Engineering Analysis with Boundary Elements 50 (2015) 69–75

Contents lists available at ScienceDirect

Engineering Analysis with Boundary Elements

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

Simulation of semiconductor devices with a local numerical approach

G. Kosec [a,b,*], R. Trobec [a]

[a] Parallel and Distributed Systems Laboratory, Jožef Stefan Institute, Jamova 39, SI-1000 Ljubljana, Slovenia
[b] Laboratoty for Multiphase Processes, University of Nova Gorica, Vipavska 13, SI-5000 Nova Gorica, Slovenia

ARTICLE INFO

Article history:
Received 31 December 2013
Received in revised form
21 July 2014
Accepted 31 July 2014

Keywords:
Semiconductor
PN junction
Drift Diffusion Model
Local numerical method
Meshless method
Parallel computation

ABSTRACT

A numerical solution of the Drift-Diffusion Model for simulation of semiconductor devices based on the local meshless numerical method is presented. Numerical difficulties inherited from convection-dominated processes and high gradients near junctions typically results in oscillations within the solution. The difficulties can be alleviated by artificial dissipation schemes or by other stabilization approaches that often require a complex computation to improve the solution convergence. We applied a simple numerical approach with a local coupling and without special treatments of nonlinearities. The proposed approach is straightforward to implement and is suitable for parallel execution. We demonstrate the efficiency of the proposed methodology on a simulation of PN junction. The results are compared against previously published data with a good agreement achieved. The applicability of the proposed methodology is confirmed with the simulation of extended tests with more complicated geometries and more intense dynamics. The computational efficiency is demonstrated through the measurement of execution time and speedup on shared memory computer architecture.

1. Introduction

Numerical simulations of semiconductor devices can significantly reduce the experimental work by predicting the behavior of realistic devices, which contributes in faster developing cycle and better performances of new devices. Simulation results can contribute to the study of material properties and better understanding of experimental measurements, which are essential in the development of compact models for integrated circuits. There are several branches of related research ranging from simulation of simple PN junctions [1], Schotky diodes [2], and transistors [3], to more specialized devices like solar cells [4,5]. The semiconductors can be simulated with different models. One possible approach is a statistical Monte Carlo Method [6] that deals with the dynamics of particles in an electric field. Basically, the Monte Carlo Method is a stochastic solution of the Boltzmann transport equation. Another class of models is based on a continuum description. The most classical approach is the Drift Diffusion Model (DDM) [7,8], where the continuity equations for carriers and the Poisson equation for potential are coupled. More sophisticated approach is the Hydrodynamic Model that solves the DDM with consideration of energy balance [9]. The Hydrodynamic Model enables capturing of more details about the internal processes of semiconductors than the sole DDM, such as carrier heating and/or velocity overshoot, however, at the cost of solving more complex systems since the number of variables is higher.

In this work, we propose a numerical approach for solving continuum semiconductor models. We solve the DDM, although the numerical methodology is identical for the Hydrodynamic model. The tackled DDM comprises three nonlinear coupled partial differential equations (PDE) that state the electrostatic potential, and electron and hole current continuities. The DDM belongs to a class of convection–diffusion problems. Besides treatment of the convection dominated regimes, the simulation of semiconductors often requires a consideration of high gradients near junctions. Numerical difficulties in such situations can be eased by stabilization schemes, e.g. upwind [10], adaptive upwind in the meshless context [11], or Residual Free Bubbles [1,12], which usually require a complex computation to improve the solution convergence. Regardless the complex formulation, computation on a coarse spatial discretization with fast convergence has been an ultimate advantage in the past. The availability of modern parallel computing platforms changes that. The computational simplicity is becoming more important factor even on the account of larger numbers of iterations and discretization points [13], since the vast number of computing units can be used. From the implementation and parallel execution point of view, the simplest and most effective numerical method is the explicit Finite Difference Method (FDM) since it is completely local and simple to compute. However, the FDM is limited to consideration of simple geometrical domains and is restricted regarding its possibilities for upgrades. The Finite Element Method (FEM) alleviates this drawback but is not ideal for massively parallel computers. Its weak formulation requires numerical integration and well defined

* Corresponding author.

http://dx.doi.org/10.1016/j.enganabound.2014.07.013
0955-7997/

neighborhood relations between discretization points [14]. However, the parallel FEM solution of DDM based problem has been recently presented in Ref. [15], where authors demonstrated the efficiency of multi-grid approach on modern computer architectures.

A promising alternative is a class of meshless methods (MM) that are based on scattered discretization points. MMs originate in the seventies with Smoothed Particles Hydrodynamics (SPH) [16] and develop further with globally formulated MMs, e.g. the Diffuse Element Method (DEM) [17], the Meshless Petrov–Galerkin method (MPG) [18], the Element Free Galerkin method (EFG) [19], and many others. The intense development in the field of the meshless methods continues, which is reflected also in several relevant recent publications [20–25].

