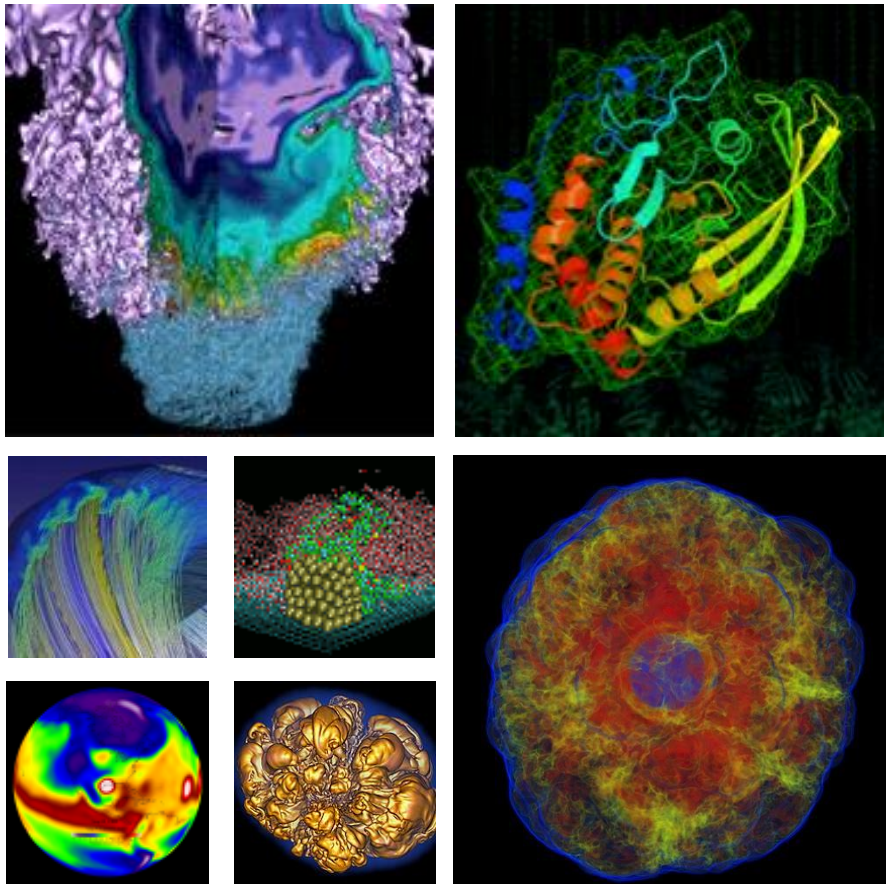
Performance Variability on Xeon Phi



Brandon Cook, Thorsten Kurth, Brian Austin, Samuel Williams, Jack Deslippe

June 22, 2017

Who cares?

- **Application developers**
  - understanding performance
  - reason effectively about optimizations
  - sound advice to application users
- **Users**
  - Efficient use of CPU allocations
  - Wasted cycles on terminated jobs
  - Correct estimates of campaign costs
- **Facilities**
  - System health
  - Advice for users
  - Utilization – scheduler efficiency

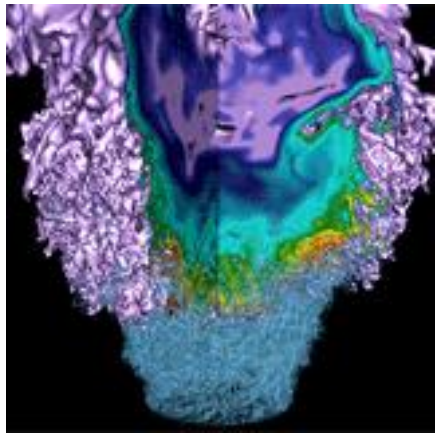
Cori at NERSC

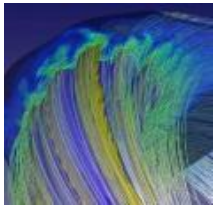
- **2388 Haswell**
  - 2x 16 core @ 2.3 GHz
  - 40 MB shared L3
  - 128 GB DDR
- **Cray Aries Interconnect**
  - dragonfly topology
- **9688 Xeon Phi (KNL) nodes**
  - 68 cores @ 1.4 GHz
  - 34 MB distributed L2
  - 96 GB DDR
  - 16 GB MCDRAM (on-package)

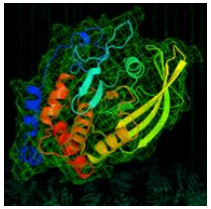
Cori at NERSC

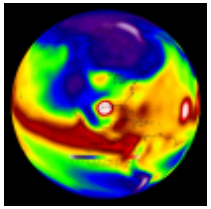
- 2388 Haswell
  - 2x 16 core @ 2.3 GHz
  - 40 MB shared L3
  - 128 GB DDR
- **Cray Aries Interconnect**
  - dragonfly topology
- **9688 Xeon Phi (KNL) nodes**
  - 68 cores @ 1.4 GHz
  - 34 MB distributed L2
  - 96 GB DDR
  - **16 GB MCDRAM (on-package)**

