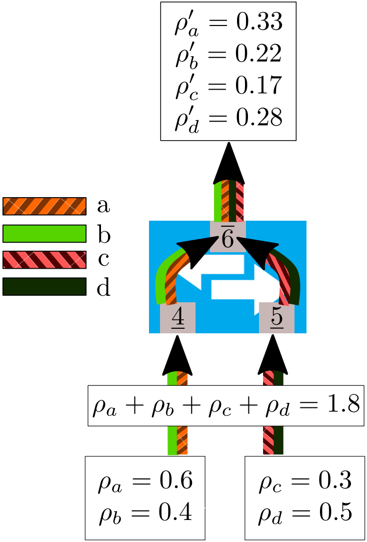
Model conflicts on switch

We want to identify the congestion factors $\rho \in [0, 1]$ which limit the available bandwidth per communication at each communication phase.

- Each communication enters a switch with a congestion factor ρ and leaves with a congestion factor ρ'
- If $\sum_i \rho_i > 1$ then an arbitration policy is required, $\rho' = \rho$ otherwise

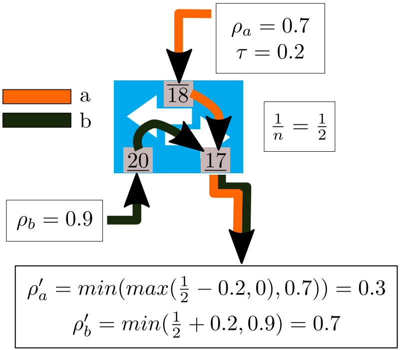
Upstream port conflict

Proportional sharing of available bandwidth



Downstream port conflict

- Round-robin policy
- Performance reduction for crossing the root complex



C^R – set of communications crossing the root complex;
n – number of grouped communication sets;
R – congestion factor of a grouped communication set;
τ – congestion factor for crossing the root complex;

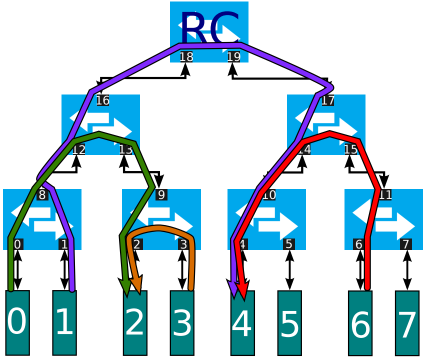
– if $C^R = \emptyset$ then

$$R' = \frac{1}{n}$$

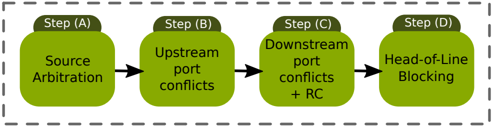
– if $C^R \neq \emptyset$ then

$$R' = \begin{cases} \min(\max(\frac{1}{n} - \tau, 0), R) & \text{if } R \text{ contains comm.} \in C^R \\ \min(\frac{1}{n} + \tau, R) & \text{otherwise} \end{cases}$$

Complete example

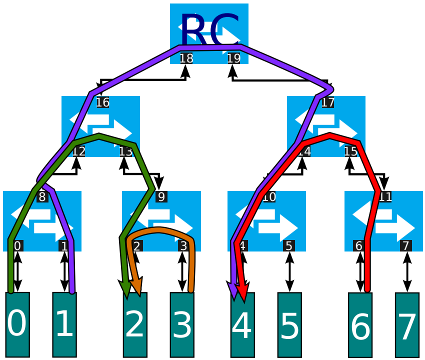


$\tau = 0.2$



comm.	Step (A)	Step (B)	Step (C)	Step (D)	
(a) $0 \to 2$	1	$1/2$	$1/2$	$3/10$	$3/10$
(b) $1 \to 4$	1	$1/2$	$3/10$	$3/10$	$3/10$
(c) $3 \to 2$	1	1	$1/2$	$1/2$	$7/10$
(d) $6 \to 4$	1	1	$7/10$	$7/10$	$7/10$