In this paper, we focus on one of the simplest class of MMs – the Local Point Interpolation Meshless method (LPIM) [26]. The LPIM is based on the approximation of the trial function over the local support domain, which is a subset of scattered neighboring discretization points. The LPIM can be easily upgraded or altered to treat anomalies such as sharp discontinuities or complex domain geometries. However, a special treatment is needed for cases with degraded distribution of local discretization points that results in ill-conditioned interpolation [27]. A popular variant of LPIM is the Local Radial Basis Function Collocation Method (LRBFCM) [28] that performs very well as long as the nodal distribution is not deformed too severely. In this paper, we employ a local refinement of nodal distribution that introduces asymmetric support domain, where the collocation approach might suffer from ill-conditioning. To avoid such problems, we use more robust variant – the Diffuse Approximate Method (DAM) [20,29].

The rest of the paper is organized as follows. In Section 2, the physical model of PN junction is described in details with corresponding boundary and initial conditions. In Section 3, the solution procedure and its implementation is described. Section 4 is devoted to the presentation and analysis of results. The paper concludes with the summary of results and directions for further work.

2. PN junction model

The PN junction is modeled by the DDM, which comprises potential Poisson equation and carrier continuity equations

$$\nabla^2\Psi(\mathbf{r}) = -\frac{q}{\varepsilon}(p(\mathbf{r}) - n(\mathbf{r}) + D(\mathbf{r})), \tag{1}$$

$$\nabla\mathbf{j}_p = -q(R(\mathbf{r}) - G(\mathbf{r})), \tag{2}$$

$$\nabla\mathbf{j}_n = q(R(\mathbf{r}) - G(\mathbf{r})), \tag{3}$$

where Ψ stands for the potential, $\mathbf{r}(x,y)$ for the position vector, q and ε for the elementary electronic charge and permittivity, n and p are the electron and hole densities, $D(\mathbf{r})$ stands for the doping defined as a difference in concentration between donors n_d and acceptors n_a, R and G stand for the recombination and generation rate, and $\mathbf{j}_n$ and $\mathbf{j}_p$ describe the electron and hole current densities. To avoid an introduction of additional boundary conditions on the junction, we use a single domain formulation, where P and N regions are defined through the spatial dependent doping function

$$D(\mathbf{r}) = \begin{cases} x < x_j; & -n_a \\ x \geq x_j; & n_d \end{cases}, \tag{4}$$

with $\mathbf{r}_j = (x_j, y_j)$ representing the position of junction. The current densities are further modeled as

$$\mathbf{j}_n(\mathbf{r}) = -q(\mu_n n(\mathbf{r})\nabla\Psi(\mathbf{r}) - D_n\nabla n(\mathbf{r})), \tag{5}$$

$$\mathbf{j}_p(\mathbf{r}) = -q(\mu_p p(\mathbf{r})\nabla\Psi(\mathbf{r}) + D_p\nabla p(\mathbf{r})), \tag{6}$$

with μ_n and μ_p standing for the electron and hole mobilities, while D_n and D_p define the electron and hole diffusivities. We exclude the recombination and generation of holes and electrons and therefore Eq. (2) simplifies to

$$\nabla\mathbf{j}_n = -\nabla\mathbf{j}_p = 0. \tag{7}$$

Considering Eq. (1) and substituting Eqs. (5) and (6) into Eq. (7), we get the final set of three coupled convection–diffusion type PDEs

$$\nabla^2\Psi(\mathbf{r}) = -\frac{q}{\varepsilon}(p(\mathbf{r}) - n(\mathbf{r}) + D(\mathbf{r})), \tag{8}$$

$$D_n\nabla^2 n(\mathbf{r}) - \mu_n(\nabla n(\mathbf{r})\nabla\Psi(\mathbf{r}) + n(\mathbf{r})\nabla^2\Psi(\mathbf{r})) = 0, \tag{9}$$

$$D_p\nabla^2 p(\mathbf{r}) + \mu_p(\nabla p(\mathbf{r})\nabla\Psi(\mathbf{r}) + p(\mathbf{r})\nabla^2\Psi(\mathbf{r})) = 0. \tag{10}$$

The presented governing system of equations is considered in a rectangular domain with height Ω_H and width Ω_W, where the horizontal walls are of the Neumann type, i.e. no electron or hole flux is allowed and the vertical walls are set to the Dirichlet boundary conditions. The boundary conditions are defined as follows:

$$n(\Omega_W, y) = n_d, \quad p(\Omega_W, y) = \frac{n_i^2}{n_d}, \quad \Psi(\Omega_W, y) = 0\ V, \tag{11}$$

$$n(0,y) = \frac{n_i^2}{n_a}, \quad p(0,y) = n_a, \quad \Psi(0,y) = \frac{kT}{q}\log\left(\frac{n_a n_d}{n_i^2}\right), \tag{12}$$

$$\left.\frac{\partial n}{\partial y}\right|_{x,0} = \left.\frac{\partial n}{\partial y}\right|_{x,\Omega_H} = \left.\frac{\partial p}{\partial y}\right|_{x,0} = \left.\frac{\partial p}{\partial y}\right|_{x,\Omega_H} = \left.\frac{\partial \Psi}{\partial y}\right|_{x,0} = \left.\frac{\partial \Psi}{\partial y}\right|_{x,\Omega_H} = 0, \tag{13}$$

where k, T and n_i stand for the Boltzmann constant, operating temperature and intrinsic carrier concentration, respectively. A scheme of the problem domain is depicted in Fig. 1.