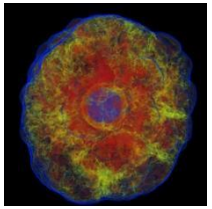
MCDRAM

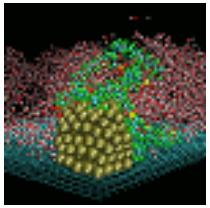




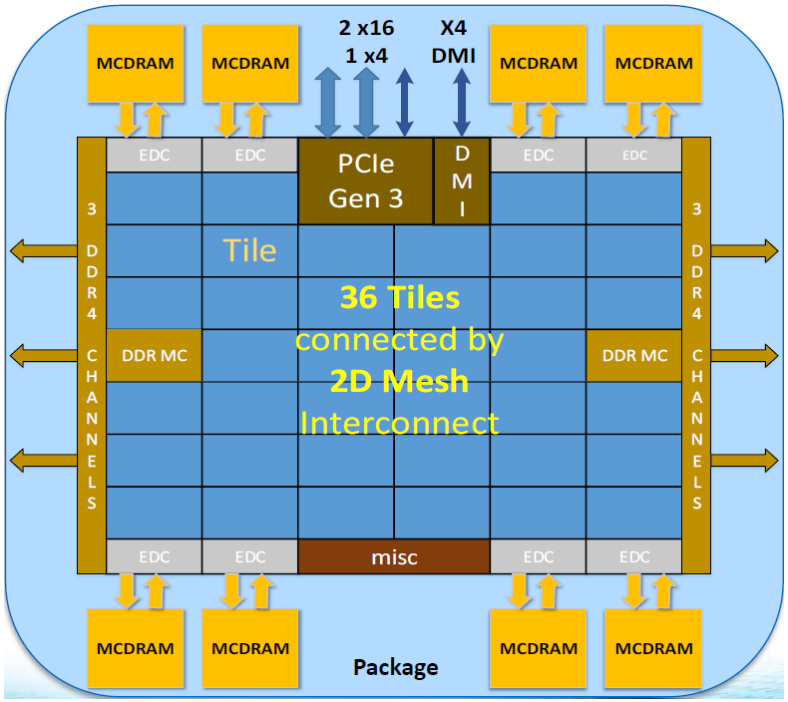








KNL is highly configurable



Cluster modes

- all-to-all
- quadrant
- SNC2/4

Memory modes

- flat
- cache
- hybrid

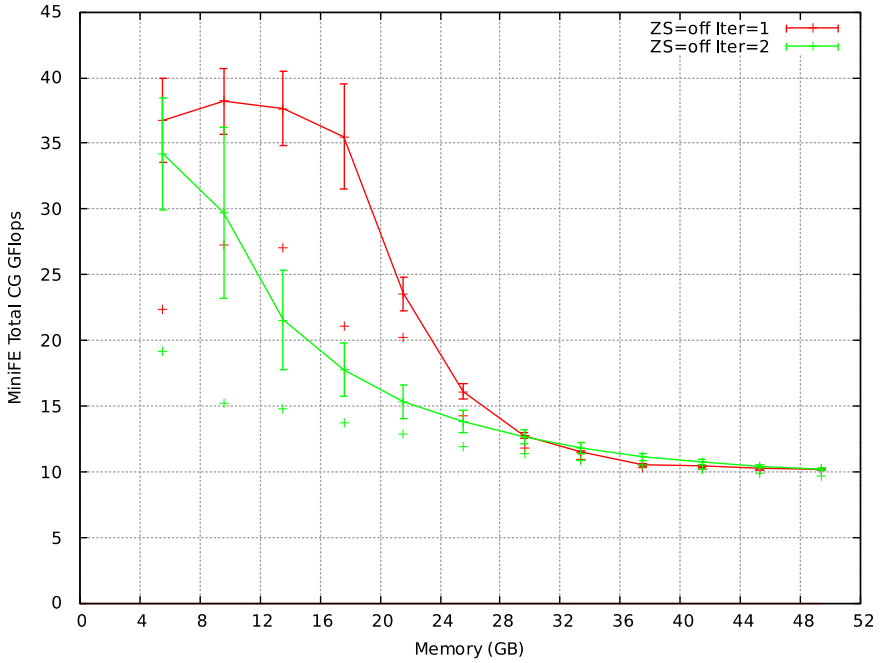
MCDRAM cache mode

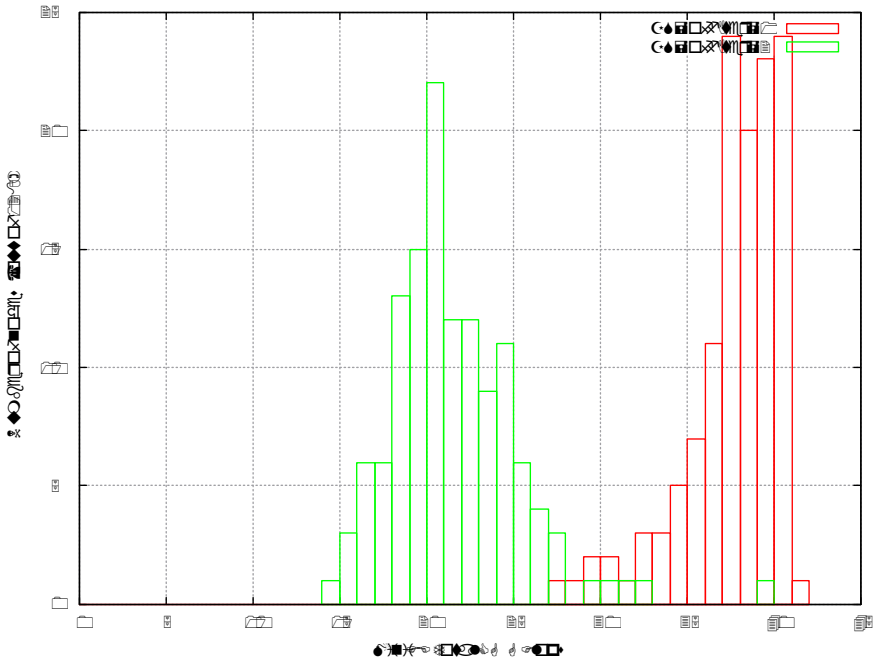
- **16GB MCDRAM cache**
- **single NUMA**
- **No code modification**
- **No NUMA programing or affinity issues (e.g. numactl)**
- **but?**

