VASP benchmark on BrENIAC

Kurt Lejaeghere – Arthur De Vos – Sam De Waele

1. Background

BrENIAC contains 580 nodes with 28 cores each, which are of the Broadwell E5-2680v4 type. Each node has 128 or 256 GB RAM and consists of 2 NUMA regions of 14 cores. The network is connected through an Infiniband EDR 2:1 connection.

To benchmark the performance of VASP (module VASP/5.4.1-intel-2016a) on BrENIAC, three very different test systems were considered:

- A doubled $Fe_{16}N_2$ unit cell with one N atom removed
  (35 atoms, 224 bands, 196 irreducible k-points, vasp_std)
  designated by tag METAL

- a Ge semiconductor surface with Pt atoms adsorbed
  (100 atoms, 336 bands, 8 irreducible k-points, vasp_std)
  designated by tag SEMI

- the metal organic framework UiO-66 with two missing linker defects
  (420 atoms, 1120 bands, 1 irreducible k-point, vasp_gam)
  designated by tag PORE

2. Optimal parallelization on 1 node

VASP has the possibility to parallelize over k-points and, for a given k-point, over electronic bands. In general, parallelization over k-points is more efficient, since it requires almost no communication between subprocesses. However, it also substantially increases the memory requirements, since the calculation of the wavefunction at 1 k-point is based on knowledge of all energy levels at that k-point. The memory needed therefore increases when more k-points are computed simultaneously (KPAR). Analogously, parallelization within 1 band occurs by grouping blocks of plane waves in diagonalization routines and allows spreading the memory even thinner. It is more favourable for the memory requirements to devote more cores to a single electronic band (NCORE), equivalent with fewer bands per node, but this behaviour is less distinct.

Table I: Walltime of a calculation of METAL, SEMI and PORE on 1 node, depending on the parallelization settings (number of k-points treated simultaneously, KPAR, and number of cores per band, NCORE).

wall time METAL [s]	NCORE = 1	NCORE = 7	NCORE = 14	NCORE 28
KPAR = 1	9863	6772	6402	6924
KPAR = 2	8654	6515	5601	
KPAR = 4	8435	6369		

wall time SEMI [s]	NCORE = 1	NCORE = 7	NCORE = 14	NCORE 28
KPAR = 1	777	770	710	687
KPAR = 2	775	769	685	
KPAR = 4	720	729		

wall time PORE [s]	NCORE = 1	NCORE = 7	NCORE = 14	NCORE 28
KPAR = 1	4900	4091	4059	3809

Table II: Memory usage per core for a calculation of METAL, SEMI and PORE on 1 node, depending on the parallelization settings (number of k-points treated simultaneously, KPAR, and number of cores per band, NCORE).

mem METAL [MB]	NCORE = 1	NCORE = 7	NCORE = 14	NCORE 28
KPAR = 1	1499	933	883	896
KPAR = 2	2002	1486	1446	
KPAR = 4	3091	2601		

