

Energy Consumption Optimization of the Total-FETI Solver and BLAS Routines by Changing the CPU Frequency

David Horak, Lubomir Riha, Radim Sojka, Jakub Kruzik, Martin Beseda
IT4Innovations National Supercomputing Center
VSB-Technical University of Ostrava
17. listopadu 15, Ostrava 70833 Czech Republic
Email: david.horak@vsb.cz

EXTENDED ABSTRACT

The energy consumption of supercomputers is one of the critical problems for the upcoming Exascale supercomputing era. The awareness of power and energy consumption is required on both software and hardware side. This poster deals with the energy consumption evaluation of the Total-Finite Element Tearing and Interconnect (TFETI) based solvers [2] of linear systems implemented in PERMON toolbox [1], which is an established method for solving real-world engineering problems, and with the energy consumption evaluation of the BLAS routines. The experiments performed in the poster deal with CPU frequency.

This work is performed in the scope of the READEX project (Runtime Exploitation of Application Dynamism for Energy-efficient eXascale computing) [6].

The measurements were performed on the Intel Xeon E5-2680 (Intel Haswell micro-architecture) based Taurus system installed at TU Dresden. The system contains over 1400 nodes that have an FPGA-based power instrumentation called HDEEM (High Definition Energy Efficiency Monitoring), that allows for fine-grained and more accurate power and energy measurements. The measurements can be accessed through the HDEEM library, allowing developers to take energy measurements before and after the region of interest.

We have evaluated the effect of the CPU frequency on the energy consumption of the TFETI solver for a linear elasticity 3D cube synthetic benchmark. On the dualized problem $MPF\lambda=MPd$, we have evaluated the effect of frequency tuning on the energy consumption of the essential processing kernels of the TFETI method. There are two main phases in TFETI – preprocessing and solve. In preprocessing it is necessary to regularize the stiffness matrix K and factorize it and to assemble the G and GG^T matrices and the second one to factorize. Both operations belong to the most time and also energy consuming operations. The solve employs the Preconditioned Conjugate Gradient (PCG) algorithm, which consists of sparse matrix-vector multiplications (by F , P , M_L , M_D matrices) and vector dot products and AXPY functions. In each iteration, we need to apply the direct solver twice, i.e., for forward and backward solves for the pseudoinverse K^+ action and for the coarse problem solution, the $(GG^T)^{-1}$ action. The multiplication by the dense Schur complement

matrix adds an additional operator with different computational characteristics, potentially increasing the exploitable dynamism.

The poster provides results for two types of frequency tuning: (1) static tuning and (2) dynamic tuning. For static tuning experiments, the frequency is set before execution and kept constant during the runtime. For dynamic tuning, the frequency is changed during the program execution to adapt the system to the actual needs of the application. The poster shows that static tuning brings up 11.84% energy savings when compared to default CPU settings (the highest clock rate). The dynamic tuning improves this further by up to 2.68%. In total, the approach presented in this paper shows the potential to save up to 14.52% of energy for TFETI based solvers, see Table 1.

Another energy consumption evaluations were done with selected Sparse and Dense BLAS Level 1, 2 and 3 routines. For benchmarking we have used a set of matrices from University Florida collection [4]. We have employed AXPY, Sparse Matrix-Vector, Sparse Matrix-Matrix, Dense Matrix-Vector, Dense Matrix-Matrix and Sparse Matrix-Dense Matrix multiplication routines from Intel Math Kernel Library (MKL) [3]. The measured characteristics illustrate the different energy consumption of BLAS routines, as some operations are memory-bounded and others are compute-bounded. Based on our recommendations one can explore dynamic frequency switching to achieve significant energy savings up to 23%, for more details see Table 2.

ACKNOWLEDGMENT

This work was supported by the READEX project – the European Union’s Horizon 2020 research and innovation programme under grant agreement No 671657; by The Ministry of Education, Youth and Sports from the National Programme of Sustainability (NPU II) project “IT4Innovations excellence in science” (LQ1602), and from the “Large Infrastructures for Research, Experimental Development and Innovations” project “IT4Innovations National Supercomputing Center” (LM2015070); by the internal student grant competition project “PERMON toolbox development II” (SP2016/178); and by the Czech Science Foundation (GACR) project no. 15-18274S.

TABLE 1: EFFICIENCY OF STATIC AND DYNAMIC TUNING FOR THE FETI SOLVE STAGE AND BOTH PRECONDITIONERS

Action	Run-time	Static Tuning Savings	Overall Static Savings	Dynamic Tuning Savings	Optimal Frequency	Overall Dynamic Savings
$M_L v$	10%	3.2%	0.32%	3.1%	2.1 GHz	0.31%
Fv with M_L	80%	12%	9.6%	1%	1.6 GHz	0.8%
$M_D v$	40%	13%	5.2%	4%	1.2 GHz	1.6%
Fv with M_D	50%	12%	6%	1%	1.6 GHz	0.5%
Pv	6%	2%	0.12%	1%	2.3 GHz	0.06%
AXPY	4%	13%	0.52%	6%	1.2 GHz	0.52%
Total savings: with M_L			10.56%			1.69%
with M_D			11.84%			2.68%

TABLE 2: EFFICIENCY OF FREQUENCY TUNING FOR BLAS 1,2,3 ROUTINES

Action	AXPY	SpMV CSR	SpMV COO	SpMM CSR	SpMM COO	DMV FLOAT
Opt.freq. [GHz]	1.2	1.3-1.7	1.3-1.8	1.8-2.4	2.1	1.2
Savings [%]	19	11-20	8-17	5-8	2.1	21-23

DMV DOUBLE	DMM FLOAT	DMM DOUBLE	SpMDM CSR-DOUBLE	SpMDM COO-DOUBLE
1.2	1.7-1.9	1.8-2.5	1.2-1.7	1.2-1.8
17-22	2-14	0-10	12-20	13-20

REFERENCES

- [1] PERMON webpages, <http://industry.it4i.cz/en/products/permon/>
- [2] Dostal, Z., Horak, D. and Kucera, R.: Total FETI – an easier implementable variant of the FETI method for numerical solution of elliptic PDE, Communications in Numer. Methods in Eng., Volume 22, number 12, pages 1155–1162, 2006.
- [3] Intel Math Kernel Library (MKL), <https://software.intel.com/en-us/intelmkl> J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
- [4] Sparse Matrix Algorithms Research at the University of Florida, containing the UF sparse matrix collection, <https://www.cise.ufl.edu/research/sparse/matrices>
- [5] Hackenberg, D., Ilsche, T., Schuchart, J., Schöne, R., Nagel, W., Simon, M., Georgiou, Y.: HDEEM: High Definition Energy Efficiency Monitoring. In: Energy Efficient Supercomputing Workshop (E2SC) (2014), <http://dx.doi.org/10.1109/E2SC.2014.13>
- [6] Run-time Exploitation of Application Dynamism for Energy-efficient eXascale computing (READEX), <http://www.readex.eu/>
- [7] Oleynik, Y., Gerndt, M., Schuchart, J., Kjeldsberg, P. G. and Nagel, W. E.: Run-Time Exploitation of Application Dynamism for Energy-Efficient Exascale Computing (READEX), Computational Science and Engineering (CSE), 2015 IEEE 18th International Conference on, Porto, 2015, pp. 347-350. doi: 10.1109/CSE.2015.55