SpMV and SpMM Performance Measurement

Elias Maciel[1], Pedro Torres[1,2], Christian E. Schaerer[1,2]

[1]Facultad Politécnica, Universidad Nacional de Asunción, P.O. BOX 2111 SL, San Lorenzo, Paraguay.
[2]Centro de Investigación en Matemática-CIMA, Asunción, Paraguay.

*eliasmacielr@gmail.com, torres.pedrozpk@gmail.com, cschaer@pol.una.py

ABSTRACT

Matrix-vector multiplication (MV) is a widely applied operation in diverse areas of science and engineering. In this work we explore the implications between execution time and data movement that take place in two particular cases of MV for which the matrix is sparse, SpMV and SpMM. Metrics are established for the evaluation of the performance attained with both routines and the results obtained with the configuration settings are compared.

SpMV

This computational kernel consists of a matrix-vector product of a sparse $m \times n$ matrix A and a dense vector $x \in \mathbb{R}^n$. [BG14]

$$y = A * x \tag{1}$$

SpMM

This operation is a generalization of SpMV in which the sparse $m \times n$ matrix A is multiplied by a dense $n \times p$ matrix B ($p \ll n$) that represents a set of dense vectors. [AW14]

$$C = A * B \tag{2}$$

MATRICES

The set of matrices used for the experiments comes from relevant study cases.

Name	Application Area	Dimension	Nonzeros
finan512	Portfolio optimization	74752x74752	335872
mcfe	Astrophysics	765x765	24382
pwt	Structural engineering	36519x36519	181313
raefsky3	Fluid/structure	21200x21200	1488768
s3dkq4m2	Finite element analysis	90449x90449	1921955
sherman5	Oil reservoir	3312x3312	20793

HARDWARE

The Table below shows the processors and compilers used for the experiments.

CPU	Cache	RAM	Compiler
AMD Phenom II X4 955	L1: 64 KB L2: 512 KB L3: 6 MB	4 GB	GCC 6.1.1
Intel Xeon E5530	L1: 64 KB L2: 512 KB L3: 8 MB	16 GB	GCC 4.5.2

