Efficiency Analysis of Intel and AMD x86_64 Architectures for Ab Initio Calculations: A Case Study of VASP

Vladimir Stegailov[1,2,3(✉)] and Vyacheslav Vecher[1,2]

[1] Joint Institute for High Temperatures of RAS, Moscow, Russia
[2] Moscow Institute of Physics and Technology (State University), Dolgoprudny, Russia
[3] National Research University Higher School of Economics, Moscow, Russia

v.stegailov@hse.ru, vecher@phystech.edu

Abstract. Nowadays, the wide spectrum of Intel Xeon processors is available. The new Zen CPU architecture developed by AMD has extended the number of options for x86_64 HPC hardware. This large number of options makes the optimal CPU choice for HPC systems not a straightforward procedure. Such a co-design procedure should follow the requests from the end-users community. Modern computational materials science studies are among the major consumers of HPC resources worldwide. The VASP code is perhaps the most popular tool for these research. In this work, we discuss the benchmark metric and results based on a VASP test model that give us the possibility to compare different CPUs and to select best options with respect to time-to-solution and energy-to-solution criteria.

Keywords: Multicore · VASP · Memory wall · Broadwell · Zen

1 Introduction

Computational materials science provides an essential part of the deployment time for high performance computing (HPC) resources worldwide. The VASP code [1–4] is among the most popular programs for electronic structure calculations that gives the possibility to calculate materials properties using the non-empirical (so called *ab initio*) methods. According to the recent estimates, VASP alone consumes up to 15-20 percent of the world's supercomputing power [5, 6]. Such unprecedented popularity justifies the special attention to the optimization of VASP for both existing and novel computer architectures (e.g. see [7]). At the same time, one can ask a question what type of processing units would be the most efficient for VASP calculations.

A large part of HPC resources installed during the last decade is based on Intel CPUs. Novel generations of Intel CPUs present the wide spectrum of multicore processors. The number Xeon CPU types for dual-socket systems is 26 for the Sandy Bridge family, 27 for Ivy Bridge, 22 for Haswell and 23 for Broadwell families. In each family, the processors share the same core type but differ by their frequency, core count, cache sizes, network-on-chip structure etc.

In March 2017, AMD released the first processors based on the novel x86_64 architecture called Zen. It is assumed that the efficiency of this architecture for HPC applications would be comparable to the latest Intel architectures (Broadwell and Skylake).

The diversity of CPU types complicates significantly the choice of the best variant for a particular HPC system. The first criterion is certainly the time-to-solution of a given computational task or a set of different tasks that represents an envisaged workload of a system under development.

Another criterion is the energy efficiency of an HPC system. Energy efficiency becomes one of the most important concerns for the HPC development today and will remain in foreseeable future [8].

The need for clear guiding principles stimulates the development of models for HPC systems performance prediction. However, the capabilities of the idealized models are limited by the complexity of real-life applications. That is why the empirical benchmarks of the real-life examples serve as a complimentary tool for the co-design and optimization of software-hardware combinations.

In this work, we present the efficiency analysis of a limited but representative list of modern Intel and AMD x86_64 CPUs using a typical VASP workload example.

2 Related Work

HPC systems are notorious for operating at a small fraction of their peak performance and the deployment of multi-core and multi-socket compute nodes further complicates performance optimization. Many attempts have been made to develop a more or less universal framework for algorithms optimization that takes into account essential properties of the hardware (see e.g. [9, 10]). The recent work of Stanisic et al. [11] emphasizes many pitfalls encountered when trying to characterize both the network and the memory performance of modern machines.

The choice of the best option among several alternative GPU-systems for running the GROMACS package is the subject of the paper of Kutzner et al. [12]. In that paper, several real life examples are considered as benchmarks of the hardware efficiency. Our paper follows a similar path but for the VASP package.

The application of *ab initio* codes requires big supercomputers and the parallel scalability of the codes becomes, therefore, an important issue. The scalability of the SIESTA code was considered in [13] for several Tier0 systems from the PRACE infrastructure (although technically quite different, SIESTA shares the same field of applications in materials science as VASP). In the previous work [14], different HPC systems were compared with respect to their performance for another electronic structure code CP2K.