Cluster modes
- all-to-all
- **quadrant**
- SNC2/4

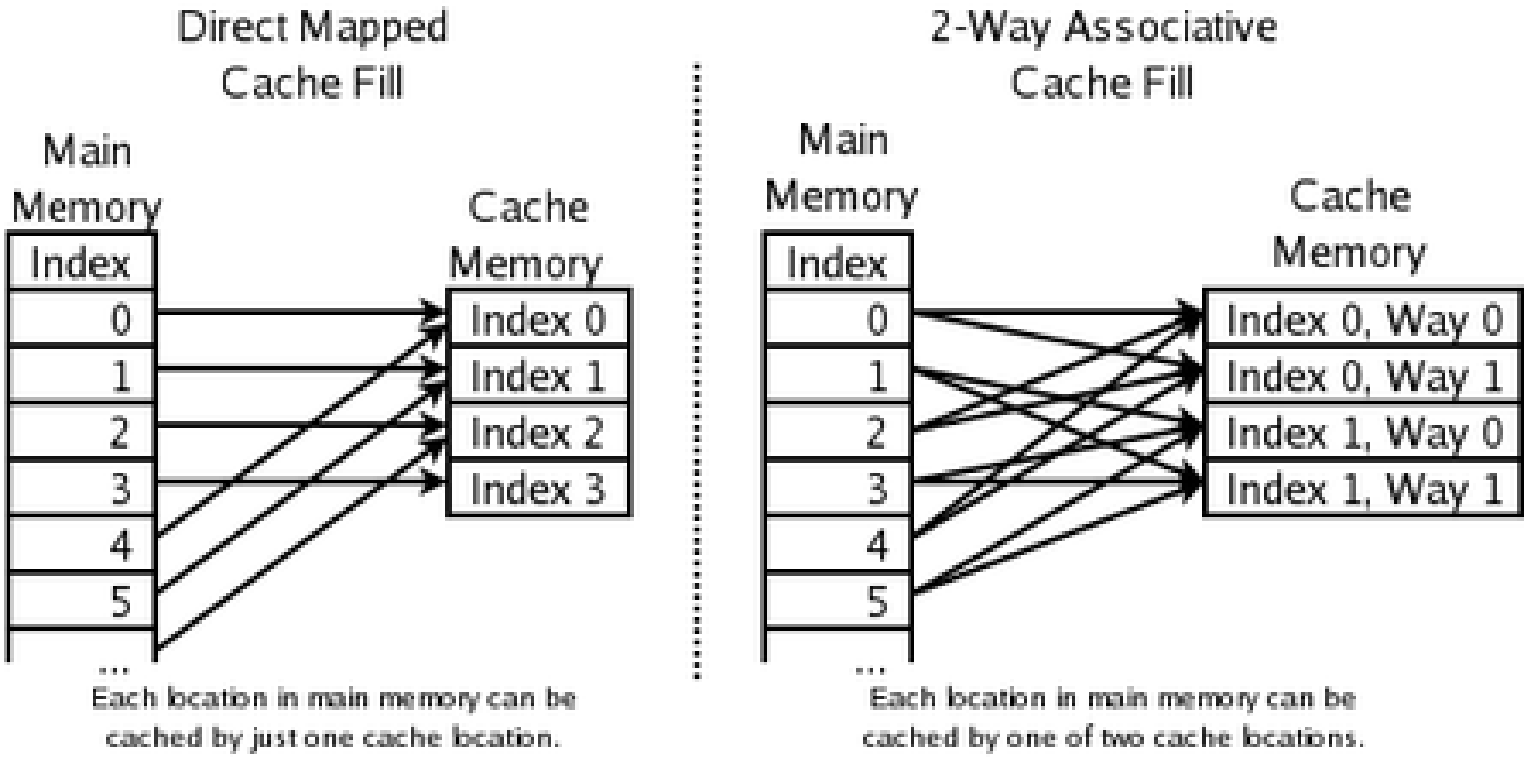
Memory modes
- flat
- **cache**
- hybrid

Variability in cache mode





Brief introduction to caches

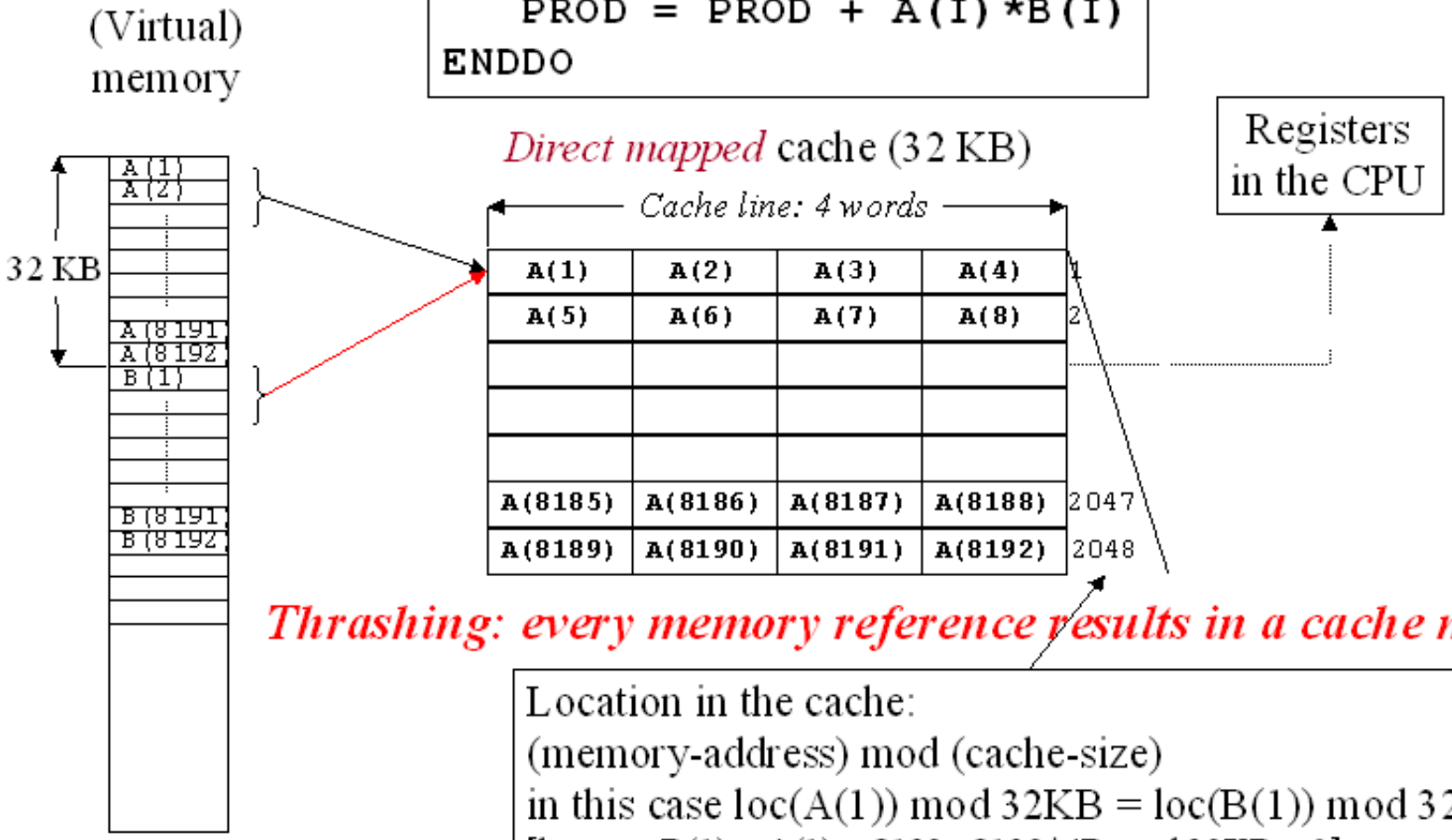


https://en.wikipedia.org/wiki/CPU_cache

KNL's MCDRAM cache is direct-mapped.

