ELSEVIER

Computational Materials Science 3 (1995) 448–456

COMPUTATIONAL MATERIALS SCIENCE

Molecular dynamics simulation of ion ranges in the 1–100 keV energy range

K. Nordlund

Accelerator Laboratory, University of Helsinki, P.O. BOX 43, FIN-00014, Finland

Received 9 September 1994; revised 17 November 1994; accepted 21 November 1994

Abstract

Binary collision approximation methods have been conventionally used to describe the slowing down of recoiling ions. In order to better understand the slowing-down process, molecular dynamics methods are more and more used in the literature. However, the computer capacity limits the usefulness of the methods in most practical cases where ion implantation in the 1–100 keV energy range is used. We present an efficient molecular dynamics method for calculating ion ranges and deposited energies in the recoil energy region 100 eV to 100 keV. By taking into account only the interactions that are involved in the slowing-down process, range and deposited energy distributions at higher energies can be simulated. The method is demonstrated by range calculations of 40 keV H atoms in Si, 40 keV He atoms in Ta and 100 eV to 10 keV Si atoms in Si.

1. Introduction

Ion implantation is commonly used in the fabrication of integrated circuits [1–3] and is a method for improving the wear behaviour of metals [4]. The doping depth is determined by the range of the implanted ions. The nature of the damage produced during implantation strongly affects the final properties of the specimen [5]. The damage production processes have been subject to extensive study in recent years, see e.g. Refs. [6–11].

The estimation of range and damage distributions is related to the description of the slowing down of energetic ions. To obtain detailed information of range and damage distributions in the low energy range (namely below about 100 keV) where elastic collisions dominate the slowing-down process, computer simulation methods have been developed [12]. Binary collision approximation (BCA) methods provide a fairly efficient means for calculating ion ranges, and have been used much since the 1960's [8,12–17]. Molecular dynamics (MD) methods describe the interactions involved in ion implantation more realistically, but require much larger amounts of computer capacity than BCA methods [18]. As computers have become more powerful, interest has started to shift towards MD methods [9,11,19]

In both BCA and classical MD simulations the interactions between atoms in the sample are described with an interatomic potential $V(\bar{r})$. If

SSDI 0927-0256(94)00085-9

the potential is assumed to depend only on the distance r between two atoms i and j, it can be written in the form

$$V_{ij}(r) = \frac{Z_i Z_j e^2}{r}\phi(r), \quad (1)$$

where the Coulombic term denotes the repulsion between the bare nuclei and the function $\phi(r)$ the screening of the nuclei due to the electron cloud [13]. The electronic stopping is taken into account as a frictional force.

In the forthcoming sections we will point out some problems of BCA simulations in calculating ion ranges, and illustrate why ordinary MD methods are not suited for calculations of range distributions of high-energy ions. We proceed by presenting new principles for calculating ion ranges and show that our method overcomes several of the drawbacks of both BCA and ordinary MD simulations.

2. Problems in binary collision approximation methods

In BCA calculations the movement of the atoms in the implanted sample is usually treated as a succession of individual collisions between the recoil ion and atoms in the sample. For each individual collision the classical scattering integral is solved by numerical integration [17].

The impact parameter used in the scattering integral is determined either from a stochastic distribution (in programs like TRIM [12,20]) or in a way that takes into account the crystal structure of the sample [17]. The former method is suitable only in simulations of implantation into amorphous materials.

Although the BCA methods have been successfully used in describing many physical processes, they have some obstacles to describing the slowing-down process of energetic ions realistically. Owing to the basic assumption that collisions are binary, problems arise when trying to take multiple interactions into account [13,17]. Also, the form of the scattering integral does not allow for incorporating angle-dependent potentials, which are necessary to describe covalently bonded materials like silicon [21,22]. These factors make it difficult to describe collision cascades realistically in BCA simulations.