The increase of power consumption and heat generation of computing platforms is a very significant problem. Measurement and presentation of the results of performance tests of parallel computer systems become more and more often

Table 1. The main features of the systems considered

CPU type	N_{cores}	$N_{mem.ch.}$	L3 (Mb)	CPU_{freq} (GHz)	$DRAM_{freq}$ (MHz)
Single socket, Intel X99 chipset					
Xeon E5-2620v4	8	4	20	2.1	2133
Core i7-6900K	8	4	20	2.1 – 3.2	2133 – 3200
Xeon E5-2660v4	14	4	35	2.0	2400
Single socket, AMD B350 chipset					
Ryzen 1800X	8	2	16	3.6	2133 – 2400
Dual socket, Intel C602 chipset (the MVS10P cluster)					
Xeon E5-2690	8	4	20	2.9	1600
Dual socket, Intel C612 chipset (the MVS1P5 cluster)					
Xeon E5-2697v3	14	4	35	2.6	2133
Dual socket, Intel C612 chipset (the IRUS17 cluster)					
Xeon E5-2698v4	20	4	50	2.2	2400
Quad socket, IBM Power 775 (the Boreasz cluster [22])					
Power 7	8	4	32	3.83	1600

evidence-based [15], including the measurement of energy consumption, which is crucial for the development of exascale supercomputers [16].

The work of Calore et al. [17] discloses some aspects of relations between power consumption and performance using small Nvidia Jetson TK1 minicomputer running the Lattice Boltzmann method algorithms. An energy-aware task management mechanism for the MPDATA algorithms on multicore CPUs was proposed by Rojek et al. [18].

Our previous results on energy consumption for minicomputers running classical MD benchmarks was published previously for Odroid C1 [19] and Nvidia Jetson TK1 and TX1 [20, 21].

3 Hardware and Software

In this work, we consider several Intel CPUs and the novel AMD Ryzen processor and compare the results with the data [22] for the IBM Power 7. The features of the systems considered are summarized in Table 1. We make use of the fact that the Intel X99 chipset supports both consumer Core series and server Xeon series Intel processors that share the same LGA 2011-3 socket. The Core i7-6900K is similar to the Xeon E5-2620v4 but allows us to vary CPU and DRAM frequencies.

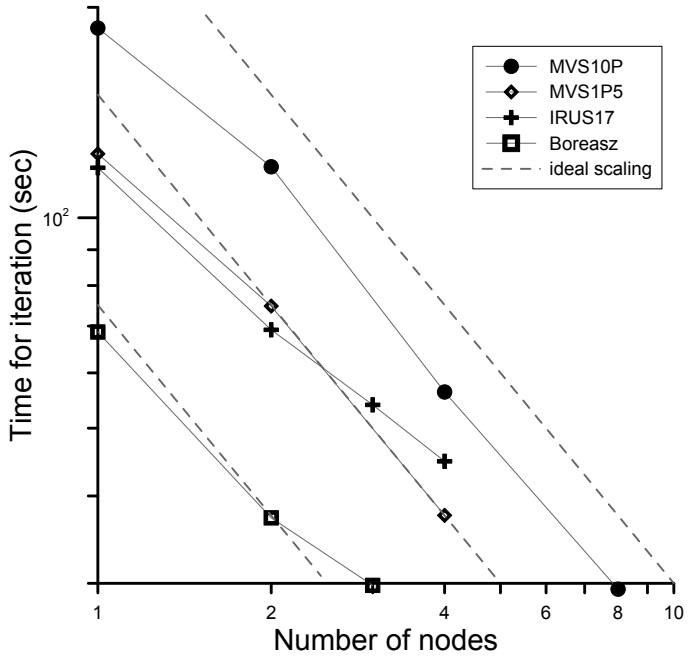

Fig. 1. Parallel scalability of the GaAs test. In all cases, 8 cores per socket are used that corresponds to 16 MPI ranks per a dual-socket node of the Xeon-based MVS10P, MVS1P5 and IRUS17 clusters and 64 MPI ranks on a quad socket node of the Power-based Boreasz cluster.

The single socket Intel Broadwell systems benchmarks are performed under Ubuntu ver. 16.04 with Linux kernel ver. 4.4.0. The single socket AMD Ryzen system is benchmarked under Ubuntu ver. 17.04 with Linux kernel ver. 4.10.0.

