TASK QUARTERLY 14 No 1–2, 35–167

MOLECULAR DYNAMICS SIMULATIONS OF ULTRAPRECISION MACHINING OF FCC MONOCRYSTALS

MONIKA RYCHCIK-LEYK[1], MICHAŁ BIAŁOSKÓRSKI[2],
JACEK DZIEDZIC[1] AND JAROSŁAW RYBICKI[1,2,3]

[1] *Department of Solid State Physics,*
Faculty of Technical Physics and Applied Mathematics,
Gdansk University of Technology,
Narutowicza 11/12, 80-233 Gdansk, Poland
jaca@kdm.task.gda.pl

[2] *TASK Computer Centre,*
Narutowicza 11/12, 80-233 Gdansk, Poland

[3] *Institute of Mechatronics, Nanotechnology and Vacuum Techniques,*
Koszalin University of Technology,
Racławicka 5–17, 75-620 Koszalin, Poland

(Received 8 December 2009; revised manuscript received 28 March 2010)

Abstract: In technical sciences, the term "machining" refers to the process of forming an object into a desired shape and size, with a desired quality of surface, by removing layers of its material by means of a cutting tool. The paper describes research on ultra-precision machining (UPM), where the abovementioned process takes place on the atomic level and involves systems (a machined object and a tool) several dozen nanometers in size. Three-dimensional computer simulations (virtual experiments) of UPM of monocrystalline copper with an infinitely hard tool were performed utilizing the classical molecular dynamics (MD) method with a many-body potential to describe the interatomic interactions. Among the examined issues were the effect of the tool shape, machining speed and depth on the obtained workmaterial surfaces, and on the stresses, slip patterns and local temperature increases generated during the process.

Keywords: nanomachining, nanocutting, nanomechanical properties, nanoplasticity, molecular dynamics

1. Introduction

Investigations of the relations between the structure of solids at the atomic level and their macroscopic mechanical properties were commenced with the advent of modern solid-state physics, understood as quantum mechanics of systems with translation symmetry. However, a detailed analytical description of mechanical properties may not extend beyond a relatively narrow class of systems with a high symmetry of both the macroscopic shape and relatively simple atomic structures. It was only the development of computational physics that made it possible to overcome these difficulties and achieve specific results for complex systems. One of the basic methods of computational physics, the so called "particle method", has been widely applied to investigate the properties of solids for more than forty years, complementing numeric calculations under the continuum theory. The method is most often commonly (not very aptly) called the molecular dynamics or MD method. The basic input data for MD calculations of mechanical properties of solids is the thorough knowledge of the initial structure (at zero external forces) and the nature of the interatomic interactions. In order to gather such data, reference must be made to the advanced atomic resolution structural research techniques on the one hand, and to quantum mechanics and chemistry on the other hand, required to describe the breaking and reorganization of chemical bonds.

Elastic properties of crystalline and amorphous solids have been investigated thoroughly using the MD method over the last 40 years. However, the first significant work related to the MD simulation of plastic properties is probably [1]. At present, the MD method makes it possible to investigate such nanomechanical properties of solids as the mechanisms of cracking, fatigue, degradation of the crystalline structure, phenomena of friction, surface scratching or nanomachining. The last of the abovementioned phenomena is the subject of this work.

Earlier nanomachining simulation algorithms will be briefly reviewed in Section 2 and the software used in this work will be described in Section 3. Sections 4 and 5 contain descriptions and discussion of the simulation results. The brief Section 6 contains conclusions and final comments.

2. Modelling of nanomachining with the molecular dynamics method

In this section, the methods and algorithms used in simulations of nanomachining by other authors will be discussed.

2.1. ARMD algorithm

In the original version of the ARMD (Area-Restricted MD) method [2], the motion of atoms was determined by integrating Newton's equations of motion. The force acting on atom i was the sum of the interactions originating from other atoms:

$$\boldsymbol{F}_i = -\sum_{j=1\,(j\neq i)}^{N} \nabla_i \phi(r_{ij}) \tag{1}$$

where ϕ was the Morse potential:

$$\phi(r_{ij}) = D\exp[-2\alpha(r_{ij}-r_0)] - 2\exp[\alpha(r_{ij}-r_0)] \tag{2}$$

The equations of motion had the following form:

$$\frac{d^2\boldsymbol{r}_i(t)}{dt^2}=\frac{\boldsymbol{F}_i(t)}{m_i},\qquad i=1,2,\ldots,N \tag{3}$$

The positions and velocities of the atoms at subsequent time steps were calculated according to the Verlet algorithm:

$$\boldsymbol{r}_i(t+\delta t)=\boldsymbol{r}_i(t)+\delta t\boldsymbol{v}_i(t)+\frac{(\delta t)^2}{2m_i}\boldsymbol{F}_i(t) \tag{4}$$

$$\boldsymbol{v}_i(t+\delta t)=\boldsymbol{v}_i(t)+\frac{\delta t}{2m_i}\left[\boldsymbol{F}_i(t+\delta t)+\boldsymbol{F}_i(t)\right] \tag{5}$$

where m_i denotes the mass of atom i, $\boldsymbol{F}_i$ – the force acting on the i^{th} atom, $\boldsymbol{r}_i$ – the position of the i^{th} atom, $\boldsymbol{v}_i$ – its speed, N – the total number of atoms, t – time, δt – the time step, D – the cohesion energy, α – the modulus of elasticity, r_0 – the equilibrium interatomic distance.

Initial velocities were drawn from the Maxwell-Boltzmann distribution and selected in such a way that the system's constant temperature was maintained in accordance with the relation:

$$\boldsymbol{v}_i^{\text{new}}=\sqrt{\frac{N_f NkT_0}{2}\left[\sum_{i=1}^{N}\frac{m_i(\boldsymbol{v}_i^{\text{old}})^2}{2}\right]^{-1}}\,\boldsymbol{v}_i^{\text{old}} \tag{6}$$

where T_0 was the system temperature, N_f – the number of degrees of freedom, k – the Boltzmann constant.

Figure 1 shows the set-up in the ARMD model. Three regions comprising various atom types can be distinguished:

- atoms from the simulation area are those taking part in the simulation and for which the above equations of motion are solved;
- thermostat atoms are responsible for the dissipation of heat from the simulation area during machining, thus keeping it at a constant temperature. The velocity of these atoms is scaled at each time step in accordance with Equation (6), where N is the number of thermostat atoms;

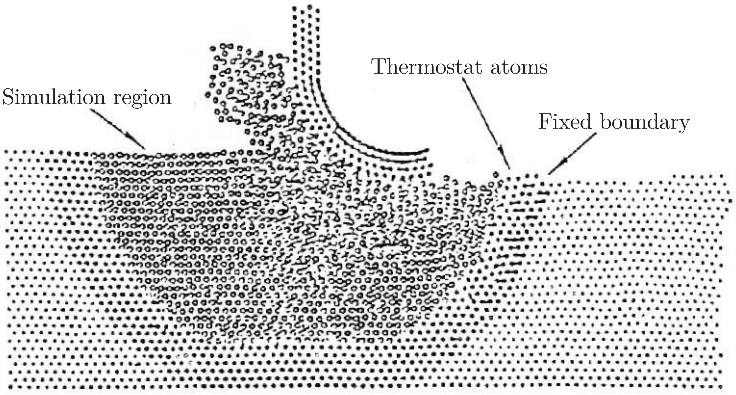

Figure 1. Overview of the simulation set-up in the ARMD method [2]

– immobilized atoms which do not take part in the simulation.

An important feature of the method is that a limited area is selected around the machining tool and the atoms in that area move together with the tool in the direction of machining. In this way, the number of atoms taking part in the simulation is reduced, keeping the simulation time and required memory manageable, which was the intention of the creators of this method.

2.2. LRMD algorithm

In the LRMD (Length-Restricted MD) method the simulated system composed of a workmaterial and a machining tool contained N atoms with the coordinates (x_i, y_i, z_i), where $i = 1,2,\ldots,N$ [3]. Similarly to the ARMD model, the machined material and the tool were divided into three regions. The set-up of the model is shown in Figure 2. The motion of atoms in the mobile region was calculated by the direct solution of the Hamilton equations, and the velocities, accelerations, momenta and forces for specific atoms were calculated according to the following relations:

$$\boldsymbol{v}_i(t) = \dot{\boldsymbol{r}}_i(t) \tag{7}$$

$$\boldsymbol{a}_i(t) = \ddot{\boldsymbol{r}}_i(t) \tag{8}$$

$$\boldsymbol{p}_i = m_i \boldsymbol{v}_i \tag{9}$$

$$\boldsymbol{F}_i = \boldsymbol{F}_{i1} + \boldsymbol{F}_{i2} + \boldsymbol{F}_{i3} + \cdots + \boldsymbol{F}_{iN} = \sum_{j=1\,(j\neq i)}^{N} \boldsymbol{F}_{ij} \tag{10}$$

where $\boldsymbol{r}_i$ is the position of the i^{th} atom, m_i – its mass, $\boldsymbol{v}_i$ – its speed, $\boldsymbol{p}_i$ – its momentum, $\boldsymbol{a}_i$ – its acceleration, and $\boldsymbol{F}_{ij}$ – the force exerted on the i^{th} atom by the j^{th} atom.

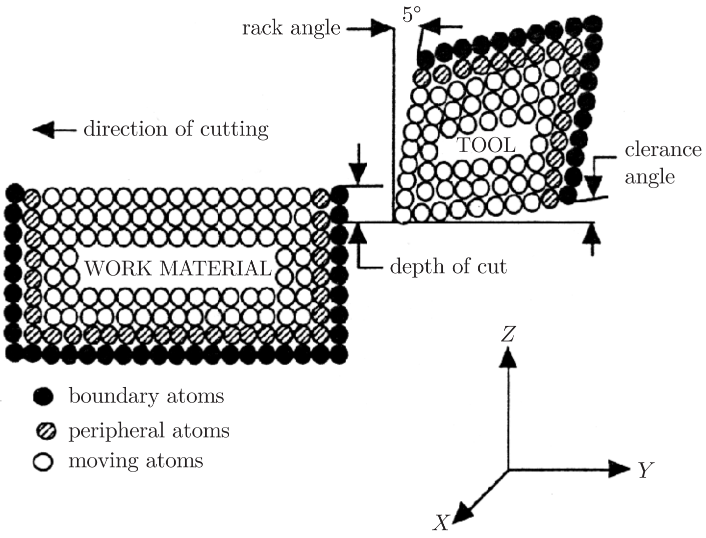

Figure 2. Overview of the simulation set-up in the LRMD algorithm [3]

Similarly as in standard MD, the force F_{ij} was calculated as a derivative of the interaction potential (which was the Morse potential):

$$\boldsymbol{F}_{x_i} = -\frac{\partial V_i}{\partial \boldsymbol{X}_i} = \dot{\boldsymbol{P}}_{x_i} \tag{11}$$

$$\dot{\boldsymbol{X}}_i = \frac{\boldsymbol{P}_{x_i}}{m_i} \tag{12}$$

where V is the potential and $\boldsymbol{X} = [x,y,z]$.

