A Meshless Method for the Numerical Solution of the 2- and 3-D Semiconductor Poisson Equation

C.J. Wordelman, N.R. Aluru, U. Ravaioli[1]

Abstract: This paper describes the application of the meshless Finite Point (FP) method to the solution of the nonlinear semiconductor Poisson equation. The FP method is a true meshless method which uses a weighted least-squares fit and point collocation. The nonlinearity of the semiconductor Poisson equation is treated by Newton-Raphson iteration, and sparse matrices are employed to store the shape function and coefficient matrices. Using examples in two- and three-dimensions (2- and 3-D) for a prototypical n-channel MOSFET, the FP method demonstrates promise both as a means of mesh enhancement and for treating problems where arbitrary point placement is advantageous, such as for the simulation of carrier wave packet and dopant cloud effects in the ensemble Monte Carlo method. The validity of the solutions and the capability of the method to treat arbitrary boundary conditions is shown by comparison with finite difference results.

keyword: finite point methods, meshless methods, mesh generation, monte carlo methods.

1 Introduction

The semiconductor Poisson equation which includes forcing terms for the potential-dependent carrier and ionized dopant concentrations is commonly used as a model equation in the field of computational electronics. Traditionally, Finite Difference (FD) and Finite Element (FE) methods have been used to solve the semiconductor Poisson equation, but recent developments in the fields of numerical methods indicate that meshless methods may be useful [Belytschko, Krongauz, Organ, Fleming, and Krysl (1999)]. In regard to the semiconductor Poisson equation, meshless methods could be used for mesh enhancement in regions where dopant or carrier concentration gradients are large without the restrictions of FD and FE methods. Meshless methods could also be combined with newly emerging point-particle techniques to treat carrier and dopant interactions in 3-D ensemble Monte Carlo simulations [Wordelman and Ravaioli (2000)]. Finally, meshless methods show potential to model carrier wave packet and dopant cloud effects in the ensemble Monte Carlo method.

In this work, the semiconductor Poisson equation has been solved with the Finite Point (FP) method [Oñate, Idelsohn, Zienkiewics, and Taylor (1996); Oñate, Idelsohn, O. C. Zienkiewics, and Sacco (1996)]. The method uses a weighted least-squares fit of the unknown approximate solution using a set of monomial base interpolating functions. The least-squares fit is used to generate a set of shape functions on which the problem at hand is solved. The FP method and the weighted least-squares approximation are described in Sec. 2 and 3. The non-linearity of the equation is treated in Sec. 4 with Newton-Raphson iteration and a technique is presented to implement Dirichlet and Neumann boundary conditions in this scheme. The semiconductor Poisson equation and its FP implementation are described in Sec. 5. The non-symmetric linear residual step is solved here using a preconditioned Generalized Minimum Residual (GMRES) method and sparse matrices. Finally, FP solutions of the semiconductor Poisson equation for a prototypical n-channel MOSFET are compared to FD results in 2- and 3-D in Sec. 6.

2 Finite point method

Meshless methods are a group of techniques useful for solving partial differential equations on irregular grids. Meshless methods originated from work with FD and FE methods, but meshless methods can treat an irregular distribution of points and require no costly mesh generation. In addition, since meshless methods use a functional basis and allow arbitrary placement of points, the solution and its derivatives may be found directly where they are needed and with greater accuracy than with FD and FE methods where differences and interpolation are required. Meshless methods include the Smooth Particle Hydrodynamics method [Moraghan (1988)], the Diffuse Element method [Nayroles, Touzot, and Villon (1992)], the Element-Free Galerkin method [Lu, Belytschko, and Gu (1994)], the Reproducing Kernel Particle (RKP) method [Liu, Jun, and Zhang (1995)], the Boundary Node method [Mukherjee and Mukherjee (1997)], the Local Boundary Integral Equations method [Zhu, Zhang, and Atluri (1998a,b)], the meshless local Petrov-Galerkin (MLPG) method [Atluri and Zhu (1998)], Cloud-Based methods [Duarte and Oden (1996)], the Partition of Unity method [Melenk and Babuska (1996)], and the Finite Point method. In this work, the Finite Point (FP) method using weighted least-squares is chosen for its simplicity and high degree of flexibility [Oñate, Idelsohn, Zienkiewics, and Taylor (1996); Oñate, Idelsohn, O. C. Zienkiewics, and Sacco (1996)].

