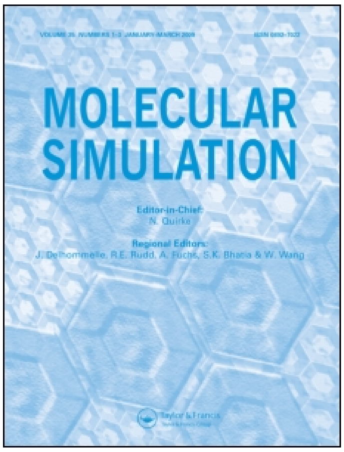

Molecular Simulation

Publication details, including instructions for authors and subscription information:
http://www.informaworld.com/smpp/title~content=t713644482

A New Efficient Method for Molecular Dynamics Simulation of Three-Body Potential Systems

Z. A. Rycerz[a][b]

[a] Central Research Laboratory, Hitachi Ltd., Tokyo, Japan [b] Department of Chemistry, The University of Western Ontario, London, Ontario, Canada

To cite this Article Rycerz, Z. A.(1995) 'A New Efficient Method for Molecular Dynamics Simulation of Three-Body Potential Systems', Molecular Simulation, 15: 6, 381 — 390

To link to this Article: DOI: 10.1080/08927029508022349

URL: http://dx.doi.org/10.1080/08927029508022349

Molecular Simulation, 1995, Vol. 15, pp. 381–390
Reprints available directly from the publisher
Photocopying permitted by license only

A NEW EFFICIENT METHOD FOR MOLECULAR DYNAMICS SIMULATION OF THREE-BODY POTENTIAL SYSTEMS

Z. A. RYCERZ[1]

Central Research Laboratory, Hitachi Ltd., Tokyo 185, Japan

(Received March 1995, accepted March 1995)

An efficient order of N molecular dynamics method for the simulation of two-body and three-body systems is presented. Due to its high speed it enables one to simulate large MD systems on a mainframe and investigate some complex processes in terms which are closer to experimental conditions (*e.g.* crystal growth or ion implantation on relatively large–by MD standards–substrates).

KEY WORDS: Three-body potentials, molecular dynamics, large systems.

1 INTRODUCTION

In recent years there has been remarkable interest in applying the molecular dynamics (MD) method to simulate covalent systems. Due to strong directionality between the bonds in such systems (*e.g.* Si, SiO_2, GaAs) they cannot be simulated by the use of a classical pair potential. Elaboration of phenomenological three-body, angle dependent, potentials [1–4] made it possible to use the MD technique for investigation of such kinds of materials. It is rather commonly believed that the MD approach is a costly one in computer resources terms. This is especially true when three-body potentials are used, since in a classical MD approach the cpu time for such potentials grows proportionally to N^3 [3]. As a result processes such as molecular beam epitaxy, ion implantation and crystal growth cannot be practically simulated by the use of a classical MD method. An example is the work [5] where the authors found they required 68 hours of Cray X-MP time to simulate the deposition process on a small silicon substrate of 14 × 16 atoms. The method described here makes it possible to perform a similar simulation in a few hours on a mainframe. However its actual value consists in the fact that it is a purely order of N method. Consequently, it enables one to simulate efficiently on a mainframe a system containing tens of thousands of particles.

---

[1] Permanent address: Department of Chemistry, The University of Western Ontario, London, Ontario, Canada N6A 5B7.

2 CALCULATION OF FORCES AND POTENTIAL ENERGY

The potential energy of a system involving two-body and three-body interactions is given by

$$\Phi = \sum_{i<j}\sum u_2(r_{ij}) + \sum_{i<j<k}\sum\sum u_3(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k) \tag{1}$$

where: u_2 denotes the two-body contribution $(r_{ij} = |\mathbf{r}_i - \mathbf{r}_j|)$ and u_3-the three-body contribution to the energy.

In a classical approach, the number of operations required for the calculation of the two-body term is proportional to N^2, while for the three-body term it is N^3. Consequently a simulation of three-body systems usually requires much more cpu time than the simulation of two-body systems. In order to solve the equations of motion it is necessary to calculate the resultant force acting upon each particle. For the potential as given by Eq. (1) the force acting on particle i equals

$$\mathbf{F}_i = -\sum_j{}' \frac{\partial u_2(r_{ij})}{\partial \mathbf{r}_j} - \sum_j \sum_k{}' \frac{\partial u_3(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k)}{\partial \mathbf{r}_i} \tag{2}$$

where primes denote omission of interactions with the same indices (i.e. $i \neq j$ ...etc).