The presented formulation describes zero biased PN junction, i.e. the PN junction without external voltage applied. After joining the P and N doped semiconductors, the electrons and holes start to diffuse. The diffusing carriers leave charged ions behind, which induce the electric field that counteracts the diffusion. In the equilibrium, the junction and its local surrounding is depleted of all carriers. The width of depletion region d can be estimated, since, in equilibrium, the charges on both sides must be the same,

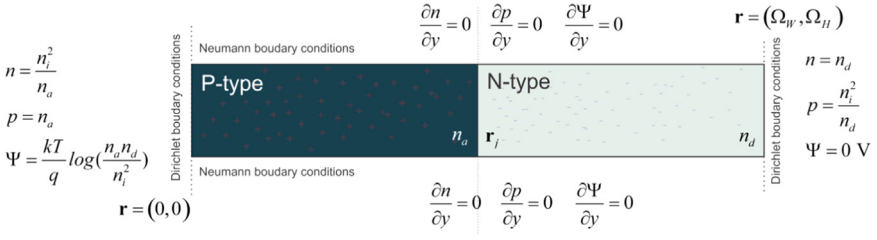

Fig. 1. Problem domain of the PN junction.

with the following relation [30]:

$$d = \sqrt{\frac{2\varepsilon}{q}\frac{n_a + n_d}{n_a n_d}\Psi(0, y)}. \tag{14}$$

3. Numerical solution procedure and implementation

The model of PN junction, described in the previous section, is governed by a non-linear and tightly coupled system of PDEs. Our goal is to design a flexible solution procedure that can be effectively implemented on modern computer architectures in order to fully exploit its computational potential. The DDM is a stationary second order problem. The most common approach would be to transform each PDE, by an appropriate spatial discretization technique, to a system of non-liner algebraic equations, solve the gathered systems by appropriate solution method, exchange the data between solutions, and iterate the process until the convergence is met. Here we address the problem by transforming the stationary PDEs to transient ones. Similar approaches are widely used in the computational fluid dynamics community to handle the pressure–velocity coupling, e.g. the artificial compressibility method [31] or the false transient method [32]. The constructed transient equations are numerically treated with the two-level explicit stepping. We get the following system:

$$\Psi_2(\mathbf{r}) = \Psi_1(\mathbf{r}) + \Delta\tau\left[\nabla^2\Psi_1(\mathbf{r}) + \frac{q}{\varepsilon}(p_1(\mathbf{r}) - n_1(\mathbf{r}) + D_1(\mathbf{r}))\right], \tag{15}$$

$$n_2(\mathbf{r}) = n_1(\mathbf{r}) - \xi\Delta\tau[\mu_n(\nabla n_1(\mathbf{r})\nabla\Psi_1(\mathbf{r}) + n_1(\mathbf{r})\nabla^2\Psi_1(\mathbf{r})) - D_n\nabla^2 n_1(\mathbf{r})], \tag{16}$$

$$p_2(\mathbf{r}) = p_1(\mathbf{r}) + \xi\Delta\tau[D_p\nabla^2 p_1(\mathbf{r}) + \mu_p(\nabla p_1(\mathbf{r})\nabla\Psi_1(\mathbf{r}) + p(\mathbf{r})\nabla^2\Psi_1(\mathbf{r}))]. \tag{17}$$

The iteration step is denoted by $\Delta\tau$. Subscripts 1 and 2 denote current and next iteration step values. From the numerical point of view the above system is expressively unbalanced. The iteration process in Eq. (15) has different convergence dynamics in comparison to Eqs. (16) and (17). To alleviate the unbalance, we include an additional relaxation parameter ξ.

The spatial discretization of the PDEs is performed by a local meshless numerical approach. The meshless spatial discretization is based on an approximation of a considered field over the local support domain, i.e. only small local sub-set of nodes is used for the approximation

$$\theta(\mathbf{r}) = \sum_{n=1}^{N_B} \alpha_n B_n(\mathbf{r}), \tag{18}$$

where N_B, α_n and B_n stand for the number of Multiquadrics basis functions, approximation coefficients and basis functions, respectively. On the approximated function, an arbitrary spatial differential operation L can be applied

$$L\theta(\mathbf{r}) = \sum_{n=1}^{N_B} \alpha_n L B_n(\mathbf{r}). \tag{19}$$

The applicability of L on the approximated functions θ is limited only by the selection of the basis functions. The computation of the coefficients and the evaluation of the differential operators can be combined in a single operation

$$L\theta(\mathbf{r}) = \sum_{n=1}^{N_S} \chi_m^L(\mathbf{r})\theta(\mathbf{r}_n), \tag{20}$$

where the differential operator χ_m^L vector is introduced as

$$\chi_m^L(\mathbf{r}) = \sum_{n=1}^{N_B} \mathbf{B}_{nm}^{-1} L(\Psi_n(\mathbf{r})). \tag{21}$$

N_S stands for the number of support domain nodes and $\mathbf{B}$ for approximation matrix (18). The approach is convenient for computer implementation since most of the complex operations can be performed in the pre-processing phase. More details about the presented method can be found in Ref. [33]. The solution procedure is schematically presented in Fig. 2. The complete locality of the introduced numerical scheme has several beneficial effects. Besides simplicity and straightforward implementation, there are substantially higher opportunities to fully exploit modern computer architectures through different parallel computing strategies [34,35], since the local approach minimizes the communication between computational nodes, which is often a limiting factor to the speedup of a parallel execution. In Fig. 2, the tasks that can be directly parallelized are marked as “parallel task”. All those parts of the code, which present bulk of the execution time, are parallelized with the shared memory parallelization. The can can be also effectively computed by multiple Graphics Processing Units (GPU) or other massively parallel architectures, since the computations are basically convolutions of local differential vectors and corresponding fields [33].

