Computer Physics Communications 189 (2015) 31–36

Contents lists available at ScienceDirect

Computer Physics Communications

journal homepage: www.elsevier.com/locate/cpc

Energy conserving, self-force free Monte Carlo simulations of semiconductor devices on unstructured meshes

M. Aldegunde *,[1], K. Kalna

Electronic Systems Design Centre, College of Engineering, Swansea University, Swansea SA2 8PP, United Kingdom

ARTICLE INFO

Article history:
Received 28 May 2014
Received in revised form
14 November 2014
Accepted 28 November 2014
Available online 8 December 2014

Keywords:
Monte Carlo simulation
Semiconductor devices
Finite element method
Self-forces
Particle–mesh coupling
FinFET

ABSTRACT

Unphysical self-forces resulting from the particle–mesh coupling occur when ensemble Monte Carlo simulations of semiconductor devices use an unstructured mesh to describe device geometry. We report on the development of a correction to the driving electric field on arbitrary meshes and show that self-forces can be virtually eliminated on a finite element mesh at a small additional computational cost. The developed methodology is included into a self-consistent 3D finite element Monte Carlo device simulator. We show the efficiency of the method simulating an isolated particle and obtaining kinetic energy conservation down to a magnitude of 10^{-10} meV. The methodology is later applied to a FinFET simulation to show what impact can be expected from the self-forces using traditional electric field interpolation strategies. We find that for a large enough ensemble of particles, the impact of self-forces on the final I_D–V_G is almost negligible.

1. Introduction

Monte Carlo (MC) methods have been widely used to simulate carrier transport in semiconductor devices [1–3]. As semiconductor devices are shrunk into deep nanoscale dimensions in order to boost their performance, the carrier transport becomes highly non-equilibrium requiring advanced physically based simulation models. The self-consistent ensemble MC is one of such methods, providing a detailed insight into transport and an accurate prediction of current characteristics of nanoscale transistors [4,5].

Novel non-planar multi-gate transistor architectures [6] such as FinFETs are replacing conventional bulk transistors during the further scaling into nanometre dimensions [7,8]. Their 3D geometry exhibits non-uniform shapes created by the fabrication process [4,9,10] which leads to new serious challenges for physically based device modelling. In order to precisely describe such fluctuating device geometries, the finite element method (FEM) delivers unrivalled advantages. Furthermore, optimised unstructured meshes can significantly reduce simulation time and memory requirements. This is essential as in self-consistent simulations the Poisson equation is solved on a 3D mesh every time step in order to update the electric field and can become the main bottleneck of the simulation process.

However, the use of unstructured meshes in a self-consistent ensemble MC simulation requires careful evaluation of the particle–mesh coupling to avoid unphysical self-force on the particles [11,1,12]. This force is felt by particles when they are alone in an infinite space with uniform permittivity. The origin of this force lays in the lack of spherical symmetry in the solution of the discrete Poisson equation for a single charge. The solution in a discretised space is based on a fixed mesh, so the values of the electrostatic potential are only defined on its nodes. However, the particles can have arbitrary positions (within the numerical precision of the computer). If a proper interpolation function is not used, the force exerted on the particle by itself can be different from zero and even become comparable to other sources of electrostatic forces in the device (other particles, ionised impurities, etc.).

The impact of self-forces in semiconductor device particle simulations has been extensively studied in the past [1,12] and various methods have been proposed to minimise them. However, most of these works have achieved a satisfactory result only for orthogonal meshes [12]. From a mathematical point of view, self-forces have been proven to be identically zero for uniform orthogonal meshes if the charge assignment and the force interpolation schemes are the same [1]. However, it must be pointed out that this is not a necessary condition as shown in [12]. In real devices, we have boundary conditions and material inhomogeneities, so the force on a particle which is alone in the device does not have to be zero. However,

* Corresponding author. Tel.: +44 2476 151797.
E-mail addresses: m.a.aldegunde-rodriguez@warwick.ac.uk (M. Aldegunde), k.kalna@swansea.ac.uk (K. Kalna).
[1] M. Aldegunde is now with the Warwick Centre for Predictive Modelling, School of Engineering, University of Warwick, Coventry CV4 7AL.

http://dx.doi.org/10.1016/j.cpc.2014.11.020
0010-4655/

although the lack of spatial accuracy to resolve the effect of this inhomogeneities will also lead to incorrect forces on the particle, this could be, in principle, avoided by the use of a finer mesh, whereas the impact of self-force cannot.

Even though MC device simulators using tetrahedral elements have been presented in the past [4,5], to our best knowledge, the problem of the self-forces has been seldom considered [12,3]. In this work, we present a methodology to evaluate and suppress the self-forces in a finite element (FE) MC device simulator based on tetrahedral elements [5]. We first study the impact of the self-force on a single particle. Later, we show the impact of the self-forces on the drive current of a 10.7 nm gate length SOI FinFET at a low drain bias because the self-force has a larger impact in these conditions.

2. Methodology

Standard methods to eliminate the self-force are based on the assumption of availability of the electric field in the nodes of the mesh [1,12]. In this case, it can be proven that under certain symmetry conditions it is possible to obtain zero self-force if the interpolation function for the electric field and the charge assignment operator are consistent [13,1]. These type of schemes are *momentum conserving*, but they do not conserve the total energy of the particles. The energy conservation can be associated to a consistent determination of the potential energy and the force on the particle, i.e., $\vec{F} = -\nabla U$. However, this condition is not satisfied for the momentum conserving scheme. This can lead to an artificial increase/decrease of the energy of the particles. Usually, this change in the total energy of the particle can be kept small by adjusting the time step. Other family of schemes, *energy conserving* [13–15], conserve the total energy of the particles, but the total momentum of the particles is not conserved. There are two sources for this non-conservation of the total momentum. On the one hand, it is the self-force, which will lead to an artificial exchange between kinetic and potential energies (and therefore an artificial change in their momentum). This can also be seen as particles having an unphysical contribution to their own potential energy. On the other hand, the lack of symmetry in the systems discretised on unstructured meshes leads to non-symmetric forces between particles ($\vec{F}_{ij} \neq \vec{F}_{ji}$ for two particles i, j) and therefore loss of the conservation of the total momentum. In the following, we will focus on the first component only, the self-force.