Molecular dynamics simulation of a three-body system containing $N > 10^3$ particles typically requires several hours of a supercomputer time when performed by a classical (*i.e.* order of N^3 or even N^2) MD program. Therefore it is a very demanding task and special attention needs to be paid to the efficiency of calculation. Three issues are especially important in this regard. The first one is that for each particle i the interaction force with a particle j ($i-j$ pair in a two-body case) or with j and k ($j-i-k$ triad in a three-body case) needs to be calculated *only once*. Consequently the algorithm should be built up in such a way that the forces $j-i$, $k-j$ etc are never calculated, but instead are taken as the *reaction* forces from the $i-j$, $j-k$ etc interactions. The second one is related to equations used for the calculation of force components. Since the energy of the system must always be calculated in any MD simulation, it is useful to express the force components (2) in terms of potential energy (1). This is especially important for three-body forces, since the two-body forces can be easily tabulated and stored in look-up tables as they depend on the distance between the particles only. For example, the x component of the three-body forces acting on particle i that originates from particles j and k ($j-i-k$ triad) can be expressed as:

$$F_{xi} = f_{xi} \cdot v_{jik} \tag{3}$$

As far as the f_{xi} is a simple factor (or a combination of simple factors) the force components can be calculated immediately from the value of v_{ijk} potential and then very quickly. The expression for f_{xi} depends on the definition of a potential. In this regard one may consider that some potentials proposed in the literature are *more suitable* for MD purposes (*i.e.* they can be calculated with a greater speed) while the others are *less suitable*. The third issue is related to the range of three-body interactions. The main task of three-body forces is to preserve some favorable directions between the closest bonds and therefore their range is always short. Usually it is shorter than that for two-body forces or at most it is equal to latter one. It is important then to adapt an efficient method to select among all neighbours only those ones that are relevant (*i.e.* that contribute to the resultant force). An ideal solution would be to deal with only those neighbours that are contained in a cut-off sphere. In fact several very efficient MD methods that are close to meeting such a requirement have been published in recent years (*e.g.* [6–11]).

Some details concerning the three problems mentioned above are discussed in the following sections.

2.1 *Three-body forces and potentials*

For more detailed discussion we have chosen the Stillinger-Weber (S–W) potential [2] since it is widely applied in the simulation of three-body systems and additionally it seems to describe the lattice dynamics more reliably than the other potentials [12]. Originally the S–W potential was invented for silicon; however later its form has been adapted to several other tetrahedral systems. The arguments and conclusions given below have more general meaning and they can be applied to other forms of three-body potentials. The S–W potential is given in the form:

–two-body term

$$u_2(r_{ij}) = \varepsilon f_2(r_{ij}) \tag{4}$$

–three-body term

$$u_3(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k) = \varepsilon f_3(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k) \tag{5}$$

where ε is a constant (unit of energy) and r is given in a dimensionless form ($r_{ij} = r_{ij}^{ph}/\sigma$; r_{ij}^{ph} in physical units, σ–unit of distance). The two-body energy factor is given by

$$f_2(r) = A(Br^{-p} - r^{-q})\exp\left(\frac{1}{r - r_{c2}}\right) \qquad r < r_{c2} \tag{6}$$

$$f_2(r) = 0 \qquad r \geq r_{c2} \tag{6a}$$

with r_{c2} being the two-body cut-off radius. The three-body energy factor is equal to

$$f_3(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k) = h(r_{ij}, r_{ik}, \Theta_{jik}) + h(r_{ji}, r_{jk}, \Theta_{ijk}) + h(r_{ki}, r_{kj}, \Theta_{ikj}) \tag{7}$$

where Θ_{jik} is the angle between the $i-j$ (length of r_{ij}) and the $i-k$ (length of r_{ik}) bonds at vortex i, etc. Provided that both r_{ij} and r_{ik} are less than the three-body cut-off radius r_{c3}, the expression for h is as follows:

$$h(r_{ij}, r_{ik}, \Theta_{jik}) = \lambda g_\gamma(r_{ij}) g_\gamma(r_{ik})(\cos\Theta_{jik} - \cos\Theta_0)^2 \tag{8}$$