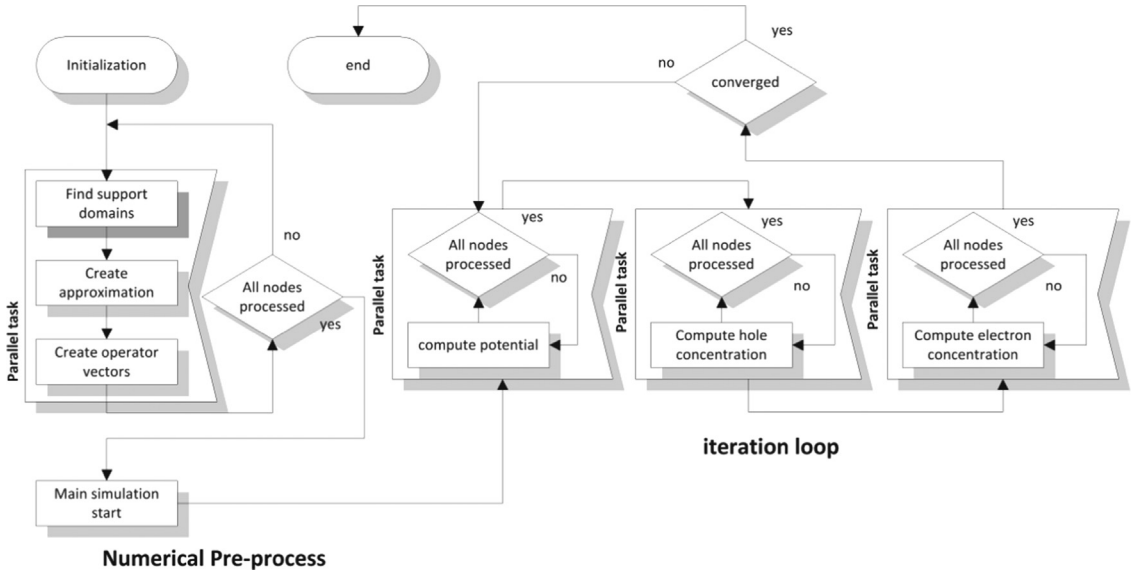

Fig. 2. Block diagram of the solution procedure.

4. Results

In this section results of numerical integration are presented. The quantitative values of related parameters are stated in Table 1. All other case specific parameters are stated in the corresponding sections.

4.1. Benchmark and convergence tests

The first numerical example, which is the same as in Ref. [1], serves as a benchmark test. The meshless solution of considered DDM is presented as a surface plot of electric potential in the left plot of Fig. 3, while the dynamics of the iterative process is shown in right plot. The iteration dynamics is presented by normalized mean values of governing fields $(\tilde{\Psi}, \tilde{p}, \tilde{n})$. The presented results are computed with N_d=40,000 regularly distributed nodes, iteration step $\Delta\tau = 10^{-14}$ and relaxation parameter ξ=0.05. We can see from Fig. 3 that the solution converges to the steady state at approximately 2×10^5 iterations ($\Delta\tau$ iterations$=2\times10^{-9}$). The selection of iteration step is conditioned by stability criteria of explicit stepping.

Table 1
Quantitative parameter values of the test problem.

ε	Permittivity	$1.036\times10^{-12}\ F/\mathrm{cm}$
q	Elementary electronic charge	1.602×10^{-19} C
n_i	Intrinsic carrier concentration	$1.350\times10^{10}\ \mathrm{cm}^{-3}$
n_a	Acceptor concentration	$10^{17}\ \mathrm{cm}^{-3}$
n_d	Donor concentration	$10^{17}\ \mathrm{cm}^{-3}$
μ_n	Electron mobility	$1400\ \mathrm{cm}^2/\mathrm{Vs}$
μ_p	Hole mobility	$450\ \mathrm{cm}^2/\mathrm{Vs}$
T	Operating temperature	300 K
D_n	Electron diffusivity	$36.192\ \mathrm{cm}^2/s$
D_p	Hole diffusivity	$11.633\ \mathrm{cm}^2/s$
Ω_W	Domain width	10^{-4} cm
Ω_H	Domain height	10^{-4} cm
k	Boltzmann constant	$1.381\times10^{-23}\ J/K$

In Fig. 4(left), the potential Ψ at the horizontal mid-line cross-section of the PN junction for different number of discretization nodes N_d is shown together with the FEM solution from Ref. [1]. It can be seen that the results do not agree with coarser nodal distributions. However, with increased number of nodes, the meshless solution converges towards the FEM solution. In the right part of Fig. 4, the convergence behavior is presented, which is characterized by the development of the maximum and the minimum of the potential with respect to the number of nodes.