Assuming a scalar problem, the differential equation to be

[1] Univ. of Illinois, Urbana, IL, USA.

solved can be written

$$L\,u = f \text{ in } \Omega \tag{1}$$

with boundary conditions

$$u = g \text{ on } \Gamma_g \tag{2}$$

$$\frac{\partial u}{\partial n} = h \text{ on } \Gamma_h \tag{3}$$

where L is the differential operator, u is the unknown function, f is the forcing term, Γ_g is the boundary region where Dirichlet conditions, g, are imposed and Γ_h is the boundary region where Neumann conditions, h, are imposed, and n is the unit outward normal vector.

Approximating the unknown function u as u^a, the weighted residual method may be applied at each point i in the solution domain as

$$\int_\Omega W_i[L\,u^a - f]d\Omega + \int_{\Gamma_g} \bar{W}_i[u^a - g]d\Gamma + \int_{\Gamma_h} \bar{\bar{W}}_i[\frac{\partial u^a}{\partial n} - h]d\Gamma = 0 \tag{4}$$

where W_i, $\bar{W}_i$, and $\bar{\bar{W}}_i$ are the weighting functions and may be defined to yield the FD, FE and the meshless techniques as special cases.

In order to keep the problem local and arrive at a banded matrix, the approximation, u^a, may be written in terms of locally defined shape functions as

$$u^a(x) = \sum_{j=1}^{n} N_j(x)u_j^h = \mathbf{N}^{\mathrm{T}}\mathbf{u}^h \tag{5}$$

where j indexes the n points in the interpolation domain (or cloud) of the point i, u_j^h is the coefficient of the jth shape function, and $N_j(x)$ is the jth shape function which satisfies

$$N_j(x) \neq 0 \text{ if } x \in \Omega_i \tag{6}$$

$$N_j(x) = 0 \text{ if } x \notin \Omega_i \tag{7}$$

Using point collocation, the validity of interpolation may be limited to the point i with $W_i = \bar{W}_i = \bar{\bar{W}}_i = \delta_i$ in Eq. 4. The governing and boundary equations can then be written

$$L\,u^a(x_i) = f(x_i) \quad \text{for i} = 1,2,\ldots,\mathrm{N}_\Omega \tag{8}$$

$$u^a(x_i) = g(x_i) \quad \text{for i} = 1,2,\ldots,\mathrm{N}_{\Gamma_g} \tag{9}$$

$$\frac{\partial u^a(x_i)}{\partial n} = h(x_i) \quad \text{for i} = 1,2,\ldots,\mathrm{N}_{\Gamma_h} \tag{10}$$

where N_Ω, N_{Γ_g}, and N_{Γ_h} are the number of points in the governing, Dirichlet and Neumann regions, respectively, and the total number of points is $N_p = N_\Omega + N_{\Gamma_g} + N_{\Gamma_h}$.

3 Weighted least-squares approximation

In the least-squares approximation, the unknown function u is approximated as u^a such that the sum of the square error of the approximation is minimized. In one dimension (1-D), the approximation function can be written as

$$u^a(x) = \sum_{k=1}^{m} p_k(x)\alpha_k = \mathbf{p}^{\mathrm{T}}\alpha \tag{11}$$

where $\mathbf{p}$ are the base interpolating functions and $\alpha = [\alpha_1, \alpha_2, \ldots, \alpha_m]^{\mathrm{T}}$ are the coefficients of the approximation on the base functions. Using quadratic interpolation and a monomial basis, the interpolating functions may be chosen in 1-D as

$$\mathbf{p}_{1\mathrm{D}} = [1, x, x^2]^{\mathrm{T}} \text{ for } m = 3 \tag{12}$$

and in 2- and 3-D as

$$\mathbf{p}_{2\mathrm{D}} = [1, x, y, x^2, xy, y^2]^{\mathrm{T}} \text{ for } m = 6 \tag{13}$$

$$\mathbf{p}_{3\mathrm{D}} = [1, x, y, z, xy, xz, yz, x^2, y^2, z^2]^{\mathrm{T}} \text{ for } m = 10 \tag{14}$$