The motion of atoms from the peripheral area, determined by solving the same Hamilton equations, was modified by a velocity reset function which acted on each atom of the peripheral area. The function represented the influence of the volume effects on the energy transfer occurring in the extended lattice model. The function had the form:

$$\boldsymbol{v}_{\boldsymbol{X}_i}^{\mathrm{new}}(t_n) = (1-w)^{1/2}\,\boldsymbol{v}_{\boldsymbol{X}_i}^{\mathrm{old}}(t_n) + w^{1/2}\boldsymbol{v}^r(\xi,T)$$

where

$$w = \frac{p_D\, t\, \omega_D \Gamma_p}{3}$$

and $\boldsymbol{v}_{\boldsymbol{X}_i}^{\mathrm{new}}(t_n)$ was a new $\boldsymbol{X}^{\mathrm{th}}$ component of velocity for the i^{th} atom at t_n, and $\boldsymbol{v}_{\boldsymbol{X}_i}^{\mathrm{old}}(t_n)$ – the old component of speed, $\boldsymbol{v}^r(\xi,T)$ – a random velocity drawn from the Boltzmann distribution at temperature T with a random number x_i, w – a parameter that controls the intensity of scaling, assuming values from 0 (no scaling) to 1 (scaling with disregard of previous velocities). Analogous action is taken for the other two velocity components.

The boundary atoms were motionless and served to reduce the surface effects and maintain the appropriate lattice symmetry.

LRMD is another method (adaptation of the classic MD) in which an attempt was made to reduce the calculation time and memory requirements. The main idea was to introduce a constant length of the processed material which was moving in accordance with the direction of machining during the simulation. In this way, it was possible to simulate systems in which the workmaterial was small in size (contained a small number of atoms), and the distances on which the machining process was observed could be infinitely long.

The idea was realized in the following way. Atoms in the machined material part (chip) have no influence on the further progress of the simulation, and therefore, they are stored in a separate array. However, their positions (coordinates) are stored in the computer's memory and used to add new atoms to the front of the workmaterial, simulating its infinite length (Figure 3).

In this way, new atoms reuse the same memory cells as they have the same coordinates (X and Z) as the rejected atoms, it is only their Y coordinates that are changed by adding an appropriate value equal to the length of the machined surface.

The machined material has a constant length, however the positions of atoms change along the direction of the machining. Owing to this, it is possible

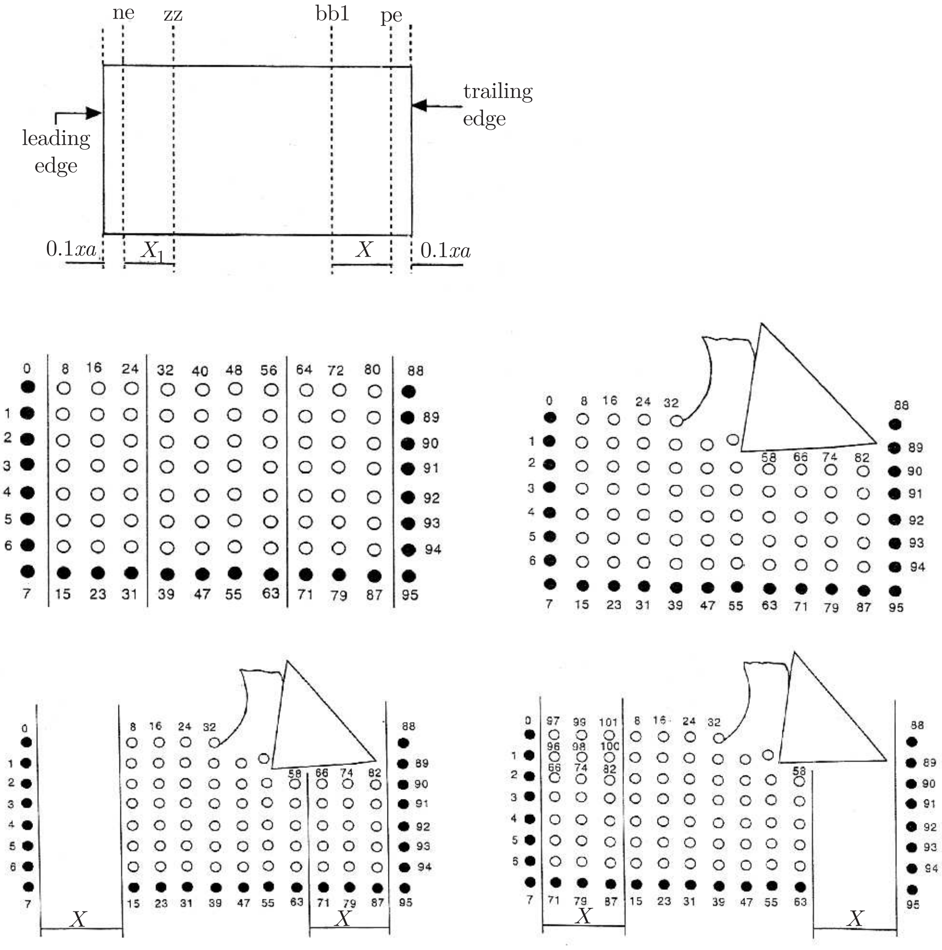

Figure 3. Principle of the LRMD method [3]

to reduce the time of calculations and the occupied memory, without limiting the number of atoms in the simulated system.

3. The *nanoMD* program and utility programs

The *nanoMD* program which is the basic simulation tool used in this work was written in C++ by M. Białoskórski in the years 2002–2005. Many utility programs serving to automate the generation the input data and the analysis of the results obtained with the *nanoMD* program were subsequently created. The main simulation program, the manner of preparing the input data and the applied methods of analyzing the simulation results will be briefly described in subsequent sections.

3.1. The nanoMD program

The *nanoMD* program comprises the main block and four auxiliary blocks:

- The main block is responsible for integrating the equations of motion and calculating the internal forces, and contains functions driving all the auxiliary blocks.
- The first auxiliary block is a database containing information about the types of atoms taking part in the simulation, relating masses and charges of specific atom types.
- The second auxiliary block contains parameterization of the potentials used to describe interactions between specific atom types.
- The task of the third auxiliary block is to ascribe external forces to atoms.
- The last of the auxiliary blocks is used when calculating short-range forces to find the closest neighbours of each atom.

Integration of the equations of motion proceeded via the Gear algorithm of the 4^{th} order, which belongs to a group of methods known as "predictor-corrector". The particle state is described by the vector:

$$\boldsymbol{x}_i(t)=\begin{pmatrix} x_i(t)\\ h\dot{x}_i(t)\\ \frac{1}{4}\ddot{x}_i(t)\\ \frac{1}{4}\ddot{x}_i(t-h)\end{pmatrix} \tag{13}$$

where h denotes the integration step. In the first approximation, the vector $\boldsymbol{y}_i(t+h)$ of the anticipated positions and their derivatives is calculated:

$$\boldsymbol{y}_i(t+h)=\mathbf{A}\,\boldsymbol{x}_i(t) \tag{14}$$

where:

$$\boldsymbol{y}_i(t+h)=\begin{pmatrix} y_i(t+h)\\ h\dot{y}_i(t+h)\\ \frac{1}{4}\ddot{y}_i(t+h)\\ \frac{1}{4}\ddot{y}_i(t)\end{pmatrix},\quad \mathbf{A}=\begin{pmatrix} 1 & 1 & \frac{4}{3} & -\frac{1}{3}\\ 0 & 1 & 3 & -1\\ 0 & 0 & 2 & -1\\ 0 & 0 & 1 & 0\end{pmatrix} \tag{15}$$

Then, the force in the next integration step $\boldsymbol{y}_i(t+h)$ is calculated from the above predicted particle position $f(\boldsymbol{y}_i(t+h))$, and the appropriate particle position vector related to the next integrating step is calculated according to the relation:

$$\boldsymbol{x}_i(t+h)=\boldsymbol{y}_i(t+h)+\boldsymbol{a}\frac{1}{2}h^2[f(\boldsymbol{y}_i(t+h))-\ddot{\boldsymbol{y}}_i(t+h)] \tag{16}$$

where

$$\boldsymbol{a}=\begin{pmatrix} \frac{1}{6}\\ \frac{5}{6}\\ 1\\ 0\end{pmatrix} \tag{17}$$

This method is fairly complicated, however its stability and accuracy make it possible to utilize a larger integration step for the equations of motion (compared to other integrators), which allows for shorter simulation walltime.

The initial velocities required to start the simulation are drawn from the Maxwell-Boltzmann distribution. In order to calculate the forces acting on specific atoms, the gradient of the potential is calculated directly (*i.e.* it is not calculated from the functional form of the potential) which speeds up the calculations significantly. The following potentials have been implemented to describe the interatomic interactions:

- the Sutton-Chen potential for atoms of *fcc* metals;
- the Finnis-Sinclair potential for atoms of *bcc* metals;
- the Morse potential for interactions between atoms of *fcc* and *bcc* metals.

The many-body Sutton-Chen potential which describes long-range effects well and belongs to class of the Finnis-Sinclair potentials [4] was used in the simulations described in this work. It was designed to model interatomic interactions in ten *fcc* metals, including but not limited to copper, aluminum and lead, and in particular to describe the interactions between atom clusters. The total energy is calculated according to the relation [5]:

$$H^{\mathrm{SC}} = \epsilon \sum_i \sum_{j>i} V(r_{ij}) - c \sum_i \sqrt{\rho_i} \tag{18}$$

where:

$$V(r_{ij}) = \left(\frac{a}{r_{ij}}\right)^n \tag{19}$$

$$\rho_i = \sum_{j \neq i} \left(\frac{a}{r_{ij}}\right)^m \tag{20}$$

where a is the lattice constant, ϵ – an energy-dimension parameter, and m, n are positive integer constants subject to the condition $n > m$. Tabulated values of the abovementioned parameters can be found in [5]. The Sutton-Chen potential was successfully used to calculate elasticity constants, cohesion energy and bulk modulus.

The *nanoMD* program allows to ascribe to each atom (or atom group) an external interaction (which is not interatomic). These interaction include:

- external force;
- external velocities;
- immobilization of atom (or atom group);
- external torque.

This creates wide possibilities of simulating such non-equilibrium processes as tension, torsion, scratching, machining or crushing of samples fixed in any given way with the help of a tool of any geometry, orientation and motion with respect to the workmaterial.

3.2. Parallelization of the nanoMD code

The initial performance tests performed on a serial workstation indicated the necessity to parallelize the *nanoMD* code. The mean calculation time for one simulation step on a serial computer was approximately 15 seconds, and the time needed to perform a single machining simulation was 250 hours. Having in mind the fact that approximately 70 of such simulations were to performed, the necessity to adapt the *nanoMD* program to a parallel processing environment became obvious. Also, the prospect of the installation of a 64-bit Itanium-2 computer of the TASK Computer Center in early 2004 made the authors revise the program code with respect to its potential incompatibilities with the 64-bit architecture.

In the first stage, the program was profiled to identify the algorithm "bottlenecks" which needed to be parallelized first. In accordance with expectations, it turned out that most of the computational workload was related to the calculation of forces. This was especially true in the case of calculations performed using the Sutton-Chen potential, where a relatively large cut-off radius and a many-body nature of the potential enforced the calculation (for each atom) of contributions to the force originating from approximately 500 neighbouring atoms. However, it was unexpectedly found out that the workload related to the identification of neighbours (Hockney's cell linked list algorithm) was only slightly smaller, or even predominant for larger cut-off radii (not used in this work). The reasons for such behavior were twofold.