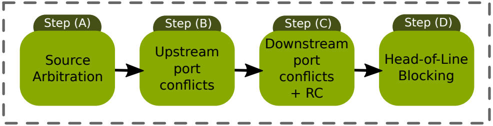
Complete example



$\tau = 0.2$

Message size: 300MB

Bandwidth: 11.6 GB/s

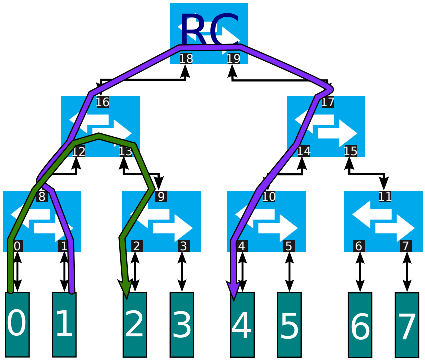


comm.	Step (A)	Step (B)	Step (C)	Step (D)	
(a) 0→2	1	1/2	1/2	3/10	3/10
(b) 1→4	1	1/2	3/10	3/10	3/10
(c) 3→2	1	1	1/2	1/2	7/10
(d) 6→4	1	1	7/10	7/10	7/10

Congestion graph step 1

comm.	cong. factor	data remaining	elapsed time
(a) 0→2	3/10	128 MB	36 ms
(b) 1→4	3/10	128 MB	36 ms
(c) 3→2	7/10	0 MB	36 ms
(d) 6→4	7/10	0 MB	36 ms

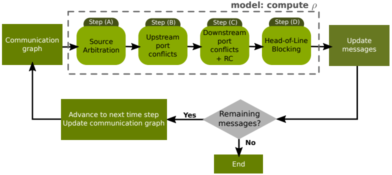
Complete example



$\tau = 0.2$

Message size: 300MB

Bandwidth: 11.6 GB/s



Congestion graph step 1

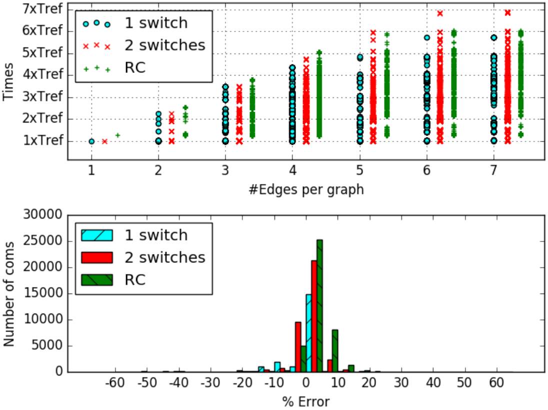
comm.	cong. factor	data remaining	elapsed time
(a) 0⟶2	$^3/_{10}$	128 MB	36 ms
(b) 1⟶4	$^3/_{10}$	128 MB	36 ms
(c) 3⟶2	$^7/_{10}$	0 MB	36 ms
(d) 6⟶4	$^7/_{10}$	0 MB	36 ms

Congestion graph step 2

comm.	cong. factor	data remaining	elapsed time
(a) 0⟶2	$^1/_2$	0 MB	65 ms
(b) 1⟶4	$^1/_2$	0 MB	65 ms

Model Validation

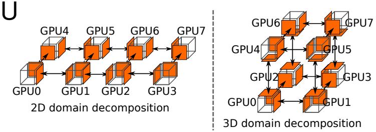
- Architecture parameters:
  $B = 11.6GB/s$
  $\tau = 0.1735$
- 22,259 graphs:
  - non-isomorphic
  - *cudaMemcpyAsync*
  - Communication pattern: scatter, gather, all-to-all
  - Entire set of graphs for subsets of GPUs
  - Randomly generated
  - $\simeq$100K communications
- Message size: 300 MB
- Time no contention:
  $T_{ref} = 25.3ms$