4.2. Increased donor concentrations

In the next analysis, the donor concentration is varied from $n_d=10^{17}\ \mathrm{cm}^{-3}$ to $n_d=10^{18}\ \mathrm{cm}^{-3}$, while the acceptor concentration is kept on $n_a=10^{17}\ \mathrm{cm}^{-3}$. The stationary results of the charge carrier densities are shown in the left part of Fig. 5, where the horizontal mid-line cross-sections of electron and holes densities are presented. Subscripts denote a setup, where 1 stands for

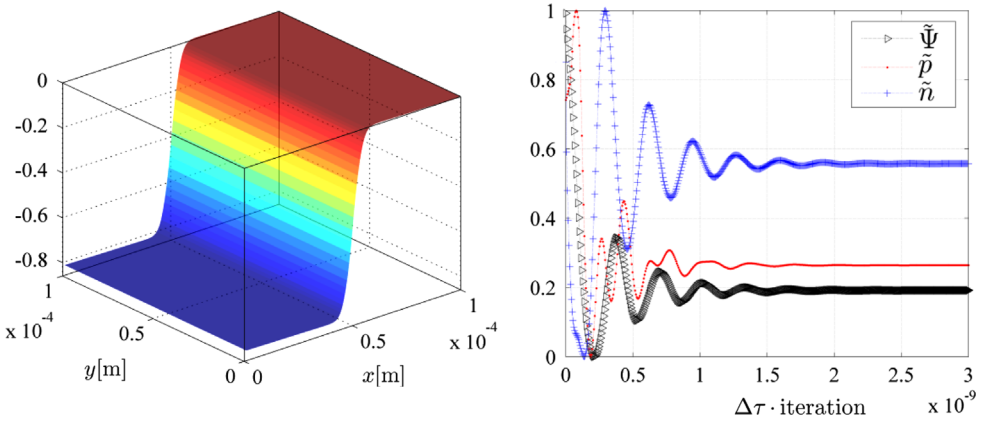

Fig. 3. Converged potential (left) and normalized development of averaged fields (right), computed on $N_d=4\times10^4$ discretization nodes.

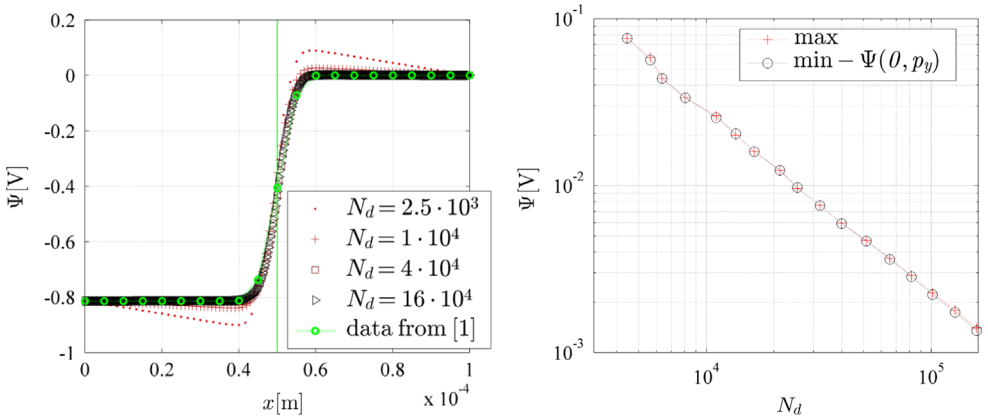

Fig. 4. Potential at horizontal mid-line cross-section of the PN junction for different number of discretization nodes together with the FEM solution from Ref. [1] (left), and potential maximum and minimum with respect to the number of nodes (right). Green line represents the position of the junction. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

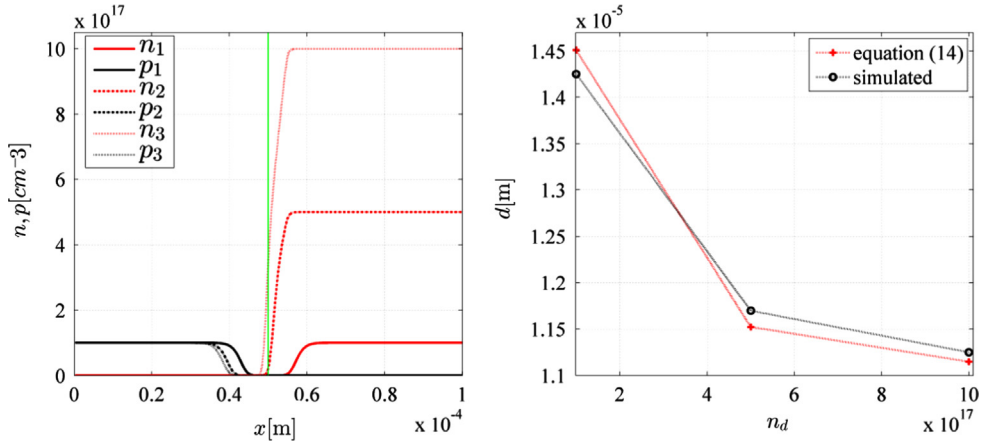

Fig. 5. Horizontal mid-line cross-section of carrier densities (left) and width of depletion region with respect to the doping (right). Green line represents the position of the junction. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