Direct-Mapped Caches: Thrashing

```
COMMON //A(8192), B(8192)
DO I=1,N
   PROD = PROD + A(I)*B(I)
ENDDO
```



Thrashing: every memory reference results in a cache miss

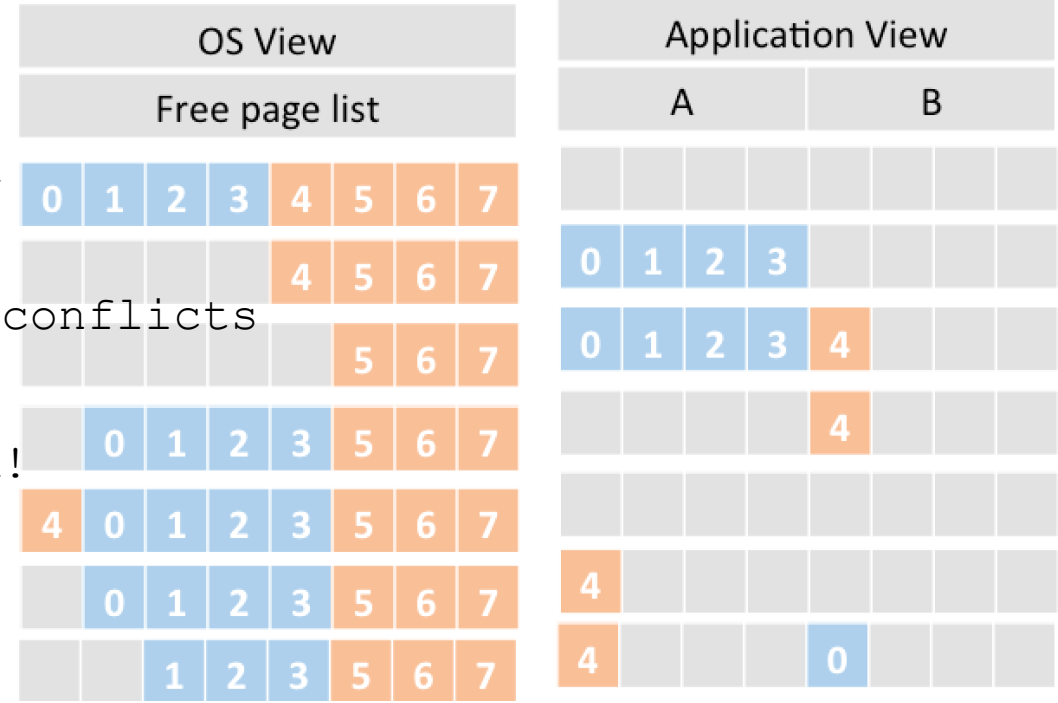
Location in the cache:
(memory-address) mod (cache-size)
in this case loc(A(1)) mod 32KB = loc(B(1)) mod 32KB
[because B(1) = A(1) + 8192; 8192*4B mod 32KB = 0]

http://sc.tamu.edu/help/power/powerlearn/html/ScalarOptnw/sld015.htm

Misses depend on free page list

- OS stores a list of free memory pages.
- Allocations are made from the top of the list.
- The free page list gets scrambled if memory is not freed in the order it was allocated.
- Consider a 16kB cache…

```
//Initial page list is sorted
A = malloc(16 kB) //4 pages
B = malloc( 4 kB) //Capacity conflicts
free(A)
free(B)           //Scrambled!
A = malloc( 4 kB)
B = malloc( 4 kB) //Conflict!
```



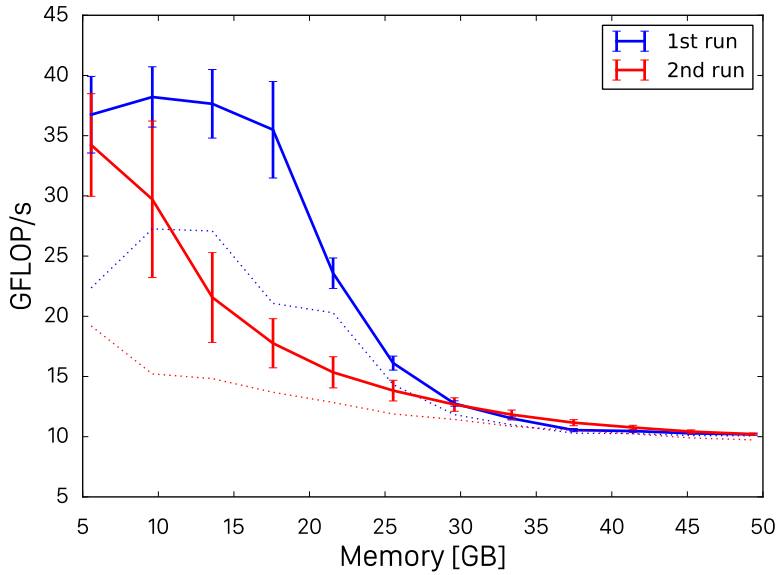
zonesort

Solution: sort the free page list*

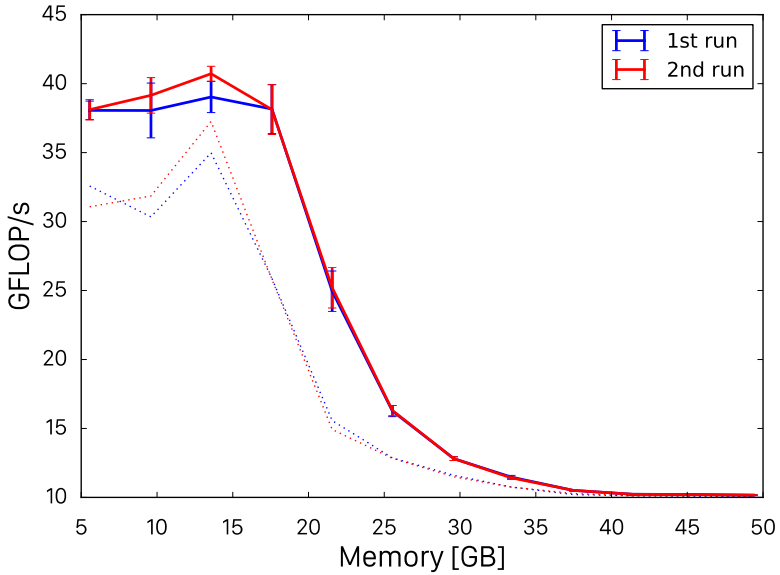
- **zonesort: kernel module provided by Intel**
- **At NERSC**
  - **called immediately before application launch**