3.1 Test Model in VASP

VASP 5.4.1 is compiled for Intel systems using Intel Fortran, Intel MPI and linked with Intel MKL for BLAS, LAPACK and FFT calls. For the AMD system, gfortran ver.6.3 is used together with OpenMPI, OpenBLAS and FFTW libraries.

Our test model in VASP is the same as used previously for the benchmarks of the IBM 775 system [22]. The model represents a GaAs crystal consisting of 80 atoms in the supercell. The Perdew–Burke–Ernzerhof model for xc-functional is used. The calculation protocol corresponds to the geometry optimization. We use the time for the first iteration of electron density optimization τ_{iter} as a target parameter of the performance metric. This parameter can serve as an adequate measure of time consumption for molecular dynamics calculations as well.

The τ_{iter} values considered in this work are about 10-100 sec and correspond mainly to a single node of an HPC cluster. At the first glance, these are not very

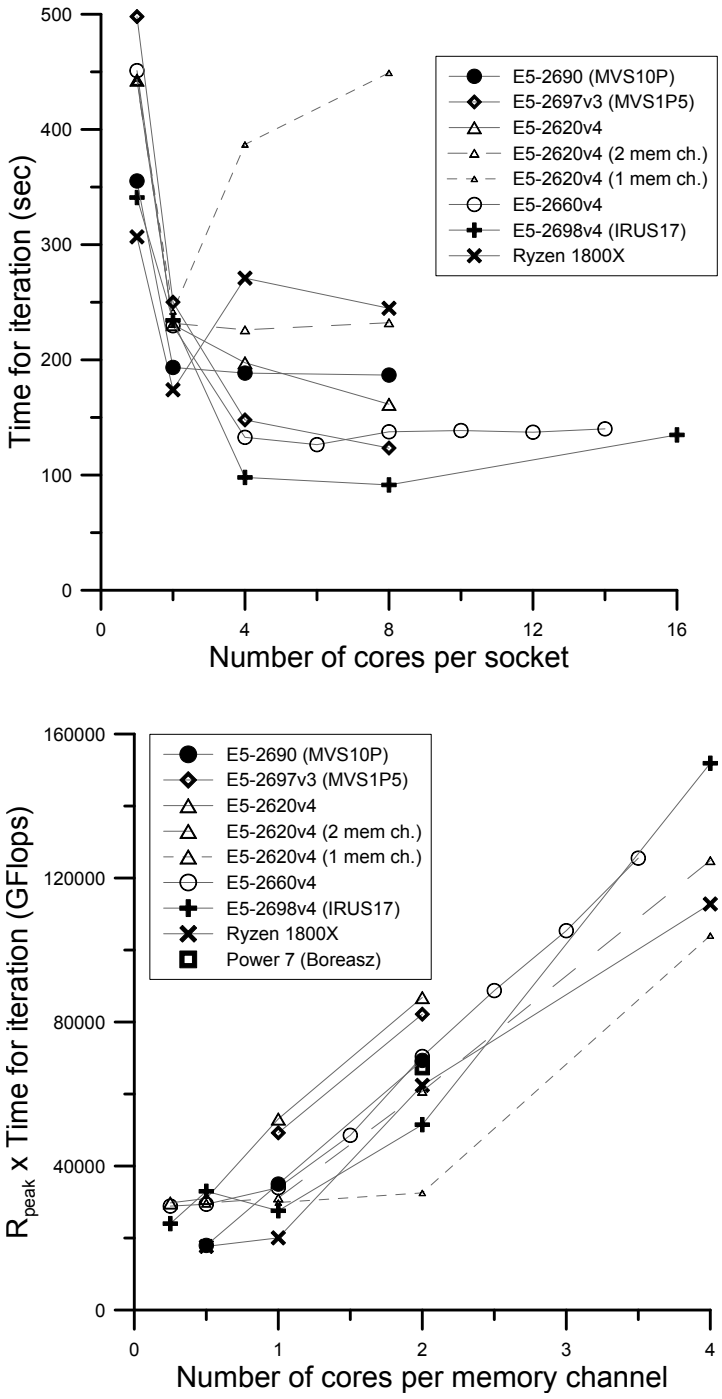

Fig. 2. Top: the dependence of the time for the first iteration of the GaAs test on the number of cores per socket. The presented time values for the single socket systems (see Table 1) is divided by 2 for comparison with the dual socket systems data. Bottom: the same data (and one point for IBM Power 7) in the reduced parameters $R_{peak}\tau_{iter}$ and $N_{cores}/N_{mem.ch.}$ (here R_{peak} is the total peak performance of the single or dual socket node).