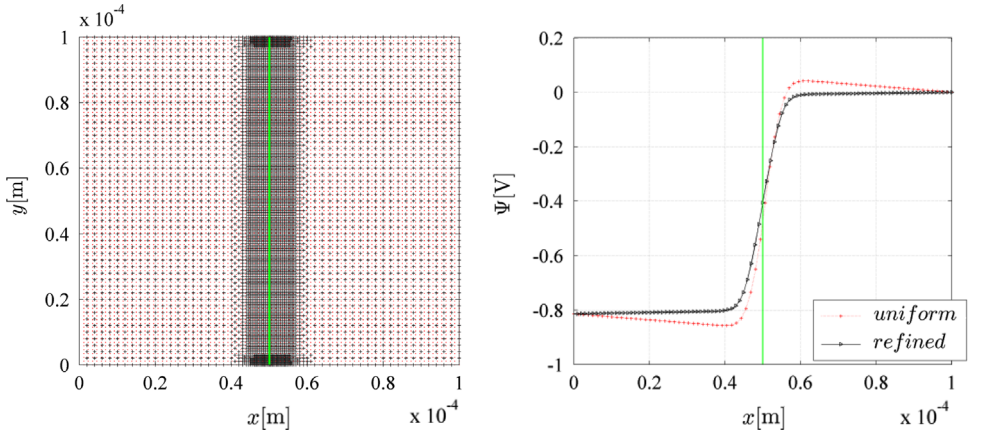

Fig. 6. Comparison of refined (black points) and regular (red points) nodal distributions (left) and corresponding results (right). The number of nodes is the same for both distributions. Green line represents the position of the junction. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

$n_d = 10^{17}$ cm^{-3}, 2 for $n_d = 5 \times 10^{17}$ cm^{-3} and 3 for $n_d = 10^{18}$ cm^{-3}. The depletion zone and its dependence on different semiconductor doping can be clearly seen. The width of depletion region d of the simulated results is shown in the right part of Fig. 5. It is computed as a distance between the maxima in the first derivatives of charge carrier densities.

4.3. Mesh refinement

From Fig. 4 it follows that using a low number of nodes results in a low quality of the solution, especially near the junction. In the presented case, the most complex part of the solution is the junction and its surrounding. Since the mesh refinement is one of the basic features of the presented numerical methodology, we exploit it by refining the nodal distribution near the junction on the account of sparser nodes on the remaining part of the domain (see the left part of Fig. 6). We use the same refinement strategy as in Ref. [36]. The results for potential at the horizontal mid-line cross-section of the PN junction are demonstrated in Fig. 6.

From the right part of Fig. 6 it is evident that with the refined nodal distribution fewer nodes are required to achieve accurate result. Naturally, the approach has its limitations. We cannot ultimately refine the discretization of junction area and leave other parts of the domain covered with too sparse nodal distribution. The width of refinement region and the minimal base nodal density are case dependant. More details about the refinement procedure can be found in Ref. [36].

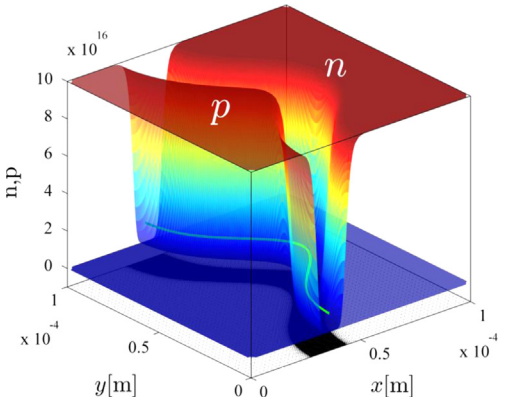

Fig. 7. Distribution of holes and electrons densities for a non-uniform PN junction on refined nodal distribution with parameters from Table 1. Green line represents the position of the junction. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

4.4. Non-uniform junction

Finally, we apply the proposed methodology on geometry closer to the real shapes of the PN junctions in the semiconductor

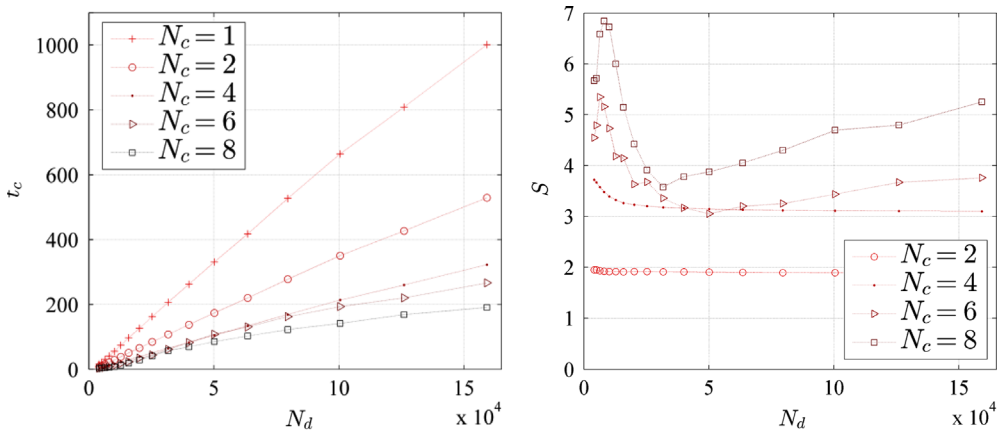

Fig. 8. The computation time (left plot) and speedup (right plot) as a function of the number of discretization nodes.

