The energy consumption optimization of the BLAS routines

Radim Sojka[1], Lubomir Riha[1], David Horak[1], Jakub Kruzik[1], Martin Beseda[1] and Martin Cermak[1,a)]

[1]*IT4Innovations, VSB-TU Ostrava, 17. listopadu 15, Ostrava 70833, Czech Republic.*

[a)]Corresponding author: martin.cermak@vsb.cz

Abstract. The paper deals with the energy consumption evaluation of selected Sparse and Dense BLAS Level 1, 2 and 3 routines. 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). 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%.

INTRODUCTION

This work is performed in the scope of the READEX project (Run-time Exploitation of Application Dynamism for Energy-efficient eXascale computing) [4]. The main goal of the participating institutions in the project is to develop an auto-tuning tool, which improves energy efficiency of extreme-scale applications by employing techniques for dynamically adapting hardware (such as CPU frequency), system-level software, and application parameters of the processing platform. A key component of the project is evaluation of current HPC applications under different system settings. In this paper the manual tuning of the BLAS routines from Intel MKL[1] version 2015, which belong to the most frequently used operations in every HPC application, is presented. For benchmarking we have used the University Florida set of matrices [2] and analyzed the characteristics of the vector AXPY (AXPY) functions, matrix-vector (MV) multiplications, and matrix-matrix (MM) multiplications, for both sparse (Sp) and dense (D) cases. We have tested following functions: cblas_sgemv (DMV float), cblas_dgemv (DMV double), cblas_sgemm (DMM float), cblas_dgemm (DMM double), mkl_dcoogemv (SpMV coo, double), mkl_dcoogemm (SpMDM coo, double), mkl_dcsrgemv(SpMV csr, double), mkl_dcsrgemm (SpMDM csr, double), mkl_dcsrmultcsr (SpMM csr, double). We can achieve energy savings by exploiting the fact that different operations have different computational characteristics and thus achieve minimal energy consumption for different CPU frequencies.

FREQUENCY TUNING AND ENERGY MEASUREMENT METHODOLOGY

The experiments performed in this paper deal with CPU frequency only. The measurements were performed on the Intel Xeon E5-2680 (Intel Haswell micro-architecture) based Taurus supercomputer installed at TU Dresden[1]. The machine contains over 1400 nodes that have an FPGA-based power measurement system called HDEEM (High Definition Energy Efficiency Monitoring), that provides fine-grained and accurate power measurements [3]. The measured data can be accessed through the HDEEM library, allowing developers to take energy measurements before and after the region of interest. We have developed a library of measurement probes on top of the HDEEM library, that can: start or stop the energy measurement and/or; change the CPU clock rate using the `cpufreq` library. With these probes we can measure the energy and average power consumption of any particular function. We have used the HDEEM library version 2.1.5.

[1] `https://doc.zih.tu-dresden.de/hpc-wiki/bin/view/Compendium/SystemTaurus`

NUMERICAL EXPERIMENTS

We have analyzed the performance of the local AXPY, MV and MM with dense (D) and sparse (Sp) matrices (i.e. D BLAS 1, 2 and 3 and Sp BLAS 2 and 3 routines). The D matrices were randomly generated for given sizes (from 300x300 to 16,000x16,000). As Sp input data we have used several matrices from the UF collection [2] with various number of nonzero values and different sparsity patterns. These were used as blocks (one block per one core) of the final block diagonal matrix. The Sp matrices are stored in two matrix formats (CSR, COO). They are square matrices of almost same dimension about 265,000. The matrix IDs and their numbers of nonzero elements are mentioned in the Fig.1.

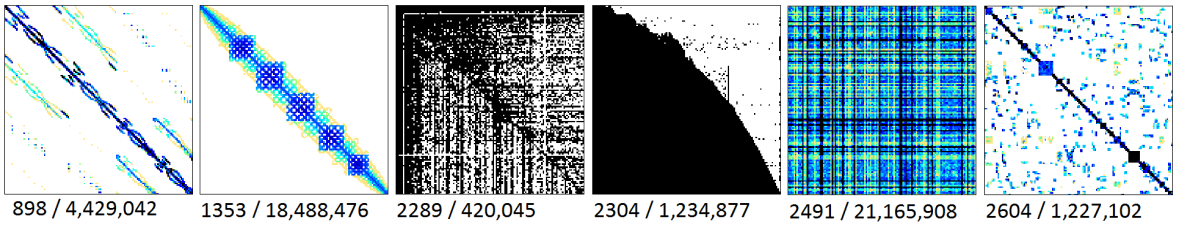

FIGURE 1. Overview of sparse matrices and their IDs/number of nonzeros from University of Florida collection used for testing.