The function, $u(x)$, is then sampled at the n points in the cloud Ω_i giving

$$\mathbf{u}^h = \begin{bmatrix} u_1^h \\ u_2^h \\ \vdots \\ u_n^h \end{bmatrix} \cong \begin{bmatrix} u_1^a \\ u_2^a \\ \vdots \\ u_n^a \end{bmatrix} = \begin{bmatrix} p_1^{\mathrm{T}} \\ p_2^{\mathrm{T}} \\ \vdots \\ p_n^{\mathrm{T}} \end{bmatrix} \alpha = \mathbf{C}\alpha \tag{15}$$

where $n = m$ for an exact fit and $n > m$ for a least-squares fit. In the $n > m$ case, the summed square error at each point is

$$J_i = \sum_{j=1}^{n} (u_j^h - u^a(x_j))^2 = \sum_{j=1}^{n} (u_j^h - \mathbf{p}_j^{\mathrm{T}}\alpha)^2. \tag{16}$$

A Gaussian weighting function of the form

$$\psi_i(x) = \frac{e^{-(x/c)^2} - e^{-(x_m/c)^2}}{1 - e^{-(x_m/c)^2}} \tag{17}$$

may be used to weight the square error, where the values x_m and c are chosen to control the cloud size and Gaussian shape [Oñate, Idelsohn, Zienkiewics, and Taylor (1996)]. It should be noted that in 2- and 3-D, x and x_m above were taken as the 2- and 3-D radius respectively and no tensor products (as introduced below) were considered. This choice was made for simplicity and maximum control over the number of points in the cloud.

As another possibility to the Gaussian function, a cubic spline function may be used to perform the weighting [Aluru (1999)]. In 3-D, the cubic spline may be defined as

$$\psi_i(x,y,z) = w(\frac{|x-x_i|}{d_x})w(\frac{|y-y_i|}{d_y})w(\frac{|z-z_i|}{d_z}) \tag{18}$$

where d_x, d_y and d_z are the dilations in each dimension and $w(v)$ is the cubic spline kernel function given by

$$w(v) = \begin{cases} \frac{2}{3} - v^2(1-\frac{v}{2}) & 0 \le v \le 1 \\ -\frac{1}{6}(v-2)^3 & 1 \le v \le 2 \\ 0 & v > 2 \end{cases} \tag{19}$$

Incorporating the chosen weighting function in Eq. 16 yields

$$\begin{aligned} J_i &= \sum_{j=1}^{n} \psi_i(x_j - x_i)\,(u_j^h - u^a(x_j))^2 \\ &= \sum_{j=1}^{n} \psi_i(x_j - x_i)\,(u_j^h - \mathbf{p}_j^{\mathrm{T}}\alpha)^2 \end{aligned} \tag{20}$$

Minimizing the error with respect to α yields

$$\alpha = \mathbf{C}^{-1}\mathbf{u}^{\mathbf{h}} = \mathbf{A}^{-1}\mathbf{B}\mathbf{u}^{\mathbf{h}} \tag{21}$$

where

$$\begin{aligned} \mathbf{B} = [&\psi_i(x_1 - x_i)\,p(x_1)\,, \psi_i(x_2 - x_i)\,p(x_2)\,, \\ &\ldots, \psi_i(x_n - x_i)\,p(x_n)] \end{aligned} \tag{22}$$

$$\mathbf{A} = \mathbf{B}\mathbf{p}^{\mathrm{T}}. \tag{23}$$

Combining the results in Eq. 5, Eq. 11, Eq. 15 and Eq. 21 gives the shape functions, $\mathbf{N}$, as

$$\mathbf{N}^{\mathrm{T}} = \mathbf{p}^{\mathrm{T}}\mathbf{C}^{-1} = \mathbf{p}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{B} \tag{24}$$

where the shape functions are found by solving the matrix subproblem $\mathbf{A}\mathbf{C}^{-1} = \mathbf{B}$ over the cloud for $\mathbf{C}^{-1}$ at each point i. It should be noted that all nonzero shape functions at the point i must be taken from the solution of the i-th subproblem for proper normalization of the shape functions.

