Journal of Computational Electronics (2019) 18:28–36
https://doi.org/10.1007/s10825-018-1274-x

Efficient two-level parallelization approach to evaluate spin relaxation in a strained thin silicon film

Joydeep Ghosh[1] · Dmitry Osintsev[2] · Viktor Sverdlov[3]

Published online: 9 November 2018

Abstract

The evaluation of the spin lifetime in an ultra-thin silicon film is a massive computational challenge because of the necessity of performing appropriate double integration of the strongly scattering momentum-dependent spin relaxation rates. We have tackled the problem by dividing the whole computation range into two levels. Our scheme in each level is based on a hybrid parallelization approach, using the message passing interface MPI and OpenMP. In the first level, the algorithm precalculates the subband wave functions corresponding to fixed energies and archives the results in a file-based cache to reduce memory consumption. In the second level, we compute the spin relaxation time by using the archived data in parallel. This two-level computation approach shows an excellent parallel speedup, and most efficient ways to maximally utilize the computational resources are described. Finally, how an application of shear strain can dramatically increase the spin lifetime is shown.

Keywords Message passing interface · Open Multi-Processing · Hybrid parallelization · Spin lifetime

1 Introduction

Continuous miniaturization of CMOS devices is the main reason behind the phenomenal increase in speed and density of modern integrated circuits (ICs). However, in this journey, growing technological challenges and soaring costs have gradually caused MOSFET scaling to an end. Utilizing electron spin as an additional degree of freedom is gaining importance for further improving the efficiency of future low-power ICs [1]. On the other hand, silicon, the main material in microelectronics, is composed of nuclei with almost zero magnetic moment, and its weak spin–orbit interaction leads to a long spin lifetime. As of the paramount importance of the ongoing shift from bulk field-effect transistors (FETs) to transistors with the channel built on ultra-thin body (UTB) silicon-on-insulator films and FinFET 3D technology for technology nodes of 14 nm and beyond, spin lifetime in such structures is a very relevant issue under scrutiny. The lower value of the spin lifetime obtained with a three-terminal injection scheme is of the order 0.1–1 ns [2], which corresponds to a spin diffusion length maximum of 2μm. Therefore, a long distance spin propagation combined with a possibility of injecting spin at room temperature [3] makes the fabrication of spin-based switching devices possible in the near future. Experimentally observed higher values of the spin relaxation in electrically gated silicon structures, however, can become a hindrance in realizing spin-driven devices [1]. These issues demand a deeper understanding of the spin relaxation mechanism fundamentals in silicon MOSFETs. It is mentioned here that the calculation of the spin relaxation rate is very computationally expensive. This is because one has to calculate the strongly scattering momentum-dependent spin relaxation rates, which, in turn, demands a highly parallelized computational approach. The computational details will be elaborated in the following sections.

Nowadays, supercomputers are playing major role in the field of computational electronics. It has become feasible to solve more and more complex problems, because high-performance computational resources are accessible for practical calculations. Message passing interface (MPI)

✉ Joydeep Ghosh
joydeepghosh@ee.iitb.ac.in

Dmitry Osintsev
dmitri.osintsev@singularis-lab.com

Viktor Sverdlov
sverdlov@iue.tuwien.ac.at

[1] Department of Electrical Engineering, Indian Institute of Technology, 400076 Mumbai, India

[2] Singularis Lab, Grushevskaya 10, 400074 Volgograd, Russia

[3] Institute for Microelectronics, TU Wien, Gußhausstraße 27–29, 1040 Wien, Austria

is a standard for writing message passing programs which functions over a wide variety of parallel computing architectures [4]. MPI provides an user a programming model where processes communicate with each other by calling library routines to send and receive messages, known as distributed memory computing. On the other hand, Open Multi-Processing (OpenMP) is an application programming interface (API) which supports multi-platform shared memory programming [5,6]. It is also possible to combine MPI and OpenMP programming models into a hybrid paradigm to exploit parallelism. As because more processor cores are dedicated to large clusters of symmetric multi-processor (SMP) nodes solving scientific problems, this hybrid programming technique combining the best of distributed and shared memory programs is gaining popularity over time. Of course, major efforts must be directed to utilize the computational power in the most effective way.

A considerable part of most algorithms in any computational problem can be parallelized by dividing the domain into independent parts, known as domain decomposition. Each part is calculated in a single MPI process without much efforts devoted to communication between separate MPI processes. However, such an approach becomes limited if each MPI process requires a high amount of memory or intensive communication. Indeed, need of a large memory requirement would depend on the selected problem as well as the chosen algorithm to solve it. In some cases, memory requirements are significantly reduced if the calculations are performed by sharing memory. However, as the number of cores per node is limited, the reduction of memory requirements leads to an unacceptable increase of the total calculation time. Nevertheless, for the class of problems for which shared memory significantly reduces the total amount of memory requirements, a combination between MPI and OpenMP approach becomes promising [6,7].

Several spin relaxation mechanisms in silicon are now discussed in brief [8–10]. In bulk silicon, Dyakonov–Perel spin relaxation mechanism is absent because of inversion symmetry in the silicon lattice. At high temperatures, the spin relaxation due to the Elliot–Yafet mechanism becomes most dominating. This mechanism is mediated by the intrinsic interactions between the electronic orbital motion with its spin. The spin–orbit interaction (SOI) does not conserve the electron spin; thus, it can generate spin flips, which is the Yafet process. When the microscopic SOI is considered, the Bloch function with a fixed spin projection is not an eigenfunction of the total Hamiltonian, and the eigenfunction contains a contribution with an opposite spin projection. This means that the SOI forces the eigenstate wave function to possess a nonzero contribution with an opposite spin projection in the fixed basis. Henceforth, even any spin-independent scattering with phonons can generate a small probability of spin flips, which is the Elliot process. In the next section, we explain the spin relaxation model in a UTB silicon film.