mem SEMI [MB]	NCORE = 1	NCORE = 7	NCORE = 14	NCORE 28
KPAR = 1	328	199	193	187
KPAR = 2	424	307	297	
KPAR = 4	643	525		

mem PORE [MB]	NCORE = 1	NCORE = 7	NCORE = 14	NCORE 28
KPAR = 1	736	406	369	352

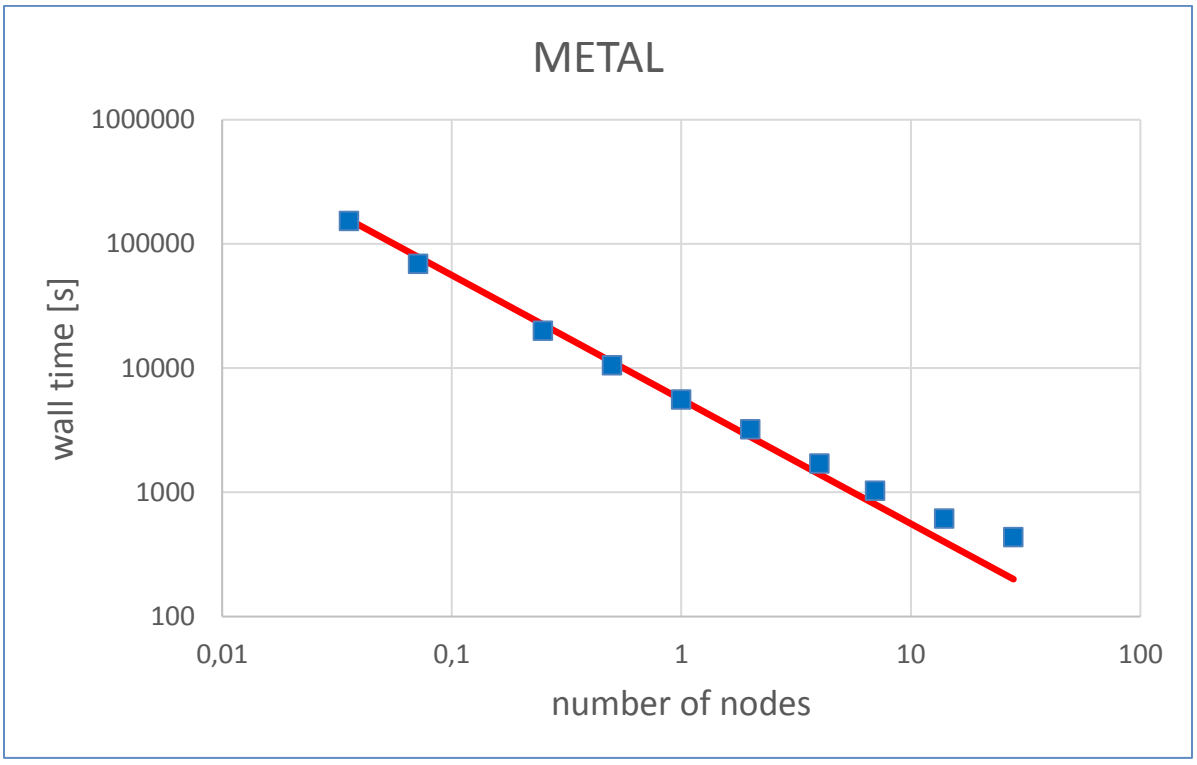
In terms of timing, we see that a higher order of k-point parallelization reduces the required wall time. However, it is not immediately clear which band parallelization is optimal. Many systems benefit from NCORE = 7 or 14, in line with the use of 1 shared memory per band, but for specific systems and number of plane waves, NCORE = 1 may become most favourable (e.g. when increasing the number of plane waves for SEMI). We can only conclude that the *best tradeoff between k-point parallelization and band parallelization needs to be tested for the particular system at hand*. This can be done quite easily, using only a few test calculations (e.g. NCORE = 1, 7, 14 and 28 at KPAR = 1 on 1 node for a representative

system and cutoff energy) and for the optimal configuration taking *KPAR as high as possible*. In addition, the guidelines for memory should be taken into account as well, since *large systems or systems with many k-points (like METAL) may suffer from too high memory requirements*. Finally, the NSIM tag does not matter too much, but NSIM = 1 is strongly discouraged, as it drastically increases the computation time (default is NSIM = 4).

In comparison to Ghent clusters, the (empty) BrENIAC machine performs exceptionally well. For the SEMI system, timings are about two times as good as the best wall times ever achieved on Muk (1378 s in 2013). The same is true in comparison to golett, one of the most recent machines on the UGent HPC (1300 s in 2016). These numbers were scaled to be comparable to the 28 cores per node of BrENIAC. Note, however, that the wall time on golett was measured on the machine in full loading (whereas the BrENIAC machine was almost empty), which has a large impact on the speed of the calculations.

3. Intra- and multinode scaling

Figure 1: Intra- and multinode scaling of the wall time for the METAL system.