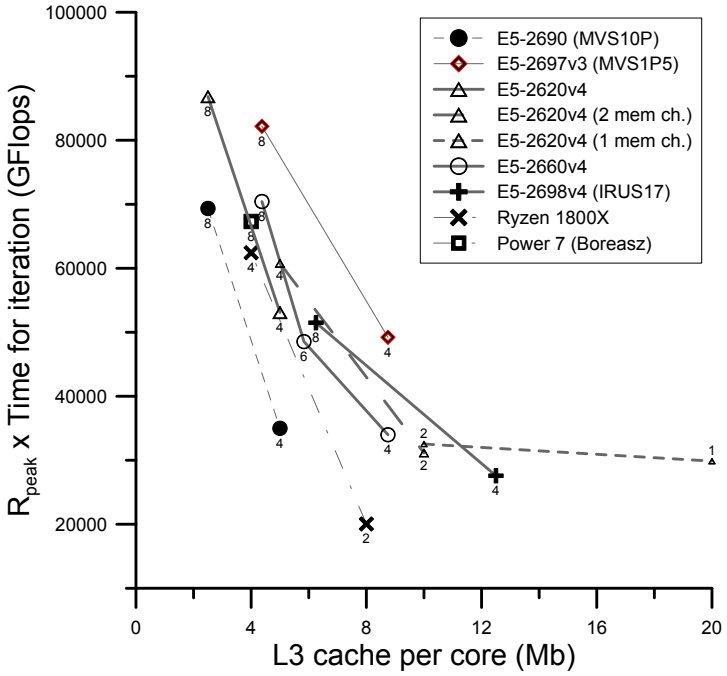

Fig. 3. The dependence of the $R_{peak}\tau_{iter}$ parameter on the L3 cache per core.

long times to be accelerated. However, *ab initio* molecular dynamics requires usually $10^4 - 10^5$ time steps and each step consists of 3-5 such iterations. That is why the decrease of τ_{iter} by several orders of magnitude is an actual problem for modern HPC systems targeted at materials science computing.

The choice of a particular test model has a certain influence on the benchmarking results. However, our preliminary tests of other VASP models show that the main conclusions of this study do not depend significantly on a particular model. In the future, a set of regression tests would be beneficial for similar analysis.

3.2 Power Consumption Measurement

For the single socket systems considered, the power consumption measurements are performed. We use APC Back-UPS Pro BR1500G-RS and the corresponding apcupsd linux driver for digital sampling of power consumed during VASP runs. In this way, we measure the total power consumption of the CPU, the memory, the motherboard and PSU. For the evaluation of the total energy consumed for one benchmark run, we multiply the average power value during the run by the time of the first iteration τ_{iter}.

4 Results and Discussion

4.1 Where is the Balance between Cores, Memory Channels and L3 Cache?

VASP 5.4.1 uses MPI for parallelization. Figure 1 illustrates the acceleration of the GaAs test considered for 1-8 nodes of the MVS10P (FDR Infiniband), MSV1P5 (FDR Infiniband) and IRUS17 (Omni-path) clusters. For the modest number of nodes considered, the acceleration is very efficient. Here we do not want to analyze the limits of parallel scalability but to show that the absolute performance of the parallel code is proportional to the performance of single cluster nodes (e.g. one can mention the similarity of the strong scaling data on Figure 1 for MVS10P and MVS1P5 that both use FDR Infiniband).

VASP is known to be both a memory-bound and a compute-bound code [7]. Modern Intel CPUs provide 4 memory channels per socket. That is why *a priori* it is not obvious how VASP performance depends on the number of cores per socket. Figure 2 shows the results of the GaAs test runs.

We see a pronounced dependence on the number of cores per socket. For majority of systems, the time per iteration saturates at 4 cores per socket and shows no significant decrease for higher core counts.