The relation of the CPU frequency and the consumed energy for the operations is depicted in following figures (Figs 2 – 11). The graphs report the mean values computed from 10 repetitions. We have decided to draw just curves of the energy consumption dependency on frequency for several matrix sizes and/or patterns. The absolute values in Joules are omitted, as we are only searching for minimum of the energy consumption. It is obvious, that operations with larger objects result in higher energy consumption. We also present the energy savings compared to execution with default CPU frequency, which for this CPU is 2.5 GHz. This provides the reader with essential information how much energy can be saved if one takes into account CPU frequency tuning.

BLAS 1 Routines

AXPY: The minimum energy consumption for AXPY processing vectors of length 100,000,000 is reached for the lowest frequency 1.2 GHz and energy savings is 19.1%, see Fig. 2. This is due to fact that BLAS 1 routines are strictly memory-bounded operations.

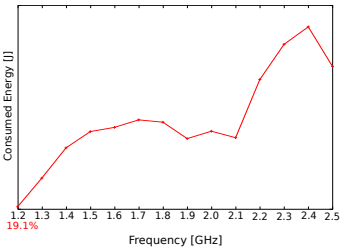

FIGURE 2. Evaluation of AXPY.

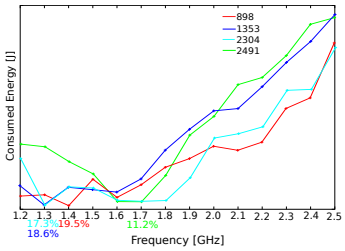

FIGURE 3. Evaluation of SpMV using CSR.

BLAS 2 Routines

SpMV: For CSR format, the minimum energy consumption is reached for the frequency interval 1.3-1.7 GHz. The optimal frequency depends on the matrix pattern. Although this operation is also memory-bounded, we can observe that the optimal frequency is higher when compared to DMV. We believe this is due to fact that the scalar operations dominate over the vector ones, similar effect is achieved also for COO format. The corresponding energy savings then range between 11.2% and 19.5%, see Fig. 3. For COO format, the minimum energy consumption is reached for the frequency interval 1.3-1.8 GHz. The corresponding energy savings then range between 8.3% and 16.9%, see Fig. 4.

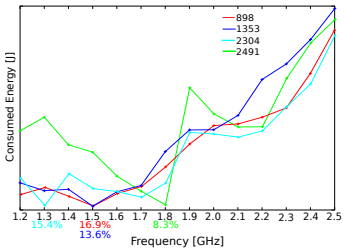

FIGURE 4. Evaluation of SpMV using COO.

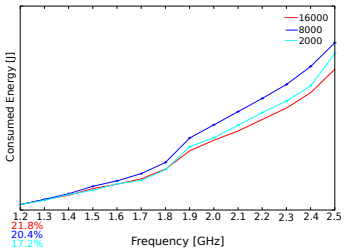

FIGURE 5. Evaluation of DMV using double precision.

DMV:For DMV with elements in double precision, the minimum energy consumption is always reached for the lowest frequency, i.e. for 1.2 GHz independently of the matrix size. The corresponding energy savings then range between 17.2% (for matrix size 2,000) and 21.8% (for matrix size 16,000), see Fig. 5. For DMV with elements in single precision, the situation is identical as for double precision. The difference is that the absolute values of the energy consumption are approximately half. The energy savings range between 21.4% and 23.2%, see Fig. 6. DMV in both, single and double precision are memory-bounded operations similarly to BLAS 1 routines.

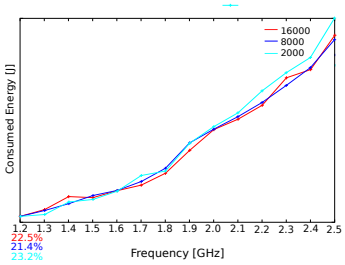

FIGURE 6. Evaluation of DMV using single precision.

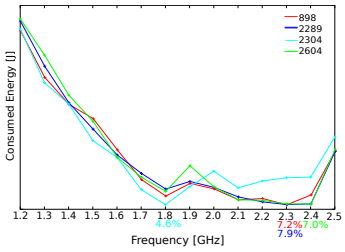

FIGURE 7. Evaluation of SpMM using CSR.