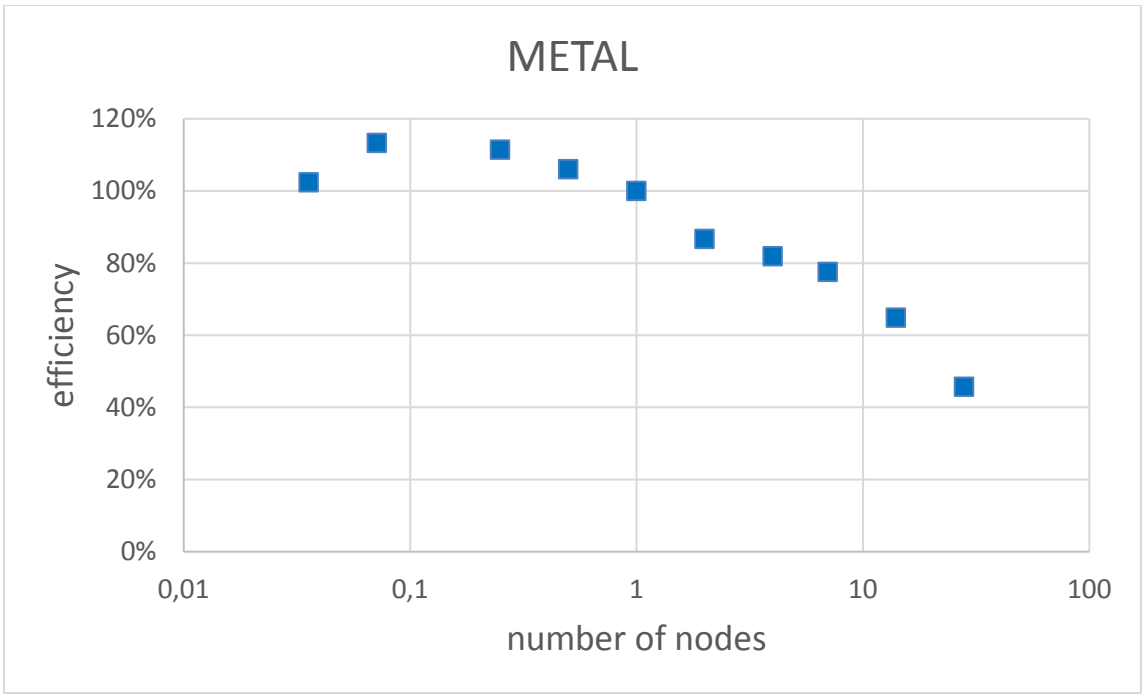
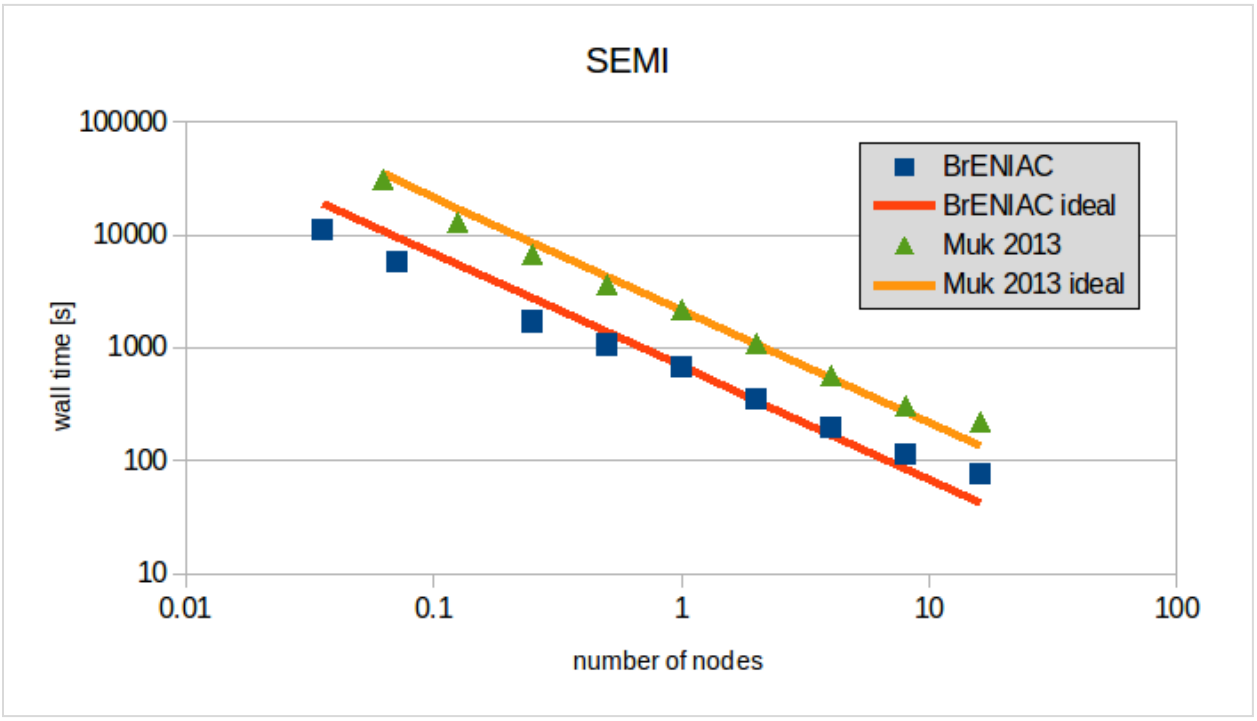