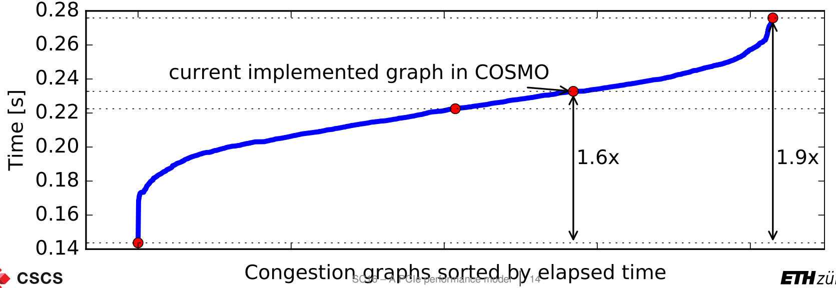


95% of communication are in range +/- 15%

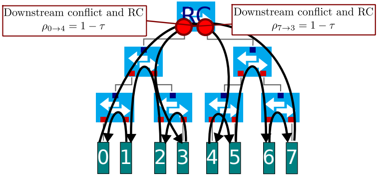
Back to the motivation – COSMO halo exchange

- Upper limit on time to solution, throughput approach
- Running mode one instance per socket (8 GPUs)
- Large domain size 256x256x80 per GPU
- One step triggers 312 halo exchanges
- Message size: 40 KB to 254 KB
- Uses MPI
- $(3!)^4 \times (2!)^4 = 20,736$ communication graphs for 2D domain

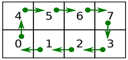


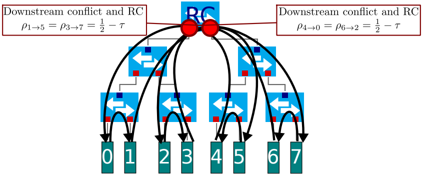


Fastest schedule for 2D decomposition

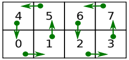


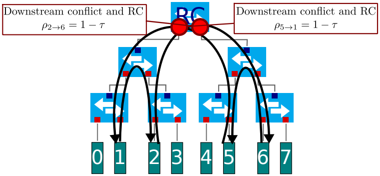
Step 1:
$0\to4,\ 1\to0,$
$2\to1,\ 3\to2,$
$4\to5,\ 5\to6,$
$6\to7,\ 7\to3$