technology. We imitate the junction shape with a Gauss function

$$x_j(y) = 0.3 \times 10^{-3} e^{-(y-0.5\times10^{-4}\ \mathrm{m}/2.5\times10^{-5}\ \mathrm{m})^2}\mathrm{m} + 0.35 \times 10^{-4}\ \mathrm{m}. \quad (22)$$

The results of numerical integration on non-uniform nodal density are presented in terms of carrier's densities on Fig. 7.

4.5. Parallel speedup

The presented numerical methodology needs more numerical iterations to reach the convergence with required accuracy as, for example, the method proposed in Ref. [1]. However, it is formulated by a local explicit approach and consequently it can be effectively parallelized. We assess its parallel computation performance by measurements of the execution time on a dual CPU Intel (R) Xeon(R) E5520 at 2.27 GHz with four computing cores per CPU. The code is written in C++ programming language and compiled with Intel 13 compiler. All parallel tasks are implemented through OpenMP API. The speedup of the parallel program is evaluated as

$$S = \frac{t_C^1}{t_C^{N_C}}, \quad (23)$$

where t_C^1 and $t_C^{N_C}$ stand for the computation time on a single core and N_C cores, respectively. The results for the computation time of 10^4 iterations and for the speedup are shown in Fig. 8. The computation time (left plot) is linearly dependent on the number of discretization nodes. Such a behavior is expected since the spatial loops are homogeneous with equal work per discretization node.

The speedup (right plot of Fig. 8) reveals also other interesting phenomena. First, maximal speedups are obtained with smaller number of discretization nodes since the complete application can be been accessed from the fast cache memories. Next, speedups are increasing with the number of discretization nodes, because the ratio between communication and computation time is decreasing and therefore communication overheads are diminishing. Finally, the high variability of the speedup curves indicates that the memory architecture plays an important role. The highest speedups in the regimes with small number of discretization nodes are consequence of accumulating private L2 caches, i.e. increased number of cores increases also the available cache, which results in unexpectedly high speedups at smaller systems. The similar effect has been already reported in Ref. [37].

5. Conclusions

We demonstrated that a local meshless based solution procedure provides adequate results in the solution of the DDM for PN junction simulation. The main incitement of this work is a construction of the solution procedure suitable for execution on modern parallel computer architectures with high number of computing units. The basic idea behind the proposed approach is to maintain locality and simplicity.

The main conclusions of our work are:

- The simulation results of the presented solution are in a good agreement with a recently published solution [1].
- The proposed numerical methodology is simple to implement and upgrade to more complex physical models and/or domains.
- The computational complexity per iteration step is reduced to the simple local convolution of spatially dependent precomputed differential vectors and corresponding field values, which enable an efficient parallelization.
- The proposed method requires more discretization nodes and iteration steps to achieve adequate convergence. However, the number of required number of nodes can be minimized with a simple refinement strategy.
- Although no special treatment near the PN junction or on boundaries, either related to spatial discretization or numerical stabilization, is used, the solution procedure behaves stable with different initial setups, as well as junction geometries.

The quantitative results obtained with smaller numbers of discretization nodes could be further improved with various numerical enhancements, e.g. smoothing of the step gradient at the PN junction, upwind techniques, augmentation of basis functions, etc. Generally, any kind of stabilization introduces additional numerical errors and computational overhead. The proposed numerical methodology enables straightforward simulations on dense spatial discretizations on parallel computers without artificial stabilizations.

Future work is in modeling of more complex systems, e.g. bipolar and monopolar transistors with complex geometries. Besides spatially dependant nodal distribution, for addressing high gradients near junctions, we will also employ temporal adaptivity. We expect that such an approach will severely reduce the required number of nodes and iterations, which will make the proposed methodology even more competitive.

Acknowledgment

The authors acknowledge the financial support from the state budget by the Slovenian Research Agency under Grant P2-0095.