The first reason is of an algorithmic nature – the algorithm requires the calculation of distances between atoms many times to identify each of the cubic cells of atoms located inside the sphere with the center in the atom whose neighbours we are trying to find and to reject atoms outside the sphere and inside the cube defining the cell. The second reason is a consequence of the cache memory model used in typical computers. Extraction from a list of neighbours of atom indices and coordinates participating in calculating the forces, results in a large number of chaotic, non-contiguous accesses to the random access memory (in general the neighbours of a given atom do no have subsequent indices). However, in typical computers the cache is optimized in such a way so as to ensure a high throughput with contiguous access.

As a result, the assignment of neighbours becomes a process limited by memory throughput, resulting in the processor's idle waiting for data from the memory. Having Amdahl's law in mind, the author decided to parallelize not only the program fragments most costly in terms of calculation but practically the whole code except for the most trivial operations, devoting special attention to the abovementioned two sections. Geometric decomposition of data and assignment of particles to processors were applied. This means that when performing calculations on N particles on k processors, each processor is assigned $\lceil N/k \rceil$ particles which will be hereinafter called local particles of the processor. If N was not divisible by k, a slightly smaller number of particles was assigned to the last processor. The particles were assigned on the basis of their indices, *i.e.* the first $\lceil N/k \rceil$ of particles was assigned to the first processor, subsequent $\lceil N/k \rceil$ of particles – to the second one, *etc.* An advantage of such an approach is simplicity and the possibility of reducing the calculation walltime by increasing locality – if care is taken that particles in the input file are grouped in such a way that subsequent indices are assigned to particles which are close in terms of topology the program's performance will improve. The possibility of data replication (a technique where the complete set of information on all particles is replicated on all processors) was abandoned in order to save precious memory.

In the adopted approach, each processor stores relevant information (position, velocity, acceleration, mass) only for local particles. This approach enables work with very large systems (up to 10^7 particles) which was not possible in the serial version due to memory limitations of the machine (the memory requirements have not changed, however, they are merely distributed evenly onto k processors). The result of assigning particles to processors is relatively well balanced operation (each processor is occupied with the same number of particles), taking good advantage of locality and simplicity of implementation compared to alternative strategies (assigning interactions to processors or cells to processors). In the adopted approach, each processor calculates forces acting only on local particles.

The contributions to the calculated forces originate mostly from local particles. The positions of particles belonging to neighbouring processors which may be required to calculate other contributions are sent collectively by means of MPI at each simulation step. Although the messages sent may even be several MB in size (for 10^6 particles and 20 processors), the communication overhead is small compared to the calculation time which is several to several dozen seconds per step in such large systems. The application of very fast InfiniBand (10 Gb/s) interconnect is the reason why the mentioned overhread is practically imperceptible.

When parallelizing the program, the adopted execution environment was MIMD with distributed memory and messages sent using the MPI standard which made it possible to disregard the details of communication between processors while maintaining high portability of the code. The force calculation module (*force*), the velocity scaling to temperature, the linked cell method (*cell*), the statistical calculation module (*stat*), determination of the stress tensor (*stress*), the module taking into account external forces (*ext_force*) and the module integrating the equations of motion (*engine*) now operate on a fully parallel basis. The data replication strategy was only adopted for global observables (those that represented properties of the whole system and not individual particles). For instance, for the calculataion of total kinetic energy, each processor calculates the contribution to energy originating from local particles, whereafter the inputs are summed ("reduced") across processors, and the sum is broadcast to each processor. As a result, each processor has a copy of global observables which simplifies certain algorithms (*e.g.* velocity scaling). In order to further increase performance, an attempt was made to parallelize the input/output by using the MPI-2 mechanisms allowing the parallel reading and saving of working files. However, difficulties of an objective nature were encountered, resulting from the network file system used on the calculation cluster. During the tests it turned out that the network file system was not able to correctly synchronize the attributes of files with a parallel access with operation on a greater number of processors (over twenty) which resulted in unexpected deadlocks. Finally, I/O was reverted to serial operation, pending a filesystem update.

All modifications were made in such a way that the program retained the capability of serial operation – the operation mode (serial or parallel) is selected during compilation by a single directive and the C++ preprocessor selects appropriate code sections, depending on whether the user has defined a certain symbol or not. The number of processors on which the program will be run (for parallel operation) does not have to be known at compilation time. From the user's point of view, running the program in the parallel version does not differ from the existing serial operation, except for the obvious necessity to submit the tasks to the batch (queueing) system. However, a certain nuance should be noted here. As floating point addition in finite precision arithmetic is not associative but depends on the order in which the terms are summed, very minor errors related to the machine's ultimate precision are made when summing the results across processors (*e.g.* when calculating total kinetic energy, pressure or force). Under these circumstances, the specific values of errors depend both on the number of processors on which the program is run, and on the order of summation which cannot be controlled in the MPI environment. It should be taken into account that the program, in formal terms, gives different results depending on how many processors it is run on. The differences are, naturally, microscopic. However, the committed relative errors are smaller than 10^{-9}, unless values close to zero are considered (*e.g.*, the abovementioned total force which should be equal to zero, could be in the order of 10^{-24} nN as a result of rounding errors, and the specific error value depends on the number of processors). This phenomenon is widely recognized and considered harmless.

As all calculations were performed using the Sutton-Chen potential, the author decided to optimize the calculation of forces corresponding to that potential. In particular, when calculating the generalized atomic density – the SIMD (vectorization) mechanisms of the Itanium-2 computer were used to a high degree (on the Intel x86 as well, albeit to a smaller degree), delegating arithmetic operations on long vectors to functions defined in the Intel MKL library which can perfectly take advantage of the pipelining mechanisms offered by the CPU. The module implementing the linked cell method enabling the user to define how often the neighbour lists are refreshed has been also optimized by allowing the possibility to refresh neighbour lists every several steps, which makes it possible to increase performance at the cost of minor deterioration in the conservation of the total energy.

The prospect of running the program on a cluster of 64-bit Itanium-2 machines enforced a precise revision of the code in terms of potential incompatibility of the program with the 64-bit architecture. A detailed line-by-line inspection of the code and runs under the supervision of the *valgrind* suite led to several bugfixes in the program which had prevented its correct operation in a 64-bit environment at an earlier time (sloppy conversion of indices into integers, ungrounded assumptions related to sizes of integer types). The program was successfully tested.

The speed-up $S(N)$ achieved owing to the program parallelization, defined as $S(N) = t_1/t_N$, where t_1 is the walltime on one processor, and t_N – the

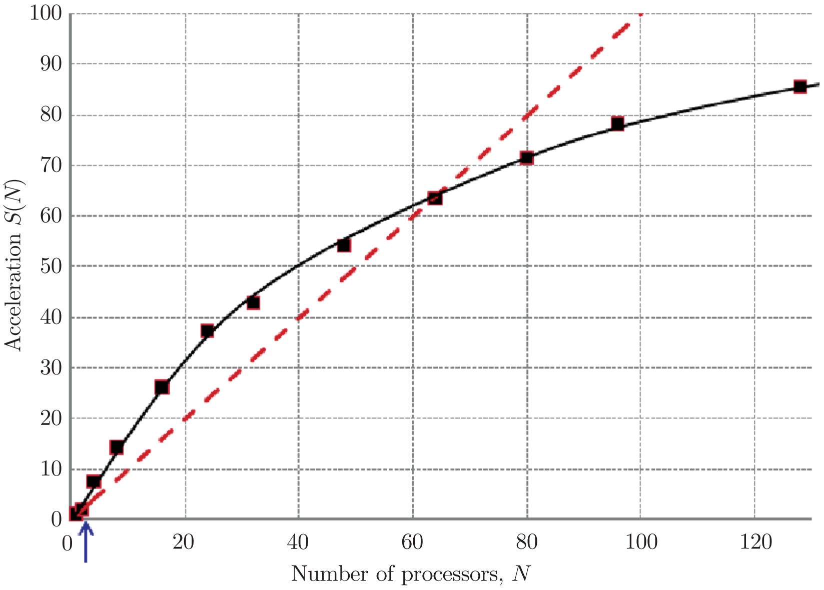

Figure 4. Code speed-up achieved owing to code parallelization for a medium-sized system (approximately 60 000 atoms). The black points represent results of measurements, the red dashed line – linear acceleration. The black line serves as a guide for the eye only. The blue arrow shows the number of processors above which the current program data fully fit in the processor's cache

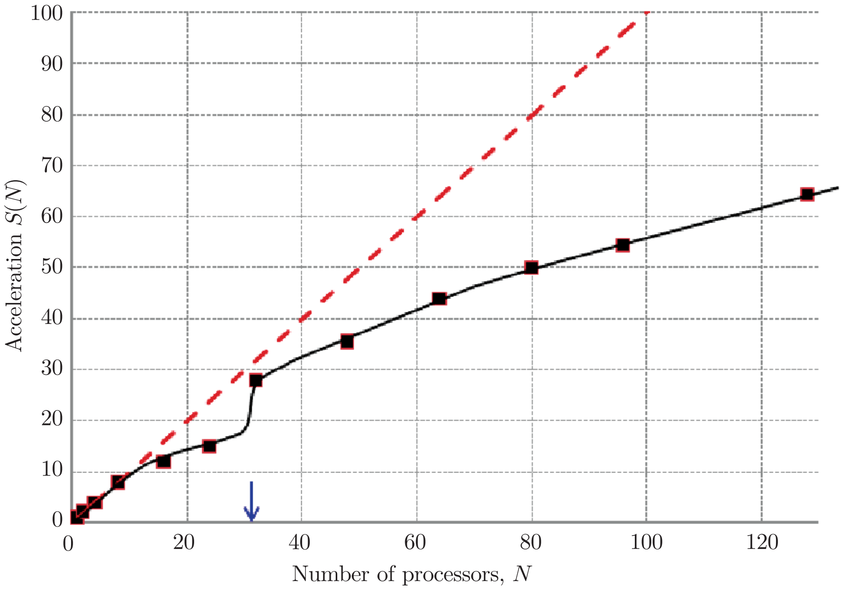

Figure 5. As in Figure 4 but for a large system (approximatelly 800 000 atoms)

walltime on N processors, was measured. Figures 4 and 5 present the results of measurements for a TASK supercomputer for two systems differing in size. The former is the 100A2 system used in machining simulation (approximately 60 000 atoms), described later in this work, the latter is a system used in the a tool degradation simulation (approximately 800 000 atoms), not discussed further in this work.

The results of measurements are in accordance with expectations. For a smaller system with a small number of processors (fewer than 64) the program achieves superlinear acceleration due to the fact that for three and more processors the working set is fully contained in the processor's cache memory, whereas the computation is inefficient (for that specific system) on one and two processors as a result of the fact that the data must be copied to and from the random access memory.

For more than 64 processors the speed-up saturates due to Amdahl's law – this is a consequence of small serial sections remaining in the code (*e.g.* input/output operation).