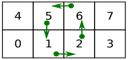


Step 2:
$0\to1,\ 1\to5,$
$2\to3,\ 3\to7,$
$4\to0,\ 5\to4,$
$6\to2,\ 7\to6$

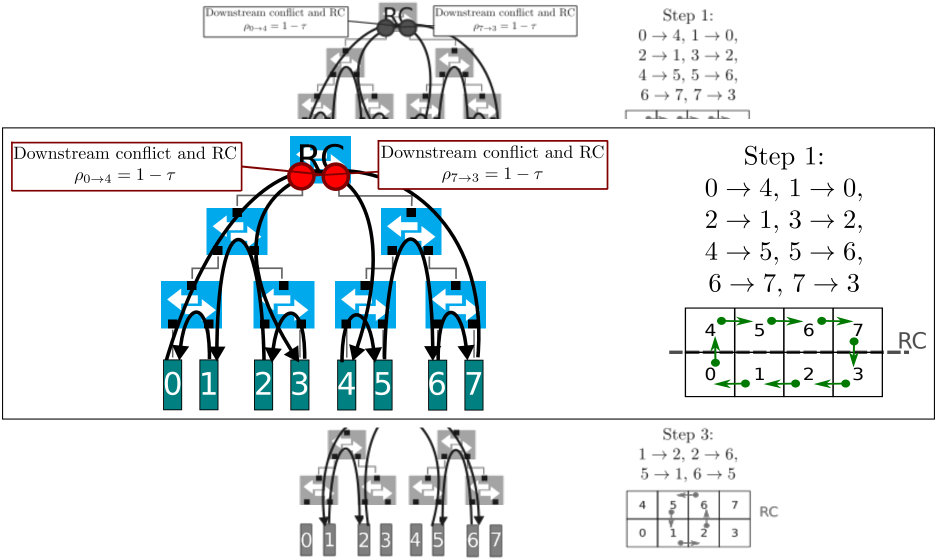




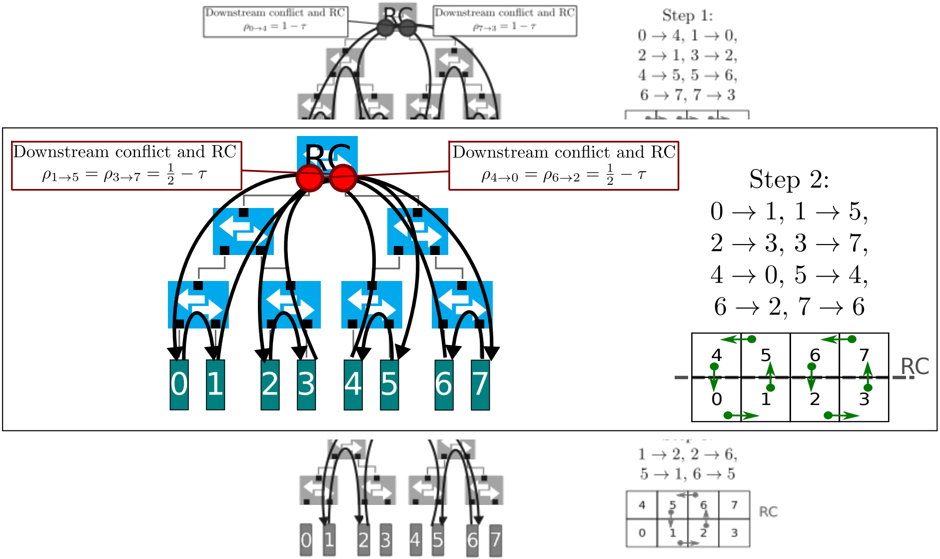
Step 3:
$1\to2,\ 2\to6,$
$5\to1,\ 6\to5$



Fastest schedule for 2D decomposition



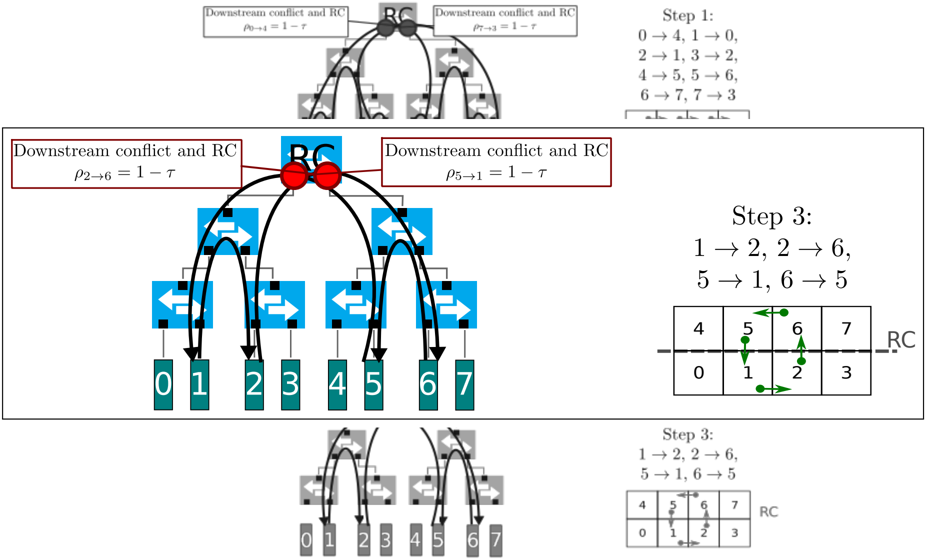
Fastest schedule for 2D decomposition



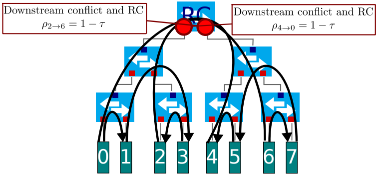
CSCS

ETH zürich

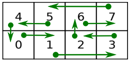
Fastest schedule for 2D decomposition

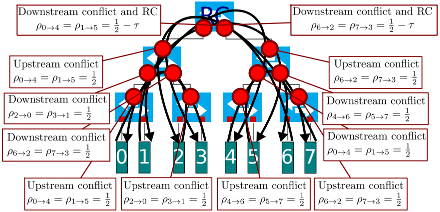


Fastest schedule for 3D decomposition

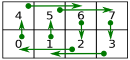


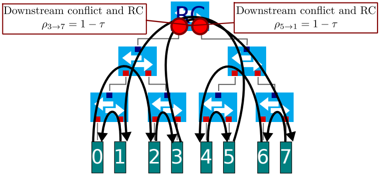
Step 1:
$0 \to 1,\ 1 \to 3,$
$2 \to 6,\ 3 \to 2,$
$4 \to 0,\ 5 \to 4,$
$6 \to 7,\ 7 \to 5$