2 Model

We calculate the spin lifetime (τ_S) in (001) ultra-thin silicon films subjected to [110] uniaxial tensile stress ε_{xy}. We take into account the main mechanisms determining the mobility in thin silicon films, namely surface roughness (SR) and electron–phonon scattering (longitudinal LA and transversal acoustic TA phonons), and analyze their role in spin relaxation. The total spin lifetime is calculated by the Matthiessen rule. In this work, both, the Elliot and the Yafet processes are taken on equal footing [11]. The spin-flip scattering processes between two [001] valleys are responsible for spin relaxation in thin (001) silicon films [11]. The unprimed electron subband energies and the wave functions are obtained with the two-band $\mathbf{k}\cdot\mathbf{p}$ Hamiltonian describing the [001] valley dispersion including intrinsic spin–orbit coupling [10,12,13]. This Hamiltonian is written at the vicinity of the X-point along the Z-axis in the Brillouin zone. As the lowest two conduction bands have their minima just k_0 away from the X-point in the Brillouin zone, a two-band $\mathbf{k}\cdot\mathbf{p}$ Hamiltonian considering only these two bands developed near the X-point describes the band dispersion and subband wave functions very well [11]. This Hamiltonian is shown below.

$$H = \begin{bmatrix} H_1 & H_3 \\ H_3 & H_2 \end{bmatrix}, \qquad (1)$$

$$H_{j=1,2} = \left[\frac{\hbar^2 k_z^2}{2m_l} + \frac{\hbar^2 \left(k_x^2 + k_y^2\right)}{2m_t} + (-1)^j \delta + U(z) \right] I \qquad (2)$$

$$H_3 = \begin{bmatrix} \frac{\hbar^2 k_0 k_z}{m_l} & 0 \\ 0 & \frac{\hbar^2 k_0 k_z}{m_l} \end{bmatrix}, \qquad (3)$$

with

$$\delta = \sqrt{\left(D\varepsilon_{xy} - \frac{\hbar^2 k_x k_y}{M}\right)^2 + \triangle_{SO}^2 \cdot (k_x^2 + k_y^2) + \Lambda_\Gamma^2}\,. \qquad (4)$$

The energy dispersion equation is given by

$$E = \frac{\hbar^2 k_z^2}{2m_l} + \frac{\hbar^2 (k_x^2 + k_y^2)}{2m_t} \pm \sqrt{\left(\frac{\hbar^2 k_z k_0}{m_l}\right)^2 + \delta^2}. \qquad (5)$$

Here (k_x, k_y, k_z) represents the $\mathbf{k}$ vector, and $U(z)$ is the confinement potential. For other parameters, refer to Table 1. $U(z)$ is approximated by an infinite square-well potential of width t. The wave functions fulfill the Schrödinger equation with Eq. 1 and satisfy the zero boundary conditions at the interfaces. Λ_Γ pertains to the unprimed subband splitting at unstrained silicon (001) films $\Lambda_\Gamma = \frac{\Delta_\Gamma \cdot k_0^3}{k_{0\Gamma}^3}$ [14]. Therefore,

Table 1 Simulation parameter list

Parameter	Value
Silicon lattice constant	$a = 0.5431\,\text{nm}$
Intrinsic spin–orbit field	$\triangle_{SO} = 1.27\,\text{meVnm}$
Shear deformation potential	$D = 14\,\text{eV}$
Acoustic deformation potential	$\Xi = 12\,\text{eV}$
Electron rest mass in silicon	m_e
Transversal effective mass	$m_t = 0.19\,m_e$
Longitudinal effective mass	$m_l = 0.91\,m_e$
M	$(m_t^{-1} - m_e^{-1})^{-1}$ Kg
Valley minimum position from X-point	$k_0 = 0.15 \cdot \frac{2\pi}{a}$
Deformation potential due to intrinsic SOI	$D_{SO} = 15\,\text{eV}/\text{k}_0$
Splitting at Γ-point	$\Delta_\Gamma = 5.5\,\text{eV}$
$k_{0\Gamma}$	$k_{0\Gamma} = 0.85 \cdot \frac{2\pi}{a}$
Autocorrelation length	$L = 15 \cdot 10^{-10}\,\text{m}$
Mean square value of the surface roughness fluctuations	$\Delta = 3 \cdot 10^{-10}\,\text{m}$

Δ_Γ defines the strength of the valley-orbit interaction with its reported value to be 5.5eV using a sp^{3}d^{5}s* spin–orbit coupled tight-binding model [15]. Once subband wave functions and subband dispersions are known, the spin relaxation matrix elements can be calculated [11,16,17]. Then, we calculate the SR and phonon-mediated spin relaxation time. Based on that, one can calculate the spin relaxation time by thermal averaging method:

$$\frac{1}{\tau} = \frac{\sum_i \int \frac{1}{\tau_i(\mathbf{K_1})} \cdot f(E)(1 - f(E))\mathrm{d}\mathbf{K_1}}{\sum_i \int f(E)\mathrm{d}\mathbf{K_1}} \tag{6}$$

where

$$f(E) = \frac{1}{1 + \exp\left(\frac{E - E_F}{K_B T}\right)}\,, \tag{7}$$

$$\int \mathrm{d}\mathbf{K_1} = \int_0^{2\pi} \mathrm{d}\tilde{\phi} \cdot \int_0^{\infty} \frac{|\mathbf{K_1}(\tilde{\phi}, E)|}{|\frac{\partial E(\mathbf{K_1})}{\partial \mathbf{K_1}}|_{\mathbf{K_1}}} \mathrm{d}E\,. \tag{8}$$

$\mathbf{K_1}$ is the in-plane subband wave vector, K_B is the Boltzmann constant, T is the temperature, $\tilde{\phi}$ denotes the wave vector direction, and E_F is the Fermi level. The term $|\frac{\partial E(\mathbf{K_1})}{\partial \mathbf{K_1}}|_{\mathbf{K_1}}$ is the derivative of the subband dispersion along $\mathbf{K_1}$ at the angle $\tilde{\phi}$. E can be expressed as $E = E_i^{(0)} + E_i(\mathbf{K_1})$, where $E_i^{(0)} = E_i(\mathbf{K_1} = 0)$. $E_i^{(0)}$ is the energy of the bottom of the subband i. Equation 7 represents the Fermi–Dirac distribution function.

