The Simulation of Diode Based on PSO Algorithm

Guofeng Pan Kewen Xia Naying Xu Fan Yang Ruixia Yang

School of Information Engineering, Hebei University of Technology, Tianjin 300130, P. R. China

Abstract

It is very difficult to solve the large-scaled in the semiconductor device simulation, sparse and ill-conditioned matrix equation exactly and quickly. So the particle swarm optimization algorithm（PSO）based on the swarm intelligent technology is presented to get the global optimal solution of the matrix equation. This proposed algorithm is mainly to optimize the position of the particle in stead of solve the matrix equation. First of all, initialize the position and velocity of particles, then renew the iterative solution, until to search the global optimum. The example to solve the potential in linearly graded p-n junctions indicates that the improved PSO algorithm introduced in the paper is superior to Newton algorithm and traditional PSO in precision, velocity of convergence and saving memory. The simulation results of diodes are accord with that of theoretical analysis, which shows the PSO algorithm is applicable and accurate.

Keywords: Semiconductor device simulation, Matrix equation, Particle Swarm Optimization algorithm (PSO)

1. Introduction of the problem

The simulation of semiconductor devices is to obtain the functional relationship of static voltage, the electron and hole concentration versus time or space, moreover, get the characteristic at any bias and other physic states by solving the basic equations. In the condition of knowing the manufacture material, feature size and adulterating distribution the simulation result can offer good reference for devices manufacture and design[1]-[2].

The basic equations are Poisson equation, carrier concentration equations and current continuity equations. The Poisson equation is as follows:

$$\varepsilon \nabla \bullet \overline{E}_l = -\varepsilon \nabla^2 \psi = \rho \qquad (1)$$

Where $\rho = -q(n - p - \Gamma)$, $\Gamma = N_d^+ - N_a^-$.

Also, The current density equation is as follows:

$$J_n = q\mu_n n \vec{E}_l + qD_n \nabla n \qquad (2)$$

$$J_p = q\mu_p p \vec{E}_l - qD_p \nabla p \qquad (3)$$

The current continuity equation is as follows:

$$-\frac{1}{q}\nabla \vec{J}_n - G_n + R_n + \frac{\partial n}{\partial t} = 0 \qquad (4)$$

$$-\frac{1}{q}\nabla \vec{J}_p - G_p + R_p + \frac{\partial p}{\partial t} = 0 \qquad (5)$$

While the feature size of the device is small enough to a certain value, there will be very large error and along with other physical phenomenon as carrier shot and electronic velocity overshot et al unaccountable non-linear transport phenomena in MOS devices. When the feature size of device (as the grid length of FET) reaches to a certainty value (0.1μm for silicon devices, 0.5μm for GaAs devices), the current continuity equation will be replaced by Boltzmann equation[3]-[4], namely:

$$\frac{\partial f}{\partial t} = -\frac{1}{\hbar}\vec{F} \bullet \nabla_k f - \vec{v}\nabla_r f$$

$$+ \sum_{k'}\{f(\vec{k'})p(\vec{k'},\vec{k}) - f(\vec{k})p(\vec{k},\vec{k'})\} \qquad (6)$$

Where f is the carrier distributing function, $p(\vec{k},\vec{k'})$ is the probability of carrier-wave vector from $\vec{k}$ to $\vec{k'}$.

It is difficult to solve the PDE, and could not get the accurate solution easily in general numerical way. So the key to semiconductor devices simulation is to solve the liner equation group precisely. Through studying the intelligent optimization algorithm PSO, we improve it and use it to solve the liner equation group. The simulation result of diode indicates that the PSO is better than numerical method in semiconductor simulation.