3. Problems in molecular dynamics simulations

In molecular dynamics simulations the time evolution of a system of atoms is calculated by solving the equations of motion numerically. In the Newtonian formalism the force $\bar{F}_i$ acting on an atom i in the system is calculated as

$$\bar{F}_i(\bar{r}_i) = \sum_{j \neq i} \bar{F}_{ij}(r_{ij}) = -\sum_{j \neq i} \nabla V_{ij}(r_{ij}), \quad (2)$$

where $\bar{F}_{ij}$ is the force acting between atoms i and j and V_{ij} (r_{ij}) is the potential energy function given in Eq. (1). The sum over j is taken over all atoms whose interaction with atom i is stronger than a threshold value $V_{\min}$ [23].

By incorporating a sum in Eq. (2) that depends on the positions of three or more atoms, one can include angular-dependent potentials in MD calculations [21].

When the forces F_i have been calculated for all moving atoms in the system, the equations of motion for the system are solved using some suitable algorithm, for instance the one given in Ref. [24]. The solution yields the change in the atom positions, velocities and accelerations over a finite time step Δt. The smaller the time step is, the more accurate is the solution of the equations of motion. After the changes have been performed the process is continued by calculating the forces in the new positions.

Thus, contrary to BCA methods, all interactions experienced by an atom are taken into consideration simultaneously in MD simulations.

Despite their wide use in other fields of physics, MD simulations have been very little used for calculating ion ranges at keV and higher energies. The main reason is that the MD methods used so far require very much larger amounts of computer capacity than BCA calculations. One reason for this is that the BCA approach of numerically solving the scattering integral requires fewer calculation steps than solving the equations of motion.

Another reason is that when calculating ion ranges with BCA simulations, it is sufficient to calculate the motion of the recoil atom. In conventional MD simulation methods (see e.g. Refs [9,11,18,25,26]) the movements of all energetic ions have been calculated. Recently some schemes where weak interactions are neglected have been presented when treating collision cascades [27,28].

In calculating ion ranges and deposited energies at high energies ($\gtrsim 1$ keV), the interactions between the recoil ion and its nearest neighbours are much stronger than the interactions between lattice atoms. It would therefore seem natural to treat only the recoil ion interactions. This approximation is hereafter called the recoil interaction approximation (RIA).

In the next section we will describe our MD method for calculating ion ranges using the RIA and relatively few atoms in the simulation cell.

4. Method for calculating ranges of high energy ions using MD simulations

To calculate ranges of ions with keV and higher energies we have modified the computer code MOLDY used previously in our laboratory to describe the slowing down of low-velocity atoms [29]. The modified code has been successfully used for range calculations in earlier publications [7,30], but to date no detailed description of the range calculation method has been given.

For efficiency the MOLDY range code uses a small simulation cell for the range calculation. However, the MOLDY code does not employ the RIA.

For performing RIA calculations a new program, MDRANGE [31], has been written. It was designed specifically for high-energy MD calculations and leaves out many redundant features in the original MOLDY code. Unless otherwise stated, the features described below apply both to the MOLDY and MDRANGE programs.

Our simulation algorithms are based on the conventional Newton formalism described above. A Verlet neighbour list [32] is employed to make the simulations more efficient. During each time step, the interactions of all atoms with their neighbours in the simulation cell are calculated. In the MDRANGE program one can also choose to use the RIA during a range calculation.

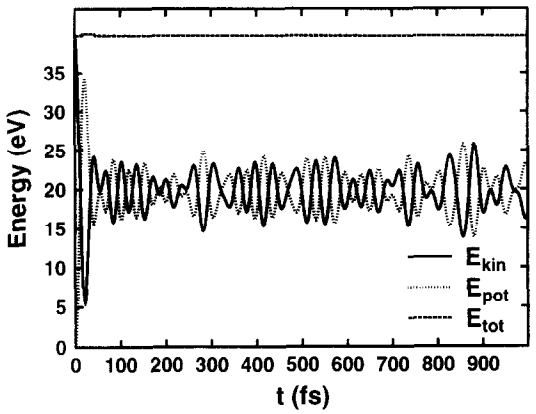

Fig. 1. Conservation of energy during an initial displacement calculation in the MDRANGE program.

