# Multilevel Parallelization Approach to Estimate Spin Lifetime in Silicon: Performance Analysis

J. Ghosh[1], D. Osintsev[2], V. Sverdlov[2], and S. Ganguly[1]
[1]Department of Electrical Engineering, Indian Institute of Technology, Mumbai, India
[2]Institute for Microelectronics, TU Wien, Gußhausstraße 27–29/E360, A–1040 Wien, Austria
e-mail: {joydeepghosh}@ee.iitb.ac.in

***Abstract*- We analyze the performance of a highly parallelized algorithm to calculate the spin lifetime in silicon thin films. The two-level parallelization algorithm is based on a hybrid parallelization approach, using the message passing interface MPI as well as OpenMP. Most efficient way to utilize the computational resources is described.**

***Key words- Spin lifetime, MPI, OpenMP, Multi-Level Parallel Computing.***

Now-a-days it is becoming possible to solve more and more complex problems, because high performance computational resources are widely accessible for the practical calculations. A lot of efforts have also been directed to utilize the computational power in the most effective way [1,2]. Practically, all systems that belong to TOP500 supercomputers are based on multi-core CPUs [3]. A considerable part of algorithms in physics can be effectively parallelized by dividing the domain into independent parts [4]. Each part can be calculated in a single MPI process without much efforts devoted to communication between separate MPI processes. Application of such approach is limited if each MPI process requires a lot of memory or intensive communication. In some cases, the memory requirements can be significantly reduced if the calculations are performed on a shared memory. For the class of problems for which shared memory can significantly reduce the total amount of memory requirements, a combination between MPI and OpenMP approach is quite promising [1,2,4].

Here we consider a problem of finding a spin lifetime ($\tau$) in (001) ultra-thin silicon films subjected to [110] uniaxial tensile stress ($\varepsilon_{xy}$) [5-10]. The considered problem belongs to a class of problems that can effectively use shared memory to reduce an on-node memory requirements in the combination with weakly coupled domain decomposition. In our earlier work, a two-level parallelization algorithm to solve the problem has been mentioned [11]. At the first stage all static wave functions and energy data are calculated and archived in a binary file as a file-based cache technique which requires about 7GB for one stress point with high accuracy (Level 1). At the second stage the spin lifetime is calculated by loading and using these data in cache memory (Level 2). In this paper, we scrutinize the performance of the algorithm in every level and explain the most efficient way to utilize the resources.

The performance is measured on the Vienna Scientific Cluster (VSC-2) [12]. Each node of the cluster has 2 processors (AMD Opteron 6132 HE, 2.2 GHz and 8 cores) and 32GB main memory. We have examined different configurations of MPI and OpenMP with a fixed number of cores 96 (i.e. number of nodes is 6). A pure-MPI based configuration in Level 1 demands maximally around 5 GB memory per node, which can be perfectly fitted in any modern supercomputer. However, the calculation time increases, when the number of threads is increased and keeping a fixed number of cores (Fig. 1). Therefore, we find that our cache approach is most efficient for a pure MPI configuration.

In the second level, we compute $\tau$ by using the archived data in parallel. For variations in our discretization parameters (energy $E$ and backscattering angle $\Phi$), the size of the archived cache is shown in Fig. 2. The smaller $E$ and $\Phi$ steps improve the computational accuracy but increase the required cache size. As this huge amount of cache is to be loaded into memory by each MPI job in Level 2, it becomes inevitable to use a hybrid MPI-OpenMP configuration for the spin lifetime computations. Fig. 3 shows maximum memory per node for 8 and 16 threaded MPI application. Each MPI process reads 7GB cache file, thus number of threads is limited by 8. The memory requirements of the application are mainly determined by the size of the serialized cache. The dependence of the total calculation time on the number of cores for a fixed threads number is shown in Fig. 4. Contrary to Fig. 1, increasing the thread numbers from 8 to 16 leads to decrease of the total calculation time for all values of the cores numbers.

This approach is tested with 416 cores and requires only around 40 min for a single relaxation time data point (around 280 core-hours). Finally, our computations show how shear strain can dramatically increase $\tau$ by orders of magnitude (Fig. 5).

## ACKNOWLEDGEMENT

We acknowledge support from the journal Electronics. The computational results presented have been partly achieved using the Vienna Scientific Cluster (VSC).

## REFERENCES

[1] G. Jost *et al.*, "Comparing the OpenMP, MPI, and Hybrid Programming Paradigms on an SMP Cluster", NAS Technical Report NAS-03-019, 2003.

[2] R Rabenseifner *et al.*, "Hybrid MPI/OpenMP Parallel Programming on Clusters of Multi-Core SMP Nodes", proc. in Euromicro International Conference on Parallel, Distributed and Network-based Processing (2009).

[3] Top500 supercomputers, http://www.top500.org/lists/2017/11/.