2. PSO algorithm (particle swarm optimization algorithm)

2.1. Introduction of PSO and its improvement

PSO algorithm is a new evolutionary computation technique, Kennedy and Eberhart proposed the PSO algorithm model firstly[5], which searches the optimal result in high dimension space with the cooperation and competition among all individuals. Usually particles of every generation will pursue two best results: one is the *pbest*, the individual best result, reached by each particle *pbest*, which stands for each particle's cognizing level to optimal direction; the other is *gbest*, which stands for the social cognizing level of the swarm, is the group best result reached by the swarm. All particles moved according to following equations:

$$v_i = v_{i-1} + \varphi_1 \cdot r_1 \cdot (pbest - x_{i-1}) + \varphi_2 \cdot r_2 \cdot (gbest - x_{i-1})$$
(7)

$$x_i = x_{i-1} + v_i$$
(8)

According to the above model, we can have two supposes.

Suppose 1: the acceleration constant $\varphi_1, \varphi_2 \to \infty$, then the maximum velocity $v \to \infty$, so the particle swarm may miss the optimum solution, and induce the algorithm non-convergence.

Suppose 2: In order to get the *gbest*, take φ_1, φ_2 little enough, so the maximum velocity v will be small enough. And the convergent velocity is so slow, and the time is so long, that the memory is not enough to do the calculation. So the particle can not continue to optimize.

Therefore, we add convergence constant χ, inertia weight ω and restraint constant α to the base model to get the improved model [6]:

$$v_i = \chi \cdot [\omega \cdot v_{i-1} + \varphi_1 \cdot r_1 \cdot (pbest - x_{i-1})$$
$$+ \varphi_2 \cdot r_2 \cdot (gbest - x_{i-1})]$$
(9)

$$x_i = x_{i-1} + a \cdot v_i$$
(10)

Where $\chi = \dfrac{2}{|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}|}$, $\varphi = \varphi_1 +$

$\varphi_2 > 4$, φ usually equals to 4.1, so λ =0.729, the process of this algorithm is as follows:

Step1 Initialize the particle swarm. Suppose accelerate constant φ_1 and φ_2, inertia weight ω, the maximum iterative times $ITER_{max}$, $ITER$ is current iterative time. In defined space R^n, initial population $X(t)$ is composed of the particles $x_1, x_2, \cdots, x_s$ by random, and the matrix of velocity $V(t)$ is composed of initial velocities $v_1, v_2, \cdots, v_s$ by random, the individual optimum solution $pbest_i$ of every particle is initialized by the original value of x_i;

Step2 Update the velocities and positions of particles according to the equation (9) and (10) to generate new population $X(t+1)$. The rule of adjusting velocity is as follows:

$$\begin{cases} v_i = V_{max}, & if & v_i > V_{max} \\ v_i = -V_{max}, & if & v_i \le V_{max} \end{cases}$$
(11)

Step3 Evaluate the fitness of every particle. Define the fitness function as following:

$$f = \frac{1}{m} \sum_{i=1}^{m} (y_i - \hat{y}_i)^2$$
(12)

Where $\hat{y}_i$ and y_i are respectively calculated result and reality result. We can calculate the fitness function according to the mean square deviation of residual $(y - Ax)$ to resolve the detail matrix equation, i.e.:

$$f(x) = \frac{1}{m} \sum_{i=1}^{m} \left(\sum_{j=1}^{n} a_{ij} x_j - y_i \right)^2$$
(13)

The smaller $f(x)$ is, the stronger the adaptability is.

Step4 With regard to every particle, compare current fitness $f(x_i)$ with the fitness of the best position in the history $f(gbest_i)$. If $f(x_i) < f(gbest_i)$, then $pbest = x_i$; Compare the current fitness of all particles in the swarm with the fitness of the best position in the population, if $f(x_i) < f(gbest_i)$, then the globally optimal solution $gbest_i = x_i$;

Step5 Check the termination condition, if met, then finish searching optimum and the solution is the current optimum individual; otherwise $t = t + 1$, turn to step 2. The termination condition supposed is reaching the maximum iterative times $ITER_{max}$ or the evaluate value is less than the given accuracy;

2.2. Application example

Now we can solve a large-scaled, sparse, and ill-conditioned matrix equation to describe the PSO application in semiconductor device simulation is feasible and effective. By calculating the potential distribution of graded p-n junction to validate the algorithm.