In order to understand the dependence on the number of memory channels, we perform tests with E5-2620v4 CPU with only 2 or 1 memory channels activated (when only 2 DIMMs or 1 DIMM are installed into the motherboard). The results confirm the crucial importance of the number of active memory channels for the VASP performance (this fact is a manifestation of the "memory wall" concept).

Performance comparison of different CPUs resembles usually a comparison of "apples and oranges". For comparison of CPUs with different frequencies and different peak numbers of Flops/cycle, it is better to use the reduced parameter of $R_{peak}\tau_{iter}$ [14, 20]. Another reduced parameter that characterizes the memory subsystem is $N_{cores}/N_{mem.ch.}$ (for simplicity we neglect here the variation of the memory bandwidth per channel). The bottom plot of Figure 2 presents the same data as shown on the upper plot in the reduced coordinates. In this way, we have eliminated the differences in floating point performance of different CPU core and the difference in the number of memory channels.

In these reduced coordinates, the scatter of data points is much smaller, and there is an evident common trend. The data point for the IBM Power 7 CPU is located at the same trend that suggests the low sensitivity of the results to the hardware and software differences between x86_64 and IBM Power systems.

The test model considered fixes the total number of arithmetic operations (Flops) required for its solution. The increase of $R_{peak}\tau_{iter}$ (that is proportional to the number of CPU cycles) shows the increase of the overhead due to the limited memory bandwidth. More CPU cycles are required for the CPU cores involved in computations to get data from DRAM.

