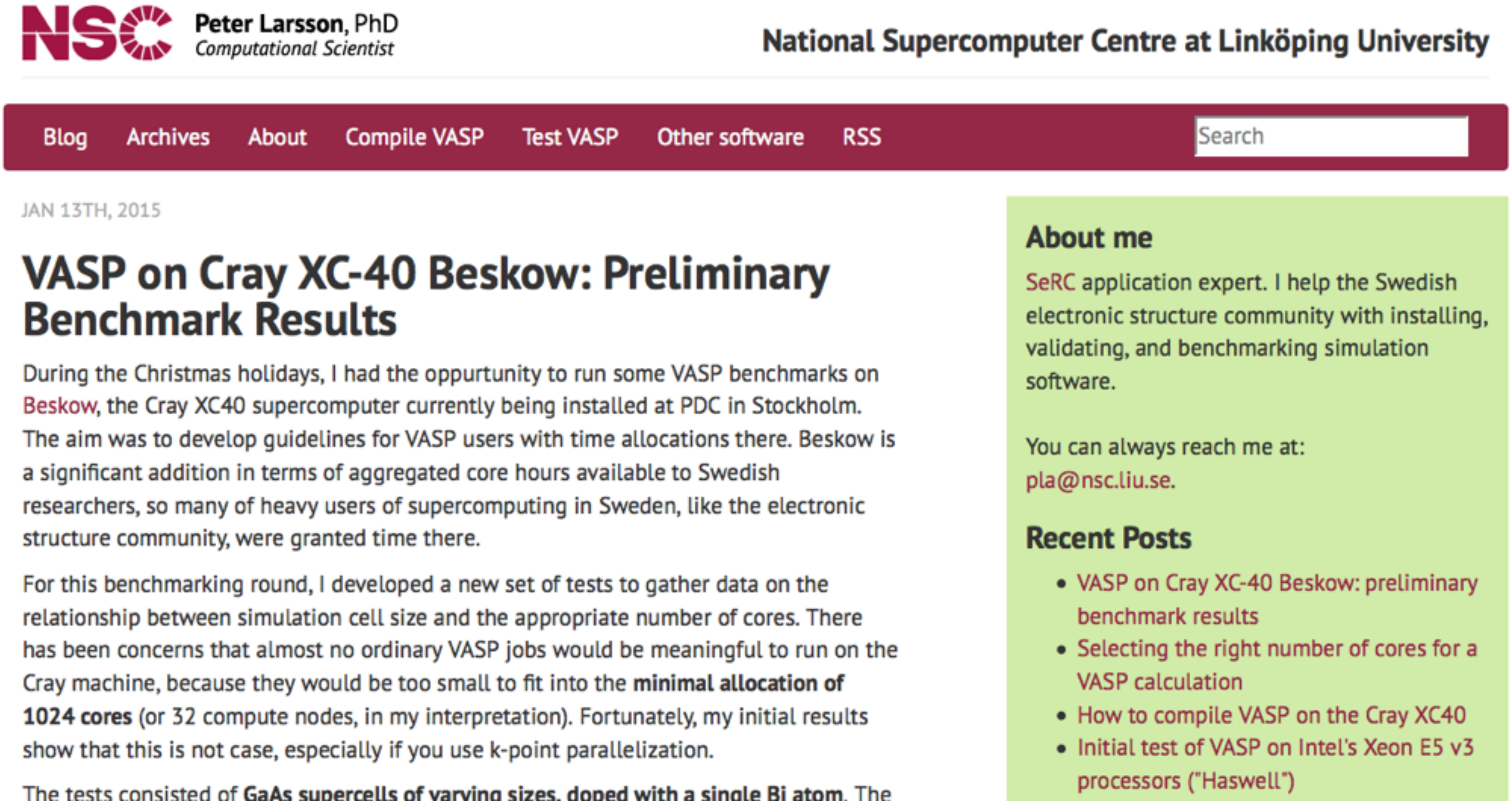
VASP runtime settings

General observations

http://www.nsc.liu.se/~pla/



Peter Larsson, PhD
Computational Scientist

National Supercomputer Centre at Linköping University

JAN 13TH, 2015

VASP on Cray XC-40 Beskow: Preliminary Benchmark Results

During the Christmas holidays, I had the oppurtunity to run some VASP benchmarks on Beskow, the Cray XC40 supercomputer currently being installed at PDC in Stockholm. The aim was to develop guidelines for VASP users with time allocations there. Beskow is a significant addition in terms of aggregated core hours available to Swedish researchers, so many of heavy users of supercomputing in Sweden, like the electronic structure community, were granted time there.