Divide p-n junction into *N* parts, *h* is interval. In every small interval the nonlinear equation can be approximated to linear equation. So the original differential equation can be rewritten as follows[1]:

$$\frac{\psi(N-1) - 2\psi(N) + \psi(N+1)}{h^2} =$$

$$-\frac{q}{\varepsilon}[mx(N) + n_i e^{-\theta\psi(N)} - n_i e^{\theta\psi(N)}]$$

$$x(N) = (N-1)h, 2 \le N \le L-1$$
(14)

The boundary condition is as follow：

$$\psi(1) = 0$$

$$\psi(L) = \frac{1}{\theta}\ln[\frac{mw}{2n_i} + \sqrt{(\frac{mw}{2n_i})^2 + 1}]$$

(15)

Where, the electron hole effective quality m=1e+21 , intrinsic electron concentration n_i=1.4e+10, and θ=38.5.

Finally, we obtain the linear equation group $Ax = B$, where A is a large-scale sparse ill-conditioned matrix of 49×49，the mesh parameter N=50, h=0.07. The matrix equation is to be solved by PSO, ordinary PSO and the Newton iteration algorithm respectively.

The parameters of improved PSO is that particle number is 13, precision is 1e-20, iterative epoch is 5000. The curve of mean-square error is shown in Fig 1. We know the convergent error is 1e-15, when the iterate epoch is 70. The precision is very high.

Fig 2 shows the potential characteristic of the p-n junction. At the right of the neutral region, the potential becomes higher but potential energy lower as the x increasing.
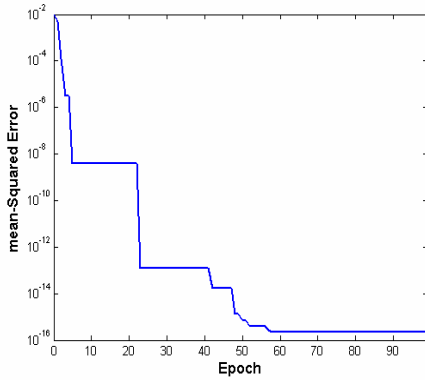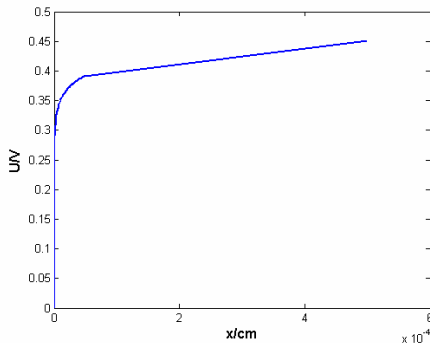

Fig. 1: The convergence error of improved PSO.

Fig. 2: Potential along x axis.

From the figure 3, we know the convergence error is 4.2008e-2, when the iterate epoch is 190, and

latter the convergent velocity become slow. Finally the error reach 4.1399e-2, when the iterate epoch is 1000, obviously, the convergent precision is not very high. And the Newton iteration method is sensitive to iterate initial-value, prone to go to local minimization, so the improved PSO is superior to Newton iteration method in semiconductor device simulation.

In order to show the superiority of improved PSO, we calculate the matrix equation by ordinary PSO too, where the parameters is similar to improved PSO, i.e. particle number is 13, precision is 1e-20, iterative epoch is 5000. convergence curve is showed in figure 4. The curve starts to converge when the epoch is 25000, and the error reaches 5.1407e-5, till 50000.
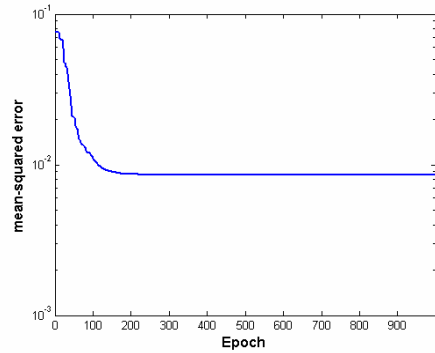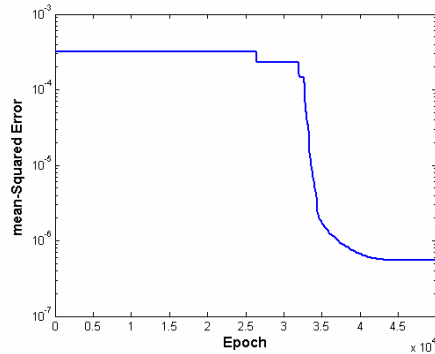