To obtain realistic thermal displacements of the atoms in the sample an initial displacement calculation is performed at the beginning of each range calculation. The atom positions in the simulation cell are initialized to a crystalline or amorphous structure. The atoms are given initial velocities in random directions according to the Maxwell velocity distribution for a desired simulation temperature T. A simulation of thermal movement is performed to yield realistic displacements from the initial sites. The simulation is carried out until the average temperature taken over the last 100 times steps yields the desired temperature T (within the error bounds).

In the initial displacement calculation periodic boundary conditions [23] and a constant time step are used. The calculation can also be used to test the stability of the solution of the equations of motion. In Fig. 1 the kinetic, potential and total energy in a simulation cell of 512 silicon atoms is shown for a 1000 fs simulation of movement at 300 K. The time step employed in the simulation was 1 fs. The simulation was performed with the MDRANGE program.

If all interactions between the lattice atoms are calculated during the actual range calculation, an attractive interatomic potential is employed to treat the interactions between lattice

atoms. In the RIA lattice–lattice interactions are not calculated. The lattice atoms which do not interact with the recoil atom are held fixed at the positions they obtained in the initial displacement calculation. When they start to interact with the recoil atom they are given the velocity vector they obtained in the initial displacement calculation.

The interaction between the recoil atom and the lattice atoms is always described by a two-body repulsive potential.

In addition to the RIA, two features in our method differ significantly from typical MD simulations: the selection process of the time step, and the treatment of the simulation cell borders.

4.1. Time step

During the actual range ("recoil event") calculation, when drastic changes occur in the maximum atom velocity in the system, a variable time step Δt is employed to speed up the simulations. Three criteria are used to determine Δt. Firstly, the time step is made inversely proportional to the recoil velocity v using a proportionality constant k_t. This approach is used e.g. in Ref. [24]. However, detailed analysis of very strong collisions in the keV energy range showed that the solution of the equations of motion did not describe strong collisions realistically if the time step was calculated from k_t alone. To obtain smaller time steps in strong collisions, the time step is also made inversely proportional to the product of the total force F the recoil atom experiences and its velocity v using a proportionality constant E_t. Since large increases of the time step make the solution of the equations of motion inexact [24], the value of the time step is never allowed to increase more than 10% from its previous value.

The final time step is determined from the criterion yielding the smallest value, that is

$$\Delta t_{\text{new}} = min\left(\frac{k_t}{v}, \frac{E_t}{Fv}, 1.1\Delta t_{\text{old}}\right). \tag{3}$$

With smaller values of k_t and E_t the error in the solution of the equations of motion gets smaller. On the other hand, the simulations get slower with small values of k_t and E_t, so it is advantageous to use the biggest possible values of E_t and k_t that do not affect the simulation results.

4.2. Simulation cell

In practical MD calculations the criterion that the potential between two atoms must be above a threshold value of V_{min} for the interaction to be taken into account is replaced by defining a critical radius r_0 so that $V(r_0) \leq V_{\text{min}}$. The interaction between two atoms is included in the calculations only if their distance from each other is less than r_0. For typical repulsive interatomic potentials, like the commonly used ZBL (universal) repulsive potential [12], r_0 typically has a value between 2 and 3 Å.

The most important criterion for selecting the size of the simulation cell during a recoil event calculation is that all atoms nearer than r_0 to the recoil atom must be present at all times during the simulation. Since the radius r_0 usually has a value around 3 Å, a simulation cell with a side length of 10–15 Å is large enough to contain all atoms that interact with the recoil atom. This typically amounts to a cell containing 50–100 atoms.

Because of efficiency considerations it is advantageous to keep the simulation cell as small as possible. On the other hand it is quite clear that a cell with a side length of 10–15 Å cannot contain the entire path of an implanted ion in the keV energy range that may move several hundreds or thousands of ångstroms in the implanted sample. Therefore, a mechanism for ensuring that the recoil atom is always surrounded by lattice atoms is needed. Using conventional periodic boundary conditions would not be realistic, since the recoil atom would move in a simulation cell damaged by its own previous motion.