Otherwise h vanishes identically. The factor $g_\gamma(r)$ is given by

$$g_\gamma(r) = \exp\left(\frac{\gamma}{r - r_{c3}}\right) \tag{9}$$

The "ideal" tetrahedral angle for Si is $\Theta_0 = 109.47°$ ($\cos\Theta_0 = -1/3$) and the parameters used in the S–W potential are as follows:

$$\begin{aligned} &\varepsilon = 50\,\text{kcal/mol} = 2.169235\,\text{eV/atom}, && \sigma = 2.0951\,\text{Å}, \\ &A = 7.049556277, \quad B = 0.6022245584, && p = 4, \\ &q = 0, \quad r_{c2} = r_{c3} = 1.8, \quad \lambda = 21, && \gamma = 1.2 \end{aligned}$$

The two-body force acting on particle i from particle j is equal to:

$$\mathbf{F}_\mathrm{i} = -\frac{\partial u_2}{\partial \mathbf{r}_\mathrm{i}} = -\frac{\partial v_{ij}}{\partial \mathbf{r}_\mathrm{i}} = A\exp\left(\frac{1}{r - r_{c2}}\right)[Br^{-(p+1)} + (Br^{-p} - 1)(r - r_{c2})^{-2}]\mathbf{e}_\mathrm{ij} \tag{10}$$

where $\mathbf{e}_\mathrm{ij}$ is a unit vector ($\mathbf{e}_\mathrm{ij}/r_{ij}$) and a constant term ε in this and the following equations have been neglected.

The three-body force acting on particle i from the $i-j$ and $i-k$ bonds is equal to:

$$\mathbf{F}_\mathrm{i} = -\frac{\partial u_3}{\partial \mathbf{r}_\mathrm{i}} = -\frac{\partial v_{jik}}{\partial \mathbf{r}_\mathrm{i}} - \frac{\partial v_{ijk}}{\partial \mathbf{r}_\mathrm{i}} - \frac{\partial v_{ikj}}{\partial \mathbf{r}_\mathrm{i}} \tag{11}$$

where v_{jik}, v_{ijk} and v_{ikj} are potentials of the $j-i-k$, $i-j-k$ and $i-k-j$ triads.

According to the argument given above, only the first term in Eq. (11) needs to be calculated during a simulation. The other two terms ought to be included as the *reaction* forces from particles j and k respectively. The first term can be expressed as:

$$-\frac{\partial v_{jik}}{\partial \mathbf{r}_\mathrm{i}} = -\frac{\partial r_{ij}}{\partial \mathbf{r}_\mathrm{i}}\frac{\partial v_{jik}}{\partial r_{ij}} - \frac{\partial r_{ik}}{\partial \mathbf{r}_\mathrm{i}}\frac{\partial v_{jik}}{\partial r_{ik}} - \frac{\partial \cos\Theta_{jik}}{\partial \mathbf{r}_\mathrm{i}}\frac{\partial v_{jik}}{\partial \cos\Theta_{jik}} \tag{12}$$

where

$$\frac{\partial r_{ij}}{\partial \mathbf{r}_\mathrm{i}} = \mathbf{e}_\mathrm{ij}, \quad \frac{\partial r_{ik}}{\partial \mathbf{r}_\mathrm{i}} = \mathbf{e}_\mathrm{ik},$$

or in respect to $\mathbf{r}_\mathrm{j}$ and $\mathbf{r}_\mathrm{k}$ derivatives as:

$$-\frac{\partial v_{jik}}{\partial \mathbf{r}_\mathrm{i}} = \frac{\partial v_{jik}}{\partial \mathbf{r}_\mathrm{j}} + \frac{\partial v_{jik}}{\partial \mathbf{r}_\mathrm{k}} \tag{13}$$