For this benchmarking round, I developed a new set of tests to gather data on the relationship between simulation cell size and the appropriate number of cores. There has been concerns that almost no ordinary VASP jobs would be meaningful to run on the Cray machine, because they would be too small to fit into the **minimal allocation of 1024 cores** (or 32 compute nodes, in my interpretation). Fortunately, my initial results show that this is not case, especially if you use k-point parallelization.

The tests consisted of **GaAs supercells of varying sizes, doped with a single Bi atom**. The

About me

SeRC application expert. I help the Swedish electronic structure community with installing, validating, and benchmarking simulation software.

You can always reach me at: pla@nsc.liu.se.

Recent Posts

- VASP on Cray XC-40 Beskow: preliminary benchmark results
- Selecting the right number of cores for a VASP calculation
- How to compile VASP on the Cray XC40
- Initial test of VASP on Intel's Xeon E5 v3 processors ("Haswell")

Efficiency:

Running as many jobs as possible within a given allocation of computer time.

Speed:

The amount of time (in real time, "human time") to run a specific simulation from the time it starts.

Time-to-solution:

The amount of time (in real time, "human time") to run a specific simulation from the time it starts + the time waiting in the queue.

Dear Professor,

We regret to inform you that we have decided to cut your time allocation from 1,000,000 core hours/month to 800,000 core hours/month.

Best regards,
SNAC

How many cores?

https://www.nsc.liu.se/~pla/blog/2015/01/12/vasp-how-many-cores/

Easy way:

Number of cores = number of atoms

How many cores?

Determined by number of bands per core:
1 band/core: usually doesn't work
2 bands/core: ca 50% parallel efficiency
8 bands/core: safe, 80%+ efficiency

- Check the number of bands (NBANDS). The **number of bands divided by 8** is a good starting guess for the number of cores to employ in your calculation.

- If you have more than one k-point, **set KPAR to the number of compute nodes** or the number of k-points, whichever is the smallest number. Then double the amount of cores determined in the previous step.

- Make a test run and **check the value of NGZ,** it should be an even number and sufficiently big (larger than 3*cores/node). Adjust either the basis set size or the number of cores/node.

NPAR & NCORE

- Applies to the Davidson/RMM-DIIS algorithms
  i.e not hybrid calculations

- Controls/activates band parallelization

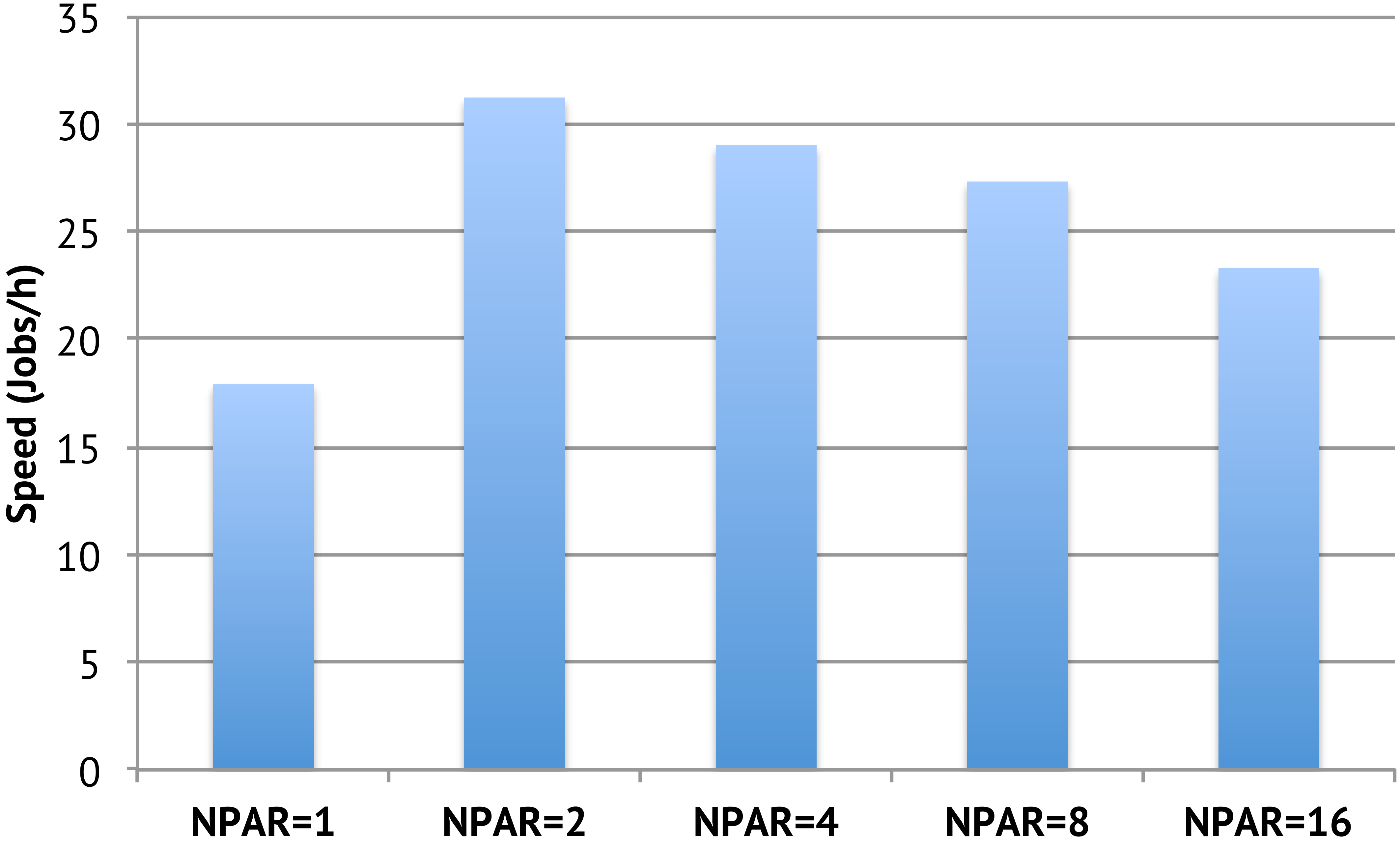
- NPAR = 1: no parallelization over bands = **BAD**
  but saves memory