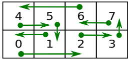


Step 2:
$0 \to 4,\ 1 \to 5,$
$2 \to 0,\ 3 \to 1,$
$4 \to 6,\ 5 \to 7,$
$6 \to 2,\ 7 \to 3$

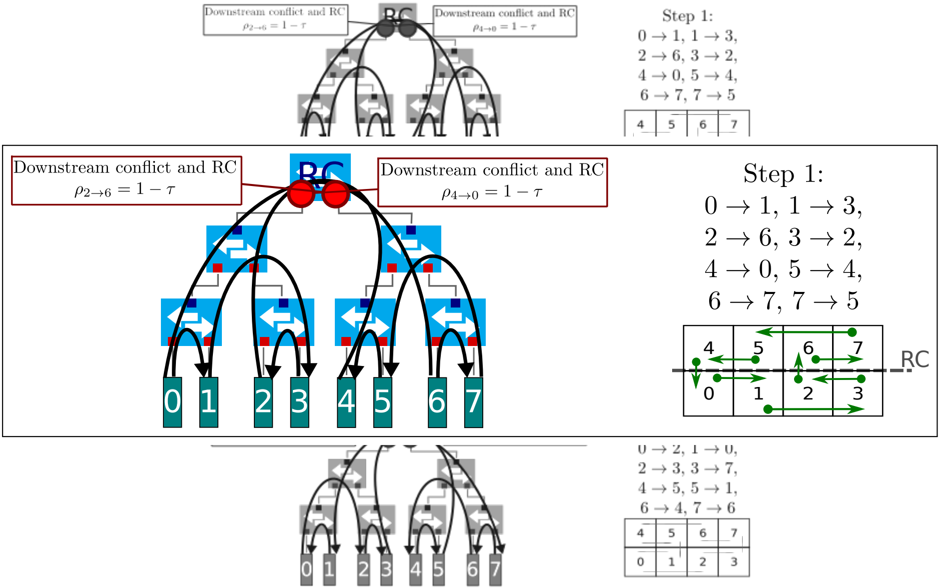




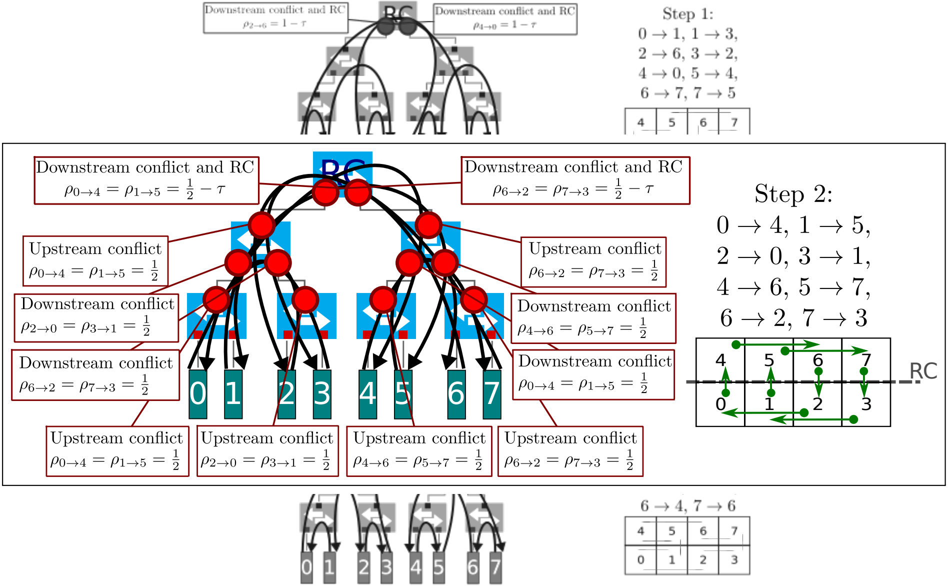
Step 3:
$0 \to 2,\ 1 \to 0,$
$2 \to 3,\ 3 \to 7,$
$4 \to 5,\ 5 \to 1,$
$6 \to 4,\ 7 \to 6$