We have developed a translation method for ensuring that the recoil atom moves in a structure unaffected by its previous motion. The method is similar to translation techniques used in BCA simulations [17], but has to our knowledge not been used earlier in MD simulations together with the RIA.

In our approach critical distances R_{S_i} are defined for the three space coordinates $i = 1,2,3$. If

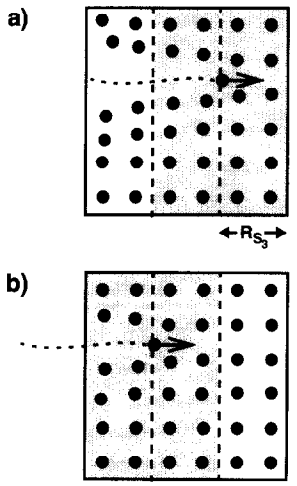

Fig. 2. Schematic two-dimensional view of the way atoms are moved during the calculation of ion ranges. In the figure a two-dimensional view of the simulation cell is shown before (a) and after (b) the atoms have been moved. In (a) the recoil atom (which is marked with a grey interior) has come closer than R_{S_3} to the simulation cell border. When this has happened, all atoms within the shaded area are moved away from the cell border, and new atoms that have not been affected by the ion movement are placed in front of the recoil atom.

the recoil atom during the simulation gets nearer than R_{S_i} to the simulation cell border, for any coordinate i, then the recoil atom and all atoms surrounding it are moved this distance R_{S_i} away from the simulation cell border. The moved atoms retain their previous velocities and accelerations. The moving of atoms creates an empty region in the simulation cell "in front of" the recoil atom, which is filled with atoms whose positions and velocities are taken from an initial displacement calculation described above. The process is illustrated in Fig. 2.

In this way movement of the recoil atom can be simulated in an arbitrarily large sample without including more than a few hundred atoms in the simulations.

The size of R_{S_i} must obviously be $\geq r_0$ and less than half the size of the simulation cell. In simulating crystalline materials it is often advantageous to select R_{S_i} to equal the size of the unit cell or some integer fraction of the unit cell size.

Open boundary conditions are applied to the cell to prevent cascade damage from spreading back in the simulation cell over the borders.

4.3. Structure of the sample

The simulation codes MOLDY and MDRANGE support several different sample structures. The atom coordinates of all atoms except the recoil atom are read in from a file at the beginning of the simulation. Crystalline and amorphous materials can be simulated by using perfect lattice positions or an amorphous structure obtained e.g. from ab initio MD calculations [33]. Any number of different isotopes or elements can be included in the structure.

In simulating polycrystalline materials the grain size is calculated using a Gaussian distribution from an average grain size and a value defining the variance from the average size. During the recoil event calculation, each time the grain boundary is reached the recoil atom is injected into a new grain. The orientation of each grain is selected randomly. In this way polycrystalline nanostructures can be simulated.

The program also supports simulation of multilayered structures with an arbitrary number of layers. For each layer the atom coordinates are read in from a separate file, and the layer is given a depth region ($z_{\min}$, $z_{\max}$) in which the recoil atom will enter the layer. The only restriction on the different layer structures is that the size of the simulation cell must always remain the same.

Several layer structures can be placed in the same z region, and given different probabilities for occurring. The type of the atoms within a layer structure can be selected randomly. These features allow simulation of ion implantation even into very complex sample structures.

In this paper results of simulations in amorphous, crystalline and polycrystalline structures are presented (cf. Section 5).

4.4. Recoil event calculation

The simulation of one recoil event is initiated by placing the recoil atom a few Å outside the simulation cell, and giving it the desired energy and velocity direction. The initial position of the atom is selected randomly. The surface of the simulation cell is set at $z = 0$; no moving of the cell atoms is performed until the z coordinate of

the recoil atom has become greater than R_{S_3}. The movement of the recoil atom is then followed until it has slowed down below some threshold energy, e.g. 1 eV. The movement of the simulation cell atoms is stored during the calculation, and taken into account when the final range of the recoil atom is calculated.