Since

$$-\frac{\partial\cos\Theta_{jik}}{\partial\mathbf{r_i}}=\frac{\partial\cos\Theta_{jik}}{\partial\mathbf{r_j}}+\frac{\partial\cos\Theta_{jik}}{\partial\mathbf{r_k}} \tag{14}$$

with

$$\frac{\partial\cos\Theta_{jik}}{\partial\mathbf{r_j}}=\frac{1}{r_{ij}}\mathbf{e_{ik}}-\frac{\cos\Theta_{jik}}{r_{ij}}\mathbf{e_{ij}} \tag{15}$$

and

$$\frac{\partial\cos\Theta_{jik}}{\partial\mathbf{r_k}}=\frac{1}{r_{ik}}\mathbf{e_{ij}}-\frac{\cos\Theta_{jik}}{r_{ik}}\mathbf{e_{ik}} \tag{16}$$

we get the three-body force acting on particle i from the $i-j$ bond:

$$\mathbf{F_{ij}}=-\frac{\partial v_{jik}}{\partial r_{ij}}\mathbf{e_{ij}}-\frac{\partial v_{jik}}{\partial\cos\Theta_{jik}}\left[\frac{1}{r_{ij}}\mathbf{e_{ik}}-\frac{\cos\Theta_{jik}}{r_{ij}}\mathbf{e_{ij}}\right] \tag{17}$$

and the three-body force acting on particle i from the $i-k$ bond:

$$\mathbf{F_{ik}}=-\frac{\partial v_{jik}}{\partial r_{ik}}\mathbf{e_{ik}}-\frac{\partial v_{jik}}{\partial\cos\Theta_{jik}}\left[\frac{1}{r_{ik}}\mathbf{e_{ij}}-\frac{\cos\Theta_{jik}}{r_{ik}}\mathbf{e_{ik}}\right] \tag{18}$$

These two forces need to be added to a particle i and subtracted as the *reaction* forces from particle j (Eq. 17) and particle k (Eq. 18), respectively.

Particularly for the S – W potential we get:

$$\frac{\partial v_{jik}}{\partial r_{ij}}=-v_{jik}R^i_{ij}\quad\text{with } R^i_{ij}=\frac{\gamma}{(r_{ij}-r_{c3})^2} \tag{19a}$$

$$\frac{\partial v_{jik}}{\partial r_{ik}}=-v_{jik}R^i_{ik}\quad\text{with } R^i_{ik}=\frac{\gamma}{(r_{ik}-r_{c3})^2} \tag{19b}$$

$$\frac{\partial v_{jik}}{\partial\cos\Theta_{jik}}=v_{jik}D^i_{\Theta}\quad\text{with } D^i_{\Theta}=\frac{2}{(\cos\Theta_{jik}-\cos\Theta_0)^2} \tag{19c}$$

Therefore Eqs. (17) and (18) can be rewritten as:

$$\mathbf{F_{ij}}=v_{jik}\left[\left(R^i_{ij}+D^i_{\Theta}\frac{\cos\Theta_{jik}}{r_{ij}}\right)\mathbf{e_{ij}}-D^i_{\Theta}\frac{1}{r_{ij}}\mathbf{e_{ik}}\right] \tag{20}$$

$$\mathbf{F_{ik}}=v_{jik}\left[\left(R^i_{ik}+D^i_{\Theta}\frac{\cos\Theta_{jik}}{r_{ik}}\right)\mathbf{e_{ik}}-D^i_{\Theta}\frac{1}{r_{ik}}\mathbf{e_{ij}}\right] \tag{21}$$

It should be pointed out that all factors in the square brackets of Eqs. (20, 21) may be calculated rapidly since fragments of them have already appeared in the calculation of v_{jik} (cf. Eqs. (8, 9) with (19)). In addition some cpu costly cpu operations (like sqrt, exp) can be easily tabulated, for example, the values of R^i expressions (Eqs. 19a, 19b).

We have examined other three-body potentials proposed in the literature with respect to the *suitability* to the approach expressed by Eqs. (20) and (21). For example, we have found that the potentials formulated by Vessal *et al.* ([4, 13]) for SiO_2 and by Vashishta *et al.* [14] (also for SiO_2) are especially suitable for MD simulation (in all these cases the factors R^i and D^i_Θ have a very simple form). On the other hand the potentials for Si given by Biswas and Hamann [1] and Tersoff [3], although they have better physical foundation than, for example, the purely empirical S–W potential, are less suitable for this approach. In both cases ([1, 3]) the force components cannot be simply expressed in terms of v_{jik} potential and therefore both potential and force components need to be calculated *independent* to some extent independently, so the computation will take longer.