The expression of the SR-limited spin relaxation rate is shown below. The SR scattering matrix elements are considered to be proportional to the product of the subband function derivatives at the interface [13,18]. The surface roughness at the two interfaces is assumed to be equal and statistically independent.

$$\begin{aligned}\frac{1}{\tau_{i,SR}(\mathbf{K_1})} &= \frac{4\pi}{\hbar(2\pi)^2} \sum_{j=1,2} \int_0^{2\pi} \pi \triangle^2 L^2 \cdot \frac{1}{\epsilon_{ij}^2(\mathbf{K_2} - \mathbf{K_1})} \cdot \\ &\quad \frac{\hbar^4}{4m_l^2} \cdot \frac{|\mathbf{K_2}|}{|\frac{\partial E(\mathbf{K_2})}{\partial \mathbf{K_2}}|} \cdot \left[\left(\frac{\mathrm{d}\psi_{i\mathbf{K}_{1\sigma}}}{dz}\right)^* \left(\frac{\mathrm{d}\psi_{j\mathbf{K}_{2-\sigma}}}{dz}\right)\right]^2_{z=\pm\frac{t}{2}} \\ &\quad \cdot \exp\left(\frac{-(\mathbf{K_2} - \mathbf{K_1})^2 L^2}{4}\right)\mathrm{d}\phi\end{aligned} \tag{9}$$

here $\mathbf{K}_1$ ($\mathbf{K}_2$) is the in-plane wave vector before (after)-scattering, ϕ is the angle between $\mathbf{K}_1$ and $\mathbf{K}_2$, ϵ is the dielectric permittivity, $\psi_{i\mathbf{K}_{1\sigma}}$ and $\psi_{j\mathbf{K}_{2\sigma}}$ are the wave functions, and $\sigma = \pm 1$ is the spin projection to the [001] axis. The rest of the notations can be found in Table 1. The detailed phonon-mediated spin relaxation time calculation methods are briefly mentioned below [11,13,14,19].

The TA-phonon-induced intravalley spin relaxation rate can be written as:

$$\begin{aligned}\frac{1}{\tau_{i,TA}(\mathbf{K_1})} &= \frac{\pi K_B T}{\hbar \rho v_{TA}^2} \sum_j \int_0^{2\pi} \mathrm{d}\phi \cdot \frac{|\mathbf{K}_2|}{|\frac{\partial E(\mathbf{K}_2)}{\partial \mathbf{K}_2}|} \\ &\quad \left[1 - \frac{|\frac{\partial E(\mathbf{K}_2)}{\partial \mathbf{K}_2}| f(E(\mathbf{K}_2))}{|\frac{\partial E(\mathbf{K}_1)}{\partial \mathbf{K}_1}| f(E(\mathbf{K}_1))}\right] \int_0^t \int_0^t \exp(-\sqrt{q_x^2 + q_y^2}|z - z'|) \\ &\quad \left[\psi^\dagger_{\mathbf{K}_{2\sigma}}(z)\tilde{M}\psi_{\mathbf{K}_{1-\sigma}}(z)\right]^* \left[\psi^\dagger_{\mathbf{K}_{2\sigma}}(z')\tilde{M}\psi_{\mathbf{K}_{1-\sigma}}(z')\right] \cdot \\ &\quad \left[\sqrt{q_x^2 + q_y^2} - \frac{8q_x^2 q_y^2 - (q_x^2 + q_y^2)^2}{q_x^2 + q_y^2}|z - z'|\right] \mathrm{d}z\mathrm{d}z'\end{aligned} \tag{10}$$

$v_{TA} = 5300\,\text{m/s}$ is the transversal phonon velocity, $\rho = 2329\,\text{kg/m}^3$ is the silicon density, t is the film thickness, (q_x, q_y)=$\mathbf{K_1} - \mathbf{K_2}$, and $\tilde{M}$ written in the two valley plus two spin projection basis is $\begin{bmatrix} 0 & 0 & \frac{D}{2} & 0 \\ 0 & 0 & 0 & \frac{D}{2} \\ \frac{D}{2} & 0 & 0 & 0 \\ 0 & \frac{D}{2} & 0 & 0 \end{bmatrix}$.

The intravalley spin relaxation rate due to LA-phonons is given by:

$$\begin{aligned}\frac{1}{\tau_{i,LA}(\mathbf{K_1})} &= \frac{\pi K_B T}{\hbar \rho v_{LA}^2} \sum_j \int_0^{2\pi} \mathrm{d}\phi \cdot \frac{|\mathbf{K_2}|}{|\frac{\partial E(\mathbf{K_2})}{\partial \mathbf{K_2}}|} \\ &\quad \left[1 - \frac{|\frac{\partial E(\mathbf{K_2})}{\partial \mathbf{K_2}}| f(E(\mathbf{K}_2))}{|\frac{\partial E(\mathbf{K_1})}{\partial \mathbf{K_1}}| f(E(\mathbf{K_1}))}\right] \\ &\quad \int_0^t \int_0^t \exp\left(-\sqrt{q_x^2 + q_y^2}|z - z'|\right) \\ &\quad \left[\psi^\dagger_{\mathbf{K}_{2\sigma}}(z)\tilde{M}\psi_{\mathbf{K}_{1-\sigma}}(z)\right]^* \left[\psi^\dagger_{\mathbf{K}_{2\sigma}}(z')\tilde{M}\psi_{\mathbf{K}_{1-\sigma}}(z')\right] \cdot \\ &\quad \frac{4q_x^2 q_y^2}{(q_x^2 + q_y^2)^{3/2}} \left[\sqrt{q_x^2 + q_y^2}|z - z'| + 1\right] dzdz'\end{aligned} \tag{11}$$

$v_{LA} = 8700\,\text{m/s}$ is the longitudinal phonon velocity.