- NPAR = number of compute nodes (x2)

- **NCORE = cores per compute node** (or cores per socket)
  much easier to use!

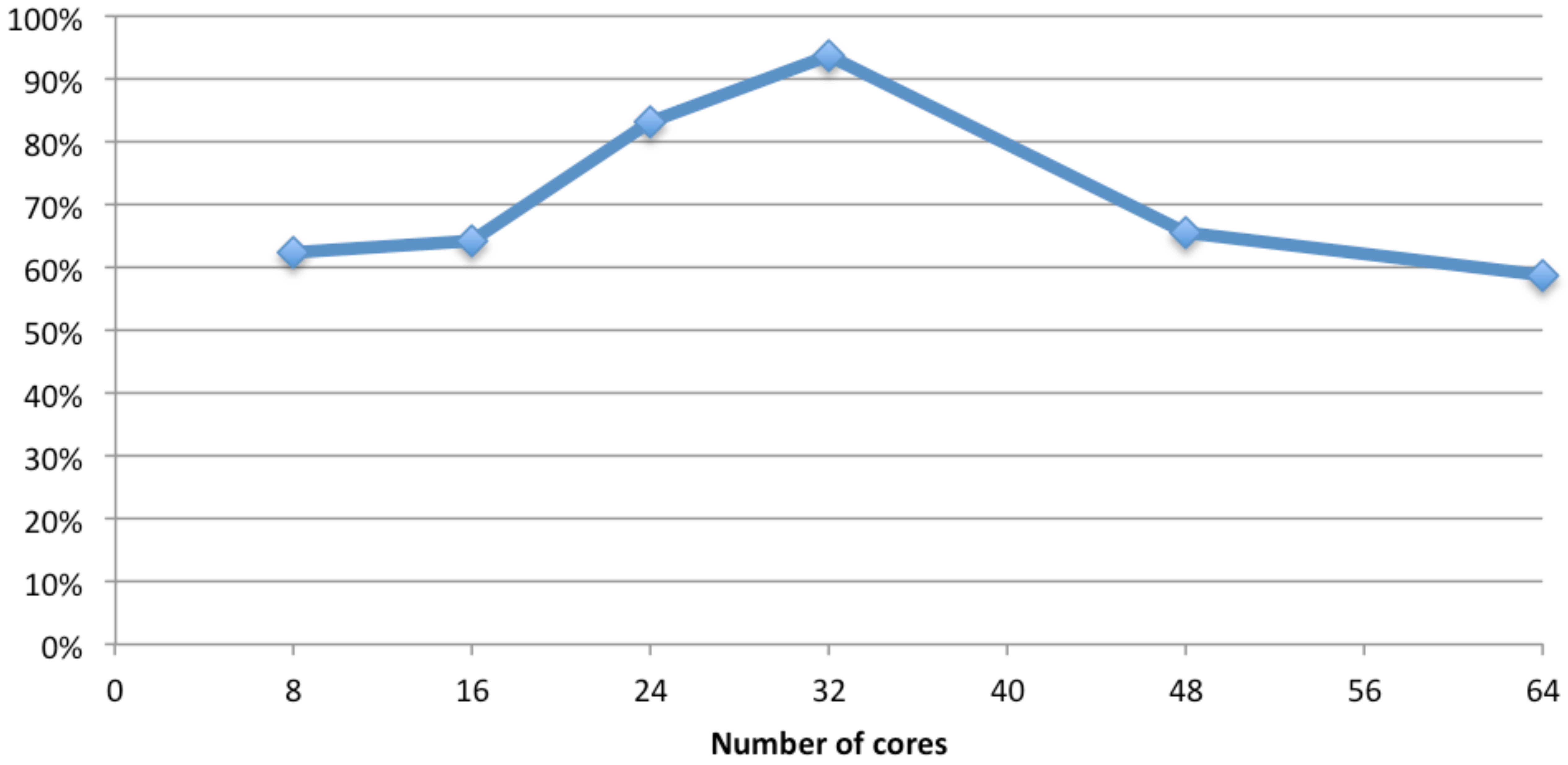
Li2FeSiO4 (128 atoms) on Triolith