3 CALCULATION OF INTERACTIONS WITH THE NEAREST NEIGHBOURS

To achieve order of N dependency for cpu time, we incorporated the pyramid method [8, 9] which has appeared to be very suitable for a treatment of three-body interactions. In this method each particle i interacts only with those neighbours which are included in the nearest neighbour (NN) list. A new list is created (updated) every NTUPDA time step of simulation and it contains some surplus neighbours (*i.e.* being outside of r_{c2} or r_{c3} cut-off spheres at the time of creation of NN list). The number of surplus neighbours is determined by the size of the pyramid and the *skin* of surplus neighbours in the NN list. For two-body interactions (Eq. 2) we need to include only *forward* neighbours since the remaining interactions (for *backward* neighbours) are included as the *reaction* forces. For three-body interactions (Eq. 3) it is necessary to include both the *forward*

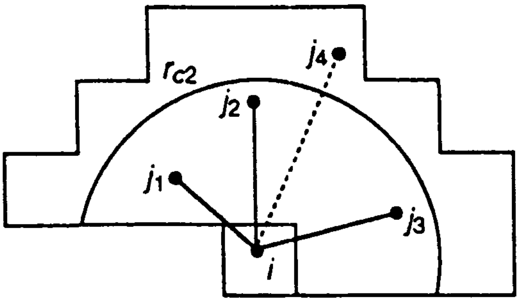

Figure 1 Schematic planar representation of two-body interactions. In this case only the *forward* pyramid is considered. The pyramid consists of several small sub-cells, the central sub-cell (containing particle i) is shown in the figure. Each central particle i interacts with its nearest neighbours ($j1, j2, j3$). The interaction with particle $j4$ (although included in the NN list) is excluded ($r_{ij} > r_{c2}$).

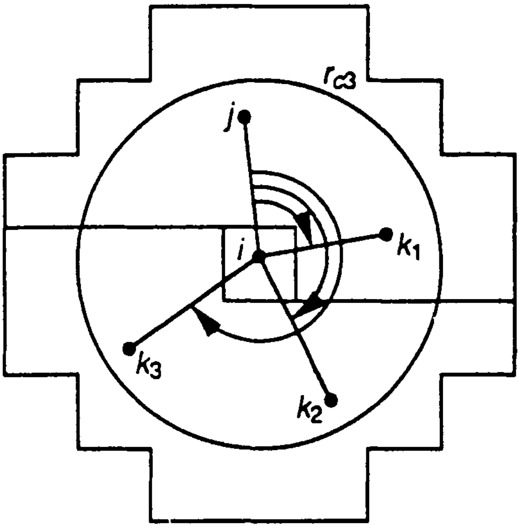

Figure 2 Schematic planar representation of three-body interactions. Both the *forward* (upper part) and *backward* (lower part) pyramids need to be considered. For the case shown in the figure the resultant force acting on a central particle i (Eqs. 20, 21) comes from six triads: for the $i-j$ bond the neighbours k_1, k_2 and k_3 are considered (marked by arrows); then k_1 becomes j (k_1-i-k_2 and k_1-i-k_3 triads) and finally the interaction for the k_2-i-k_3 triad is calculated.

and *backward*·neighbours. For this reason it is necessary to include two NN lists: *forward* and *backward*. The first one is used for both two-body and three-body interactions while the second one only for three-body ones. An interaction diagram for two-body forces is given in Figure 1, while that for three-body forces in Figure 2.

4 COMPARISON WITH THE EXISTING METHODS AND DISCUSSION

In the algorithm presented each particle in the system interacts with n_2 two-body neighbours and $(n_3-1)\cdot n_3/2$ "three-body neighbours" (for $r_{c3}=r_{c2}, n_3 \approx 2n_2$) and therefore the cpu time per time step of simulation can be expressed as:

$$t_{st} = [t_2 n_2 + t_3 n_3(n_3-1)/2] \times N \qquad (22)$$

where t_2 and t_3 represent cpu times for calculation of two-body (t_2) and three-body (t_3) forces for a single $i-j$ pair. Tests show that for the S–W potential $t_3 \approx 0.6 t_2$ and $n_2 \approx 4$, and therefore