The intervalley spin relaxation rate due to acoustic phonons (containing Elliot and Yafet contributions) is to be included as well:

$$\begin{aligned}\frac{1}{\tau_{i,Y,LA}(\mathbf{K_1})} &= \frac{\pi K_B T}{\hbar \rho v_{LA}^2} \sum_j \int_0^{2\pi} \mathrm{d}\phi \cdot \frac{|\mathbf{K_2}|}{|\frac{\partial E(\mathbf{K_2})}{\partial \mathbf{K_2}}|} \\ &\left[1 - \frac{|\frac{\partial E(\mathbf{K_2})}{\partial \mathbf{K_2}}| f(E(\mathbf{K_2}))}{|\frac{\partial E(\mathbf{K_1})}{\partial \mathbf{K_1}}| f(E(\mathbf{K_1}))}\right] \\ &\int_0^t \left[\psi^\dagger_{\mathbf{K}_{2\sigma}}(z) M' \psi_{\mathbf{K}_{1-\sigma}}(z)\right]^* \left[\psi^\dagger_{\mathbf{K}_{2\sigma}}(z) M' \psi_{\mathbf{K}_{1-\sigma}}(z)\right] dz.\end{aligned} \tag{12}$$

$M' = \begin{bmatrix} M_{ZZ} & M_{SO} \\ M^\dagger_{SO} & M_{ZZ} \end{bmatrix}$ with $M_{ZZ} = \begin{bmatrix} \Xi & 0 \\ 0 & \Xi \end{bmatrix}$. We also have $M_{SO} = \begin{bmatrix} 0 & D_{SO}(r_y - ir_x) \\ D_{SO}(-r_y - ir_x) & 0 \end{bmatrix}$, $(r_y, r_x) = \mathbf{K_1} + \mathbf{K_2}$ (ref: Table 1).

The major contribution to spin relaxation in bulk silicon is due to optical phonon scattering between the valleys residing at different crystallographic axes which also includes primed subbands, known as f-process. However, their contribution can be safely neglected for a film thickness of less than 3nm, which is the case under scrutiny. This is because due to the rather high energies of the primed subbands in relation to the unprimed subbands, in such a case, the optical phonon transitions become rare [16,17].

As the spin relaxation matrix elements strongly depend on the wave vectors (c.f. Eq. 9 to 12), the only way to calculate the spin lifetime is to perform multi-dimensional integrals over the energy E and ϕ without using any approximations. A suitable discretization scheme in order to evaluate the integrals is now discussed [20]. The intersubband spin relaxation matrix elements are characterized by very narrow and sharp peaks, known as spin hot spots [11,13,14]. At this condition, we have $D\varepsilon_{xy} - \frac{\hbar^2 k_x k_y}{M} = 0$, and thus, the value of δ in Eq. 4 attains its minimum. The equivalent subband splitting is at its minimum at the spin hot spots, which signifies a maximum mixing between up- and down-spin eigenstates. At spin hot spot condition, this subband splitting is determined by the terms $\triangle_{SO}$ and Δ_Γ, resulting in a strong spin relaxation. Indeed, the spin hot spots need to be resolved by using a very fine mesh. Therefore, we have estimated that the energy step value $\triangle E$ should be upper-bounded by 0.5 meV. The lower limit of the integral over E is zero, and we also identify that it is sufficient to set the corresponding upper limit to be 0.7 eV. Since the Fermi energy E_F has a value of an order of 0.1 eV, the upper limiting value 0.7 eV (c.f. Eq. 7) will be sufficient to diminish the influence of the energy integral on the relaxation time. Thus, this particular simulation setup requires around 1400 points. The step value for ϕ, or $\triangle\phi$, needs to be set smaller than 0.5°, where the lower and upper limits of the inner integral over ϕ are 0° and 360°, respectively. Thus, this integral on before- and after-scattering directions at fixed energies requires almost 1000 points each. This means, the scattering matrix elements and the derivative of the dispersion energy over the wave vector must be calculated numerically for almost 1,400,000 times. Additionally, to compute the matrix elements, the eigenfunction problems for the 4×4 Hamiltonian matrix must be solved for the two wave vectors before and after scattering for a broad range of parameters. This necessitates the development of a highly parallelized computational framework.

The major computational difficulties to perform the multi-dimensional integrals and the need for the development of our two-level algorithm are now elaborated. A straightforward way for obtaining wave function and energies in a certain subband and valley for a known wave vector is described in [21]. Here, in contrast, we have to address the inverse problem: for a fixed energy E and angle of the wave vector ϕ we search the wave function. We employ a Nelder–Mead method to solve this problem using nlopt [22]. The procedure for obtaining wave functions starts at an initial value of the $\mathbf{K}$ vector: $|\mathbf{K}|$=1nm^{-1}. The optimization routine returns the derivative of the calculated energy for a given parameter values of the target energy. At each step, the length of the $\mathbf{K}$ vector is adjusted by the library. Now, because of the integral over E in Eq. 8, the search of the wave functions has to be performed for several times at a fixed wave vector direction (i.e., having constant ϕ). Moreover, the integration over ϕ is present in Eqs. 8 to 12. Considering the above-mentioned discretization scheme that we must have, the numerical spin relaxation time calculation becomes prohibitively expensive.

By using a standard adaptive integration technique, we found that a month of calculations on 20 cores, or 15000 core hours total, is required to evaluate a single data point of the spin relaxation time at a certain value of ε_{xy}. One can successfully resolve this difficulty by introducing a file-based cache technique with logarithmic size complexity. This means, we calculate and archive all static wave functions and energy data to a binary file (file-based cache) at the first level, and perform the spin lifetime calculations by loading those data in memory at the second level.

Indeed, a standard adaptive integration technique which includes irregular steps of the integration domain can be used to evaluate the integrals. However, in this technique, it is very hard to reuse any previously calculated data in the current calculations. In our approach, a regular but fine grid for integration is used instead; thus, the cache that consists of recently calculated points can be more efficiently used for successive calculations.

3 Simulation results

Our two-level parallelization algorithm is shown below. At the first level, all static wave functions and energy data are calculated and archived in a binary file as a file-based cache technique in parallel. This is known as serialization process. At the second level, the spin lifetime is calculated by deserializing the cache and calculating the spin relaxation rates. Since the values in the cache depend only on film thickness t and ε_{xy}, the dependence of spin lifetime on the Fermi energy and temperature (c.f. Eq. 6) can easily be computed without recalculating the cache. Therefore, this technique helps to save a significant number of core hours. Serialization and deserialization processes are performed by using the Boost Serialization library [23]. The performance is measured on the Vienna Scientific Cluster (VSC-2) [24]. VSC-2 consists of 1314 nodes having high-performance InfiniBand (IB) for network communications. Each node of the cluster has 2 processors (AMD Opteron 6132 HE, 2.2 GHz and 8 cores), and 32 GB main memory.