benefit of zonesort for MiniFE

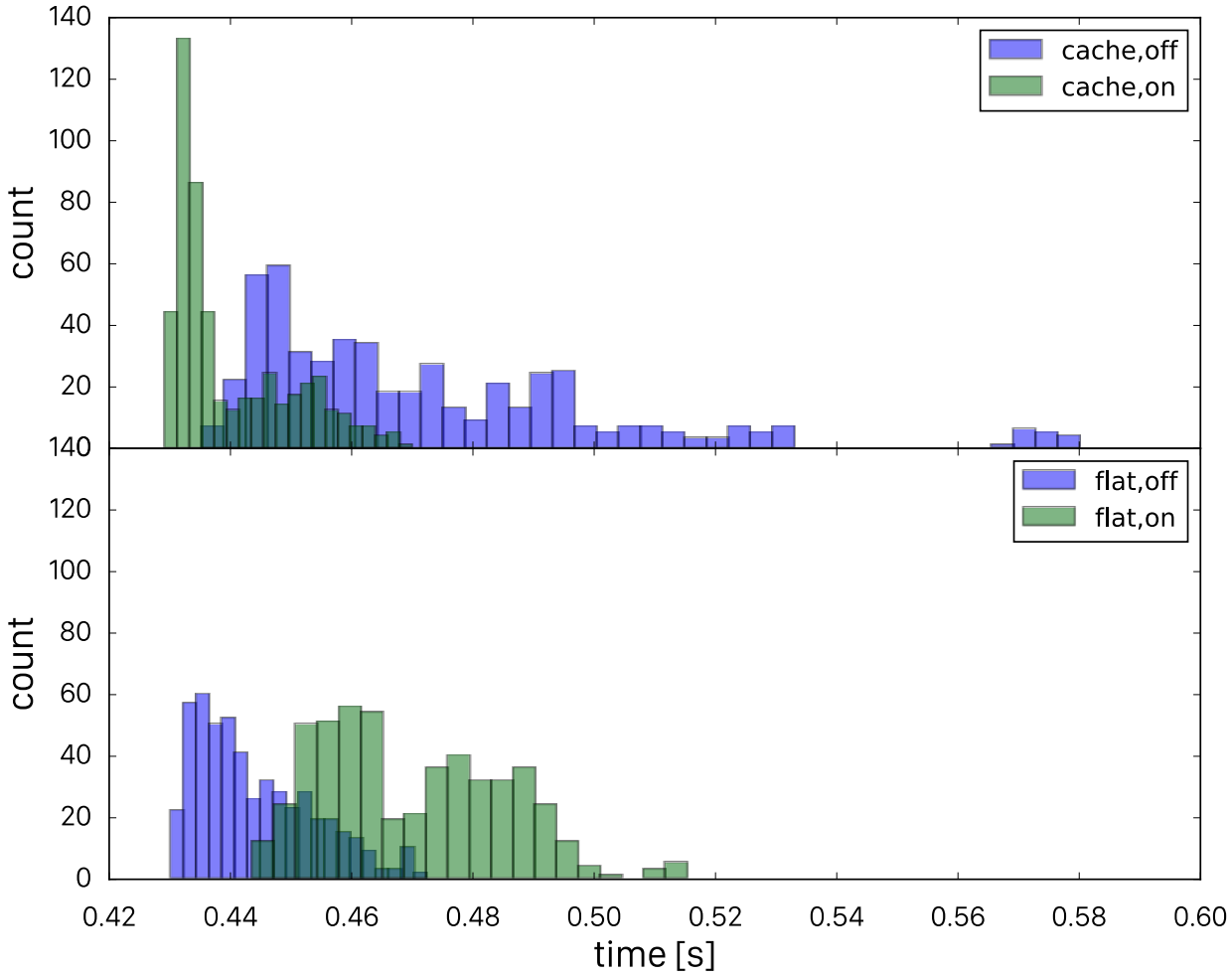
zonesort off



zonesort on



effect of zonesort for HPGMG



High Performance Geometric Multi-Grid

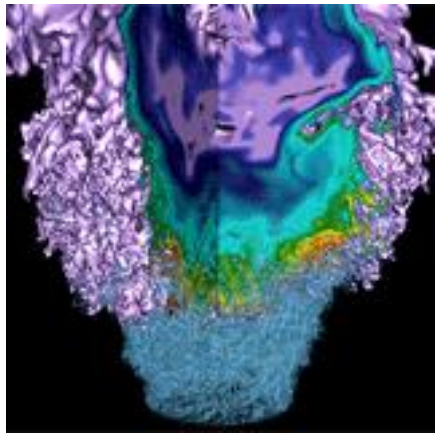
Highly instrumented

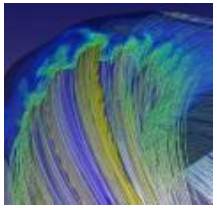
Perfectly load balanced problem

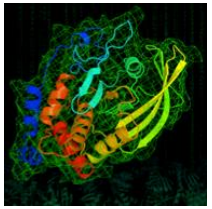
“smooth time”

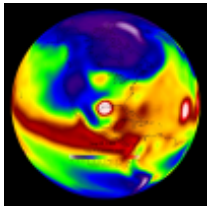
256^3 grid per rank

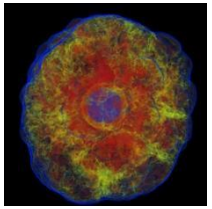
Job placement

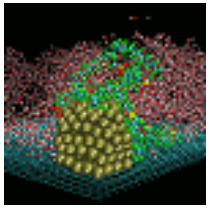












A Xeon Phi issue?

A Xeon Phi issue?

	Haswell	Xeon Phi
Flops	1.2 TFlops	3 TFlops
Memory Bandwidth	~100 GB/s	~400 GB/s
Memory Capacity	128 GB	96 GB
Capacity / bandwidth	1.28 s	0.24 s