Standard DFT / 4 compute nodes / 64 cores

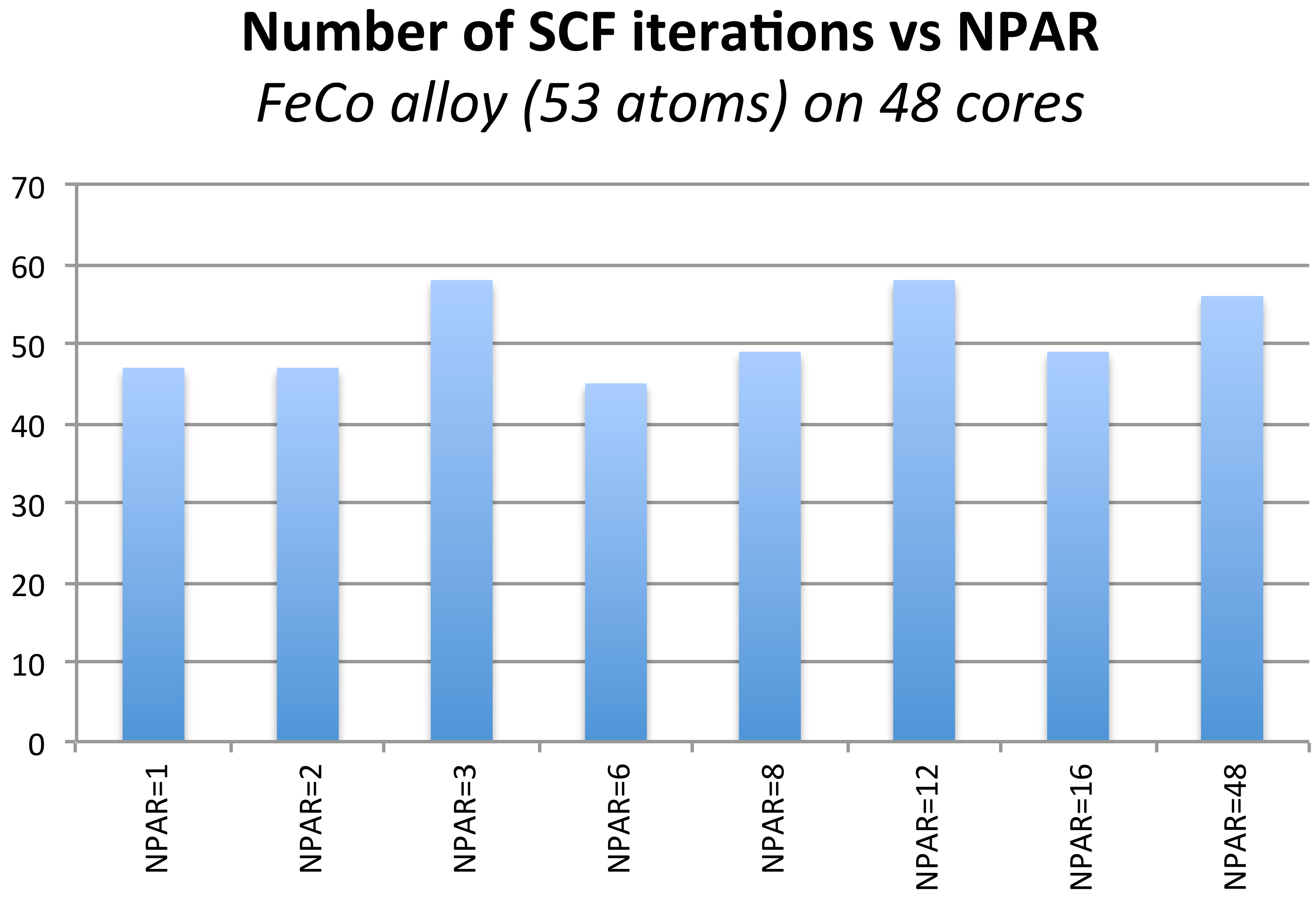


Speed-up with optimal NPAR

FeCo alloy (53 atoms)



NPAR: caveat


Number of SCF iterations vs NPAR