Finally, a matrix equation is obtained

$$\mathbf{K}\mathbf{u}^h = \mathbf{b} \tag{25}$$

where the coefficient matrix $\mathbf{K} \in \Re^{N_p \times N_p}$ is constructed by inserting $L\mathbf{N}^{\mathrm{T}}$, $\mathbf{N}^{\mathrm{T}}$, and $\frac{\partial \mathbf{N}^{\mathrm{T}}}{\partial n}$ for the governing, Dirichlet, and Neumann regions respectively and the right-hand side vector $\mathbf{b} \in \Re^{N_p \times 1}$ is set to f, g and h at each point i as per Eq. 8-10. Since the shape functions are locally defined or $n << N_p$, the coefficient matrix $\mathbf{K}$ is sparse, but because the distance between cloud points varies and the cloud size is not necessarily uniform, symmetry of the coefficient matrix is not achieved. The solution vector $\mathbf{u}^h \in \Re^{Np \times 1}$ are the coefficients of the solution on the shape function basis and Eq. 5 must be applied to find the approximate solution, $u^a(x)$.

It should be noted that the general antisymmetry of the coefficient matrix, $\mathbf{K}$, in the FP method and the coefficient matrix of other meshless methods does not occur in the FD method and some FE methods and introduces several limitations currently. First, the antisymmetry precludes the use of the conjugate gradient method commonly employed for FD implementations of the semiconductor Poisson equation. (The conjugate gradient method is the basis of the FD solver used for comparison in this work [Trellakis, Galick, Pacelli, and Ravaioli (1997)]). This implementation of the FP method utilizes the iterative preconditioned GMRES method [Saad (1994)] to treat the antisymmetry. Second, the antisymmetry of the coefficient matrix precludes the solution of real eigenvalue problems such as the time-independent Schrödinger equation which is often coupled to the semiconductor Poisson equation for self-consistent solution. Other issues concerning the coefficient matrix are those of reordering and bandedness. In this work, no attempt is made to enforce strict banding and sparse matrices are used for storing and manipulating of the coefficient matrix. For performance reasons, reordering is used to concentrate non-zero coefficients along the diagonal. It should be further noted that care must be taken to limit the cloud size and the resulting total error in the least-squares fit. In this work, the cloud size is limited to 2-3 times the number of base interpolating functions.

4 Newton-Raphson iteration and boundary conditions

Nonlinearity in the governing equation [Eq. 1] may be treated by Newton-Raphson iteration. If the forcing term is moved to the left-hand-side, the general differential equation is satisfied when the function $\omega(u) = Lu - f(u)$ is zero. The standard Newton Raphson step for u is then

$$\Delta u = -\frac{\omega(u)}{\omega'(u)} = -\frac{Lu - f}{\frac{\partial}{\partial u}(Lu - f)} = -\frac{\Re(u)}{L\,(\frac{\partial u}{\partial u}) - \frac{\partial f}{\partial u}} \tag{26}$$

or

$$(L - \frac{\partial f}{\partial u})\Delta u = -\Re(u) \tag{27}$$

where $\Re(u) = Lu - f$ is the residual. Writing Eq. 27 as a vector equation and using $\mathbf{u} = \mathbf{N}^{\mathrm{T}}\mathbf{u}^h$ and $\Delta\mathbf{u} = \mathbf{N}^{\mathrm{T}}\Delta\mathbf{u}^h$ yields

$$(L - \frac{\partial f}{\partial u})\mathbf{N}^{\mathrm{T}}\Delta\mathbf{u}^h = -\Re(\mathbf{u}) \tag{28}$$

or

$$(L\mathbf{N}^{\mathrm{T}} - \frac{\partial f}{\partial u}\mathbf{N}^{\mathrm{T}})\Delta\mathbf{u}^h = -\Re(\mathbf{u})\,. \tag{29}$$