More flops & lower memory capacity / bandwidth

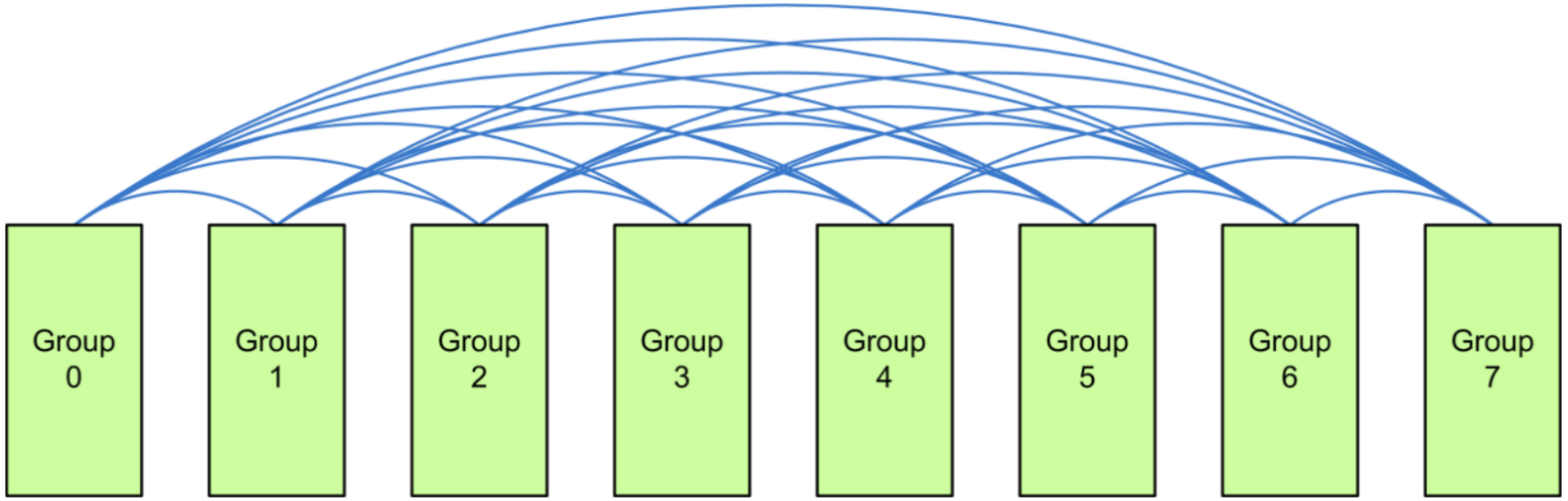
+

same network

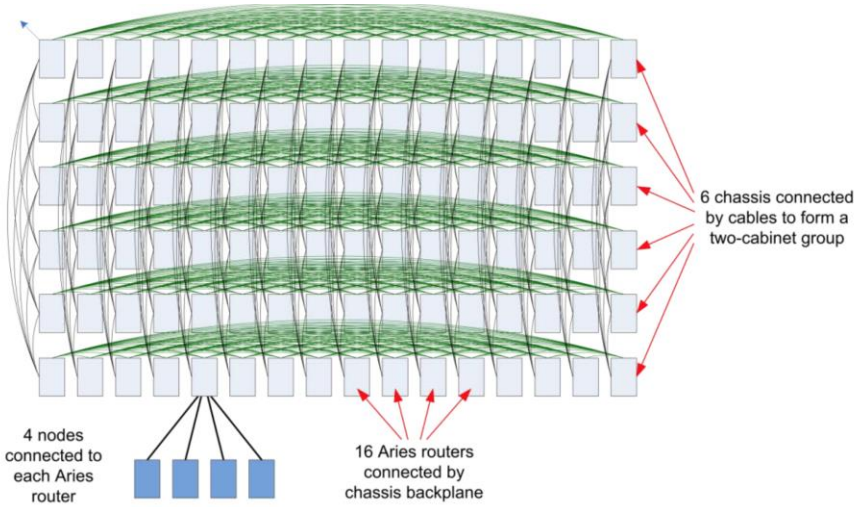
=

more pressure on network!