Figure 2: Intra- and multinode scaling of the wall time for the SEMI system (BrENIAC 2016 and Muk 2013). The red and orange lines denote the ideal scaling behaviour.



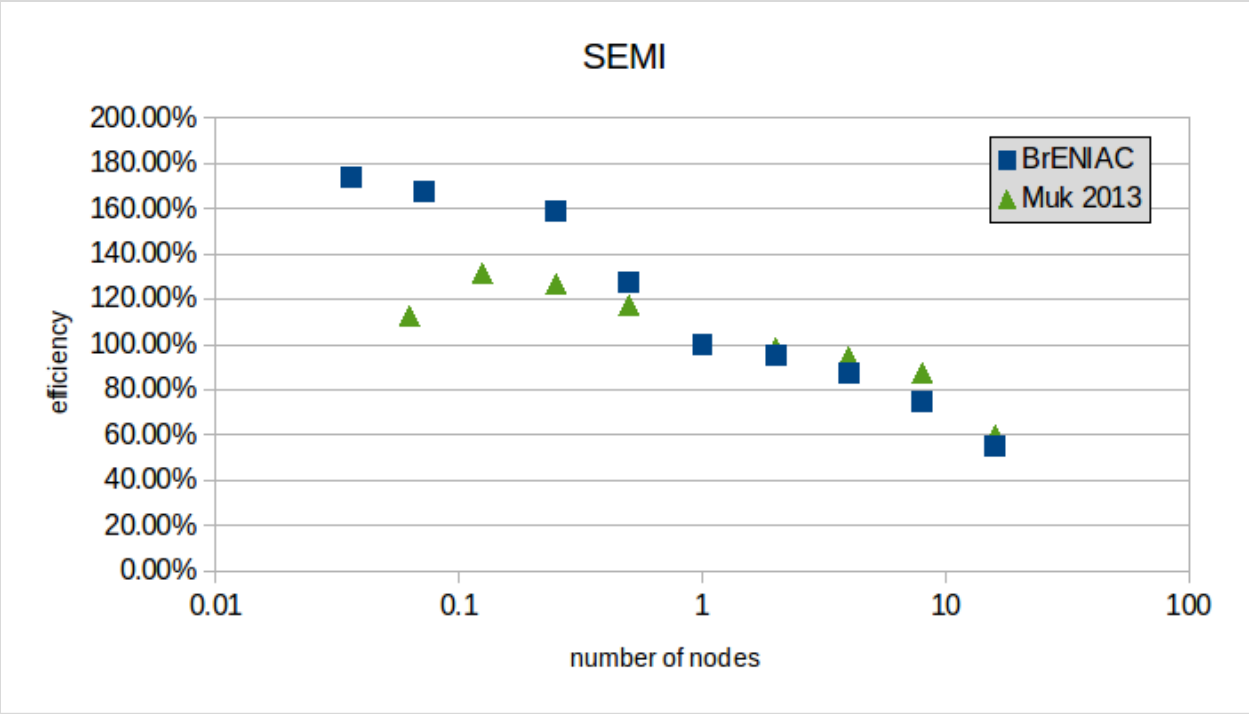
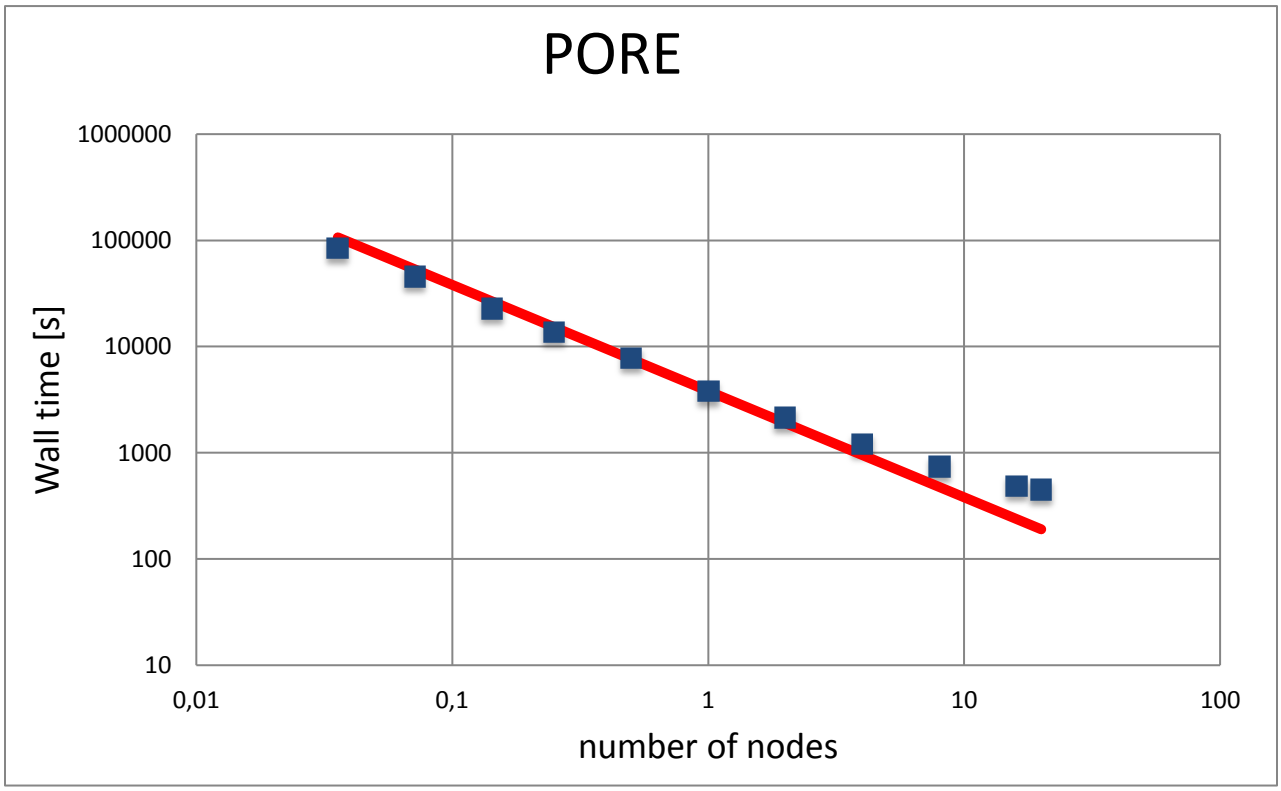
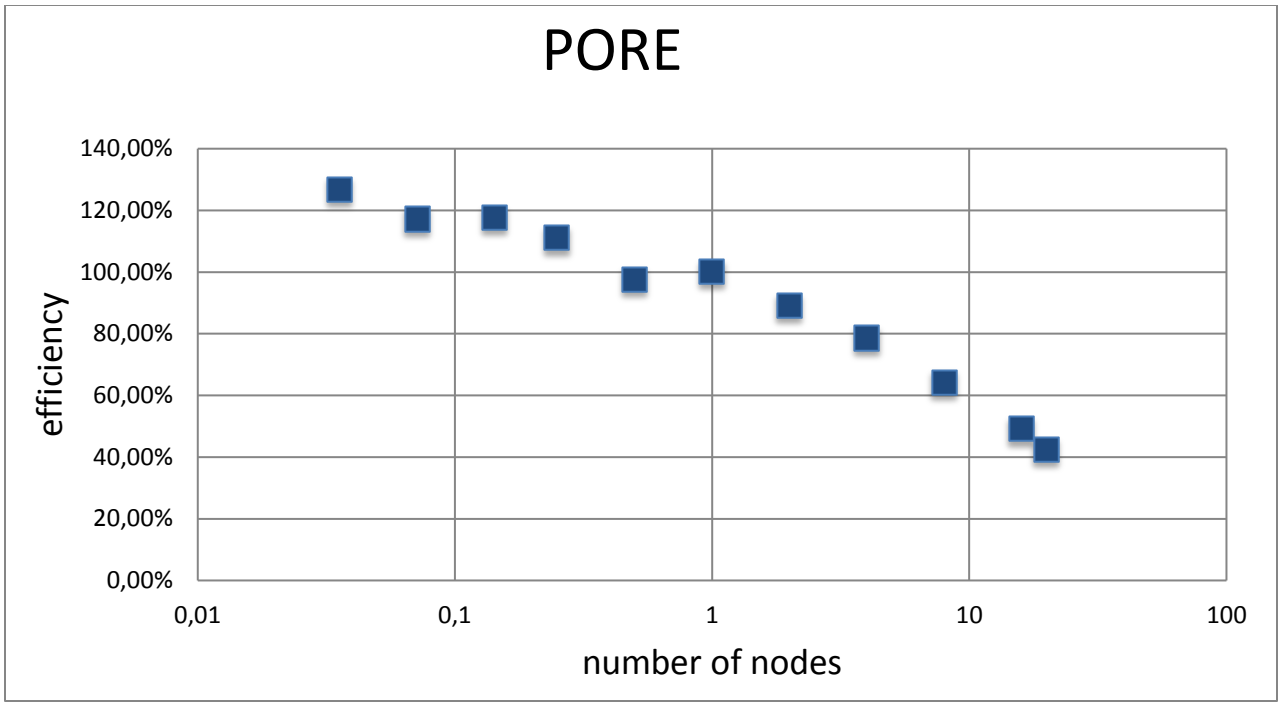