Fig. 3: Convergence curve of Newton algorithm.

Fig. 4: Convergence curve of ordinary PSO.

Obviously, the improved PSO is superior to ordinary PSO and Newton iteration method at precise and convergent velocity. So the improved PSO is practicable.

3. The simulation of diode

In order to show the effectiveness of the improved PSO, we simulate and analysis the more complicated device, diode.
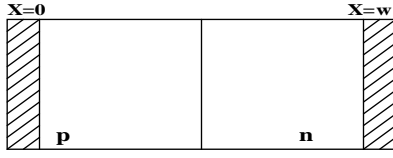
Fig. 5: Structure of p-n diode.

The effective impurity concentration equation is $\Gamma(x) = -N_A e^{-m_1 x^2} + N_B + N_D e^{=m_2(w-x)^2}$. Fig 6 shows the curve.
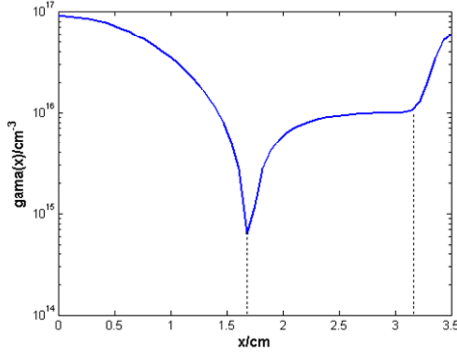

Fig. 6: Adulteration along x axis.

The solving process as follows epochs:

1) First, divide the diode along *x* axis. The main point is N = 1 to L corresponding to the p-n junction side point x= 0 and x = W. Furthermore, assistance point M is between N and N+1.Now let N=51.
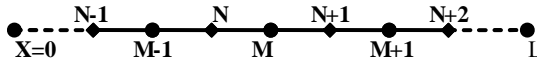

Fig.7 Compartmentalize along x axis

2) Calculate the differential coefficient of p、n and ψ at these assistance lattice point, to improve the differential approximate precision. The five equations can be written as:

$$J_p(M) = \frac{q}{h}[\lambda_{p1}(M)p(N) + \lambda_{p2}(M)p(N+1) \tag{11}$$

$$J_n(M) = \frac{q}{h}[\lambda_{n1}(M)n(N) + \lambda_{n2}(M)n(N+1) \tag{12}$$

$$\frac{1}{q} \cdot \frac{J_p(M) - J_p(M-1)}{h} - G(N) + U(N) = 0 \tag{13}$$

$$\frac{1}{q} \cdot \frac{J_n(M) - J_n(M-1)}{h} + G(N) - U(N) = 0 \tag{14}$$

$$\gamma_1(N)\psi(N-1) + \gamma_2(N)\psi(N) +$$
$$\gamma_3(N)\psi(N+1) = -\frac{q}{\varepsilon}[\Gamma(N) + p(N) - n(N)] \tag{15}$$

The Taylor expansion of p、n、ψ are neglected the high-order terms. The liner continuity equations come into being. At last, the equations can be:

$$A(N)\delta y(N-1) + B(N)\delta y(N) +$$
$$C(N)\delta y(N+1) = F(N), (0 \le N \le L-1) \tag{16}$$

3) Confirm the initial boundary conditions, $\delta y(1) = \delta y(L) = 0$, and initial value $y^0(N) = [p^0(N), n^0(N), \psi^0(N)]^T$, y^0 is the first iteration value, and then according to $y^1 = y^0 + \delta y$, get the value of δy with CPSO, and then get y^1, y^2 et al, till $\delta y/y < \varepsilon$ (ε =1e-10).