With a large number of processors such remaining serial sections predominate the program execution time. A sudden surge in efficiency for the second system is with 32 processors when the current data begins to fit in the cache memory. Here the drop in the effectiveness as a result of serial fragments of the program (and, perhaps, communication overhead) is so perceptible that superlinear acceleration cannot be achieved. However, the set goal was accomplished with better results than expected – in typical machining simulations the adopted parallelization strategy allowed for a decrease of the calculation time for one simulation step from 15 seconds to 0.14 seconds for operation on 64 processors.

3.3. Systems under study

The systems under study in this work used six different shapes for the tool with the geometries shown in Figure 6. For all tools, three orientations of the machined material were considered. The numbers of atoms and characteristic dimensions of the system are given in Tables 1–3, while Figure 6 describes the geometries and how the characteristic dimensions were measured.

A typical simulation comprised the following stages:

1. tool generation and relaxation;
2. generation of the machined material and its relaxation;
3. interorientation of the material and the tool;
4. simulation of machining;
5. analysis of results.

The first three stages can be classified as pre-processing, stage 4 denotes the actual simulation and stage 5 consisted in post-processing the results.

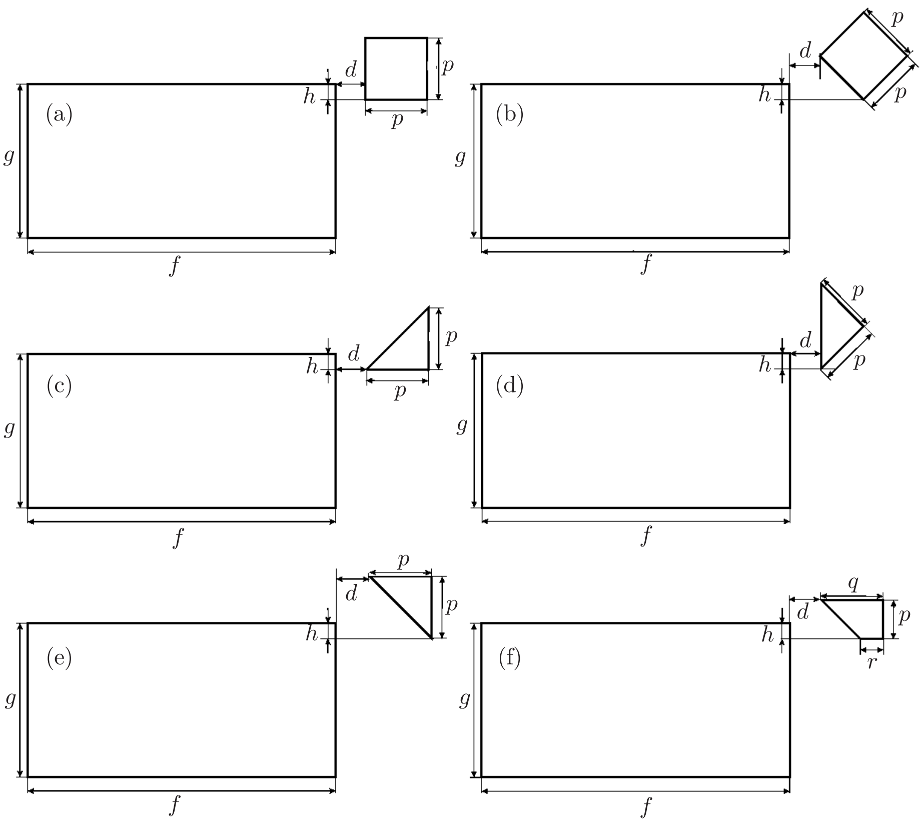

Figure 6. Schematic presentation of the tool orientation and characteristic dimensions of the system. The figure shows a projection onto the xy plane: e – width of the workmaterial (along the z direction, not shown here), f – length of the workmaterial (along the x (machining) direction), g – thickness of the workmaterial (along the y direction), d – initial distance between the workmaterial and the tool, h – machining depth, p, q, r – characteristic tool dimensions, a – doubled distance between (100) planes, (Cu lattice constant $a = 3.61$ Å), b – doubled distance between (110) layers ($b = 3.61\sqrt{2} = 2.56$ Å), c – distance between (111) layers ($c = 3.61(\sqrt{2}/\sqrt{3}) = 2.96$ Å)

3.4. Visualization methods employed during the analysis of simulation results

A direct result of typical MD simulation is a set of system configurations in form of Cartesian coordinates and velocities of specific atoms at subsequent instants. Although this data contains full information about the structure and state of the simulated material, it is directly illegible and requires appropriate treatment. Unfortunately, the standard methods of structure analysis, such as globally calculated radial or angular distribution functions, distributions of coordinated numbers, Voronoi analysis or ring analysis are almost useless in the case of performed nanomachining simulations.

Table 1. Geometries of simulated systems, (100) workmaterial orientation

(100)	$d=40\,\text{Å}$, $e=57.92\,\text{Å}=16a$, $f=144.80\,\text{Å}=40a$, $g=72.40\,\text{Å}=20a$, $N_{wrkm}=53136$			
	N_{tool}	N_{sys}	h	tool dimensions
A1	4624	57760	$3.62\,\text{Å}=1a$	$p=28.96\,\text{Å}$
A2	4624	57760	$7.24\,\text{Å}=2a$	$p=28.96\,\text{Å}$
A5	4624	57760	$18.10\,\text{Å}=5a$	$p=28.96\,\text{Å}$
B1	4624	57760	$3.62\,\text{Å}=1a$	$p=28.96\,\text{Å}$
B2	4624	57760	$7.24\,\text{Å}=2a$	$p=28.96\,\text{Å}$
B5	4624	57760	$18.10\,\text{Å}=5a$	$p=28.96\,\text{Å}$
C1	2448	55584	$3.62\,\text{Å}=1a$	$p=28.96\,\text{Å}$
C2	2448	55584	$7.24\,\text{Å}=2a$	$p=28.96\,\text{Å}$
C5	2448	55584	$18.10\,\text{Å}=5a$	$p=28.96\,\text{Å}$
D1	2448	55584	$3.62\,\text{Å}=1a$	$p=28.96\,\text{Å}$
D2	2448	55584	$7.24\,\text{Å}=2a$	$p=28.96\,\text{Å}$
D5	2448	55584	$18.10\,\text{Å}=5a$	$p=28.96\,\text{Å}$
E1	2448	55584	$3.62\,\text{Å}=1a$	$p=28.96\,\text{Å}$
E2	2448	55584	$7.24\,\text{Å}=2a$	$p=28.96\,\text{Å}$
E5	2448	55584	$18.10\,\text{Å}=5a$	$p=28.96\,\text{Å}$
F1	4624	57760	$3.62\,\text{Å}=1a$	$p=28.96\,\text{Å}$ $q=43.44\,\text{Å}$ $r=14.48\,\text{Å}$
F2	4624	57760	$7.24\,\text{Å}=2a$	$p=28.96\,\text{Å}$ $q=43.44\,\text{Å}$ $r=14.48\,\text{Å}$
F5	4624	57760	$18.10\,\text{Å}=5a$	$p=28.96\,\text{Å}$ $q=43.44\,\text{Å}$ $r=14.48\,\text{Å}$

3.5. Visualization of results

The *nanoMD* program generates the following raw results: positions of all particles (here, recorded every 100 steps), averaged squared velocities for all particles (recorded every 1000 steps), total forces acting on the workmaterial and the tool (mean values over 100 steps) and components of the stress tensor (averaged over cuboid blocks and time, over 500 steps). In order to best interpret the phenomena taking place in the observed systems, it is important to adopt a coherent system of preparing the results on the basis of the raw data.

The authors decided to use an in-house visualization system developed in the Department of Solid State Physics in which the POV-ray program [6] is used to generate images in which atoms are represented as spheres, colored according to values of certain observables.

Table 2. Geometries of simulated systems, (110) workmaterial orientation

(110)	$d = 40\,\text{Å}$, $e = 57.92\,\text{Å} = 16a$, $f = 143.35\,\text{Å} = 56b$, $g = 71.67\,\text{Å} = 28b$, $N_{wrkm} = 51520$			
	N_{tool}	N_{sys}	h	tool dimensions
A1	4624	56144	$2.56\,\text{Å} = 1b$	$p = 28.96\,\text{Å}$
A2	4624	56144	$5.12\,\text{Å} = 2b$	$p = 28.96\,\text{Å}$
A5	4624	56144	$12.80\,\text{Å} = 5b$	$p = 28.96\,\text{Å}$
B1	4624	56144	$2.56\,\text{Å} = 1b$	$p = 28.96\,\text{Å}$
B2	4624	56144	$5.12\,\text{Å} = 2b$	$p = 28.96\,\text{Å}$
B5	4624	56144	$12.80\,\text{Å} = 5b$	$p = 28.96\,\text{Å}$
C1	2448	53968	$2.56\,\text{Å} = 1b$	$p = 28.96\,\text{Å}$
C2	2448	53968	$5.12\,\text{Å} = 2b$	$p = 28.96\,\text{Å}$
C5	2448	53968	$12.80\,\text{Å} = 5b$	$p = 28.96\,\text{Å}$
D1	2448	53968	$2.56\,\text{Å} = 1b$	$p = 28.96\,\text{Å}$
D2	2448	53968	$5.12\,\text{Å} = 2b$	$p = 28.96\,\text{Å}$
D5	2448	53968	$12.80\,\text{Å} = 5b$	$p = 28.96\,\text{Å}$
E1	2448	53968	$2.56\,\text{Å} = 1b$	$p = 28.96\,\text{Å}$
E2	2448	53968	$5.12\,\text{Å} = 2b$	$p = 28.96\,\text{Å}$
E5	2448	53968	$12.80\,\text{Å} = 5b$	$p = 28.96\,\text{Å}$
F1	4624	56144	$2.56\,\text{Å} = 1b$	$p = 28.96\,\text{Å}$ $q = 43.44\,\text{Å}$ $r = 14.48\,\text{Å}$
F2	4624	56144	$5.12\,\text{Å} = 2b$	$p = 28.96\,\text{Å}$ $a = 43.44\,\text{Å}$ $r = 14.48\,\text{Å}$
F5	4624	56144	$12.80\,\text{Å} = 5b$	$p = 28.96\,\text{Å}$ $q = 43.44\,\text{Å}$ $r = 14.48\,\text{Å}$

An advantage of this system is its automatedness – subsequent snapshots representing the system states are automatically processed one after another, producing numbered images in the PNG format as output. An automatic system of editing images obtained in this way permits to easily create synthetic reviews of the simulation progress. Using the ImageMagick package [7], it is easy to generate animations that illustrate that progress. In the following paragraphs, the quantities used to illustrate the results will be discussed with a detailed description of their treatment (scaling, averaging).