3.1 Algorithm of spin relaxation time calculation

Algorithm for spin lifetime calculations:

First level:

- (**1**) Divide the range of angle ϕ into sub-domains for each MPI process.
- (**1.1**) Divide the range of energy E into sub-domains for each OpenMP thread.
- (**1.1.1**) Calculate the derivatives at the interface $(\frac{d\psi}{dz})_{z=\pm\frac{t}{2}}$, and $\frac{|\mathbf{K}|}{|\frac{\partial E(\mathbf{K})}{\partial \mathbf{K}}|}$ in parallel (MPI, OpenMP).
- (**2**) Collect all the cached values at the master MPI process.
- (**3**) Archive the cache to a binary file.

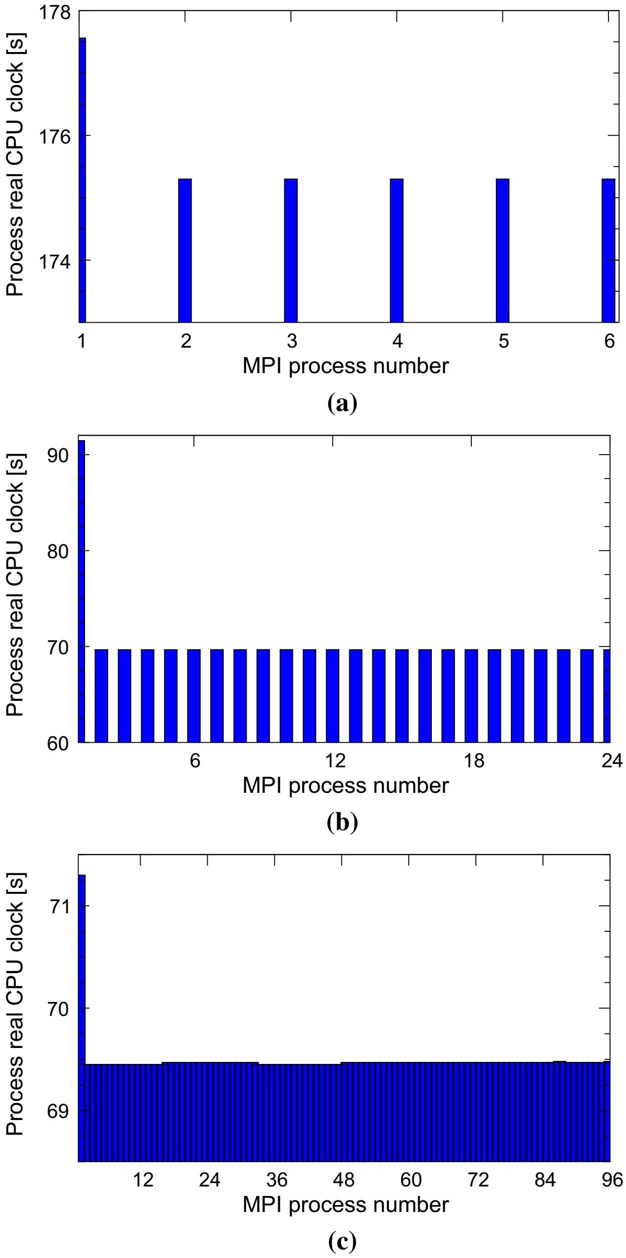
Second level:

- (**4**) Load archived cache by the master MPI process.
- (**5**) Divide the range of ϕ into sub-domains for each MPI process.
- (**5.1**) Divide the range of E into sub-domains for each OpenMP thread.
- (**5.1.1**) Calculate Eq. 6 for a given range of values in parallel (MPI, OpenMP).
- (**6**) Collect all calculated relaxation rates into the final relaxation rate.

3.2 Load balancing of the MPI jobs

In our measurements of computational time, we measure wall clock central processing unit (CPU) time spent by the current MPI process, rather than user CPU time (i.e., the time spent to execute the user code). The wall clock time is the elapsed time which additionally includes the time spent waiting for the process′s turn on the CPU. The Boost Chrono library is used to measure the runtime of the code [25].

For the first level, 96 cores have been taken to analyze load on each MPI process (i.e., the load distribution). We have tested several configurations with different numbers of MPI processes and OpenMP threads. Since a node on the VSC-2 has 2 processors each 8 cores, the maximum 16 threads have been tested. Figure 1a shows a configuration which has 6 MPI jobs, where each job has 16 threads (6x16 MPIxOpenMP configuration). Since the first MPI process is



(a)

(b)

(c)

Fig. 1 Load distribution for the cache calculation part on 96 cores for different system configuration is shown for: **a** 6 MPI processes and 16 OpenMP threads **b** 24 MPI processes and 4 OpenMP threads, and **c** 96 MPI processes. A good load balancing is demonstrated for all configurations

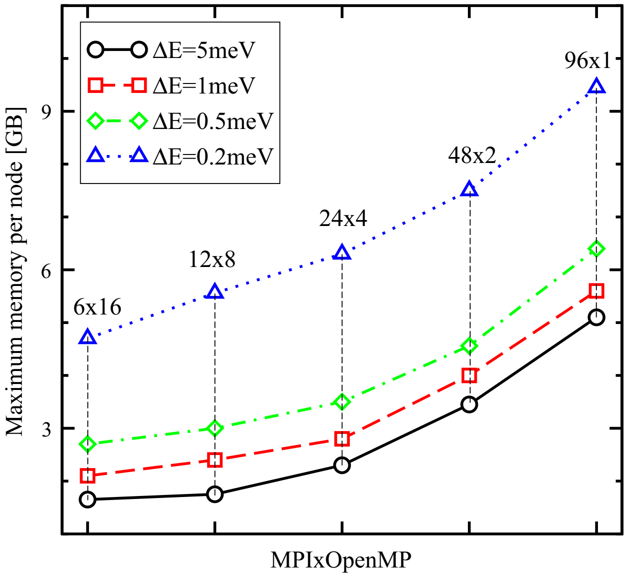


Fig. 2 Dependence of the maximum memory/node for different values of $\triangle E$ and different MPIxOpenMP configurations is shown