4) Analysis: matrix equation (16) shows the inner characteristic of diode.

Figure 8 shows the inner potential distribution curve ,which indicates the potential is smaller gradually from p to neutral region, but bigger from neutral region to n. In neutral region the potential is not smooth owing to the non-uniform adulteration.

Figure 9 shows the carrier concentration curve, in which curve 1 describes the concentration of p, curve 2 describes that of n by improved PSO algorithm; And curve 3 and 4 is the result of reference. We can see that in p region (x=0~1.703μm), the hole concentration is 9e+16(curve), electron concentration is 2.1778e+3; but in neutral region (x=1.703~3.2895μm) electron and hole are basically equal to 1e+10; in n region (x=3.2895~3.5μm), electron concentration is 5.9994e+016, hole concentration is 3.267e+3.

The simulation result is mostly accord with the theoretical analysis. So it shows the improved PSO is a good algorithm in semiconductor devices simulation.
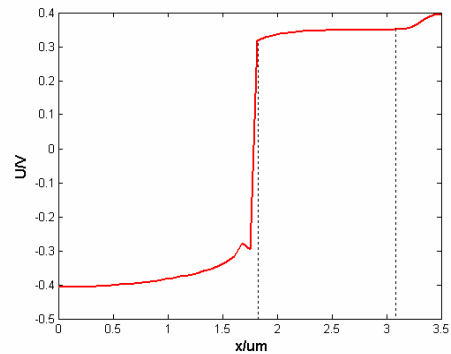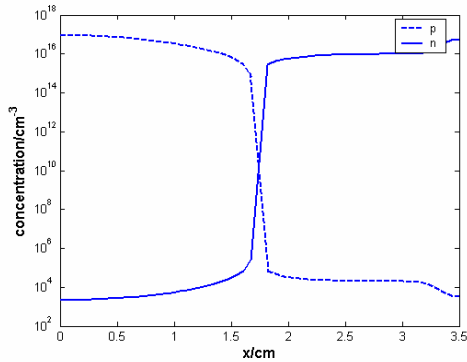

Fig. 8: Potential along x axis.

Fig. 9: Concentration of carriers along x axis.

4. Conclusion

The above actual simulation example shows that the improved PSO is superior to ordinary PSO and Newton iteration method in matrix equation solving, moreover, can be used in semiconductor devices simulation applicably and effectively. The further exploration to the improved PSO is to simulate the more complicated quantum devices, and the feasibility of imbedding in simulation software.

Acknowledgement

References

[1]   Y. He and T.L. Wei, The Computer Simulation Method of Semiconductor Devices, *Beijing: Publishing House of Science*, pp.1-18, 1989.

[2]   M. Shi, Semiconductor Devices Physics, *Beijing: Publishing House of Electronics Industr*, pp.41-48, 1987.

[3]   Z.C. Li, Y. L. Jiang and R. Z. Zhang, Neural Network Based Charge Density Quantum Correction of Nanoscale MOSFETs, *Chinese Journal of Semiconductors*, 27: 438-442, 2006.

[4]   J. Wu, L.F. Yang, Q. G. Liu, P. Wu, T. L. Wei, The Parallel Simulation of Semiconductor Devices, *Electronic Devices*, 22: 80-86, 1999.

[5]   J. Kennedy, R. C. Eberhart, Particle swarm optimization, *Prof. Of the IEEE Int. Conf. Neural Networks*, pp. 1942-1948, 1995.

[6]   Y. Shi, R. Eberhart, A modified particle swarm optimizer, *Prof. of IEEE World Congress on Computational Intelligence*, pp. 69-73, 1998.