3.6. Analysis of thermomechanical properties

3.6.1. Temperature field

Due to a naturally broad range of velocities of single particles, it was necessary to present values averaged over both time and space when visualizing

Table 3. Geometries of simulated systems, (111) workmaterial orientation

(111)	$d = 40\,\text{Å}$, $e = 57.92\,\text{Å} = 27c$, $f = 144.10\,\text{Å} = 195c$, $g = 71.67\,\text{Å} = 28b$, $N_{wrkm} = 50272$			
	N_{tool}	N_{sys}	h	tool dimensions
A1	4624	54896	$2.56\,\text{Å} = 1b$	$p = 28.96\,\text{Å}$
A2	4624	54896	$5.12\,\text{Å} = 2b$	$p = 28.96\,\text{Å}$
A5	4624	54896	$12.80\,\text{Å} = 5b$	$p = 28.96\,\text{Å}$
B1	4624	54896	$2.56\,\text{Å} = 1b$	$p = 28.96\,\text{Å}$
B2	4624	54896	$5.12\,\text{Å} = 2b$	$p = 28.96\,\text{Å}$
B5	4624	54896	$12.80\,\text{Å} = 5b$	$p = 28.96\,\text{Å}$
C1	2448	52720	$2.56\,\text{Å} = 1b$	$p = 28.96\,\text{Å}$
C2	2448	52720	$5.12\,\text{Å} = 2b$	$p = 28.96\,\text{Å}$
C5	2448	52720	$12.80\,\text{Å} = 5b$	$p = 28.96\,\text{Å}$
D1	2448	52720	$2.56\,\text{Å} = 1b$	$p = 28.96\,\text{Å}$
D2	2448	52720	$5.12\,\text{Å} = 2b$	$p = 28.96\,\text{Å}$
D5	2448	52720	$12.80\,\text{Å} = 5b$	$p = 28.96\,\text{Å}$
E1	2448	52720	$2.56\,\text{Å} = 1b$	$p = 28.96\,\text{Å}$
E2	2448	52720	$5.12\,\text{Å} = 2b$	$p = 28.96\,\text{Å}$
E5	2448	52720	$12.80\,\text{Å} = 5b$	$p = 28.96\,\text{Å}$
F1	4624	54896	$2.56\,\text{Å} = 1b$	$p = 28.96\,\text{Å}$ $q = 43.44\,\text{Å}$ $r = 14.48\,\text{Å}$
F2	4624	54896	$5.12\,\text{Å} = 2b$	$p = 28.96\,\text{Å}$ $q = 43.44\,\text{Å}$ $r = 14.48\,\text{Å}$
F5	4624	54896	$12.80\,\text{Å} = 5b$	$p = 28.96\,\text{Å}$ $q = 43.44\,\text{Å}$ $r = 14.48\,\text{Å}$

and analyzing the local temperature. The squared velocities on the basis of which the temperature was calculated were calculated as mean values (squares) of 1000 steps for each of the particles. Then, these were averaged over cuboid blocks $3a \times 3a \times e$ in size, where e was the workmaterial width along Z, and a – the lattice constant (3.62Å). For simulations with the highest machining speed (1000m/s) the mean values were calculated from 100 steps as the number of simulation steps was ten times smaller.

The values presented in the figures may contain minor artifacts resulting from the fact that the intervals at which the average squared velocities were recorded during the simulation were, in general, different from the intervals in which the positions were recorded (the squared velocities were recorded not as often to save disk space). As a result, it sometimes happened that the configuration of atoms from step t was combined with colors corresponding to the temperatures averaged from step $t \pm \Delta t$, where Δt could be as much as 500 steps (corresponding to 1.25 ps). Due to the temperature averaging over 1000-step intervals, this effect is not of any particular importance.

3.6.2. Stress fields

The components of the Cauchy atomic stress tensor σ were calculated as

$$\sigma_i = \frac{1}{2\Omega_i} \sum_{j \neq i} \boldsymbol{r}_{ij} \otimes \boldsymbol{f}_{ij} \tag{21}$$

where $\otimes$ denotes the tensor product and the sum runs over all atoms j located within a radius of $r_{cut} = 12\,\text{Å}$ from atom i, and Ω_i is an arbitrary volume associated with atom i. In practice, in order to be released from the necessity to specify Ω_i, the stress tensor components were averaged over blocks of $5\,\text{Å} \times 5\,\text{Å} \times 5\,\text{Å}$ and over time (500 steps). Attention should be drawn to the fact that adopting a constant volume $5^3\,\text{Å}^3$ in the formula results in unreliable results for the "cubes" in which the number of atoms was significantly smaller than in the bulk material. As a result, underestimated values (in terms of magnitude) of three observables calculated on the basis of the stress tensor, *i.e.*, the hydrostatic pressure (one third of the tensor's trace), shear stresses (xy-component), invariant J_2 of the stress tensor deviator (the von Mises invariant) are observed on the material's edges where the averaging volumes were only partially filled with atoms.

As atoms in figures are presented by means of opaque spheres, direct visualization of the values of the observables in the deeper layers (along Z), other that those closest to the eye, became impossible. For these reasons, we present values additionally averaged over the whole system width (that is, along Z). The above concerns the values calculated on the basis of the stress tensor that were mentioned at the end of the preceding paragraph.

A few words of explanation are in order, regarding the atoms marked in a light-pink color in the figures presented in the following sections. This color represents atoms for which it was, for any reason, impossible to define the stress tensor components. This may have occurred due to minor differences between steps in which the positions of particles were recorded and steps in which averaged stress tensor values were recorded, when some of the cubes were totally emptied of particles as a result of "migration" of particles between the cubes. As a result of the shortcomings in the *nanoMD* program related to the handling of periodic boundary conditions some particles ($< 0.1\%$) were not ascribed correctly to any of the cubes when recording the stress tensor component. Intentional distinguishing of those particles by an intense color is unfavorable from the aesthetic point of view, however, it allows their number to be easily monitored.

3.7. Analysis of structural variations

The plastic properties of crystalline materials are closely related to the occurrence of structural defects therein, *i.e.* mainly dislocation. Actual dislocations are unlike their extreme model edge and screw dislocation types and are a mixture of both, they form loops, split and join forming a spatial lattice (the so-called Frank lattice) [8, 9].

Dislocations are often characterized with the help of the so-called Burgers vector. For the edge dislocation the vector is perpendicular to the dislocation line, and for the screw dislocation – it is parallel thereto. In a general case, the mixed dislocation line forms any angle with its Burgers vector.

The Burgers vector for a given dislocation section (between the Frank lattice nodes) remains constant, and in the lattice bond it splits into dislocations for which the sum of Burgers vectors is equal to the Burgers vector of the line entering the lattice bond. Ideal and partial dislocations are distinguished in terms of the Burgers vector length. In the former case, the Burgers vector length is an integer multiple of the lattice constant, in the latter case – it is a fractional multiple. During deformation, the motion of mass is accompanied by the motion of dislocations classified as conservative (slipping) whereby the dislocation is moving in a plane containing the dislocation line and its Burgers vector and by the non-conservative motion (climbing) whereby the dislocation leaves the slip plane in a perpendicular direction to the Burgers vector.

Plastic deformation in a crystal occurs owing to shifts of atomic planes with respect to each other. The plane where the motion occurs is the so called slip plane and the crystal sections which it divides remain intact. The slip plane and direction are usually the plane and direction of the most dense packing. The preferred plane for *fcc* crystals considered in this work is $\{111\}$ and the preferred direction is $\langle 110\rangle$.

It is impossible to specify the Burgers vectors experimentally for specific dislocations in a plastically deformed crystal, and reference must be made to averaged values. The situation is almost as complicated in numerically simulated crystals that have undergone plastic deformation. This results most of all from the intuitive Burgers contour concept (a loop including a single dislocation line). The applied simplified methods of structural analysis of crystals in an elastic and plastic deformation state will be discussed below.

3.7.1. Absolute displacement of atoms from initial equilibrium positions

One of the simplest measures of changes occurring in a machined workmaterial is the scalar (absolute) displacement, $\delta_i(t)$, of each atom with respect to its initial position calculated as the length of the vector joining the i^{th} atom's position at instant t with its initial position:

$$\delta_i(t) = |\boldsymbol{r}_i(t) - \boldsymbol{r}_i(t=0)| \qquad (22)$$

For atoms oscillating around their initial lattice positions this quantity will be close to zero, and definitely significantly smaller than the distance between the closest neighbours (2.56Å). However, it should be noted that values even of the order of several Å do not have to indicate local plastic deformation at all, but can equally well indicate elastic deformation of large magnitude – for instance at the moment when the machined material bends in response to the attractive forces originating from the approaching tool (the tool deforms in a similar way when it does not consist of immobilized atoms). Atoms at the edge of the deformed material may reach displacements of even several Å, which is caused by a local increase in the lattice constant, and not by plastic deformation.

When calculating $\delta_i(t)$, periodic boundary conditions along the Z axis were taken into consideration. Where the atoms are intentionally moved at a constant speed (as was the case in the study of tool degradation), the calculated displacement was corrected for this motion.

3.7.2. Slip vectors

At the beginning of this century a new parameter was proposed describing plastic deformation, the so called slip vector [10, 11]. The slip vector carries relevant information concerning Burgers vector, however, it is significantly easier to calculate.

The slip vector is calculated for each atom separately according to the formula:

$$\boldsymbol{S}_i = \frac{1}{N_s}\sum_{j\neq i}^{N}(\boldsymbol{r}_{ij}-\boldsymbol{r}_{ij}^0) \tag{23}$$

where $\boldsymbol{r}_{ij}$ and $\boldsymbol{r}_{ij}^0$ are, respectively, the vectors linking atoms i with all its N closest neighbours j, in a given simulation step and a reference simulation step, and N_s is the number of slipped neighbours. The initial configuration was adopted as the reference, *i.e.* $\boldsymbol{r}_{ij}^0$ refers to the non-deformed configuration. The original neighbours of central atom i do not have to be neighbours in the deformed configuration. Their absence amongst the current neighbours indicates displacement of atoms. The discussed parameter is potentially very useful, most of all due to the freedom of selecting the reference moment of counting and the resulting possibility of identifying the moment of origin of dislocation, speed and the direction of its displacement in the crystal.

The cut-off radius where neighbours of each atom were contained was adopted as equal to 2.6Å. The distance threshold (between the atom position in a deformed system and the position in a non-deformed system) above which atoms were considered as “slipped” was adopted as equal to 25% of the lattice constant. In the figures presented in the following sections, atoms with a zero slip vector are presented in dark gray, and are somewhat translucent, so that the outlines of slip planes, extending into the material (along Z) could be seen.

3.7.3. Falk-Langer coefficient

The coefficient $D^2_{\min}$, defined by Falk and Langer in [12], permits the identification of the fragments of the system in which irreversible shear transformation occurred. It is calculated according to the formula:

$$D_i^{\min} = \sqrt{\sum_{j\neq i}^{N_{\text{neigh}}}\left[\boldsymbol{r}_{ij}-\boldsymbol{r}_i^0\left(\boldsymbol{r}_{ij}\cdot\frac{\boldsymbol{r}_i^0}{|\boldsymbol{r}_i^0|}\right)\right]^2} \tag{24}$$

Following the authors of the method, for the calculation of $D^2_{\min}$, the cut-off distance for the determination of neighbours was equal to the cut-off radius used for the potential during the simulation (12 Å). We comment on the fact that that the quantity $D^2_{\min}$ is not normalized with respect to the number of atom-neighbours included in the sum in Equation (24). The consequences of this fact are twofold.