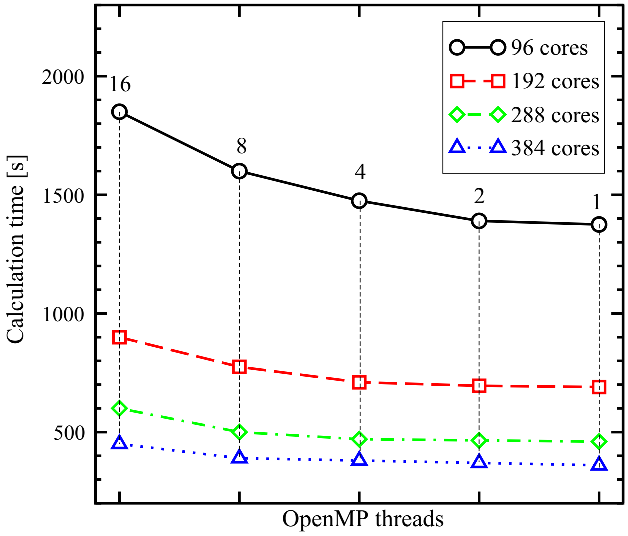


Fig. 3 Dependence of the total cache calculation time on different number of threads for fixed total core numbers is shown

responsible for the collection of all results and serializing the cache, it executes longer than other processes. Next, we reduce the number of threads, while the number of MPI processes is increased (Fig. 1b). The configuration for Fig. 1b is 24 MPI jobs with 4 threads on each (24x4 MPIxOpenMP configuration). Insignificant deviation is observed on 9th, 10th, and 12th processes, but the whole load figure looks the same as in Fig. 1a. Then, we test a full MPI realization as in Fig. 1c to find that the previous trends remain unchanged. The throughput of the IB is much higher than the message size and that is why the number of communication points does not influence the collection and serialization time.

Now, we check the memory consumption of the first level of the algorithm. Figure 2 shows the dependence of the maximum memory per node required for different values of $\triangle E$ and configurations of MPIxOpenMP. Total number of cores is 96 and kept fixed. We observe that the memory requirement per node increases when the number of threads is reduced for all considered values of $\triangle E$. In the worst case, all energy points in a fixed direction should be calculated by a single thread (full MPI realization, 96x1 MPIxOpenMP). If a single thread is used, the memory requirement becomes 5 GB by considering $\triangle E$=5meV, which is around three times larger as compared to maximum threaded case (i.e., 6x16 MPIxOpenMP configuration). More realistic requirements are shown in Fig. 2 when $\triangle E$=0.5 meV. In the full MPI configuration scheme, it requires 6.3 GB memory, whereas in 6x16 MPIxOpenMP configuration it demands 2.6 GB. It is further observed that even very accurate calculations (when $\triangle E$=0.2 meV) require less than 10 GB of memory. Thus, memory limitations are not an issue considering any modern supercomputer using this particular simulation setup. The speed of computation is now investigated.

Figure 3 shows the dependence of the total cache calculation time on different number of parallel threads, where the number of core used is kept as a parameter. We find that, for 96 parallel cores the total calculation time reduces from 1800 to 1356 s (around 30% reduction), while the number of threads reduces from 16 to 1. The same trend is followed when multiple number of cores are used (192, 288, and 384). The performance decrease of a hybrid approach can be attributed to the data locality issues arising in shared memory techniques.

Figure 4, as can be obtained from Fig. 3, demonstrates the dependence of the calculation time on the total number of cores for a fixed thread count. As the sub-domains of each MPI process are not correlated, the calculations in one domain do not influence on the other. An increment of the number of cores leads to the lossless reduction of the total calculation time. This explains a perfect scalability as demonstrated in the figure. This scalability, however, is limited by the number of points in the angle integral (c.f. Eq. 8). This is because the angle step $\Delta\phi$ has been chosen based on the number of cores (multiple of 96) to obtain an optimum load distribution condition at the first step of the algorithm, thereby utilize all the resources.

3.3 Spin relaxation time calculation

The spin relaxation calculations start from the deserialization of the cache process (i.e., second level of the algorithm). As the deserialized object is to be stored in the memory, the size of the cache strictly determines the number of parallel MPI jobs on a single node. Figure 5 shows the size of serialized

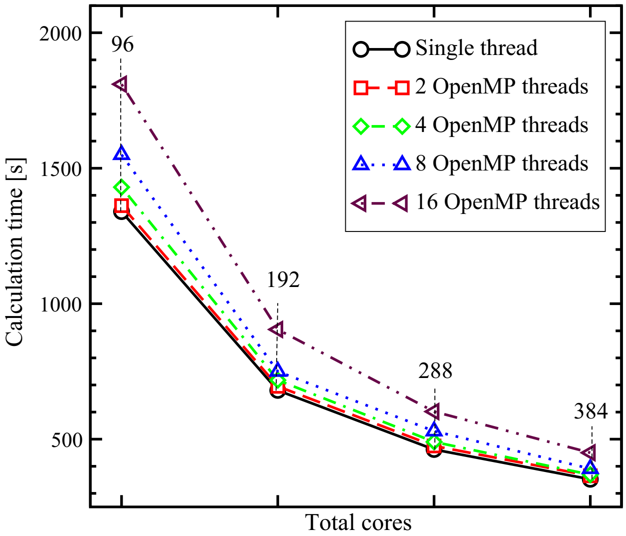


Fig. 4 Dependence of the calculation time for a fixed threads count on total number of cores is shown (ref. Fig. 3)