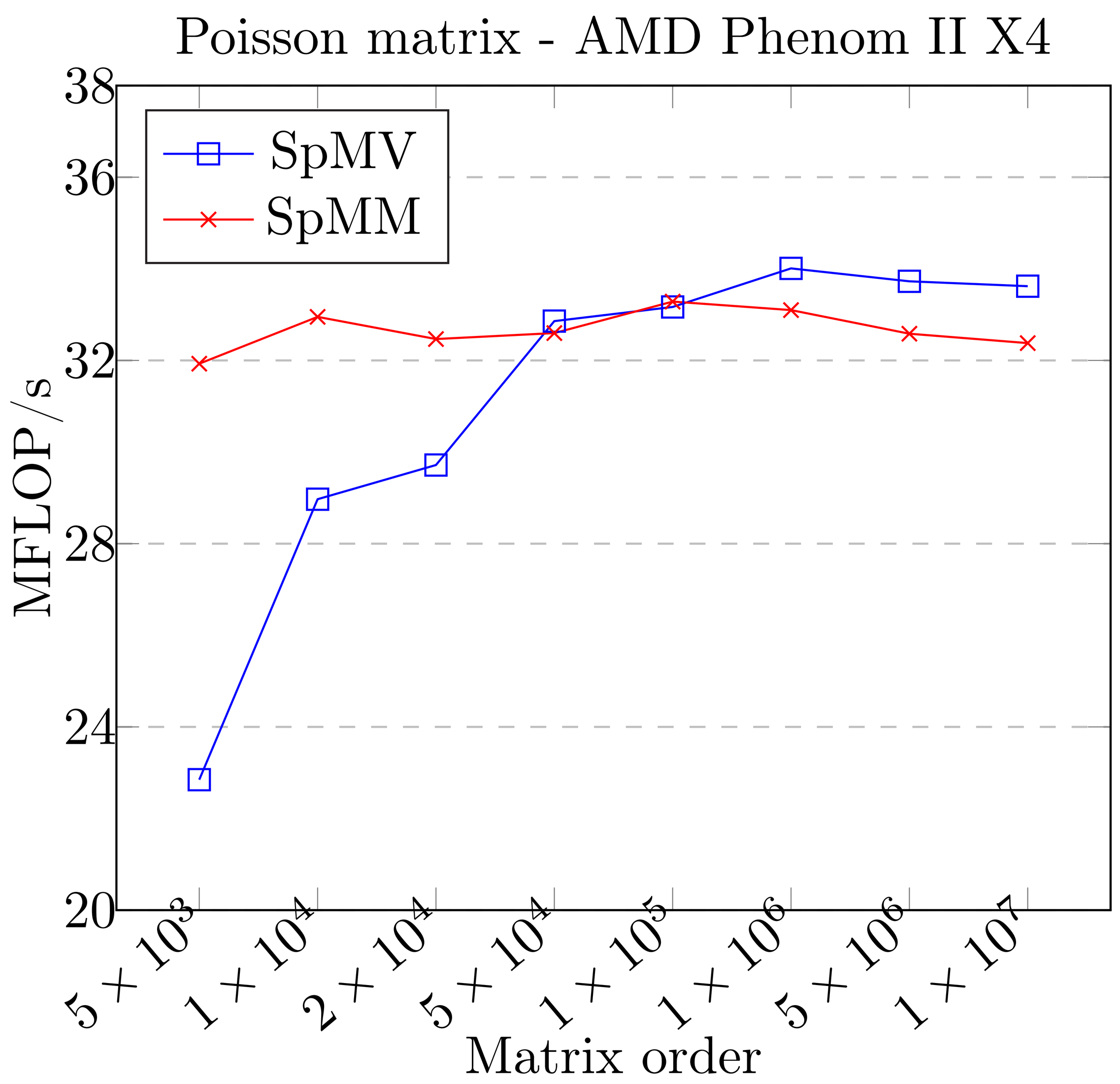
RESULTS

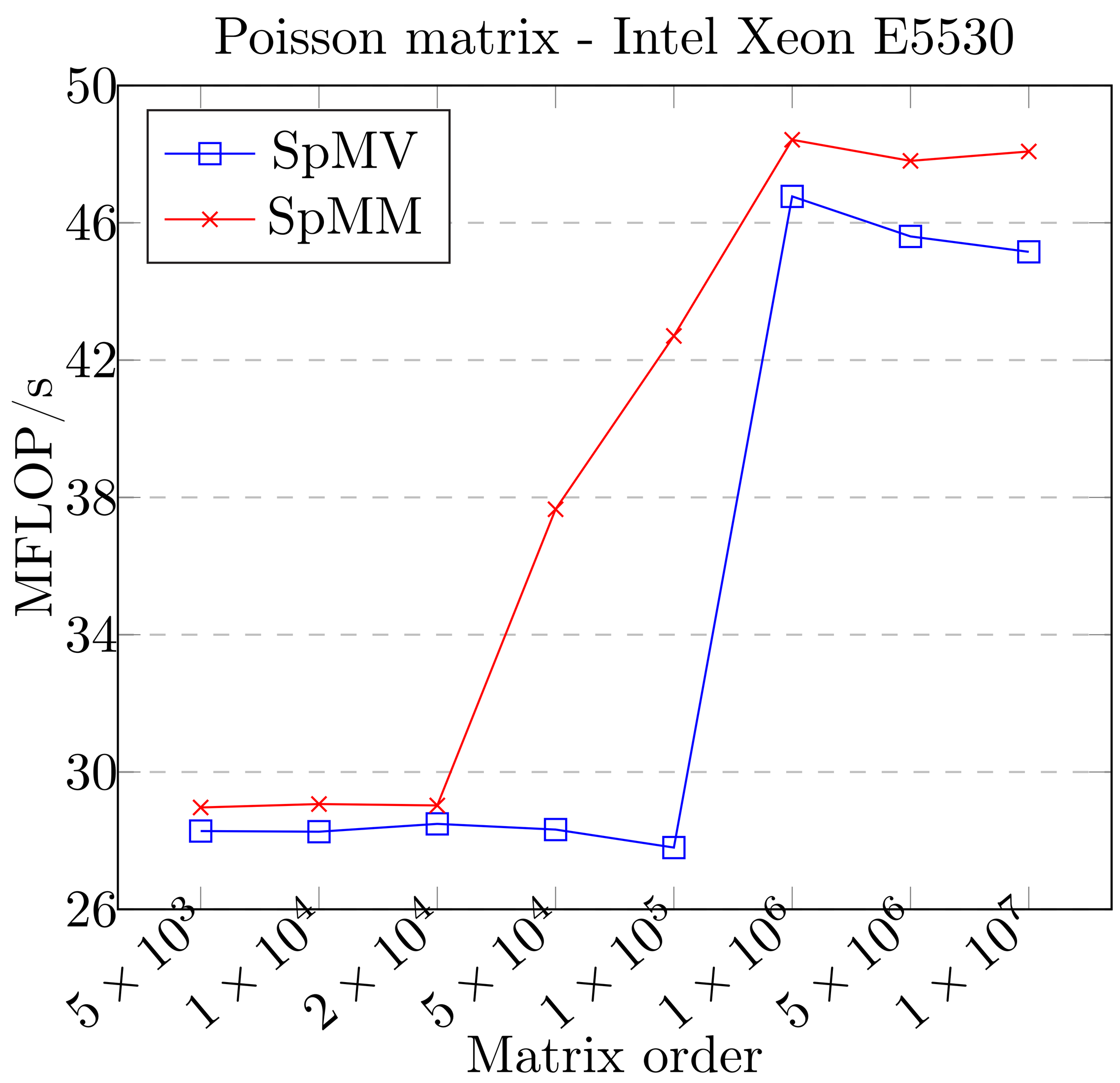
		MFLOP/s	
Matrix	**Dimension**	**SpMV**	**SpMM** ($p = 10$)
finan512	74752x74752	32.416	37.2906
mcfe	765x765	34.7158	48.2067
pwt	36519x36519	27.9292	33.8493
raefsky3	21200x21200	50.1089	51.5167
s3dkq4m2	90449x90449	46.7826	49.4118
sherman5	3312x3312	28.4741	39.2228

Table 1: Tests on AMD Phenom II X4

		MFLOP/s	
Matrix	**Dimension**	**SpMV**	**SpMM** ($p = 10$)
finan512	74752x74752	42.2557	53.2811
mcfe	765x765	45.6382	66.2403
pwt	36519x36519	36.2499	45.1379
raefsky3	21200x21200	81.3873	83.6199
s3dkq4m2	90449x90449	73.1416	77.6655
sherman5	3312x3312	32.9084	55.4644

Table 2: Tests on Intel Xeon E5530





CONCLUSIONS

The SpMM kernel outperforms SpMV when the cache size is enough to hold the matrices involved in the computation.

ACKNOWLEDGEMENTS

CES acknowledges PRONII-CONACyT-Paraguay. This work was partially supported by Project CONACyT: 14-INV-186 CABIBESKRY. EM thanks Polytechnic School, National University of Asuncion-FPUNA.

REFERENCES

[BG14] Ballard, G., Carson, E., Demmel, J., Hoemmen, M., Knight, N. and Schwartz, O., 2014. Communication lower bounds and optimal algorithms for numerical linear algebra. *Acta Numerica, 23*, pp.1-155.

[AW14] Abu-Sufah, W. and Ahmad, K., 2014. Performance of Sparse Matrix-Multiple Vectors Multiplication on Multicore and GPUs. In *International Conference for High Performance Computing, Networking, Storage and Analysis, Supercomputing 2014 (SC 14)* (pp. 933-938).