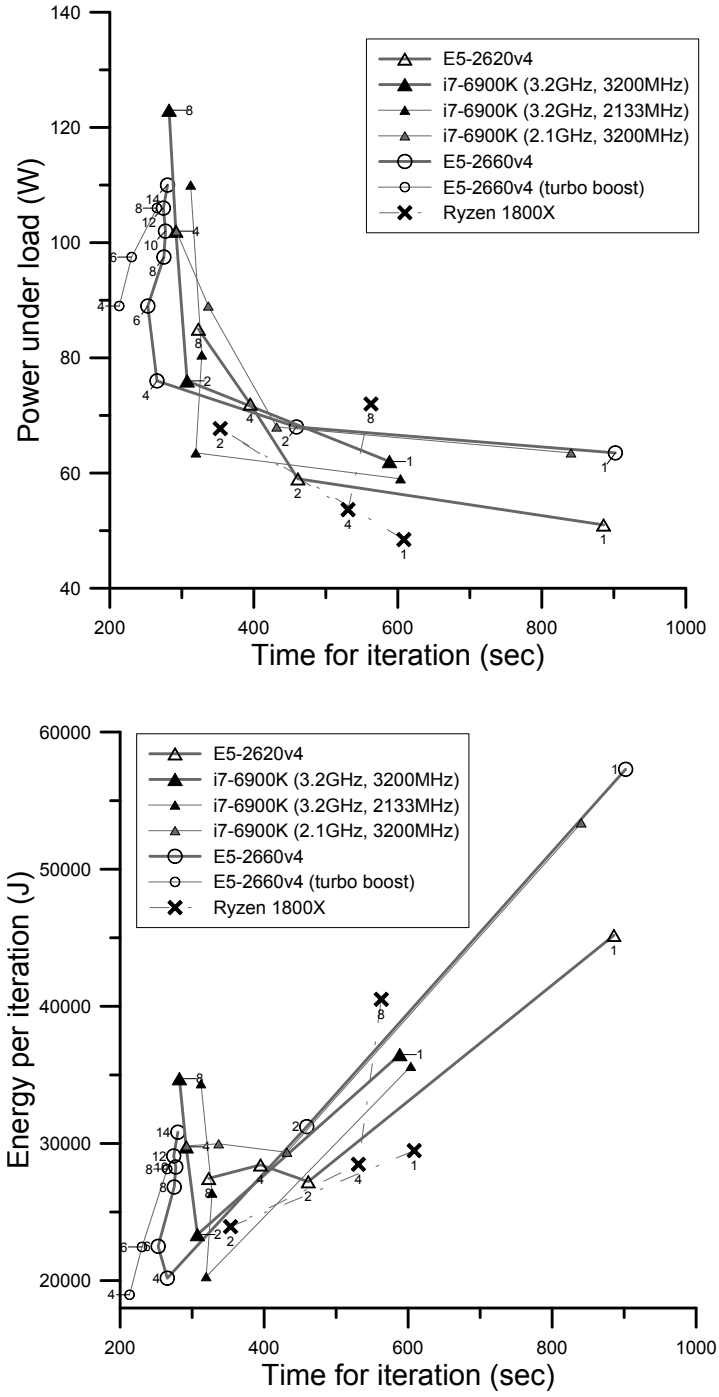

Fig. 4. The average power draw and energy consumption of the single socket systems under the VASP test model load. The number of active cores is shown near each data point.

The remaining scatter of data points at the bottom plot of Figure 2 can be partially attributed to different L3 cache sizes of the CPUs considered. We select the data points from Figure 2 that correspond to $N_{cores}/N_{mem.ch.} = 1\text{-}2$, and plot the $R_{peak}\tau_{iter}$ values as a function of the L3 cache size per core (Figure 3). There is a visible trend: the larger is the L3 cache size per core, the smaller is the $R_{peak}\tau_{iter}$ value. The precise analysis of the data structures used by VASP and their caching is beyond the scope of this paper. However, it is evident that the main VASP computational kernel (composed of the MKL routines) accesses continuous blocks of data in DRAM, and L3 cache mechanism provides efficient acceleration.

Remarkably, the point for the IBM Power 7 benchmark corresponds to this trend very well. The rightmost point (that corresponds to the benchmark with 1 core of E5-2620v4 with 1 active memory channel) is not located at the main trend, because in this combination, presumably, all available L3 cache can not be utilized effectively.

4.2 Optimization of the Energy-to-Solution

For the single socket systems considered (see Table 1) the power consumption measurements are performed together with the VASP model test runs. The results are summarized in Figure 4 that shows the average power and the total consumed energy as functions of τ_{iter}.

The experiments with Core i7-6900K shows that

- increasing DRAM frequency from 2133 to 3200 MHz results in 10 percent higher power draw but gives about 10 percent smaller times for iteration for 4 and 8 cores;
- decreasing CPU frequency from 3.2 to 2.1 GHz results in 20 percent smaller power draw but gives only about 4 percent larger times for iteration for 8 cores.

Comparing E5-2620v4 (with 8 cores in total) and E5-2660v4 (with 14 cores in total), we conclude that non-active cores do not contribute significantly to the power draw during VASP test runs.

AMD Ryzen shows a competitive level of power consumption. However, the increase of average power consumption after the transition from 1 to 2 cores for AMD Ryzen is more pronounced than for Intel Broadwell CPUs considered. The probable reason is the activation of both quad-core CPU-Complexes (CCX) of the Ryzen 1800X CPU.

In most cases, there is a minimum in energy consumption for a given CPU. This minimum is mainly connected with the reduction of τ_{iter}. Beyond this minimum when more cores come into play, further acceleration is connected with essentially higher power draw, or there is no acceleration at all.

The most power-efficient and energy-efficient case among the variants considered is the use of 4 cores of E5-2660v4, especially in the turbo boost mode.

5 Conclusions

In this work, we have considered several Intel CPUs (from Sandy Bridge, Haswell and Broadwell families), the novel AMD Ryzen CPU and used the data on IBM Power 7 for comparison. In all the cases, we have used the test VASP model of GaAs crystal as a benchmark tool. Complimentary power consumption measurements have been carried out as well.

Additionally to the variation of the CPU types, we have considered the variations in the number of active memory channels, the CPU and DRAM frequencies.

For comparison of different cases, we have used three reduced parameters: 1) the time for iteration normalized by the floating point peak performance $R_{peak}\tau_{iter}$, 2) the number of CPU cores per memory channel $N_{cores}/N_{mem.ch.}$ and 3) the L3 cache size per core.

The benchmark results correlate with these reduced parameters quite well. This fact allows us to make several conclusions on optimal VASP performance.

For VASP, the optimal number of cores per memory channel is 1-2. Using more that 2 cores per channel provides no acceleration.

For $N_{cores}/N_{mem.ch.}$ = 1-2, VASP performance increases significantly with the increase of the L3 cache per core. Each additional Mb of L3 cache per core reduces the time-to-solution by 30-50 percent.