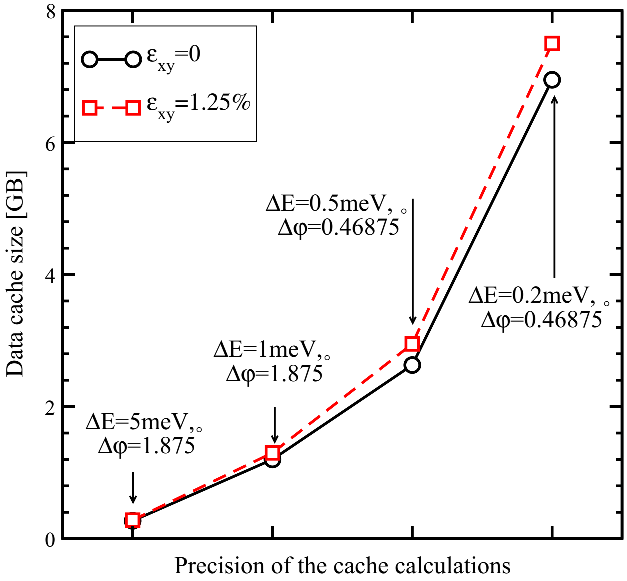


Fig. 5 Dependence of the size of cache calculated in the first level on the precision of the calculations fixed by energy and angle steps for different ε_{xy} values is shown

cache for different values of energy and angle steps $\triangle E$ and $\triangle\phi$. For $\triangle E$=5 meV and $\triangle\phi$=1.875°, the size of the cache is around 270 MB. This size allows 16 parallel MPI processes being executed on a computational node. However, as pointed out earlier, the step value $\triangle E$ should be at least 0.5meV and $\triangle\phi$ should be at least 0.5°. For such parameters, the size of the serialized cache grows up to 3 GB. Such size of the serialized cache imposes restrictions on the number of parallel executed MPI jobs on a single node. The even smaller energy step makes the cache as big as 7 GB or even 7.5 GB in the dependence on the input parameters. Theoretically, only three processes can work together on a single node leading to a significant loss of the computational resources. Hence, it becomes inevitable to use a hybrid MPI-OpenMP configuration, albeit its execution performance limitations.

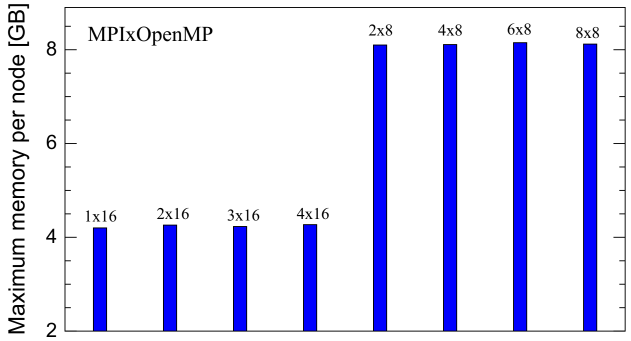


Fig. 6 Maximum required memory/node used by VSC-2 for different MPI and OpenMP configurations (second level) is shown. Each MPI process loads 4 GB cache

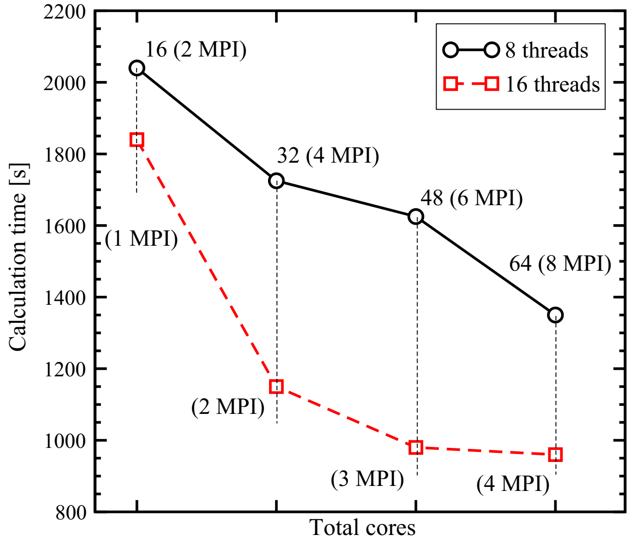


Fig. 7 Dependence of the total spin relaxation calculation time (second level) on total number of cores for 8 and 16 threads per MPI process is shown

Figure 6 shows maximum memory per node for 8 and 16 threaded MPI applications. Each MPI process reads 4 GB cache file; thus, the number of parallel executed MPI jobs is limited by 8. The memory requirements of the computations in the second level of the algorithm are mainly determined by the size of the serialized cache, and hence, the memory footprint of the algorithm itself can be neglected. Doubling of MPI jobs per node requires double memory space.

Figure 7 shows the dependence of the total calculation time on the number of cores as well as the number of threads, corresponding to the cases as in Fig. 6. This illustrates that increasing the total number of cores at a fixed number of threads decreases the demand on computing time, which is

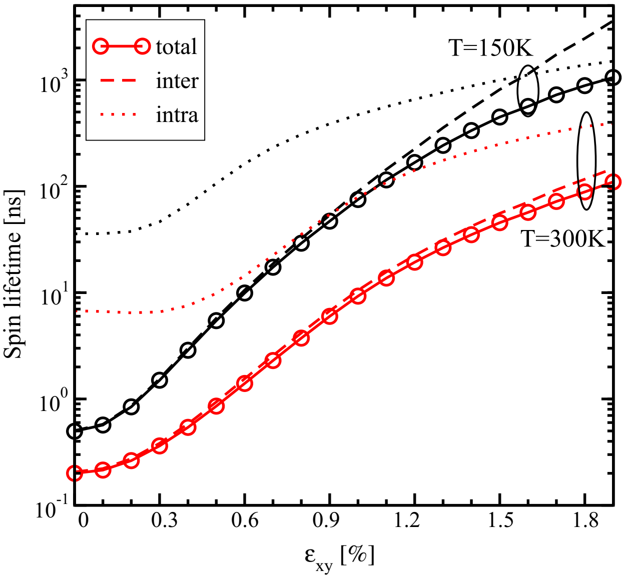


Fig. 8 Spin relaxation time with the respective inter- and intra-subband components at two distinct temperatures is shown. The electron density N_S=10^{12}cm^{-2}, t=2.7nm, Δ_Γ=5.5eV

further reduced when the number of threads is increased. This is in contrast to Figs. 3 and 4, as an increment in the thread numbers from 8 to 16 leads to decrement 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).

3.4 Spin lifetime

Finally, we calculate the spin lifetime τ_S as a function of shear strain ε_{xy}, c.f. Fig. 8. The temperature T is shown as a parameter. The valley splitting in relaxed silicon films characterized by the parameter Δ_Γ is also taken into account. We show the spin-flip caused by the intra- and inter-unprimed subband scattering. The major contribution to τ_S comes from the intersubband processes due to the presence of the spin hot spots [11,26,27]. Shear strain moves the spin hot spots to high energy outside of the states occupied by carriers, leading to a sharp increase of spin lifetime. This trend remains similar even at lower value of T, although the value of τ_S goes significantly higher as the phonon scattering rate is suppressed at lower temperatures. It is further noticed that at higher stress, the intra-subband component also turns out to be non-negligible.