First, the range of values of $D^2_{\min}$ is a function of the adopted cut-off radius, hence the specific values of the coefficient do not have a clearly stated physical meaning. The scale of colors and the threshold below which the spheres representing the atoms were not colored and left semi-transparent was selected by trial and error. The second consequence is that atoms located close to the surfaces have a smaller number of neighbours than volume atoms, and hence a smaller number of terms in the sum in Equation (24), which results in lowered $D^2_{\min}$ values for these atoms. This effect is visible particularly on chip edges.

Typical distributions of the discussed quantities are shown in Figure 7. The general observations were as follows:

- the absolute displacement field shows well the elastic motions of the machined material which do not result in dislocation;
- the slip vector shows where the material defects are located,
- the $D_{\min}$ coefficient shows where the machined material has been plastically deformed,
- hydrostatic pressure (obtained as one third of the trace of the stress tensor) – makes it possible to notice that the whole machined material is "under pressure" and that the pressure is lower only in front of the tool, within the chip (the machined material is relaxing),
- the shear stress field shows that the shearing occurs directly to the south-west of the tool,
- the von Mises invariant is the least useful in the interpretation of the results.

3.8. Chip identification and characterization

In many cases the information about the chip, its size, temperature or structure will be of interest. The following method of identification of atoms belonging to the chip was developed: moving horizontally from the left side of the workmaterial the first atom which is located at least $2a$ above the initial upper free surfaces of the workmaterial is found. From that point a plane is set to the left lower section of the tool which is the chip-nonchip boundary. Another plane is set behind the tool, inclined at the angle of 45°, to take into account the chip atoms that have stuck to the tool. All atoms that are found inside the determined angle are treated as belonging to the chip. The structure is illustrated in Figure 8. In fact, all the methods of analysis of results described above may also be applied to the chip alone.

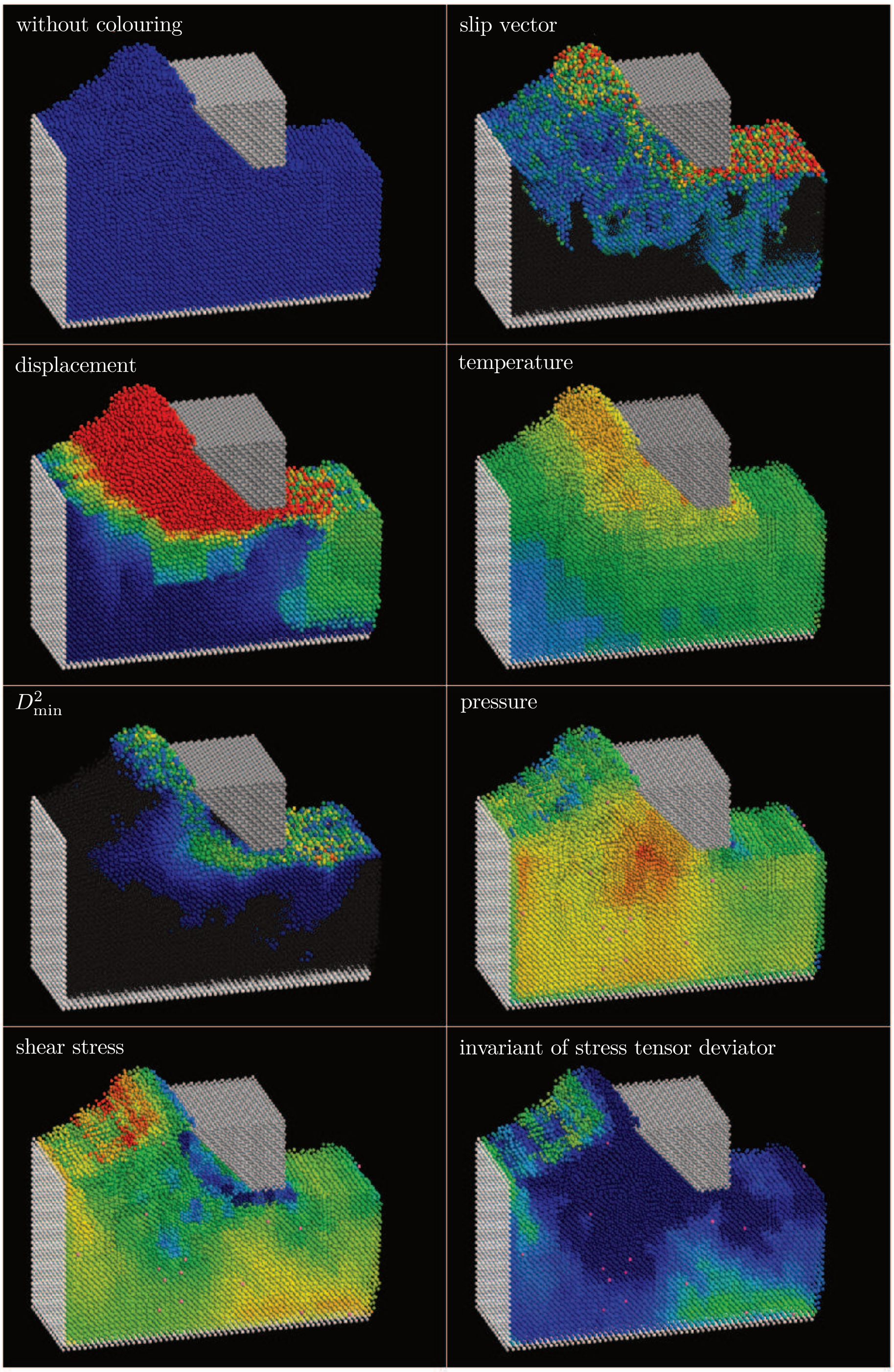

Figure 7. Typical distributions of monitored quantities

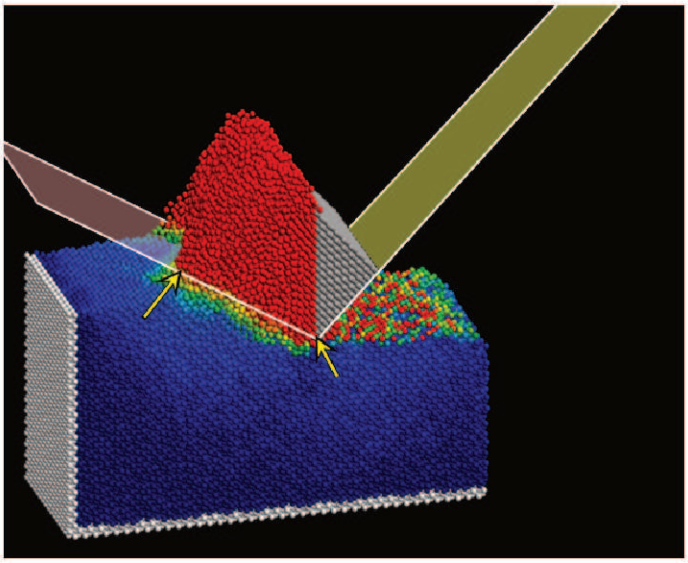

Figure 8. A schematic illustrating the principle of chip identification.

4. Test runs and the selection of optimum parameters for production runs

In this section, selected test runs will be discussed. The objective of these was to get a rough idea of the general behaviour of the system and to select the optimum parameters for the subsequent production runs. In particular, the test calculations were limited to only one orientation of the machined monocrystal (100), one machining depth, varying the system's initial temperature (*cf.* Section 4.1). and the machining speed (*cf.* Section 4.2). The machining depth was selected arbitrarily as one, two and five doubled distances between crystallographic planes parallel to the upper crystal surface, namely 3.61Å for (100), $3.61\sqrt{2}/2 = 2.5527$Å for (110) and $3.61(\sqrt{2}/\sqrt{3}) = 2.96$Å for (111). The effect of the machining speed and the system's initial temperature was investigated only for the case of intermediate depth.

4.1. Effect of the system's initial temperature on the nanomachining run

Test simulations were carried out for initial temperatures of 100, 300, 500, 700, 900 and 1100K, with a machining speed of $v = 100$m/s, tool type A and a depth of $2a$.

Figures 9–14 show the fields of absolute displacement of atoms for these systems. For initial temperatures not higher than 500K, the fields of absolute displacement are in accordance with intuition and no comments are required. Starting from the initial temperature of 700K, increased displacements of atoms located at the right vertical free surface of the workmaterial are observed in later snapshots. This effect is not related directly to the machining process, but results from the thermally induced tendency to round the material edges on free surfaces. At the initial temperature of 1100K some atoms “flowed out” of the area limited by immobilized atoms and found themselves below them (see Figure 14).

The temperature distributions are shown in Figures 15–20.

It should be noted that the colorscale varies between the plots: whereas the most intense blue color always corresponds to 300K, the most intense red color corresponds to 900K in the first three figures, to 1200K in the next two figures and to 1500K in the last figure. As the tool is gradually moving, the whole volume of the machined material heats up, with the highest temperatures occuring at the point of contact between the tool and the workmaterial (the chip and the material located directly below the machining tool).

Figure 21 shows changes of the mean temperature of the machined material for different initial temperatures of the system. As can be seen, the mean temperature of the initially hot (1100K) machined material does not change appreciably, while the temperature of the initially cold (100K) material increased significantly during machining (by approximately 400K). Intermediate initial temperatures correspond to intermediate slopes of monotonous temperature increases. The effect can be easily explained by looking at the graphs in Figure 22 which show that the heated material is more plastic and gives little resistance to the tool which means that the tool performs less work on the heated material.

The total number of atoms with a non-zero slip vector as function of time in systems of different initial temperatures is shown in Figure 23.

Figure 24 illustrates the accumulation of atoms in the chip in the same systems.

It can be seen that the behaviour for low initial temperatures (100, 300 and 500K) is quantitatively different than for high initial temperatures (700, 900 and 1100K). In the low-temperature range, the first atoms with a non-zero slip vector appear when the tool gets into immediate contact with the machined material and the chip begins to be identifiable several picoseconds later. In the high-temperature range the first slips occur immediately when the simulation has started when the distance between the tool and the workmaterial is larger than the interaction cut-off. Hence, the motion of those atoms has no connection with the machining process, and is related to the spontaneous rounding of the right upper edge of the workmaterial.

Moreover, the chip begins to form before the moment of contact between the tool and the workmaterial (*cf.* the two upper panels of Figures 13 and 14). Hence, what is observed here is an artifact resulting from the spontaneous instability of the machined material, irrelevant to actual machining. Moreover, at high temperatures (700K, and particularly at 1100K) huge numbers of atoms comparable with the total number of atoms in the simulated system are subject to slipping.