FeCo alloy (53 atoms) on 48 cores
70
60
50
40
30
20
10
0
NPAR=1
NPAR=2
NPAR=3
NPAR=6
NPAR=8
NPAR=12
NPAR=16
NPAR=48

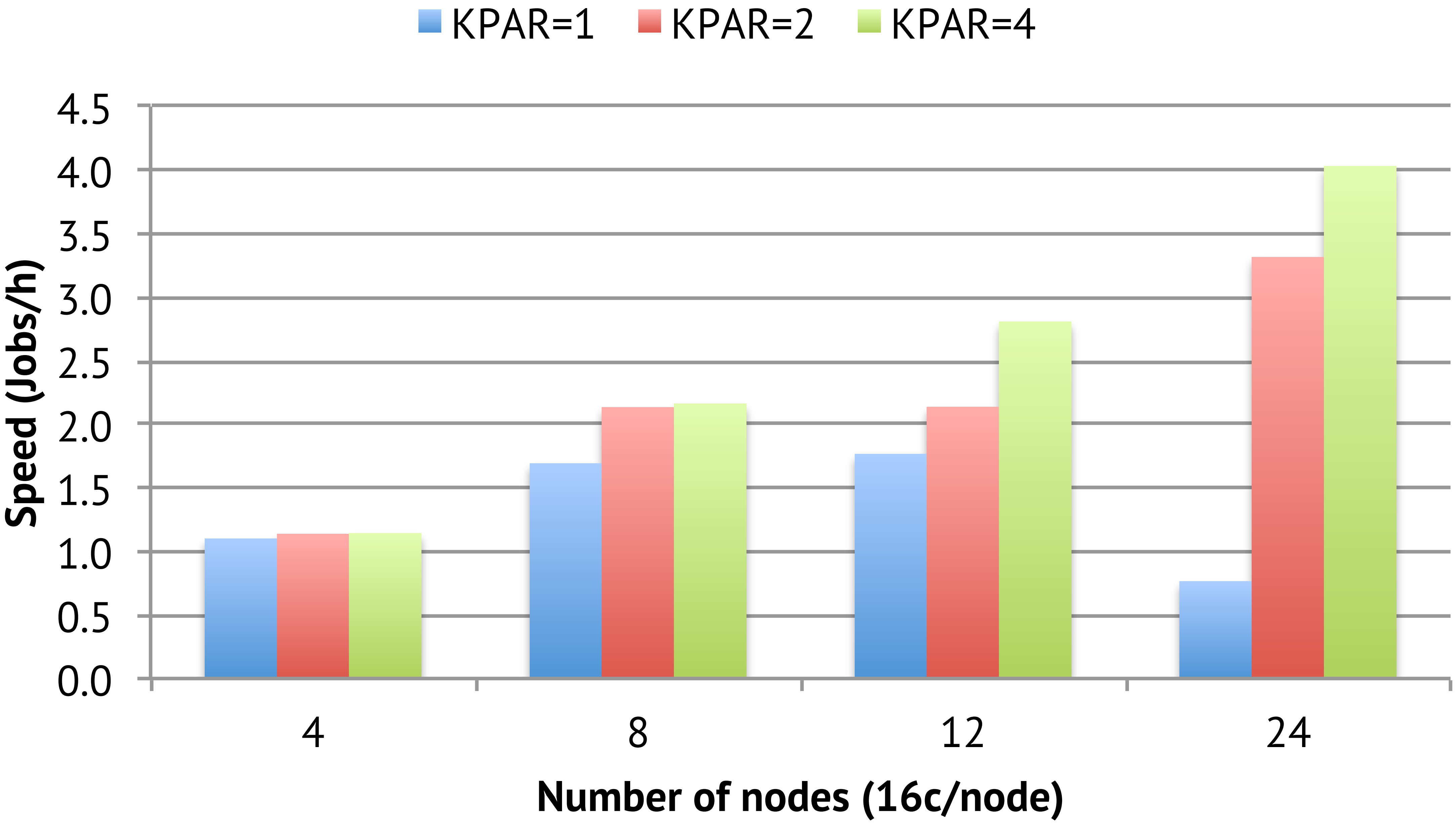
K-point parallelization ("KPAR")