If the general form in Eq. 25 is inserted above, then

$$(\mathbf{K} - \frac{\partial f}{\partial u}\mathbf{N}^{\mathrm{T}})\Delta\mathbf{u}^h = -\Re(\mathbf{u}) \tag{30}$$

or

$$\mathbf{K}'\Delta\mathbf{u}^h = \Delta\mathbf{b} \tag{31}$$

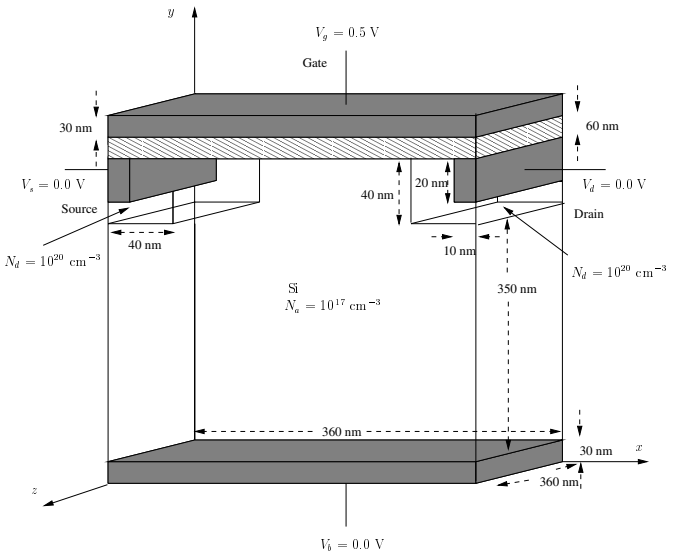

Figure 1 : Diagram of the proto-typical MOSFET treated in this work.

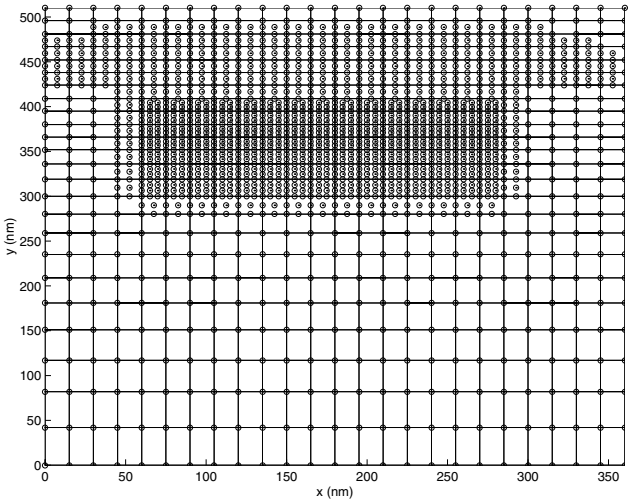

Figure 2 : The 2-D point distribution is shown where dotted circles show the FP distribution and connected lines show the FD mesh. The FP mesh density matches the (25x25) FD mesh in the bulk of the MOSFET but is doubled in the upper portion of the device and tripled in the channel region.

where $\mathbf{K}^{l} = \mathbf{K} - \frac{\partial f}{\partial u}\mathbf{N}^{\mathrm{T}}$ and $\Delta b = -\Re(\mathbf{u}) = -(\mathbf{K}\mathbf{u}^{h} - \mathbf{b})$.

In this approach, the shape function coefficients may be found initially to satisfy the Dirichlet conditions with $\mathbf{K}\mathbf{u}_{\mathrm{o}} = \mathbf{b}$ and may be iterated from $\mathbf{u}^{l} = \mathbf{u}_{\mathrm{o}}$ with $\mathbf{u} = \mathbf{u} + \Delta\mathbf{u}$ by enforcing $\mathbf{K}^{l}\Delta\mathbf{u} = 0$ at each step, such that the Dirichlet conditions are maintained. The Neumann conditions, on the other hand, may be treated identically to the governing equations where Eq. 31 will set the variation of the slope as the negative residual slope and will, thereby, effectively force the slope to zero with each step.