The increase of CPU frequency gives diminishing returns but increases significantly the power draw. The increase of DRAM frequency results in the proportional rise of the power draw and in the proportional acceleration.

Comparing different CPUs at the same level of performance, we conclude that CPUs with larger L3 cache size need less power and consume less energy.

Acknowledgments. The authors are grateful to Dr. Maciej Cytowski and Dr. Jacek Peichota (ICM, University of Warsaw) for the data on the VASP benchmark [22].

The authors acknowledge Joint Supercomputer Centre of Russian Academy of Sciences (http://www.jscc.ru) and Shared Resource Center "Far Eastern Computing Resource" IACP FEB RAS (http://cc.dvo.ru) for the access to the supercomputers MVS10P, MVS1P5 and IRUS17.

The work was supported by the grant No. 14-50-00124 of the Russian Science Foundation. A part of the equipment used in this work was purchased with financial support of HSE and using the President of Russian Federation grant for young researchers MD-9451.2016.8.

References

1. Kresse, G., Hafner, J.: Ab initio molecular dynamics for liquid metals. Phys. Rev. B **47**, 558–561 (1993). DOI 10.1103/PhysRevB.47.558. URL `http://link.aps.org/doi/10.1103/PhysRevB.47.558`

2. Kresse, G., Hafner, J.: Ab initio molecular-dynamics simulation of the liquid-metal-amorphous-semiconductor transition in germanium. Phys. Rev. B **49**, 14,251–14,269 (1994). DOI 10.1103/PhysRevB.49.14251. URL http://link.aps.org/doi/10.1103/PhysRevB.49.14251
3. Kresse, G., Furthmuller, J.: Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set. Computational Materials Science **6**(1), 15 – 50 (1996). DOI http://dx.doi.org/10.1016/0927-0256(96)00008-0. URL http://www.sciencedirect.com/science/article/pii/0927025696000080
4. Kresse, G., Furthmüller, J.: Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set. Phys. Rev. B **54**, 11,169–11,186 (1996). DOI 10.1103/PhysRevB.54.11169. URL http://link.aps.org/doi/10.1103/PhysRevB.54.11169
5. Iain Bethune. "Ab initio molecular dynamics," Introduction to Molecular Dynamics on ARCHER, 2015. URL: https://www.archer.ac.uk/training/course-material/2015/06/MolDy_Strath/AbInitioMD.pdf
6. Max Hutchinson. "VASP on GPUs. When and how," GPU technology theater, SC15, 2015. URL: http://images.nvidia.com/events/sc15/pdfs/SC5107-vasp-gpus.pdf
7. Zhao, Z., Marsman, M.: Estimating the performance impact of the MCDRAM on KNL using dual-socket Ivy Bridge nodes on Cray XC30. In: Proceedings of the Cray User Group — 2016 (2016). URL https://cug.org/proceedings/cug2016_proceedings/includes/files/pap111.pdf
8. Kogge, P., Shalf, J.: Exascale computing trends: Adjusting to the "new normal" for computer architecture. Computing in Science & Engineering **15**(6), 16–26 (2013). DOI 10.1109/MCSE.2013.95
9. Burtscher, M., Kim, B.D., Diamond, J., McCalpin, J., Koesterke, L., Browne, J.: Perfexpert: An easy-to-use performance diagnosis tool for hpc applications. In: Proceedings of the 2010 ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis, SC '10, pp. 1–11. IEEE Computer Society, Washington, DC, USA (2010). DOI 10.1109/SC.2010.41. URL https://doi.org/10.1109/SC.2010.41
10. Rane, A., Browne, J.: Enhancing performance optimization of multicore/multichip nodes with data structure metrics. ACM Trans. Parallel Comput. **1**(1), 3:1–3:20 (2014). DOI 10.1145/2588788. URL http://doi.acm.org/10.1145/2588788
11. Stanisic, L., Mello Schnorr, L.C., Degomme, A., Heinrich, F.C., Legrand, A., Videau, B.: Characterizing the Performance of Modern Architectures Through Opaque Benchmarks: Pitfalls Learned the Hard Way. In: IPDPS 2017 - 31st IEEE International Parallel & Distributed Processing Symposium (RepPar workshop). Orlando, United States (2017). URL https://hal.inria.fr/hal-01470399
12. Kutzner, C., Páll, S., Fechner, M., Esztermann, A., de Groot, B.L., Grubmüller, H.: Best bang for your buck: GPU nodes for GROMACS biomolecular simulations. Journal of Computational Chemistry **36**(26), 1990–2008 (2015). DOI 10.1002/jcc.24030. URL http://dx.doi.org/10.1002/jcc.24030
13. Corsetti, F.: Performance analysis of electronic structure codes on HPC systems: A case study of SIESTA. PLOS ONE **9**(4), 1–8 (2014). DOI 10.1371/journal.pone.0095390. URL http://dx.doi.org/10.1371%2Fjournal.pone.0095390
14. Stegailov, V.V., Orekhov, N.D., Smirnov, G.S.: HPC hardware efficiency for quantum and classical molecular dynamics. In: V. Malyshkin (ed.) Parallel Computing Technologies: 13th International Conference, PaCT 2015, Petrozavodsk,