BLAS 3 Routines

SpMM: For SpMM in CSR format, the minimum energy consumption is reached for the frequency interval 1.8-2.4 GHz. This operation utilizes caches more efficiently and can be considered as compute-bounded operation. The optimal frequency depends on the matrix pattern. The corresponding energy savings then range between 4.6% and 7.9%, see Fig. 7.

DMM: For DMM with elements in double precision the minimum energy consumption is reached for the frequency interval 1.8-2.5 GHz. For this frequency range maximum energy consumption of the DMV was achieved. The optimal frequency depends on the matrix size, as with its increase the optimal frequency increases, as this operation becomes more and more compute-bounded. The corresponding energy savings then range between 0% and 9.7%, see Fig. 8. For DMM with elements in single precision, the minimum energy consumption is reached for the frequency

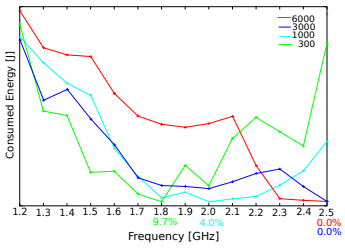

FIGURE 8. Evaluation of DMM using double precision.

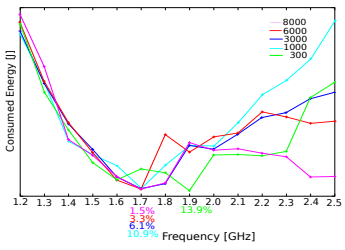

FIGURE 9. Evaluation of DMM using single precision.

interval 1.7-1.9 GHz. The optimal frequency depends on the matrix size, as with its increase the optimal frequency

decreases, which is opposite to the case with double precision. The corresponding energy savings then range between 3.3% and 13.9%, see Fig. 9.

Action	AXPY	SpMV CSR	SpMV COO	SpMM CSR	SpMM COO	DMV FLOAT	DMV DOUBLE	DMM FLOAT	DMM DOUBLE	SpMDM CSR-DOUBLE	SpMDM COO-DOUBLE
Opt.freq. [GHz]	1.2	1.3-1.7	1.3-1.8	1.8-2.4	2.1	1.2	1.2	1.7-1.9	1.8-2.5	1.2-1.7	1.2-1.8
Savings [%]	19	11-20	8-17	5-8	2.1	21-23	17-22	2-14	0-10	12-20	13-20

TABLE 1. Efficiency of frequency tuning for BLAS 1,2,3 routines.

SpMDM: For SpMDM where Sp matrix is stored in CSR format, D matrix is in double precision and its sizes are (size(SpM,2),100), the minimum energy consumption is reached for the frequency interval 1.2-1.7 GHz. The energy savings are between 11.9% and 20.2%, see Fig. 10. For SpMDM where Sp matrix is stored in COO format, D matrix

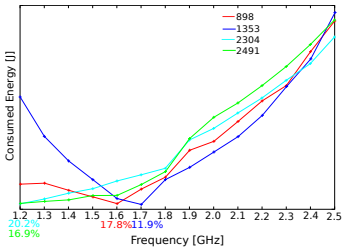

FIGURE 10. Evaluation of SpMDM using CSR and double precision.

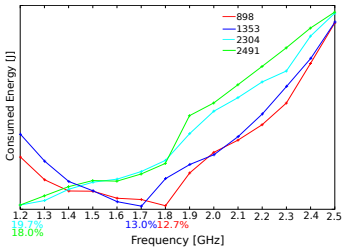

FIGURE 11. Evaluation of SpMDM using COO and double precision.

in double precision and sizes are (size(SpM,2),100), the minimum energy consumption is reached for the frequency interval 1.2-1.8 GHz. In both last cases shift of the minimum to higher frequencies with increasing concentration of nonzeros around the main diagonal can be observed. The energy savings are between 12.7% and 19.7%, see Fig. 11.

CONCLUSION

We have presented our initial energy consumption evaluation of selected sparse and dense BLAS routines on one computational node. Presented optimal frequencies are able to achieve significant energy savings up to 23%. The complex overview of recommended optimal frequencies and associated savings related to the default frequency 2.5 GHz is then depicted in the Tab. 1.

ACKNOWLEDGMENTS

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

REFERENCES

[1] Intel Math Kernel Library (MKL), https://software.intel.com/en-us/intel-mkl

[2] Sparse Matrix Algorithms Research at the University of Florida, containing the UF sparse matrix collection, https://www.cise.ufl.edu/research/sparse/matrices/

[3] 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`

[4] Run-time Exploitation of Application Dynamism for Energy-efficient eXascale computing (READEX), `http://www.readex.eu/`