- Controls parallelization over k-points (version 5.3.2+)
- **Great, you can run on at least 2x number of cores**
- Especially great for hybrid-DFT jobs
- Not so great for metals with 100-1000s of k-points

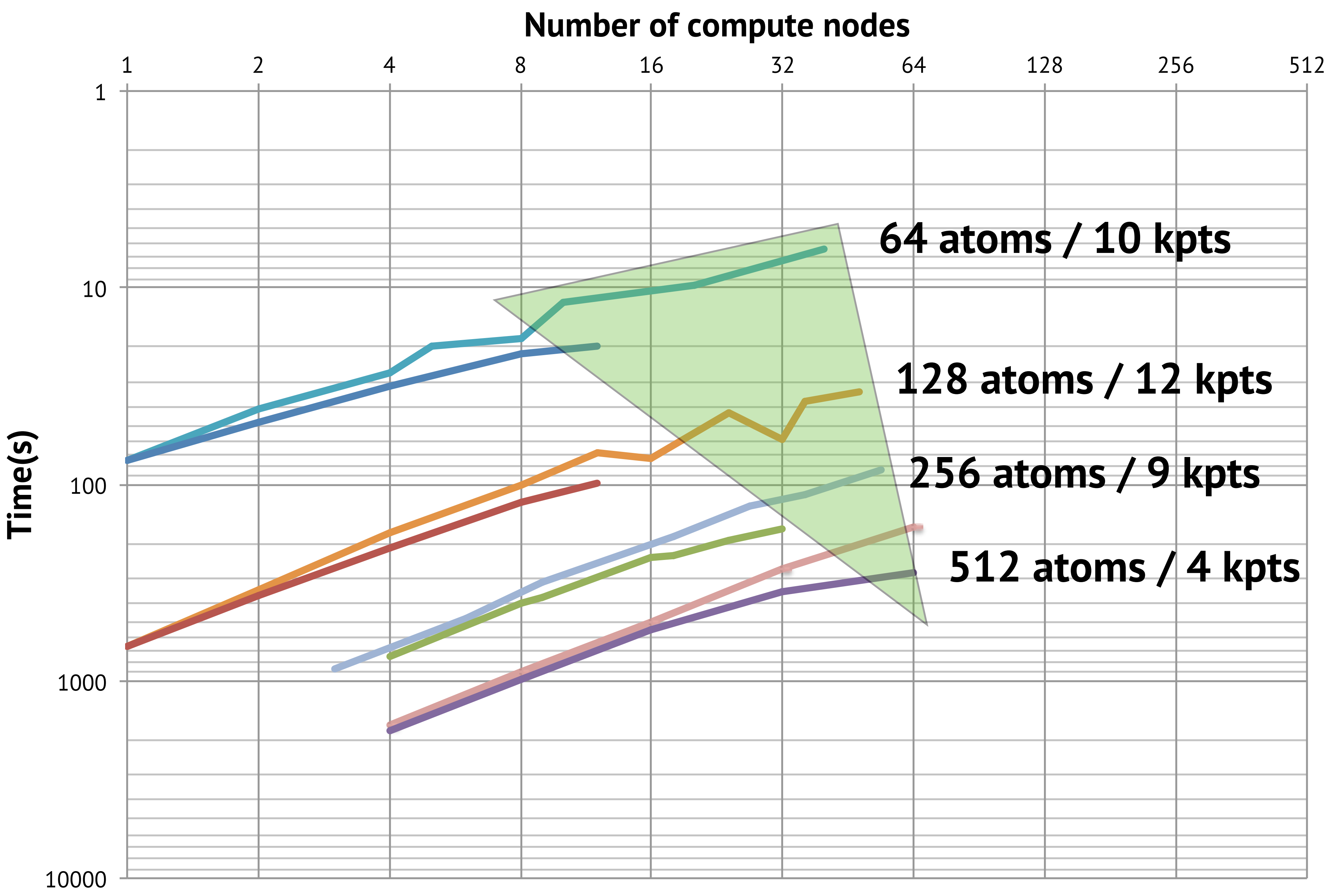
KPAR = *"number of k-points treated in parallel"*
cores / KPAR = *"group size" = cores/node?*
KPAR = nodes