$$t_{st} \approx 5 t_2 n_2 \times N \qquad (23)$$

In practice then, the simulation of a three-body system requires only about 5 times more cpu time than the simulation of a simple two-body system. What is, however, much more important is that for a three-body system also the cpu time is strictly proportional to the total number of particles N. On the Hitac M-880, a single processor mainframe, the speed of simulation for Si with the S–W potential is about 10^4 particles per cpu second and therefore the simulation of an $N = 10^5$ system requires about 10 cpu seconds per time step. For potentials which are *less suitable* for this approach (like for example, [1] or [3]) the cpu time is still proportional to N; however it can be 50% to 200% longer, depending on the potential used. The program based on the algorithm described is fast not only for large N but it is also much faster than other known programs for small N. Biswas *et al.* [15] report that during an hour of cpu time of a Cray X-MP their program performed 12×10^3 MD time steps for an $N = 216$ silicon system. In our case for the same system the program performs 180×10^3 time steps on a much slower mainframe. It should be pointed out here that the present version of the program has been optimized for a scalar processor and its acceleration on a vector processor is expected to be rather modest (order of 2 or so). On the other hand, its properly written vector version (*e.g.* according to the rules given in [10, 11]) would result in an acceleration order of 5 to 10 giving the speed of $(1-2) \times 10^6$ steps per cpu hour for an $N = 216$ silicon system on a supercomputer such as the Hitac S-820. The disadvantage of a vectorized version is that it is hardware dependent and therefore cannot be, in contrast to a scalar one, easily transfered between different vector computers.

Dodson and Taylor [16] concluded that an extension of their MD simulation (performed on a small silicon substrate) into actual simulation of crystal growth is not possible because of the amount of computer time required. Schneider *et al.* [5] reported they needed 68 hours of Cray X-MP time to deposit 2500 atoms on a 14×16 silicon substrate. We performed a simulation of the deposition process on such a subtrate, and in our case we needed only about 4 minutes of mainframe cpu time to deposit one monolayer containing 224 atoms during 2000 steps of simulation.

Furthermore by the use of the method described it is possible to investigate more complex processes under conditions that are closer to those existing in a real experiment. We have performed an MD simulation of the deposition process on a 1×2 Si(001) reconstructed surface of a large silicon substrate (ca. 200×200 Å). The substrate contained about 30×10^3 atoms. Depositing one full monolayer on such a huge system required on the Hitac M-880 mainframe 1.5 cpu hour at the deposition rate of one monolayer per 2000 MD steps (the results of that work will be published soon). A similar simulation by the use of a classical MD program (*e.g.* [5, 16]) is not possible since it would most likely take several months of Cray cpu time. An additional feature of the program is that it may calculate the long-range Coulomb interactions by the Ewald method [17] and therefore it can be used to simulate a system such as SiO_2. This part of the calculation is also of the order N [18], and when included the total cpu time grows by a factor order of two.

There is a discussion in the literature [15, 19] about the efficiency of MD versus MC (Monte Carlo) methods in the simulation of three-body systems. Some authors claim that in certain applications one method is more efficient than the other by an order of magnitude. Since the MD method presented here is orders of magnitude faster than other methods, we believe that this dispute is resolved in the favour of MD, at least for the time being.

5 SUMMARY

An efficient method for molecular dynamics simulation of three-body systems has been presented. Due to its high speed performance and $0(N)$ cpu time dependency, it opens new possibilities in the application of the MD technique to investigate complex processes in such materials like Si, GaAs and others that cannot be properly described by simple two-body spherical potentials. As far as the author is aware this is probably the first $0(N)$ MD method for three-body potentials described in the literature. The method seems to be particularly profitable whenever the examination of much larger MD systems, than those available by using classical MD methods, is required. Examples of this are phonon propagation in semiconductors [16], investigation of phase transition, structural and dynamical properties of stepped surface structures [20–23].

The program based on the method described has been recently adopted for the Pentium 90 MHz DOS micro computer. Its speed is 1600 particles per cpu second and therefore the simulation of a three-body system containing order of 300 or more particles requires less cpu time on a PC than on the Cray X-MP. It does not mean, of course, that the MD approach does not require powerful computers anymore, but it means that some less demanding tasks such as a simulation of relatively small systems (from hundreds to thousands of particles) can be performed in reasonable time on a PC or on a workstation.

Acknowledgements

I am grateful to the Hitachi Central Research Laboratory for the award of a visiting fellowship. I would like to express my gratitude to Dr. S. Itoh of the Hitachi Central Research Laboratory and to Mr. J. Kitakami of the Hitachi VLSI Engineering, Hitachi Ltd. for helpful discussions.

References

[1] R. Biswas and D. R. Hamann, "Interatomic potentials for silicon structural energies", *Phys. Rev. Lett.*, **55**, 2001 (1985).