5 Semiconductor Poisson equation

The semiconductor Poisson equation may be written as

$$\nabla(\varepsilon\nabla\phi) = -\rho[\phi] \tag{32}$$

$$= -q(-n[\phi] + p[\phi] + N_D^{+}[\phi] - N_A^{-}[\phi]) \tag{33}$$

where ϕ is the electrostatic potential, n and p are the electron and hole concentrations, and N_D^{+} and N_A^{-} are the ionized donor and acceptor concentrations.

In terms of the differential equation in Eq. 1, we see $L = \nabla\varepsilon\nabla$, $u = \phi$ and $f = -\rho$. These terms may be substituted into Eq. 31 for the nonlinear case, or, if a ϕ-independent charge distribution is employed, Eq. 25 may be used directly. As discussed in Sec. 3, sparse matrices were used to store the shape function and coefficient matrix and the iterative preconditioned GMRES method was used to treat the antisymmetric coefficient matrix in performing the solve.

6 Numerical results

The procedure outlined above was used to solve for the electrostatic potential profile in a prototypical unipolar n-channel MOSFET in 2- and 3-D, and results are compared for corresponding FD problems (see Fig. 1). In both examples at $T = 300$ K, the semiconductor Poisson equation was solved in silicon with a substrate of uniform boron doping at $N_a = 10^{17}\mathrm{cm}^{-3}$ and source and drain implants of uniform arsenic doping at $N_d = 10^{20}\mathrm{cm}^{-3}$. Dirichlet conditions were imposed for equilibrium bias conditions of $V_g = 0.5$ V and $V_s = V_d = V_b = 0$ V and the built-in bias. Neumann conditions were imposed along all unbiased edges to force the electric field to zero.

The 2-D example shown in Figs. 2 and 3 demonstrates the utility of the FP method for enhancement of nonuniform rectilinear meshes. The mesh density matches that the FD mesh used for comparison in the bulk region of the MOSFET, but is doubled or quadrupled in the upper portion and channel regions of the device respectively (see Fig. 2). Using adaptively determined constants in the Gaussian weighting function Eq. 17 for the 2-D case, a cloud size of $n \cong 12$ is maintained. The results for the conduction band edge are compared with those from a corresponding FD solve in Fig. 3. Note that the boundary conditions along the sides and at the contacts are enforced and good agreement with the FD solution is obtained.

It should be noted that the use of point collocation (see Sec. 2) in the FP method calls for the solution of equation Eq. 32 rather than its commonly-used box integrated form. This relaxes the constraint that mesh lines should fall on the mate-

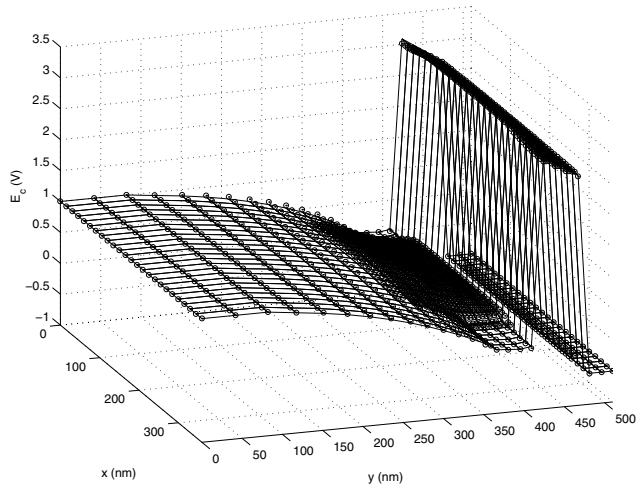

Figure 3 : The base of the conduction band is shown for the 2-D example where dotted circles show the FP solution and connected lines show the FD solution.