We thus conclude that systems with too high initial temperatures may not be the subject of realistic simulations of the nanomachining process. Moreover,

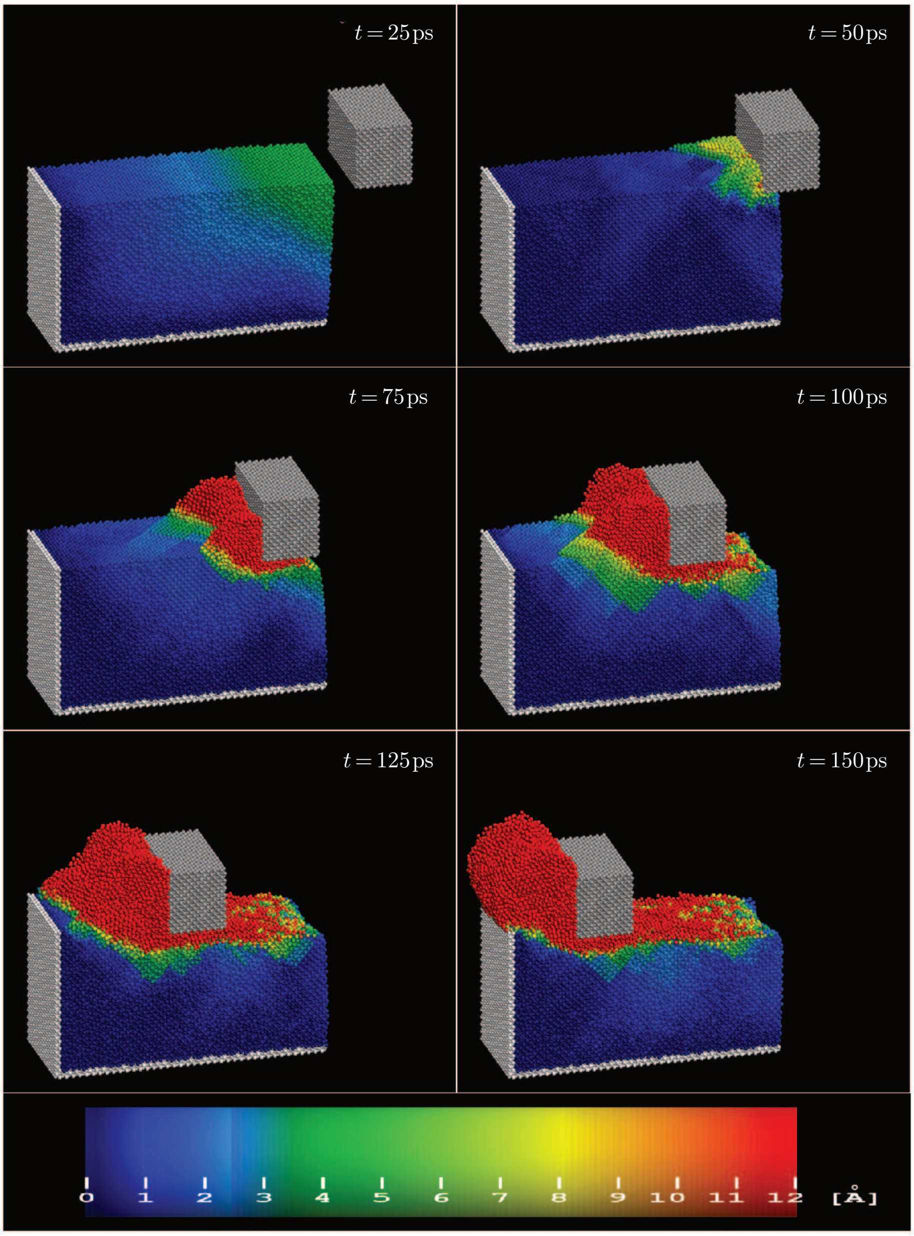

Figure 9. Atomic configurations at selected instants for the initial system temperature of $T = 100$K. Machining depth $2a$, machining speed $v = 100$m/s

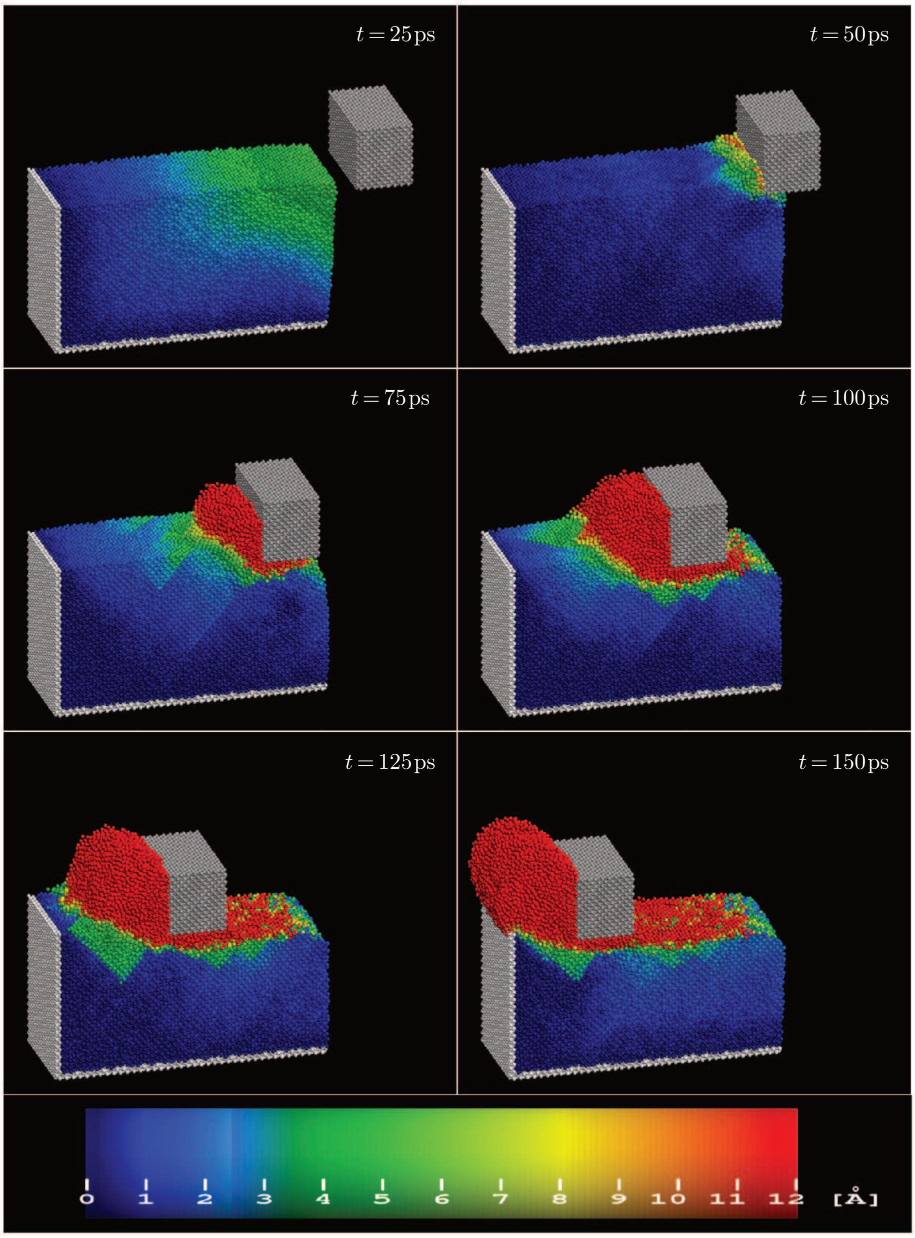

Figure 10. Atomic configurations at selected instants for the initial system temperature of $T = 300$K. Machining depth $2a$, machining speed $v = 100$m/s

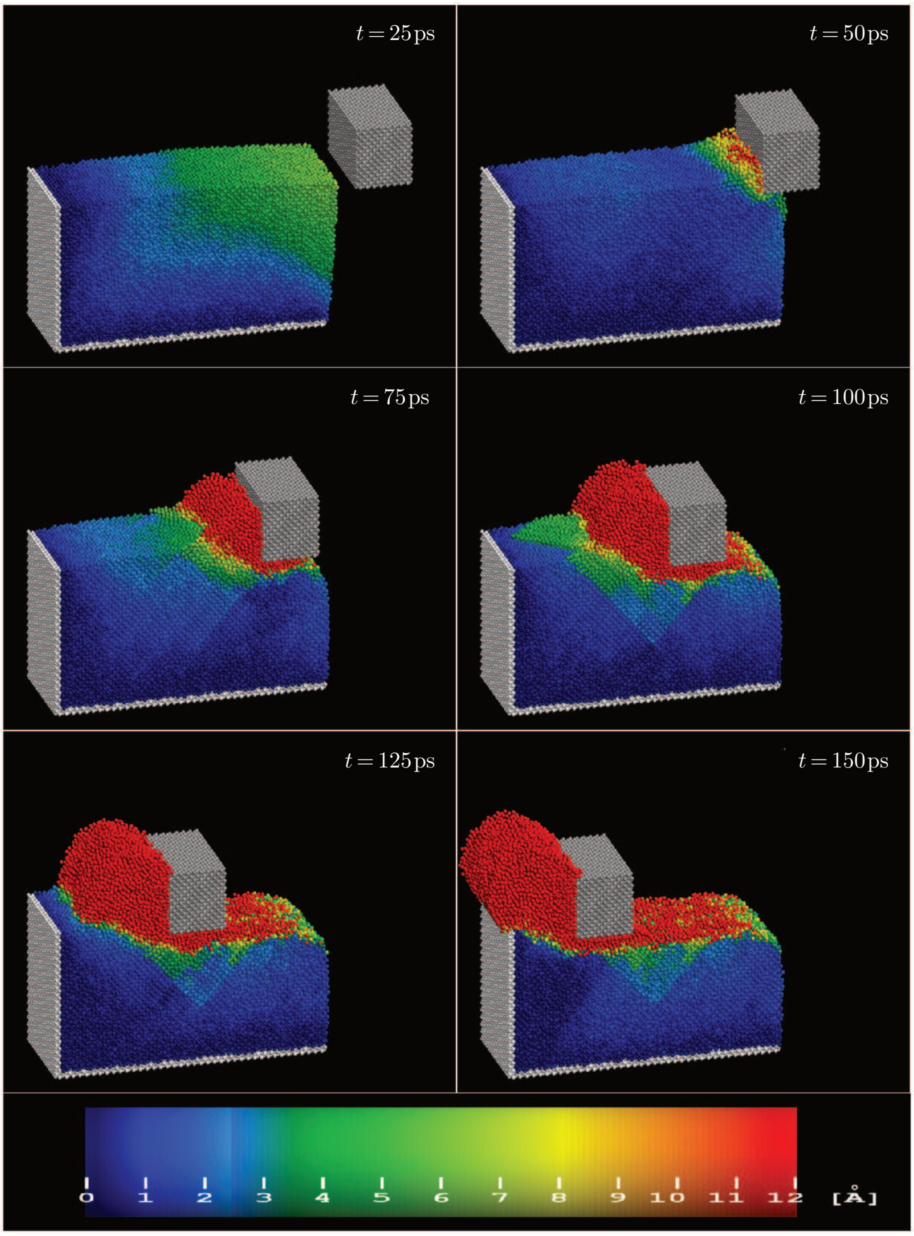

Figure 11. Atomic configurations at selected instants for the initial system temperature of $T = 500$K. Machining depth $2a$, machining speed $v = 100$m/s