The electronic stopping S_e is incorporated in the simulations by subtracting the recoil atom velocity with

$$\Delta v = \Delta t \frac{S_e}{m} \tag{4}$$

(where m is the ion mass) at each time step.

To obtain a reliable range distribution, the ranges of at least a few thousand recoils are simulated.

To obtain a theoretical estimate of primary damage the nuclear and electronic deposited energies $F_{D_n}(z)$ and $F_{D_e}(z)$, respectively, are also calculated during the range simulation. The energy losses of the recoil atom are evaluated for each time step and stored in arrays as a function of the depth. The nuclear energy loss is calculated by subtracting the electronic energy loss obtained from Eq. (4) from the total energy loss of the recoil ion. The final deposited energy distributions are obtained by taking the average of the distributions of each individual recoil event.

5. Test of the method and discussion

To test our simulation method a number of implantation simulations were carried out. The ions and ion energies were selected to test implantation at ion velocities encompassing two orders of magnitude. Different pairs of ions and sample structures were selected to test implantation of both light and heavy ions into both light and heavy backings. All the implantations used for the tests have been previously studied in our laboratory [7,25,30,34]. The simulations are listed in Table 1.

For the implantations in silicon (numbers 1, 2 and 4–6 in the table) the angle of incidence was

Table 1
Range results for the various simulations. The error given for the range is the statistical error of the simulation, and does not take into account the uncertainties of the nuclear and electronic stoppings. The MOLDY program was used to perform full MD simulations, whereas the MDRANGE program used the RIA. The efficiency factor in the last column tells how many events were calculated in one CPU hour on a Decstation 3100 using the Ultrix operating system. Since the TRIM91 program is usually run on a PC platform, the TRIM efficiency factor is given for simulations performed on an IBM PC-compatible 386SX computer with a 33 MHz clock speed

Implantation	Program	Mean range (Å)	Straggling (Å)	Effect (events/CPUh)
1. 10 keV ^{30}Si → a-Si	MOLDY	172 ± 3	93 ± 2	81
(10 keV ≙ 2.53 Å/fs)	MDRANGE	170 ± 1	87 ± 1	1200
	TRIM	185 ± 1	82 ± 1	4700
2. 40 keV ^{1}H → c-Si	MOLDY	4580 ±20	727 ± 7	2.9
(40 keV ≙ 27.8 Å/fs)	MDRANGE	4555 ± 6	738 ± 3	51
	TRIM	4140 ±10	664 ± 3	6300
3. 40 keV He → Ta	MOLDY	1540 ±20	810 ±10	2.5
(40 keV ≙ 13.9 Å/fs)	MDRANGE	1490 ±20	760 ±10	46
	TRIM	1280 ±10	602 ± 3	1400
4. 1 keV ^{30}Si → c-Si	MOLDY	31.2 ± 0.2	21 ± 0.1	550
(1 keV ≙ 0.80 Å/fs)	MDRANGE	32.5 ± 0.3	22 ± 0.2	7400
5. 300 eV ^{30}Si → c-Si	MOLDY	11.6 ± 0.1	6.3 ± 0.1	830
(300 eV ≙ 0.44 Å/fs)	MDRANGE	12.1 ± 0.1	6.2 ± 0.1	13000
6. 100 eV ^{30}Si → c-Si	MOLDY	5.09 ± 0.03	2.50 ± 0.01	900
(100 eV ≙ 0.25 Å/fs)	MDRANGE	5.35 ± 0.02	2.29 ± 0.01	30000

tilted 6° from the [001] crystal direction against the [010] direction. The initial position of the recoil atom was selected randomly over a square with a side length of 5.43 Å to account for all possible incidence positions in a unit cell in silicon. The square was located 2.7 Å above the cell surface.

The initial atom coordinates in implantation 2 were the coordinates for amorphous silicon (a-Si) given in Ref. [33].