The potential energy of a particle p, U_p, in a semiconductor device can be decomposed, in general, as

$$U_p = U_{other\ particles} + U_{background\ charge} + U_{applied\ bias} + U_{image\ charges}, \quad (1)$$

which includes the contribution of particle p only through the image (polarisation) charges induced in the contacts or material boundaries. However, when we solve the Poisson equation, we include all the charges and then this potential is used in the calculation of the potential energy which is, in turn, used to calculate the driving force. In the absence of material discontinuities it can be written as:

$$U_p = q_p\psi(\vec{r}_p) = U_{self} + U_{other\ particles} + U_{background\ charge} + U_{applied\ bias} + U_{image\ charges}, \quad (2)$$

where q_p is the charge of particle p and $\psi(\vec{r}_p)$ is the electrostatic potential in the position of the particle p, $\vec{r}_p$. Eq. (2) includes the contribution of p's own Coulomb energy, U_{self}. Since we are solving numerically, there is no divergence in the Coulomb energy in the position of the particles. During the simulation, the driving force on a particle is then given by

$$\vec{F}(\vec{r}_p) = -\nabla U_p = -q_p\nabla\psi(\vec{r}_p) = \vec{F}_{self} + \vec{F}_{other\ particles} + \vec{F}_{background\ charge} + \vec{F}_{applied\ bias} + \vec{F}_{image\ charges}, \quad (3)$$

which, therefore, includes a contribution from the particle's own Coulomb energy, the self-force $\vec{F}_{self}$. A numerical method which preserves the spherical symmetry of the Coulomb potential energy around the particle will produce zero self-force since the gradient of this contribution will be zero. In this case, the contribution to the force, $\vec{F}_{self}$, will disappear and the method will be momentum conserving. However, the contribution to the potential energy, U_{self}, is still present and therefore the potential energy of the particle cannot be properly calculated. On the other hand, an energy conserving method, which verifies $\vec{F} = -\nabla U$ exactly, will conserve the total energy of the particle but both the potential energy and the force in the position of p will, in general, have a contribution from p itself.

The only way to properly account for the potential energy and force on a particle p would be removing the contribution of p from the fields used in the calculations. This can be accomplished in two ways. The first way would be the solution of one different Poisson equation for each particle with modified boundary conditions to account for the image charges [16,17]. However, this is computationally very expensive in self-consistent simulations as it would multiply the cost of the simulation by the number of particles in the domain, typically several thousands in semiconductor device simulations. The alternative is to find a correction to the potential energy and force which removes the contribution from the particle for which we are calculating the energy/force. As most semiconductor device Monte Carlo simulators use the finite difference method and momentum conserving schemes, there have not been attempts at using correction forces with unstructured meshes to the best of our knowledge. Since the environment of every node will be different depending on the mesh, it is not possible to find a general correction which can be applied in every case and a specific corrections for each mesh and node is required.

For a given mesh and for every node, we calculate the *reference electrostatic potential* for a unit charge assigned to the node p, $\psi^{p,R}$:

$$\nabla^2\psi^{p,R} = \delta(\vec{r} - \vec{r}_p), \quad (4)$$

$$\psi^{p,R}|_{\partial\Omega_D} = \frac{k_C}{|\vec{r} - \vec{r}_p|}, \quad (5)$$

where $\partial\Omega_D$ is the external boundary of the domain, with Dirichlet boundary conditions applied. This reference potential mimics the potential that the particle would create in an infinite domain without any other charge and with the boundary condition $\lim_{|\vec{r}-\vec{r}_p|\to\infty}\psi^{p,R} = 0$. The potential energy in this case should be zero and, therefore, any energy in $\vec{r}_p$ will be the self-potential energy of this particle, $U_{p,self} = \psi^{p,R}(\vec{r}_p)$. Similarly, any force in $\vec{r}_p$ will be the self-force, $\vec{F}_{self} = -\nabla\psi^{p,R}(\vec{r}_p)$.

To calculate the reference potential, we use the Ritz–Galerkin approximation and apply the FEM based on tetrahedral elements with piecewise linear base functions θ_i [18]. We then obtain the following weak formulation of Eq. (4) [19]:

$$\sum_{j=1}^{K}\psi_j^{p,R}\int_\Omega \nabla\theta_j\cdot\nabla\theta_i d\Omega = -\sum_{j=1}^{K}\int_\Omega \delta(\vec{r} - \vec{r}_p)\theta_i d\Omega = -\theta_i(\vec{r}_p) = -\delta_{ip}, \quad \forall i = 1,\ldots,K, \quad (6)$$

where K is the number of nodes in the mesh and we have used the fact that p is in the position of a node.

During the simulation, we solve the Poisson equation to obtain the electrostatic potential at every time step:

$$\nabla(\varepsilon(\vec{r})\nabla\psi(\vec{r})) = \sum_p q_p\delta(\vec{r} - \vec{r}_p) + \rho(\vec{r}), \quad (7)$$

$$\psi|_{\partial\Omega_{D_i}} = \psi_D + V_{ext,i}, \quad (8)$$

$$\nabla_n\psi|_{\partial\Omega_N} = 0, \quad (9)$$