4 Conclusion

We have described a two-level parallelization algorithm to calculate the silicon spin lifetime. The computational trade-off with respect to accuracy of our simulation set up, memory consumption, and calculation time is analyzed. The suggested algorithm precalculates wave functions and energies in the first level, and computes the spin relaxation rate by using the precalculated data in the second level. In each level, the calculations are performed in parallel. We have explained how the first level is best performed through a pure MPI scheme. We also have elaborated how the second level should be efficiently performed by a hybrid approach due to memory demands, although a full OpenMP realization could be more convenient. Finally, we conclude that shear strain routinely used to enhance mobility can also be used to hugely boost spin lifetime.

Acknowledgements The computational results presented have been partly achieved using the Vienna Scientific Cluster (VSC).

References

1. Li, J., Appelbaum, I.: Modeling spin transport in electrostatically-gated lateral-channel silicon devices: role of interfacial spin relaxation. Phys. Rev. B **84**, 165318 (2011)
2. Jansen, R.: Silicon spintronics. Nat. Mat. **11**, 400–408 (2012)
3. Dash, S.P., Sharma, S., Patel, R.S., de Jong, M.P., Jansen, R.: Electrical creation of spin polarization in silicon at room temperature. Nature **462**, 491–494 (2009)
4. MPI 1.1 Standard, http://www-unix.mcs.anl.gov/mpi/mpich. Accessed 2015
5. L. Dagum, R. Menon, OpenMP: An industry standard API for shared-memory programming. In: Proc. in IEEE Computational Science and Engineering, pp. 46–55 (1998)
6. Jost, G., Jin, H., an Mey, D. Hatay F. F.: Comparing the OpenMP, MPI, and hybrid programming paradigms on an SMP cluster. NAS Technical Report NAS-03-019, pp. 1–10 (2003)
7. Tang, S., Lee,B. -S., He B.: Speedup for multi-level parallel computing. In: Proc. in International Parallel and Distributed Processing Symposium Workshops & Ph.D. Forum, pp. 537–546 (2012)
8. Fabian, J., Matos-Abiaguea, A., Ertlera, Ch., Stano, P., Zutic, I.: Spintronics: fundamentals and applications. Rev. Mod. Phys. **76**, 323–410 (2004)
9. Zutic, I., Fabian, J., Das Sarma, S.: Semiconductor spintronics. Acta Phys. Slovaca **57**, 567–907 (2007)
10. Li, P., Dery, H.: Spin-orbit symmetries of conduction electrons in silicon. Phys. Rev. Lett. **107**, 107203 (2011)
11. Sverdlov, V., Selberherr, S.: Silicon spintronics: progress and challenges. Phys. Reports **585**, 1–40 (2015)
12. Song, Y., Dery, H.: Analysis of phonon-induced spin relaxation processes in silicon. Phys. Rev. B **86**, 085201 (2012)
13. Ghosh, J., Osintsev, D., Sverdlov, V., Selberherr, S.: Variation of spin lifetime with spin injection orientation in strained thin silicon films. ECS Trans. **66**(5), 233–240 (2015)
14. Ghosh, J., Osintsev, D., Sverdlov, V., Selberherr, S.: Enhancement of electron spin relaxation time in thin SOI films by spin injection orientation and uniaxial stress. J. Nano Res. **39**, 34–42 (2016)
15. Boykin, T.B., Klimeck, G., Eriksson, M.A., Friesen, M., Coppersmith, S.N., von Allmen, P., Oyafuso, F., Lee, S.: Valley splitting in strained silicon quantum wells. Appl. Phys. Lett. **84**, 115–117 (2004)

16. Osintsev, D.: Modeling Spintronic Effects in Silicon (Dissertation). Institute for Microelectronics, TU Wien (2014)
17. Ghosh, J.: Modeling Spin-Dependent Transport in Silicon (Dissertation). Institute for Microelectronics, TU Wien (2016)
18. Ghosh, J., Osintsev, D., Sverdlov, V., Selberherr S.: Dependence of spin lifetime on spin injection orientation in strained silicon films. In: Proc. in Joint International EUROSOI Workshop and International Conference on Ultimate Integration on Silicon, pp. 285–288 (2015)
19. Fischetti, M.V., Ren, Z., Solomon, P.M., Yang, M., Rim, K.: Six-band $\mathbf{k} \cdot \mathbf{p}$ calculation of the hole mobility in silicon inversion layers: dependence on surface orientation. strain, and silicon thickness. J. Appl. Phys. **94**, 1079–1095 (2003)
20. Ghosh J., Osintsev D., Sverdlov V., Weinbub J., Selberherr S. Evaluation of spin lifetime in thin-body FETs: a high performance computing approach. In: Lirkov I., Margenov S., Waśniewski J. (eds.) Large-Scale Scientific Computing. LSSC 2015. Lecture Notes in Computer Science, vol. 9374. Springer, Cham (2015)
21. Sverdlov, V.: Strain-induced Effects in Advanced MOSFETs. Springer, Wien - New York (2011)
22. Nonlinear Optimization, https://nlopt.readthedocs.io/en/latest/. Accessed 2015
23. Boost Serialization Library, www.boost.org/libs/serialization/. Accessed 2015
24. Vienna Scientific Cluster, http://www.vsc.ac.at/systems/vsc-2/. Accessed 2015
25. Boost Chrono Library, www.boost.org/libs/chrono/. Accessed 2015
26. Ghosh, J., Osintsev, D., Sverdlov, V., Selberherr, S.: Intersubband spin relaxation reduction and spin lifetime enhancement by strain in SOI structures. Microelectron. Eng. **147**, 89–91 (2015)
27. Ghosh, J., Sverdlov, V., Selberherr S.: Influence of valley splitting on spin relaxation time in a strained thin silicon film. In: Proc. in International Workshop on Computational Electronics, pp. 1–4 (2015)