[2] F. H. Stillinger and T. A. Weber, "Computer simulation of local order in condensed phases of silicon", *Phys. Rev.*, **B31**, 5262 (1985).

[3] J. Tersoff, "New empirical model for the structural properties of silicon", *Phys. Rev. Lett.*, **56**, 632 (1986).

[4] B. Vessal, M. Amini, D. Fincham and C. R. A. Catlow, "Water-like melting behaviour of SiO_2 investigated by the molecular dynamics simulation technique", *Phil. Mag.*, **60**, 753 (1989).

[5] M. Schneider, I. K. Schuller and A. Rahman, "Epitaxial growth of silicon: A molecular dynamics simulation", *Phys. Rev.*, **B36**, 1340 (1987).

[6] D. C. Rapaport, "Large-scale molecular dynamics simulation using vector and parallel computers", *Comput. Phys. Report*, **9**, 1 (1988).

[7] G. S. Grest, B. Dunweg and K. Kremer, "Vectorized link cell fortran code for molecular dynamics simulation for a large number of particles", *Computer Phys. Commun.*, **55**, 269 (1989).

[8] Z. A. Rycerz and P. W. M. Jacobs, "Molecular dynamics simulation program of order N I MDPYRS1: Scalar version, short-range interactions", *Computer Phys. Commun.*, **60**, 53 (1990).

[9] Z. A. Rycerz, "Molecular dynamics simulation program of order N, II. MDSPNL: Pyramid with neighbour list, short-range interactions", *Computer Phys. Commun.*, **61**, 361 (1990).

[10] Z. A. Rycerz and P. W. M. Jacobs, "A vectorized program of order N for molecular dynamic simulation of condensed matter, II. MDSLAB1: Slab, short-range interactions", *Computer Phys. Commun.*, **62**, 145 (1991).

[11] Z. A. Rycerz, "Acceleration of molecular dynamics simulation of order N", *Computer Phys. Commun.*, **60**, 297 (1990).

[12] E. R. Cowley, "Lattice dynamics of silicon with empirical many-body potentials", *Phys. Rev. Lett.*, **60**, 2379 (1988).

[13] B. Vessal, M. Leslie and C. R. A. Catlow, "Molecular dynamics simulation of silica glass", *Molecular Simulation*, **3**, 123 (1989).

[14] P. Vashista, R. K. Kalia, J. P. Rino and I. Ebbsjo, "Interaction potential for SiO_2: A molecular dynamics study of structural correlations", *Phys. Rev.*, **B41**, 12197 (1990).

[15] R. Biswas, G. S. Grest and C. M. Soukoulis, "Generation of amorphous-silicon structure with use of molecular dynamics simulation", *Phys. Rev.*, **B36**, 7437 (1987).

[16] B. W. Dodson and P. A. Taylor, "Interaction of 10 eV silicon beam with the Si(111) surface: A molecular dynamics study", *J. Mater. Res.* **2**, 805 (1987).

[17] P. P. Ewald, "Die Berechnung optischer und elektrostatischer Gitterpotentiale", *Ann. Physik*, **64**, 253 (1921).

[18] Z. A. Rycerz, "Calculation of the Coulomb interactions in condensed matter simulation", *Molecular Simulation*, **9**, 327 (1992).

[19] S. Clarke, M. R. Wilby and D. D. Vvedensky, "Theory of homoepitaxy on Si(001). I. Kinetics during growth", *Surface Sci.*, **255**, 91 (1991).

[20] O. L. Alerhand, A. N. Beker, J. D. Joannopoulos, D. Vanderbilt, R. J. Hammers and J. E. Demuth, "Finite temperature phase diagram of vicinal Si(100) surfaces", *Phys. Rev. Lett.*, **64**, 2406 (1990).

[21] N. C. Barnet, T. L. Einstein and C. Rottman, "First order transition between surface phases with different step structures", *Phys. Rev. Lett.*, **66**, 961 (1991).

[22] T. W. Poon, S. Yip, P. S. Ho and F. F. Abraham, "Equilibrium structures of Si(100) stepped surfaces", *Phys. Rev. Lett.*, **65**, 2161 (1990).

[23] E. Pehlke and J. Tersoff, "Phase diagram of vicinal Si(001) surfaces", *Phys. Rev. Lett.*, **67**, 1290 (1991).