Russia, August 31-September 4, 2015, Proceedings, pp. 469–473. Springer International Publishing, Cham (2015). DOI 10.1007/978-3-319-21909-7_45. URL http://dx.doi.org/10.1007/978-3-319-21909-7_45

15. Hoefler, T., Belli, R.: Scientific benchmarking of parallel computing systems: Twelve ways to tell the masses when reporting performance results. In: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, SC '15, pp. 73:1–73:12. ACM, New York, NY, USA (2015). DOI 10.1145/2807591.2807644
16. Scogland, T., Azose, J., Rohr, D., Rivoire, S., Bates, N., Hackenberg, D.: Node variability in large-scale power measurements: Perspectives from the Green500, Top500 and EEHPCWG. In: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, SC '15, pp. 74:1–74:11. ACM, New York, NY, USA (2015). DOI 10.1145/2807591.2807653. URL http://doi.acm.org/10.1145/2807591.2807653
17. Calore, E., Schifano, S.F., Tripiccione, R.: Energy-performance tradeoffs for HPC applications on low power processors. In: S. Hunold, A. Costan, D. Giménez, A. Iosup, L. Ricci, M.E. Gómez Requena, V. Scarano, A.L. Varbanescu, S.L. Scott, S. Lankes, J. Weidendorfer, M. Alexander (eds.) Euro-Par 2015: Parallel Processing Workshops: Euro-Par 2015 International Workshops, Vienna, Austria, August 24-25, 2015, Revised Selected Papers, pp. 737–748. Springer International Publishing, Cham (2015). DOI 10.1007/978-3-319-27308-2_59. URL http://dx.doi.org/10.1007/978-3-319-27308-2_59
18. Rojek, K., Ilic, A., Wyrzykowski, R., Sousa, L.: Energy-aware mechanism for stencil-based MPDATA algorithm with constraints. Concurrency and Computation: Practice and Experience pp. e4016–n/a (2016). DOI 10.1002/cpe.4016. URL http://dx.doi.org/10.1002/cpe.4016. Cpe.4016
19. Nikolskiy, V., Stegailov, V.: Floating-point performance of ARM cores and their efficiency in classical molecular dynamics. Journal of Physics: Conference Series **681**(1), 012,049 (2016). URL http://stacks.iop.org/1742-6596/681/i=1/a=012049
20. Nikolskiy, V.P., Stegailov, V.V., Vecher, V.S.: Efficiency of the Tegra K1 and X1 systems-on-chip for classical molecular dynamics. In: 2016 International Conference on High Performance Computing Simulation (HPCS), pp. 682–689 (2016). DOI 10.1109/HPCSim.2016.7568401
21. Vecher, V., Nikolskii, V., Stegailov, V.: GPU-Accelerated Molecular Dynamics: Energy Consumption and Performance, pp. 78–90. Springer International Publishing, Cham (2016). DOI 10.1007/978-3-319-55669-7_7. URL http://dx.doi.org/10.1007/978-3-319-55669-7_7
22. Maciej Cytowski. "Best Practice Guide — IBM Power 775," PRACE, November 2013. URL: http://www.prace-ri.eu/IMG/pdf/Best-Practice-Guide-IBM-Power-775.pdf