Figure 3: Intra- and multinode scaling of the wall time for the PORE system. The red line denotes the ideal scaling behaviour.





By performing the calculations on a few cores up to multiple nodes, we note that the computational efficiency proceeds in 2 steps. On the one hand, it remains most efficient to perform calculations on 1 or a few cores, and up to the use of an entire node, the efficiency steadily declines. This intranode scaling differs significantly for different systems, however, with poor scaling for SEMI and almost ideal scaling for METAL. *The multinode scaling, however, is quite efficient*, and parallelization over 8 nodes leads to wall times that are still 60-80 % of the efficiency of a single node. Beyond 16 nodes, efficiency drops below 50 %, and calculations are only advisable if they cannot be calculated within 72h on fewer nodes. This behaviour is similar for all tested systems, despite their large diversity, and in line with tests on Muk in 2013 (see Figure 2). We may therefore conclude that it is *not meaningful to perform such scaling tests time and again; only the optimal parallelization settings on 1 node need to be examined when considering a new system*.

As a final note concerning the parallelization settings in multinode calculations, we remark that it is best not to parallelize 1 k-point or 1 band over multiple nodes. Using KPAR equal to the number of nodes (or higher) decreases the computational load significantly, because k-point parallelization requires little communication. For the SEMI system on 2 nodes, for example, a k-point-parallelized calculation (KPAR = 4, NCORE = 14) takes 359 s, while a band-parallelized calculation (KPAR 1, NCORE = 14) takes 408 s. For the METAL system, the difference is huge: 14 002 s for KPAR = 4 and NCORE = 14, compared to 43 229 s for KPAR = 1 and NCORE = 14.