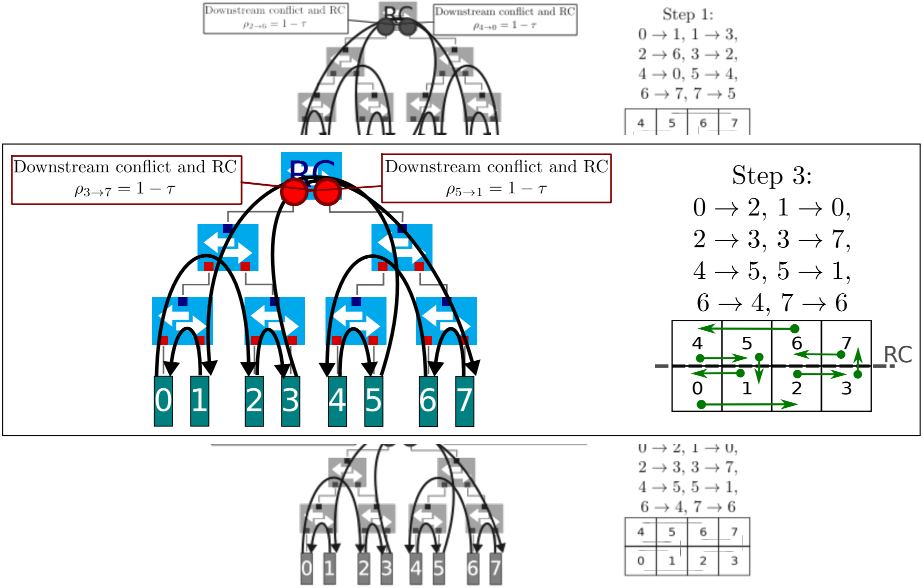
Fastest schedule for 3D decomposition



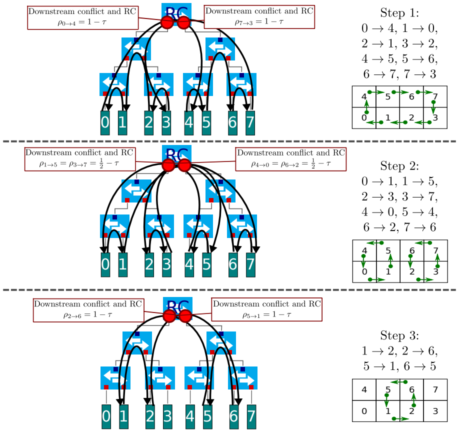
Fastest schedule for 3D decomposition



Fastest schedule for 3D decomposition



COSMO improvement – fastest schedule



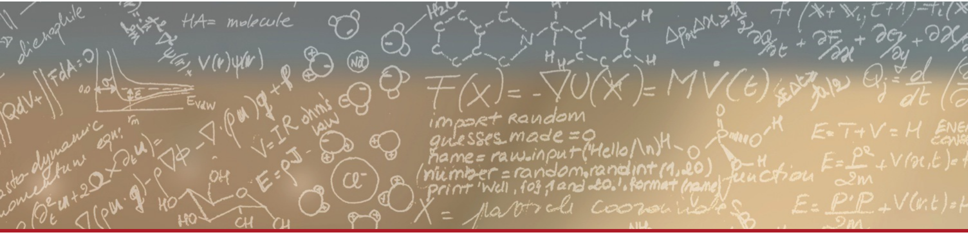
COSMO gain: 5.6% per halo exchange step,
gain is limited by MPI 2-sided overhead.