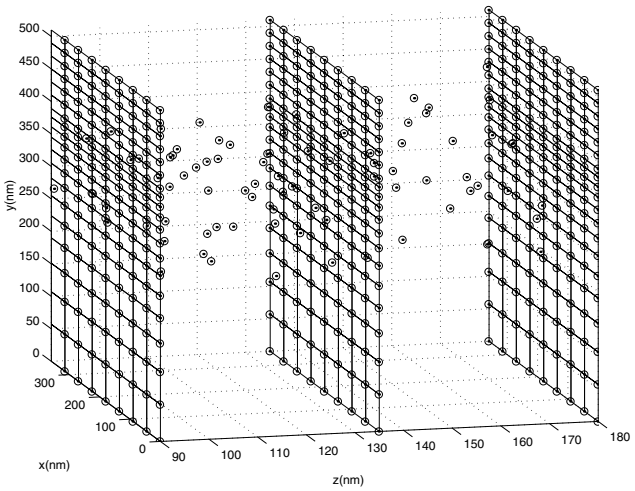

Figure 4 : The center of the 3-D point distribution is shown where dotted circles show the FP distribution and connected lines show the FD mesh plane used for comparison in Fig. 5. The FP distribution matches the FD mesh but 80 additional points are randomly placed in the channel region to demonstrate the flexibility of the method in treating problems where arbitrary point placement is advantageous.

rial boundaries for highest accuracy, but introduces the corresponding constraint that sufficient points must be used near boundaries to adequately resolve them.

The 3-D example shown in Fig. 4 and 5 demonstrates the utility of the FP method to place points arbitrarily on a nonuniform rectilinear background mesh. While rectilinear enhancement (as done in the 2-D example) would be even more useful in 3-D, this example identifies a more powerful application of meshless methods, namely that meshless methods allow solution points to be inserted independently from the background mesh geometry. This functionality, for instance, could be applied to model carrier wave packet and dopant cloud effects within ensemble Monte Carlo simulations. In the 3-D example, 80 random points in the channel region are added to a $9 \times 20 \times 9$ background. The cubic spline weighting function is employed and a cloud size of $n \cong 30$ is maintained. The FP solution is compared with those of a $9 \times 20 \times 9$ FD mesh and good agreement is achieved.

7 Conclusion

The weighted least-squares FP method is applied to the solution of the semiconductor Poisson equation. The non-linearity is treated with Newton-Raphson iteration and a technique is presented to implement Dirichlet and Neumann boundary conditions in this scheme. Sparse matrices are employed to store the shape function and coefficient matrices and the nonsymmetric linear residual step is solved by using the GMRES method. Comparison of solutions in 2- and 3-D with those of obtained with the FD method indicates that the FP method can treat both Dirichlet and Neumann boundaries in nonlinear problems, while it provides the flexibility of solving the semiconductor Poisson equation on an irregular grid. The numerical examples considered here demonstrate that the FP method shows promise both as a means for enhancement of nonuniform meshes with fewer constraints than with FD and FE methods and as a means to treat physical problems where arbitrary point placement is advantageous, such as the simulation of carrier wave packet and dopant cloud effects in the ensemble Monte Carlo method. The further application of the FP method and other meshless methods toward the problems outlined in this work are currently in progress.

Acknowledgement: This work was supported by the United States Department of Energy under the CRADA with Semiconductor Research Corporation (SRC contracts 97-CJ-816 and 98-SJ-406), by the National Center for Computational Electronics (NCCE) and by Los Alamos National Laboratory.

References

Aluru, N. R. (1999): A reproducing kernel particle method for meshless analysis of microelectromechanical systems. *Comput. Mechanics*, vol. 23, pp. 3–47.

Atluri, S.; Zhu, T. (1998): A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics. *Comput. Mech.*, vol. 22, pp. 117–127.