During the initial displacement calculation the time step Δt was given the constant value 2 fs. During the range calculation Δt was determined from Eq. (3) with $k_t = 0.1$ Å and $E_t = 300$ eV. These values we obtained from a number of test simulations where k_t and E_t were varied.

The structure of Ta in implantation 3 was assumed to be polycrystalline with a grain size much larger than the implantation range. Polycrystallinity was modeled by selecting the implantation angles randomly (the polar angle θ between 0 and 45°, the azimuthal angle ϕ between 0 and 360°) to describe implantation in all crystal directions. The interatomic potential and electronic stopping employed for all the simulations were those given by Ziegler et al. in Ref. [12].

To give an impression of how much the results of our method and conventional BCA methods differ, the high-energy test simulations (implantations 1–3) were also carried out with the TRIM program commonly used to calculate ion ranges [20]. The program uses the same electronic stopping and repulsive interatomic potential as the MD simulations. Since the version of the TRIM program used (TRIM91) does not take into account the crystal structure of the sample, the results are not directly comparable. Inspection of Table 1 shows that the MD and TRIM results differ by up to 20%.

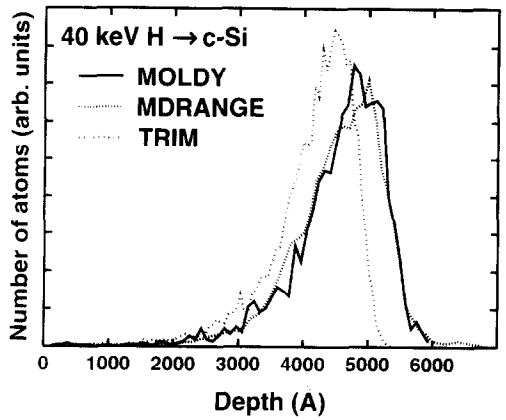

Fig. 4. MD and BCA range profiles of 40 keV H implantation in c-Si.

5.1. Test of the RIA method

The validity of the RIA was tested by comparing RIA results to results of full MD calculations (where all interactions are taken into account). In Figs. 3–5 and Table 1 the results of RIA simulations have been compared with results from full MD simulations. The results from the full MD

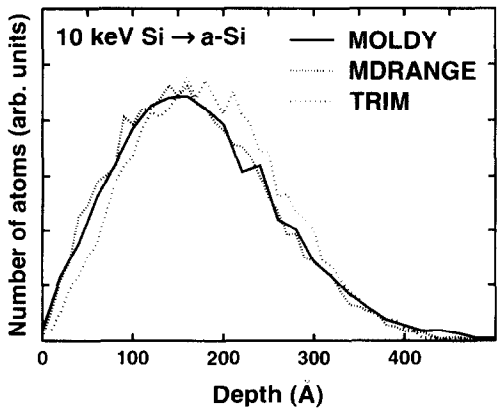

Fig. 3. MD and BCA range profiles of 10 keV Si implantation in a-Si.

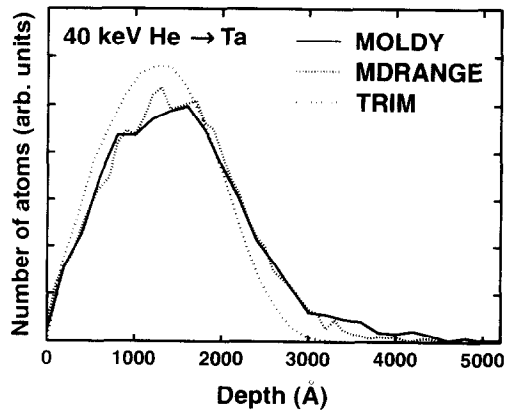

Fig. 5. MD and BCA range profiles of 40 keV He implantation in Ta.

simulations in Figs. 3 and 4 have been successfully compared with experimental range profiles in earlier works at our laboratory, see Refs. [7,30].

A very good correspondence between the full MD and RIA results can be seen in the figures and table, indicating that the RIA method is valid at keV energies.