MgO (63 atoms) on Triolith

HSE06 hybrid calculation with 4 k-points



GaAsBi supercells on Beskow: KPAR vs no KPAR



NSIM

- Blocking mode for RMM-DIIS algorithm
(so only applies for ALGO=Fast/VeryFast)

- Not terribly important nowadays (±5%).

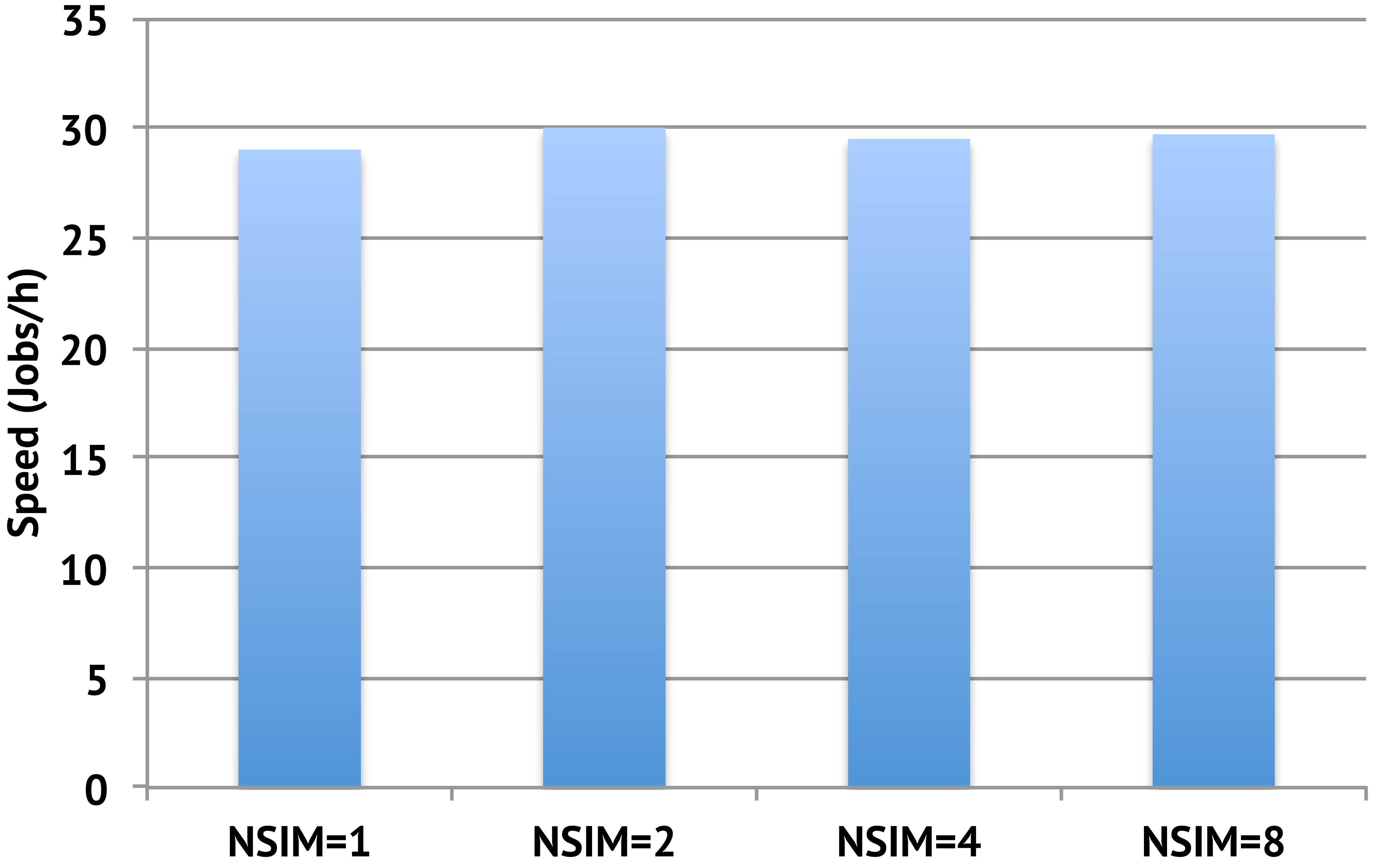
- **NSIM = 1** for single-node jobs on Triolith

- **NSIM = 2** on Beskow

- **NSIM = 4** for older systems (incl. AMD). This is VASP's default.