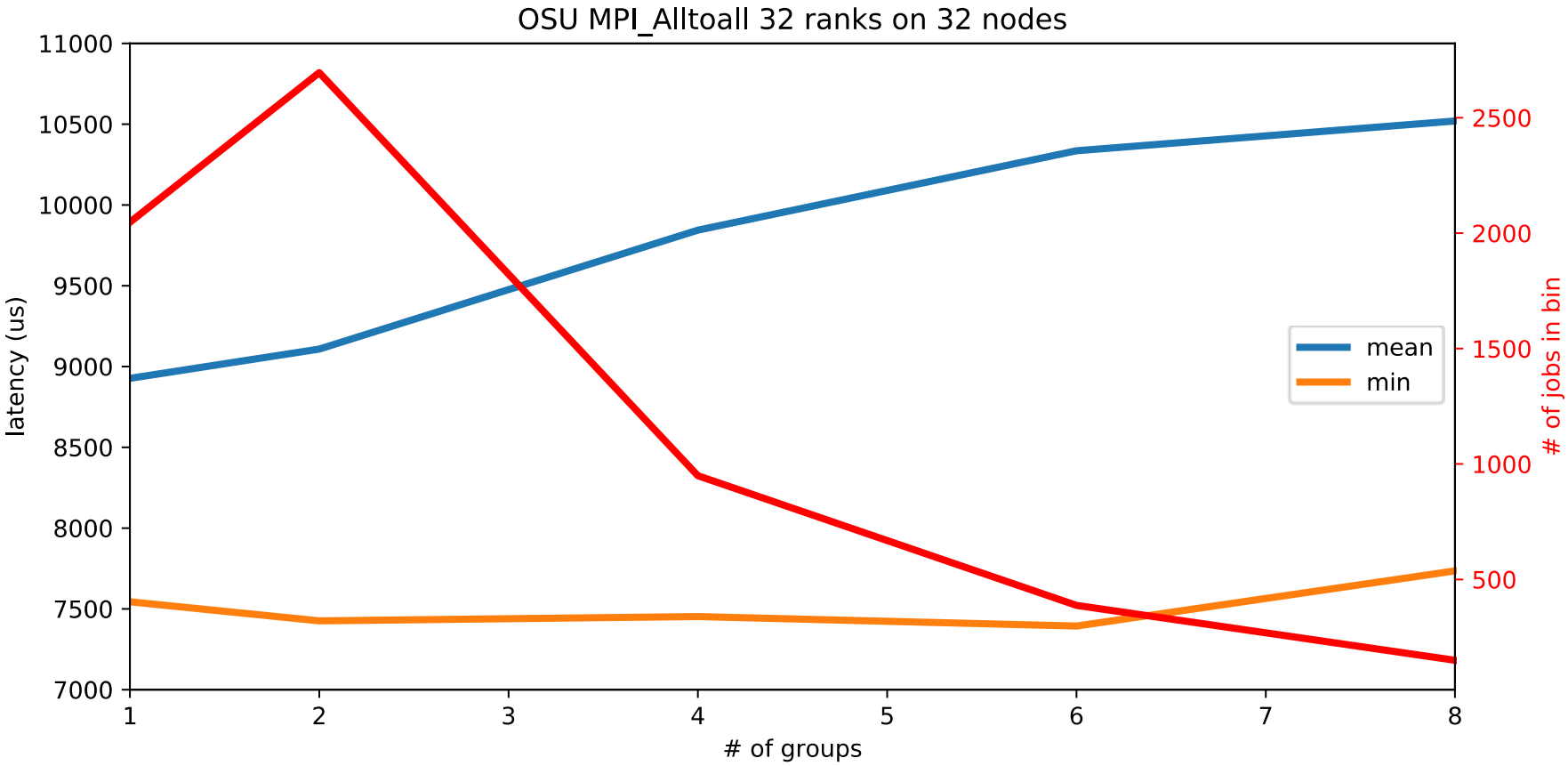
Aries topology



~386 nodes per group



Impact of # of groups



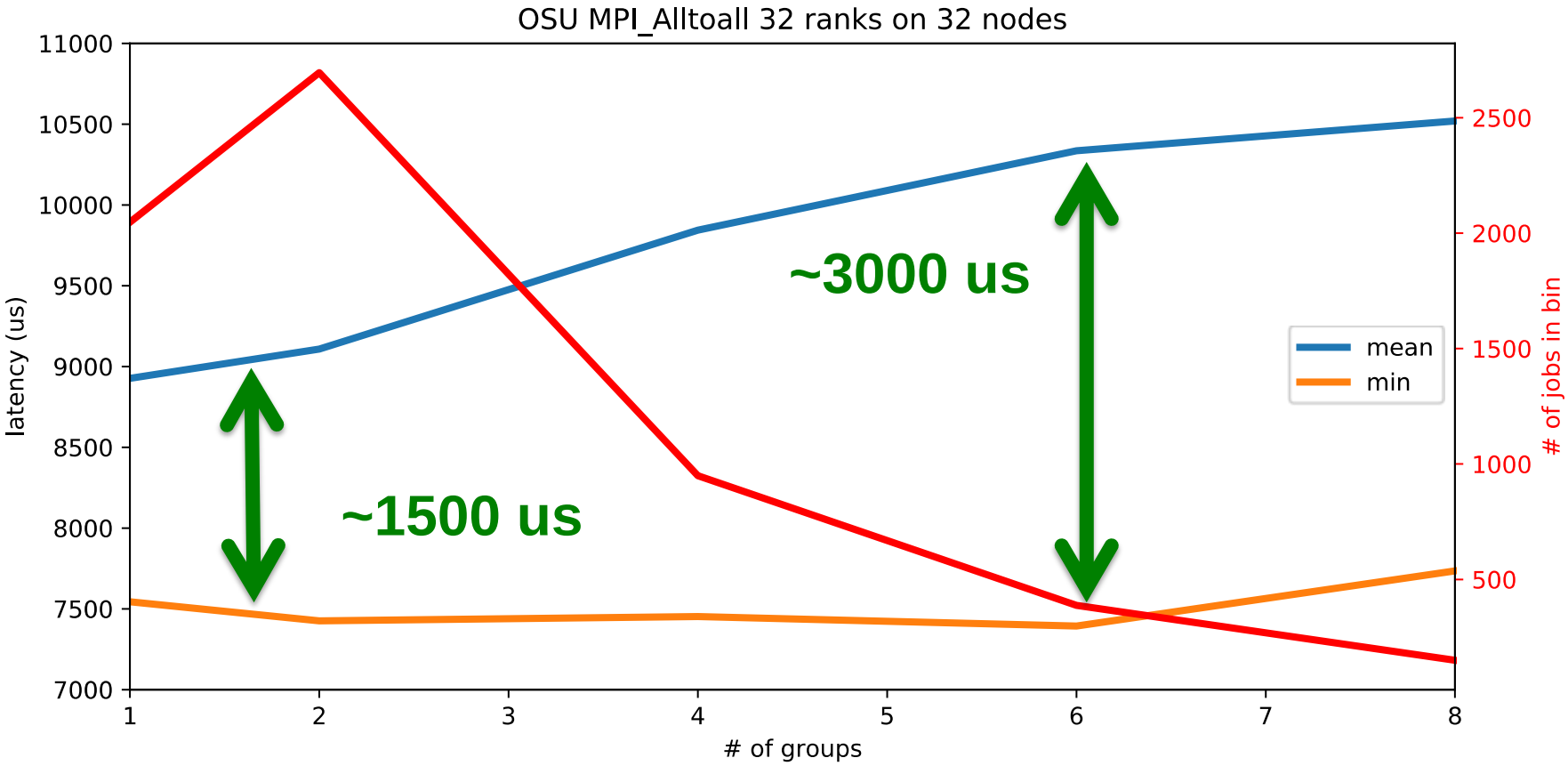
scheduler topology awareness

sbatch --switches=<count>[@<max-time>]