At lower energies the interaction energy between the recoil ion and lattice atoms approaches the energies by which the lattice atoms interact with each other. Therefore one would expect the RIA to break down at sufficiently low energies. Simulations of implantation of ^{30}Si into c-Si were performed to find out the low energy limit of the validity of the RIA. The results are given in Table 1. It was found that RIA simulations give practically the same results as full MD calculations at implantation energies as low as 100 eV. Below about 300 eV there is a statistically significant difference in the mean range, but since it is less than one fourth of the interatomic distance 2.35 Å, the difference is of very little physical importance.

5.2. Calculation of deposited energies

The calculation of deposited energies using the TRIM91 and MDRANGE programs is demonstrated in Fig. 6. The range and deposited nuclear energy distributions for implantation of 10 keV ^{30}Si in amorphous silicon are shown. The deposited nuclear energy is the energy deposited to primary knock-on atoms by the recoil atom. The deposited energy calculated using the MDRANGE program shows a distinct peak near the sample surface which is not visible in the profile calculated with the TRIM program. The total nuclear deposited energies per implanted ion were 8400 and 7500 eV in the MDRANGE and TRIM simulations, respectively.

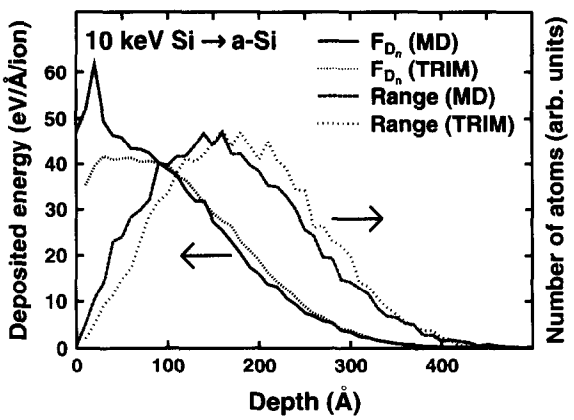

Fig. 6. Deposited energies and range profiles for implantation of 10 keV ^{30}Si in a-Si calculated with the TRIM and MDRANGE programs.

The difference shows how the different ways of describing amorphous silicon lead to differences in primary damage distributions.

5.3. Efficiency of the simulations

In the last column of Table 1 efficiency figures are given for the simulations. The numbers show that the use of the RIA in the MD simulations is highly advantageous. The RIA simulations performed with MDRANGE are on the average 15 times faster than the corresponding MOLDY simulations. The TRIM program, although run on a slower computer (see the table caption) still performs far better than the MD simulations.

Besides the obvious advantage of shorter calculation times, the RIA method also increases the range of physical problems that can be treated with MD simulations. For instance, implantation at high energies which are much used in the semiconductor industry [1] can now be simulated realistically. Also, the size of the implanted sample is not limited in our method.

We also believe that our method can be developed further to treat, for instance, full collision cascades efficiently. Some provisions have already been made in writing the MDRANGE code to enable it to be expanded to calculate the movement of primary knock-on atoms.

6. Conclusions

We have presented the first practical method based on molecular dynamics simulations to calculate ion ranges and deposited energies in the 1–100 keV energy range. The method combines cell translation techniques and the RIA within

the framework of molecular dynamics simulations.

We calculated range distributions of 40 keV H implantation of Si, 40 keV He implantation of Ta and 100 eV–10 keV Si implantation of Si with the MD method. The results showed that one can ignore interactions between lattice atoms when calculating the distribution of primary damage during ion implantation.

We also presented principles by which one can simulate implantation in polycrystalline or complexly damaged materials.

We conclude that our algorithms offer the most realistic practical method for calculating the distribution of primary damage in keV ion implantations.

Acknowledgement

This work was supported by a grant from the University of Helsinki.