References

[1] Simpson RN, Bordas SPA, Asenov A, Brown AR. Enriched residual free bubbles for semiconductor device simulation. Comput Mech 2012;50:119–33.
[2] Chand S, Kaushal P, Osvald J. Numerical simulation study of current–voltage character of a Schottky diode with inverse doped surface layer. Mater Sci Semicond Process 2012;16:454–60.
[3] Tessler N, Roichman Y. Two-dimensional simulation of polymer field-effect transistor. Appl Phys Lett 2001;79:2987–9.
[4] Gharghi M, Bai H, Stevens G, Sivoththaman S. Three-dimensional modeling and simulation of p–n junction spherical silicon solar cells. IEEE Trans Electron Devices 2006;53:1355–63.
[5] Deinega Alexei, John Sajeev. Finite difference discretization of semiconductor drift-diffusion equations for nanowire solar cells. Comput Phys Commun 2012;183:2128–35.
[6] Jacoboni C, Lugli P. The Monte Carlo Method for Semiconductor Device Simulation. Wien: Springer; 1989.
[7] Burgler JF, Bank RE, Fichtner W, Smith, RK. A new discretization scheme for the semiconductor current continuity equations. In: Proceedings of the IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 8; 1989. pp.479–489.
[8] Carey GF, Pardhanani AL, Bova SW. Advanced numerical methods and software approaches for semiconductor device simulation. VLSI Des 2000;10:391–414.
[9] Grasser T, Tang TW, Kosina H, Selbherherr S. A Review of hydrodynamic and energy-transport models for semiconductor device simulation. Proc IEEE 2003;91:251–74.
[10] Courant R, Isaacson E, Rees M. On the solution of nonlinear hyperbolic differential equations by finite differences. Commun Pure Appl Math 1952;5:243–55.
[11] Lin H, Atluri SN. Meshless local Petrov Galerkin method (MLPG) for convection–diffusion problems. Comput Model Eng Sci 2000;1:45–60.
[12] Brezzi F, Bristeau MO, Franca LP, Mallet M, Rogé G. A relationship between stabilized finite element methods and the Galerkin method with bubble functions. Comput Methods Appl Mech Eng 1992;96:117–29.
[13] Flynn MJ, Mencer O, Milutinovic V, Rakocevic G, Stenstrom P, Trobec R, Valero M. Communications of the ACM 2013;56(5):39–42.
[14] Zienkiewicz OC, Taylor RL, Zhu JZ. The Finite Element Method: Its Basis and Fundamentals. Oxford: Elsevier; 2005.
[15] Lin TP. Improving multigrid performance for unstructured mesh drift-diffusion simulations on 147,000 cores. Int J Numer Methods Eng 2012;91:971–89.
[16] Gingold RA, Monaghan JJ. Smoothed particle hydrodynamics: theory and application to non-spherical stars. Mon Not R Astron Soc 1977;181:375–89.
[17] Nayroles B, Touzot G, Villon P. Generalizing the finite element method: Diffuse approximation and diffuse elements. Comput Mech 1992;10:307–18.
[18] Atluri SN, Shen S. The Meshless Method. Encino: Tech Science Press; 2002.
[19] Belytschko T, Lu YY, Gu L. Element-free Galerkin methods. Int J Numer Methods Eng 1994;37:229–56.
[20] Wang CA, Sadat H, Prax C. A new meshless approach for three dimensional fluid flow and related heat transfer problems. Comput Fluids 2012;69:136–46.
[21] Duan Y, Rong F. A numerical scheme for nonlinear Schrödinger equation by MQ quasi-interpolation. Eng Anal Bound Elem 2013;37:89–94.
[22] De Chowdhury S, Sannasiraj SA. SPH Simulation of shallow water wave propagation. Ocean Eng 2013;60:41–52.
[23] Arefmanesh A, Najafi M, Musavi SH. Buoyancy-driven fluid flow and heat transfer in a square cavity with a wavy baffle—Meshless numerical analysis. Eng Anal Bound Elem 2013;37:366–82.
[24] Guo YM. An overrange collocation method. Comput Model Eng Sci 2011;73:1–22.
[25] Stevens D, Power H. A scalable meshless formulation based on RBF hermitian interpolation for 3D nonlinear heat conduction problems. Comput Model Eng Sci 2010;55:111–45.
[26] Wang JG, Liu GR. A point interpolation meshless method based on radial basis functions. Int J Numer Methods Eng 2002;54:1623–48.
[27] Lee CK, Liu X, Fan SC. Local muliquadric approximation for solving boundary value problems. Comput Mech 2003;30:395–409.
[28] Šarler B. From global to local radial basis function collocation method for transport phenomena. Adv Meshfree Tech 2007:257–82.
[29] Trobec R, Kosec G, Šterk M, Šarler B. Comparison of local weak and strong form meshless methods for 2-D diffusion equation. Eng Anal Bound Elem 2012;36:310–21.
[30] Shur MS, Kwok KNG. Physics of Semiconductor Devices. 3rd ed.. Hoboken: Wiley-Interscience; 2007.
[31] Malan AG, Lewis RW. An artificial compressibility CBS method for modelling heat transfer and fluid flow in heterogeneous porous materials. Int J Numer Methods Eng 2011;87:412–23.
[32] Lee TS. A false transient approach to steady state solution of fluid flow through vascular constrictions. Comput Mech 1991;7:269–77.
[33] Kosec G, Zinterhof P. Local strong form meshless method on multiple graphics processing units. Comput Model Eng Sci 2013;91:377–96.
[34] Trobec R, Šterk M, Robič B. Computational complexity and parallelization of the meshless local Petrov-Galerkin method. Computers & Structures 2009;87:81–90.
[35] Bollig EF, Flyer N, Erlebacher G. Solution to PDEs using radial basis function finite-differences (RBF-FD) on multiple GPUs. J Comput Phys 2012;231:7133–51.
[36] Kosec G, Šarler B. H-adaptive local radial basis function collocation meshless method. Comput Mater Contin 2011;26:227–53.
[37] Kosec, G, Trobec, R. A parallel meshless numerical approach for the solution of transport phenomena. In: Proceedings of the Eleventh International Conference on Computational Structures Technology, Dubrovnik; 2012.