Belytschko, T.; Krongauz, Y.; Organ, D.; Fleming, M.; Krysl, P. (1999): Meshless methods: An overview and re-

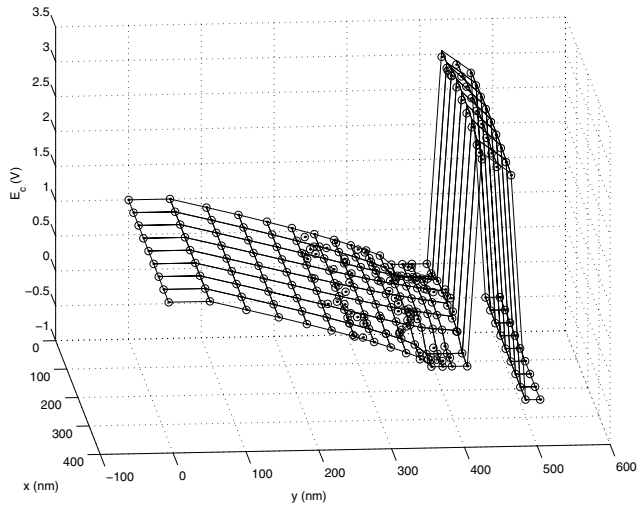

Figure 5 : The base of the conduction band is shown for the 3-D example in the $z = 135 \pm 40$ nm region where individual points show the FP solution and connected lines show the FD solution.

cent developments. *Comput. Metho. Appl. Mech. Eng.*, vol. 129, pp. 3–47.

Duarte, C.; Oden, J. (1996): An h-p adavptive method using clouds. *Comput. Meth. Appl. Mech. Eng.*, vol. 139, pp. 237–262.

Liu, W.; Jun, S.; Zhang, Y. (1995): Reproducing kernel particle methods. *Int. J. Numer. Methods fluids*, vol. 20, pp. 1081–1106.

Lu, Y.; Belytschko, T.; Gu, L. (1994): A new implementation of the element free Galerkin method. *Comput. Methods Appl. Mech. Engrg.*, vol. 113, pp. 397–414.

Melenk, J.; Babuska, I. (1996): The partition of unity finite element method: Basic theory and applications. *Comput. Appl. Mech. Eng.*, vol. 139, pp. 289–314.

Moraghan, J. (1988): An introduction to SPH. *Comput. Phys. Comm.*, vol. 48, pp. 89–96.

Mukherjee, Y.; Mukherjee, S. (1997): The boundary node method for potential problems. *Int. J. Numer. Methods Eng.*, vol. 40, pp. 797–815.

Nayroles, B.; Touzot, G.; Villon, P. (1992): Generalizing the FEM: Diffuse approximation and diffuse elements. *Comput. Mech.*, vol. 10, pp. 307–318.

Oñate, E.; Idelsohn, S.; O. C. Zienkiewics, R. L. T.; Sacco, C. (1996): A stabilized finite point method for analysis of fluid mechanics problems. *Comput. Methods Appl. Mech. Engrg.*, vol. 139, pp. 315–346.

Oñate, E.; Idelsohn, S.; Zienkiewics, O. C.; Taylor, R. L. (1996): A finite point method in computational mechanics. Applications to convective transport and fluid flow. *Int. J. Numer. Methods Engrg.*, vol. 39, pp. 3839–3866.

Saad, Y. (1994): *SPARSKIT: a basic tool kit for sparse matrix computations*. `http://www.cs.umn.edu/Research/arpa/SPARSKIT/`.

Trellakis, A.; Galick, A. T.; Pacelli, A.; Ravaioli, U. (1997): Iteration scheme for the solution of two-dimensional schrödinger-poisson equations in quantum structures. *J. Appl. Phys.*, vol. 81, pp. 7880–7884.

Wordelman, C.; Ravaioli, U. (2000): Integration of a particle-particle-particle-mesh algorithm with the ensemble Monte Carlo method for the simulation of ultra-small semiconductor device. *IEEE Trans. Electron Devices. to appear.*

Zhu, T.; Zhang, J.; Atluri, S. (1998): A local boundary integral equation (LBIE) method in computational mechanics and a meshless discretization approach. *Comput. Mech.*, vol. 21, pp. 223–235.

Zhu, T.; Zhang, J.; Atluri, S. (1998): A meshless local boundary integral equation (LBIE) method for solving nonlinear problems. *Comput. Mech.*, vol. 22, pp. 174–186.