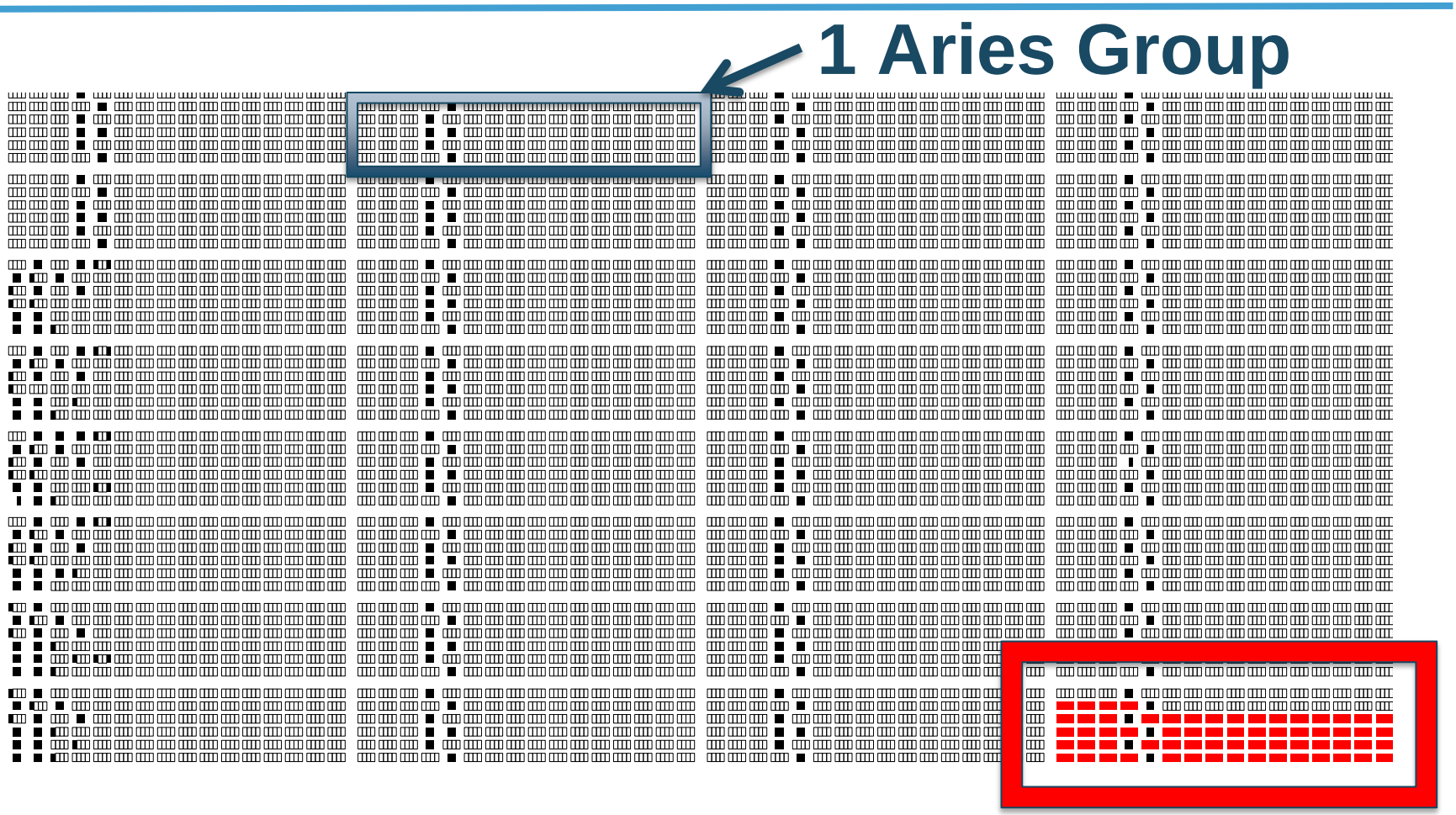
<count> = # of groups

<max-time> = time to wait for constraint

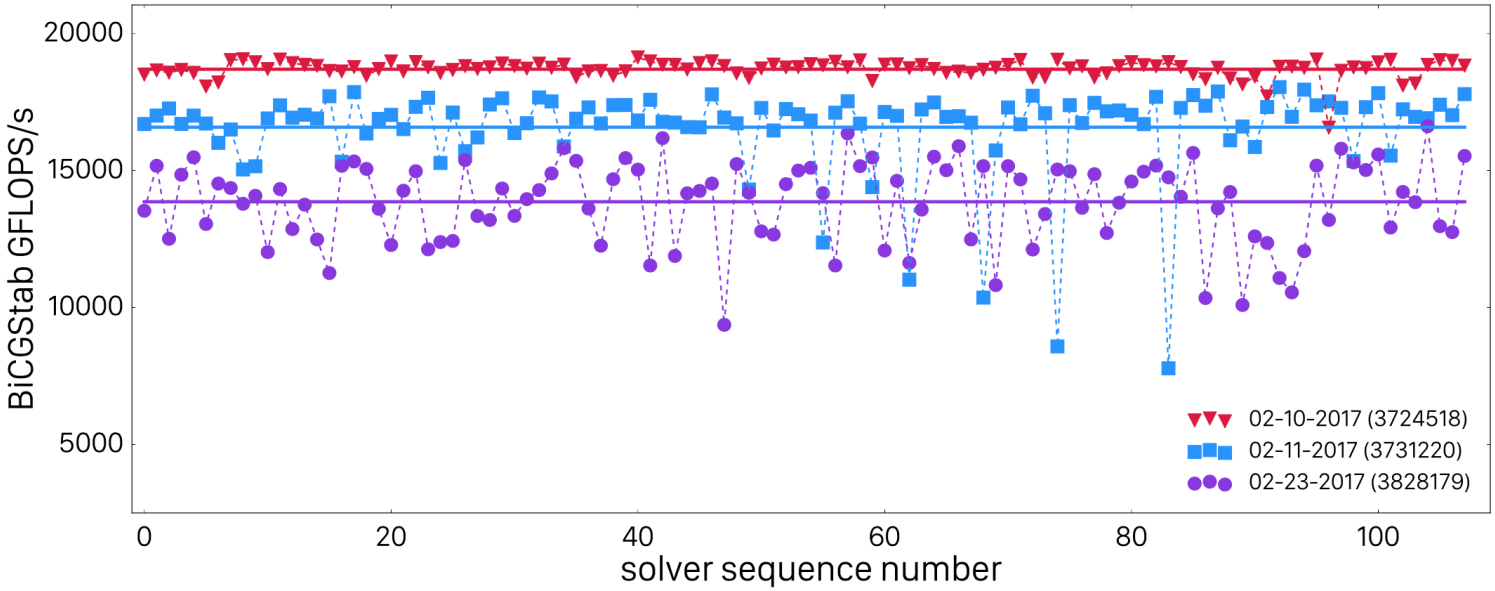
Impact of # of groups

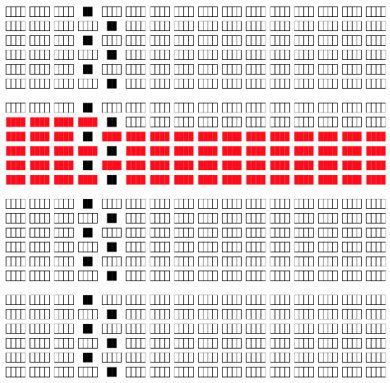


Job Placement

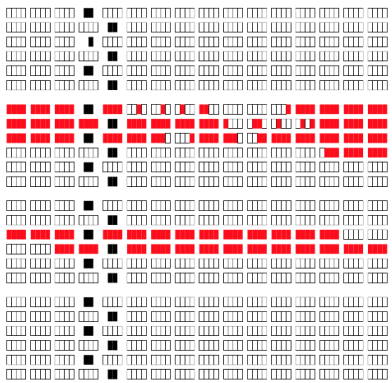


Chroma HMC – 256 nodes

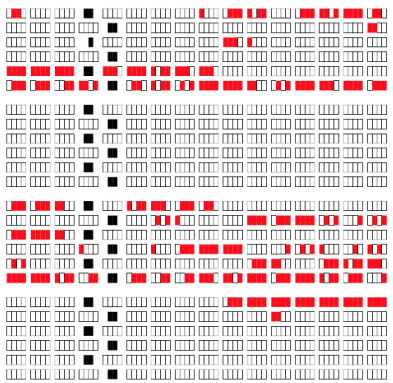




(b) job 3724518



(c) job 3731220



(d) job 3828179

Summary

- **MCDRAM Cache**
  - direct map cache
  - leads to cache conflicts
  - Intel zonesort
- **Job placement**
  - Bad placement introduces extra hops for data
  - Bad placement increases potential for interference
  - SLURM topology control helps (# of nodes < 350)
- **Not covered in this talk**
  - **IO! (burst buffer on compute fabric)**
  - Identification of network “Aggressors”
  - frequency scaling (DVFS)

National Energy Research Scientific Computing Center