[4] S. Tang *et al.*, "Speedup for Multi-Level Parallel Computing", proc. in International Parallel and Distributed Processing Symposium Workshops & PhD Forum (2012).

[5] J. Ghosh *et al.*, "Intersubband Spin Relaxation Reduction and Spin Lifetime Enhancement by Strain in SOI Structures", Microelectronic Engineering 147, 89 (2015).

[6] J. Ghosh *et al.*, "Variation of Spin Lifetime with Spin Injection Orientation in Strained Thin Silicon Films", ECS Transactions 66:5, 233 (2015).

[7] J. Ghosh *et al.*, "Injection Direction Sensitive Spin Lifetime Model in a Strained Thin Silicon Film", proc. in Simulation of Semiconductor Processes and Devices, p.277 (2015).

[8] J. Ghosh *et al.*, "Influence of Valley Splitting on Spin Relaxation Time in a Strained Thin Silicon Film", proc. in International Workshop on Computational Electronics, p.1-4 (2015).

[9] J. Ghosh *et al.*, "Enhancement of Electron Spin Relaxation Time in Thin SOI Films by Spin Injection Orientation and Uniaxial Stress", Journal of Nano Research 39, 34 (2016).

[10] V. Sverdlov and S. Selberherr, "Silicon Spintronics: Progress and Challenges", Physics Reports 585, p.1-40 (2015).

[11] J. Ghosh *et al.*, "Evaluation of Spin Lifetime in Thin-Body FETs: A High Performance Computing Approach", Lecture Notes in Computer Science 9374, 285 (2015).

[12] Vienna Scientific Cluster, http://www.vsc.ac.at/systems/vsc-2/.

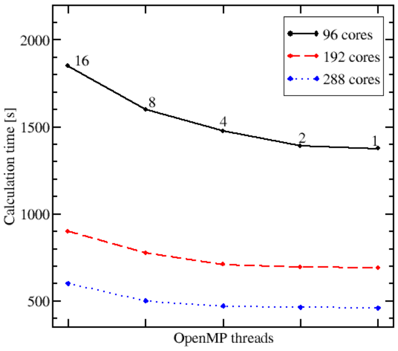

Fig. 1. Dependence of calculation time (Level 1) on different number of threads for a fixed core number is shown.

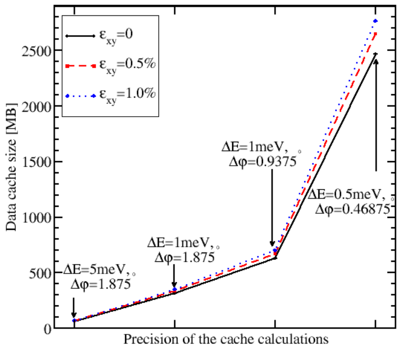

Fig. 2. Dependence of the size of cache on the precision of the calculations fixed by energy and angle steps for different $\varepsilon_{xy}$ values is shown. Sample thickness *t*=1.5nm.

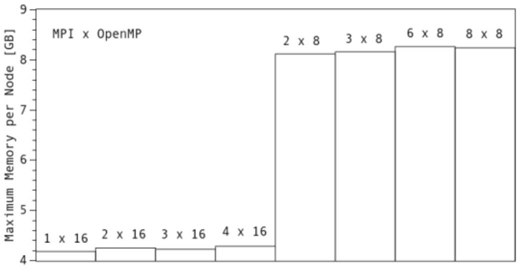

Fig. 3. Maximum required memory per node as reported by VSC-2 for different configurations is shown (Level 2).

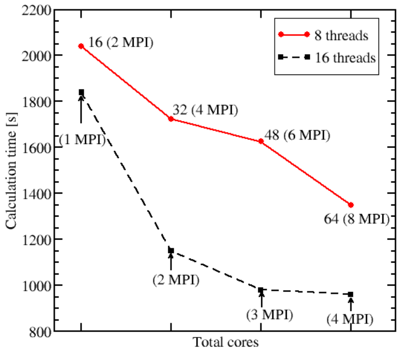

Fig. 4. Dependence of the total calculation time of the spin relaxation on total cores for 8 and 16 threads per MPI process.

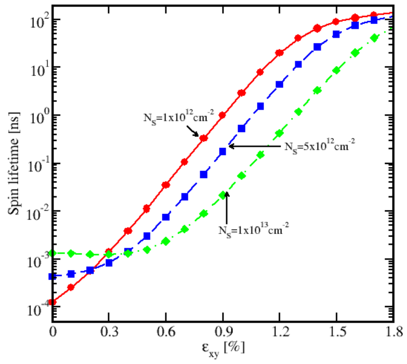

Fig. 5. Variation of spin lifetime with $\varepsilon_{xy}$ is shown. *t*=1.36nm, and electron concentration $N_s$ is a parameter.