Conclusion

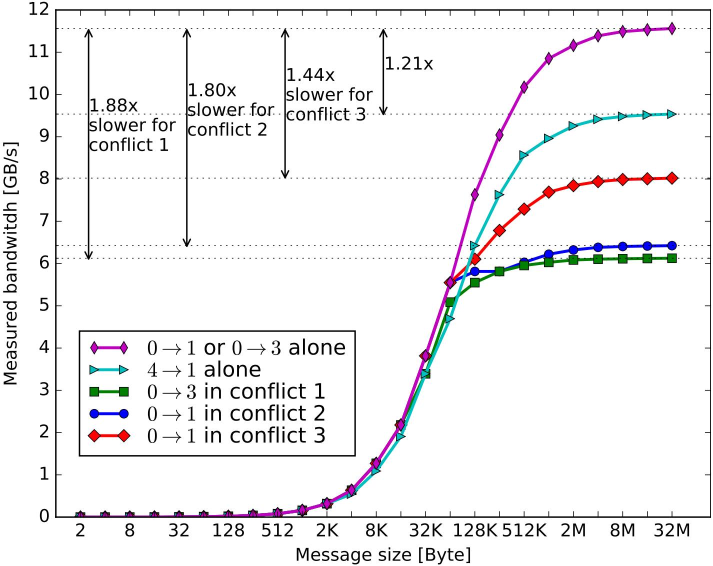
- \- Latency not modeled
- \- MPI 2-sided overhead not modeled (use one-sided?)
- \+ Captures all PCIe features including congestion
- \+ Simple model only 2 parameters (B and τ)
- \+ Precise for large messages
- \+ Design of topology-aware algorithms
- \+ COSMO halo exchange performance gain

CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

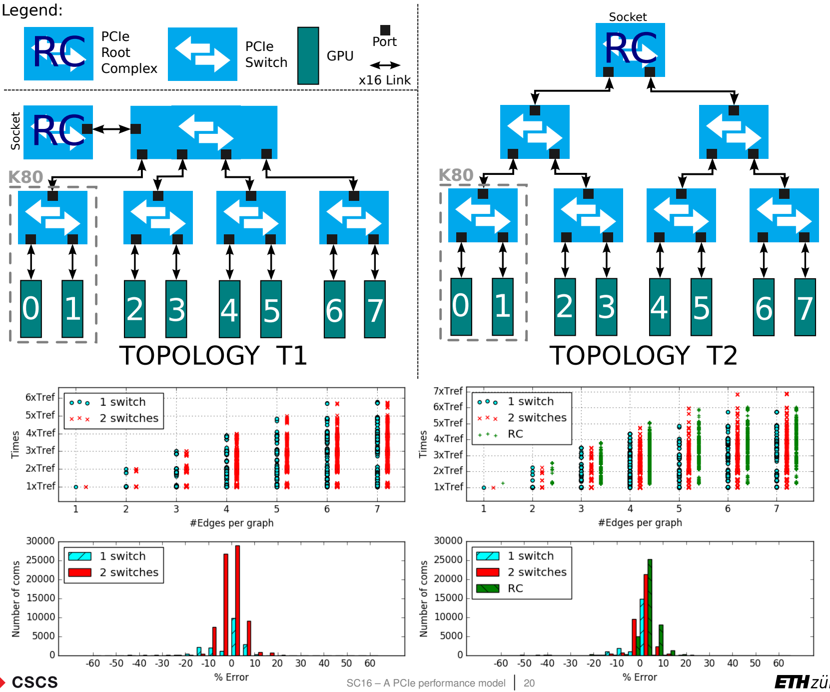
ETH zürich



Thank you for your attention.


1.88x slower for conflict 1
1.80x slower for conflict 2
1.44x slower for conflict 3
1.21x
Measured bandwidth [GB/s]
Message size [Byte]
0 → 1 or 0 → 3 alone
4 → 1 alone
0 → 3 in conflict 1
0 → 1 in conflict 2
0 → 1 in conflict 3
0 1 2 3 4 5 6 7 8 9 10 11 12
2 8 32 128 512 2K 8K 32K 128K 512K 2M 8M 32M

$$\tau = 1 - {}^{1}/_{1.21}$$


Legend:
RC
PCIe Root Complex
PCIe Switch
GPU
Port
x16 Link
Socket
K80
TOPOLOGY T1
TOPOLOGY T2
1 switch
2 switches
RC
Times
#Edges per graph
Number of coms
% Error