References

[1] J. Asher, Nucl. Instr. Meth. Phys. Res. B 89 (1994) 315.
[2] B. Holländer et al., Nucl. Instr. Meth. Phys. Res. B 84 (1993) 218.
[3] M.-A. Hasan et al., J. Appl. Phys. 65 (1989) 172.
[4] D.M. Rück, D. Boos and I.G. Brown, Nucl. Instr. Meth. Phys. Res. B 80/81 (1993) 233.
[5] J. Narayan, D. Fath, O.S. Oen and O.W. Holland, J. Vac. Sci. Technol. A 2 (1984) 1303.
[6] J.K.N. Lindner, Nucl. Instr. Meth. Phys. Res. B 84 (1994) 153.
[7] J. Keinonen et al., Nucl. Instr. Meth. Phys. Res. B 88 (1994) 382.
[8] N. Hayashi et al., Phys. Rev. Lett. 70 (1993) 45.
[9] A.M. Mazzone, Nucl. Instr. Meth. Phys. Res. B 83 (1993).
[10] T.A. Tombrello, Nucl. Instr. Meth. Phys. Res. B 83 (1993) 508.
[11] T.D. de la Rubia and M.W. Guinan, Phys. Rev. Lett. 66 (1991) 2766.
[12] J.F. Ziegler, J.P. Biersack and U. Littmark, in: The Stopping and Range of Ions in Matter (Pergamon, New York, 1985) Vol. 1.
[13] K. Gärtner, M. Nitshke and W. Eckstein, Nucl. Instr. Meth. Phys. Res. B 83 (1993) 87.
[14] L. Leblanc and G.G. Ross, Nucl. Instr. Meth. Phys. Res. B 83 (1993) 15.
[15] R.G. Vichev and D.S. Karpuzov, Nucl. Instr. Meth. Phys. Res. B 83 (1993) 345.
[16] J.L. Klatt, R.S. Averbach, D.V. Forbes and J.J. Coleman, Phys. Rev. B 48 (1993) 17629.
[17] M.T. Robinson and I.M. Torrens, Phys. Rev. B 9 (1974) 5008.
[18] M. Hou and Z. Pan, Nucl. Instr. Meth. Phys. Res. B 90 (1994) 469.
[19] T. Diaz de la Rubia et al., Nucl. Instr. Meth. Phys. Res. B 80/81 (1993) 86.
[20] J.P. Biersack and L.G. Haggmark, Nucl. Instr. Meth. 174 (1980) 257.
[21] F.H. Stillinger and T.A. Weber, Phys. Rev. B 31 (1985) 5262.
[22] J. Tersoff, Phys. Rev. B 37 (1988) 6991.
[23] D.W. Heermann, Computer Simulation Methods in Theoretical Physics (Springer, Berlin, 1986).
[24] R. Smith and D.E. Harrison Jr., Computers in Physics Sep/Oct 1989 (1989) 68.
[25] K. Nordlund, J. Keinonen and A. Kuronen, Physica Scripta (1994), to be published.
[26] H. Zhu and R.S. Averbach, Nucl. Instr. Meth. Phys. Res. B 83 (1993) 334.
[27] R. Smith, J. Don E. Harrison and B.J. Garrison, Phys. Rev. B 40 (1989) 93.
[28] M. Sayed, J.H. Jefferson, A.B. Walker and A.G. Gullis, Nucl. Instr. Meth. Phys. Res. B (1994), to be published.
[29] J. Keinonen et al., Phys. Rev. Lett. 67 (1991) 3692.
[30] P. Torri, J. Keinonen and K. Nordlund, Nucl. Instr. Meth. Phys. Res. B 84 (1994) 105.
[31] The full C language source code for the MDRANGE program is available from the author. E-mail requests should be directed to kai.nordlund@helsinki.fi.
[32] L. Verlet, Phys. Rev. 159 (1967) 98.
[33] D.A. Drabold, P.A. Fedders, O.F. Sankey and J.D. Dow, Phys. Rev. B 42 (1990) 5135.
[34] P. Haussalo, J. Keinonen, U.-M. Jäske and J. Sievinen, J. Appl. Phys. 75 (1994) 7770.