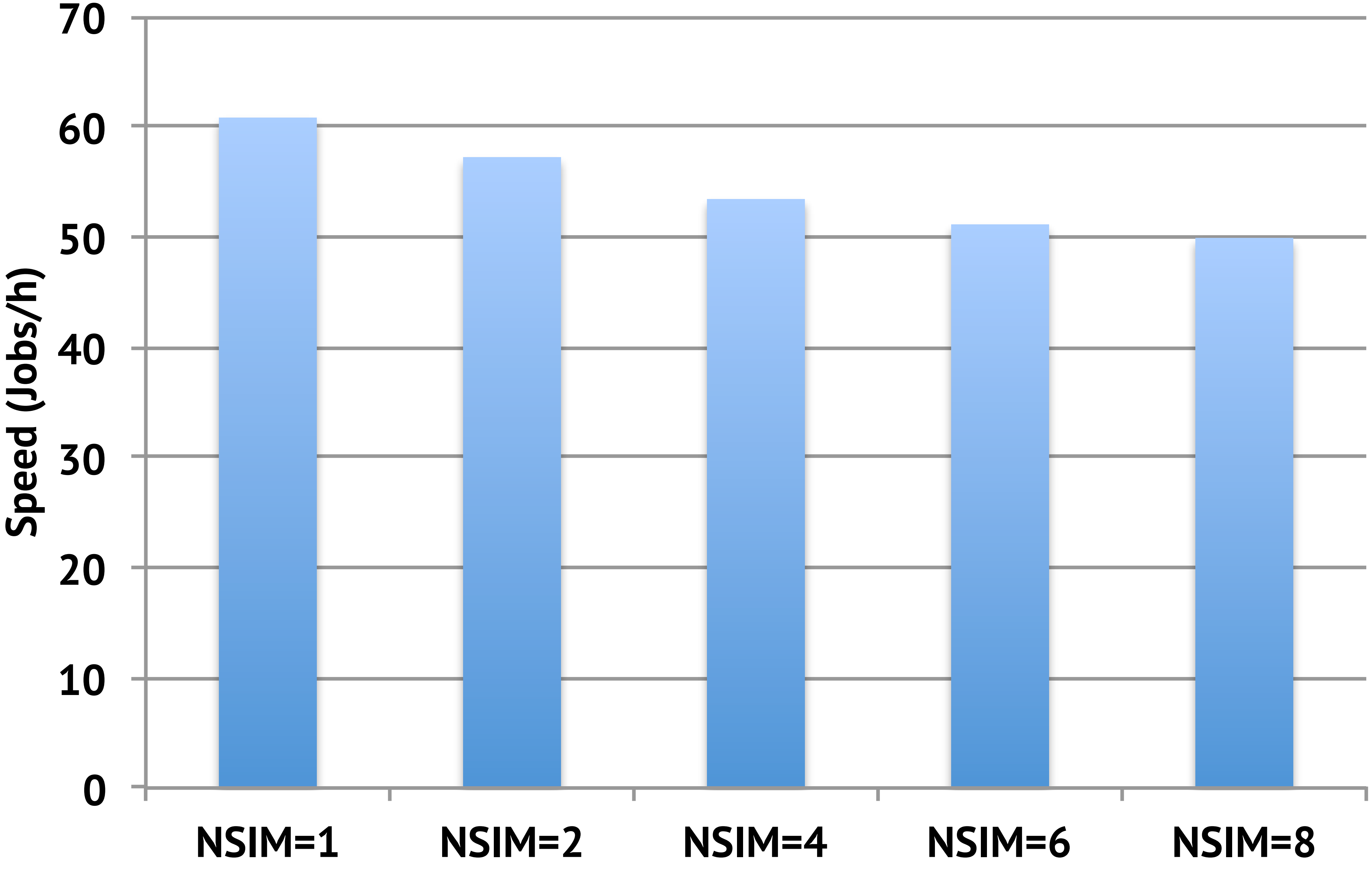
Li2FeSiO4 (128 atoms) on Triolith

Standard DFT / 4 compute nodes / 64 cores



PbSO4 (24 atoms) on Triolith

Standard DFT / 1 compute node / 16 cores


Speed (Jobs/h)
NSIM=1
NSIM=2
NSIM=4
NSIM=6
NSIM=8