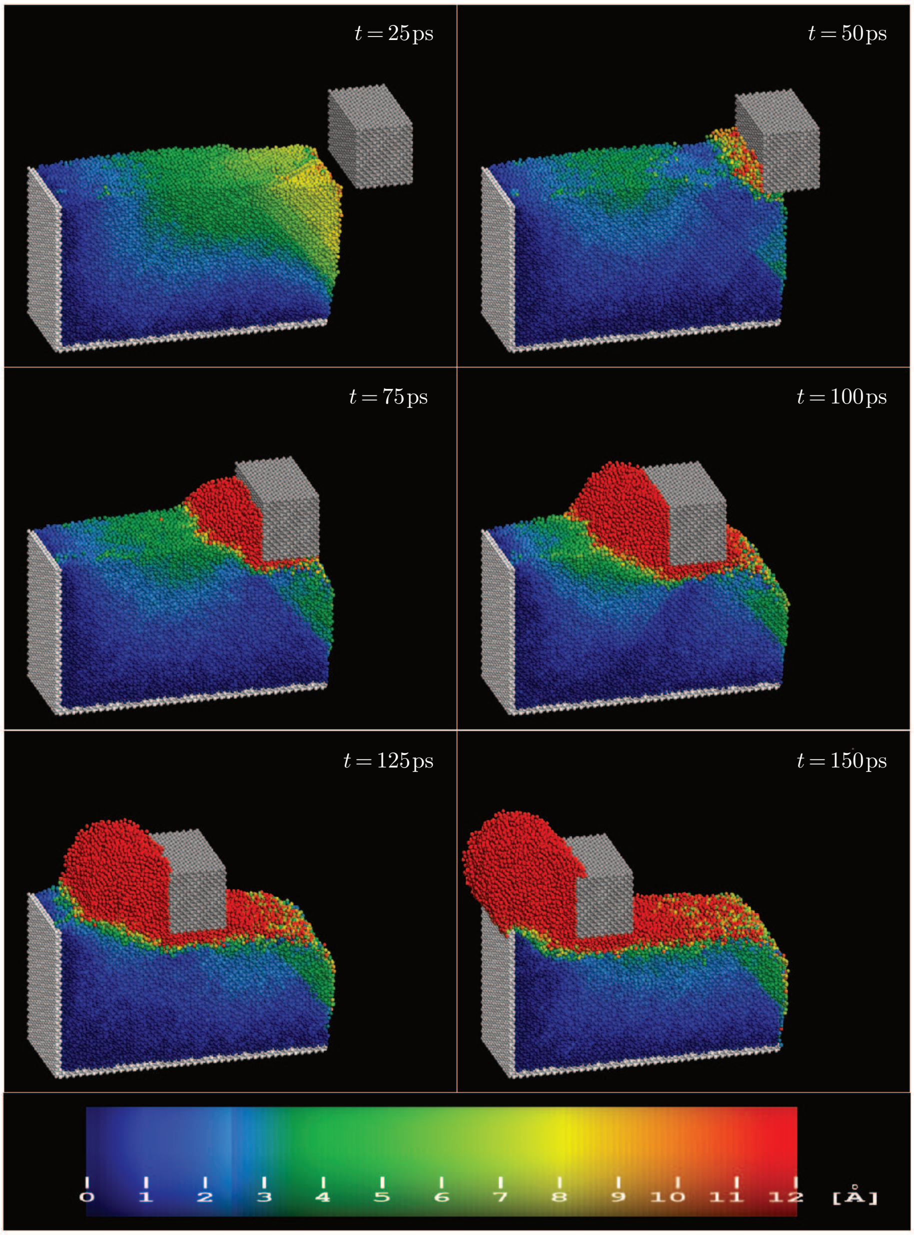

Figure 12. Atomic configurations at selected instants for the initial system temperature of $T = 700$K. Machining depth $2a$, machining speed $v = 100$m/s

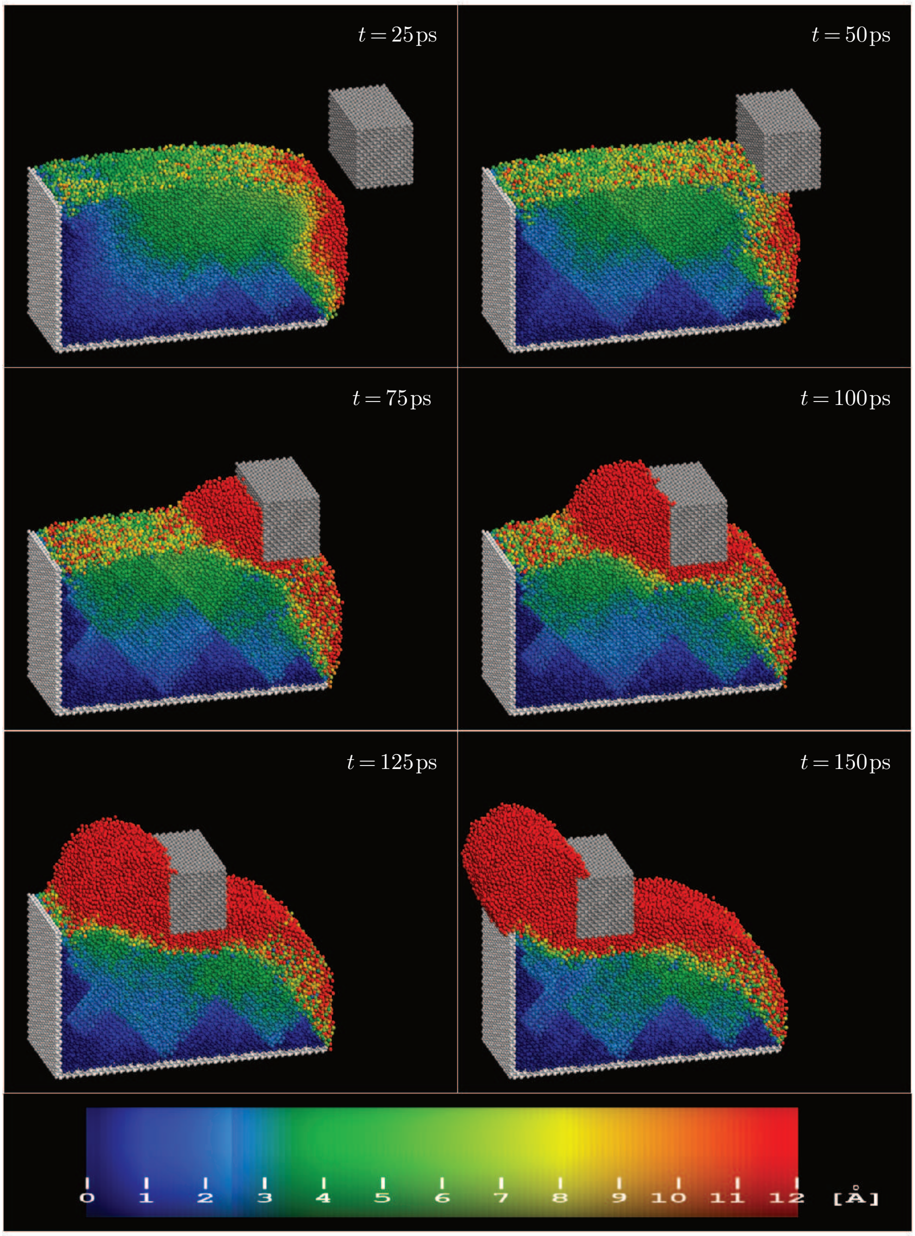

Figure 13. Atomic configurations at selected instants for the initial system temperature of $T = 900$K. Machining depth $2a$, machining speed $v = 100$m/s

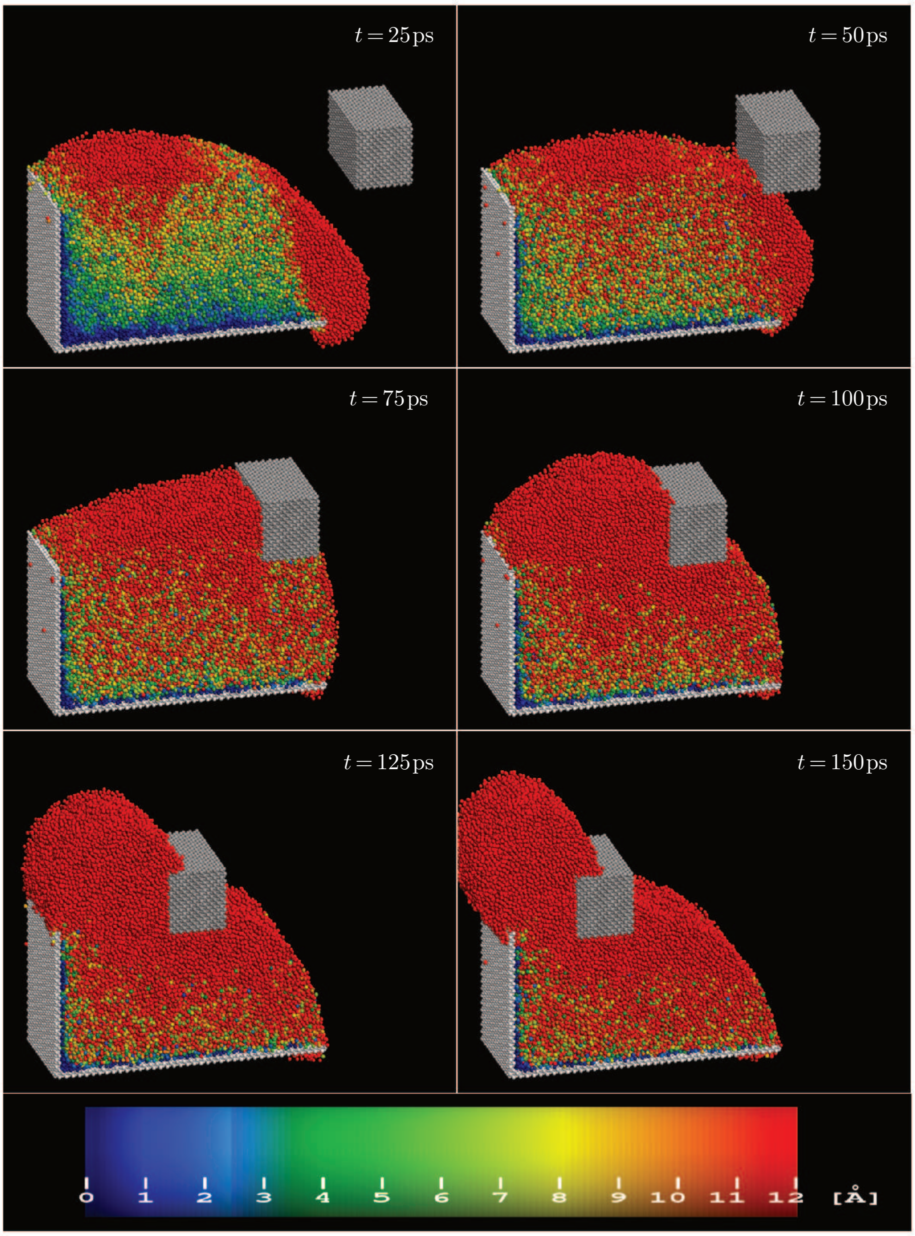

Figure 14. Atomic configurations at selected instants for the initial system temperature of $T = 1100$K. Machining depth $2a$, machining speed $v = 100$m/s

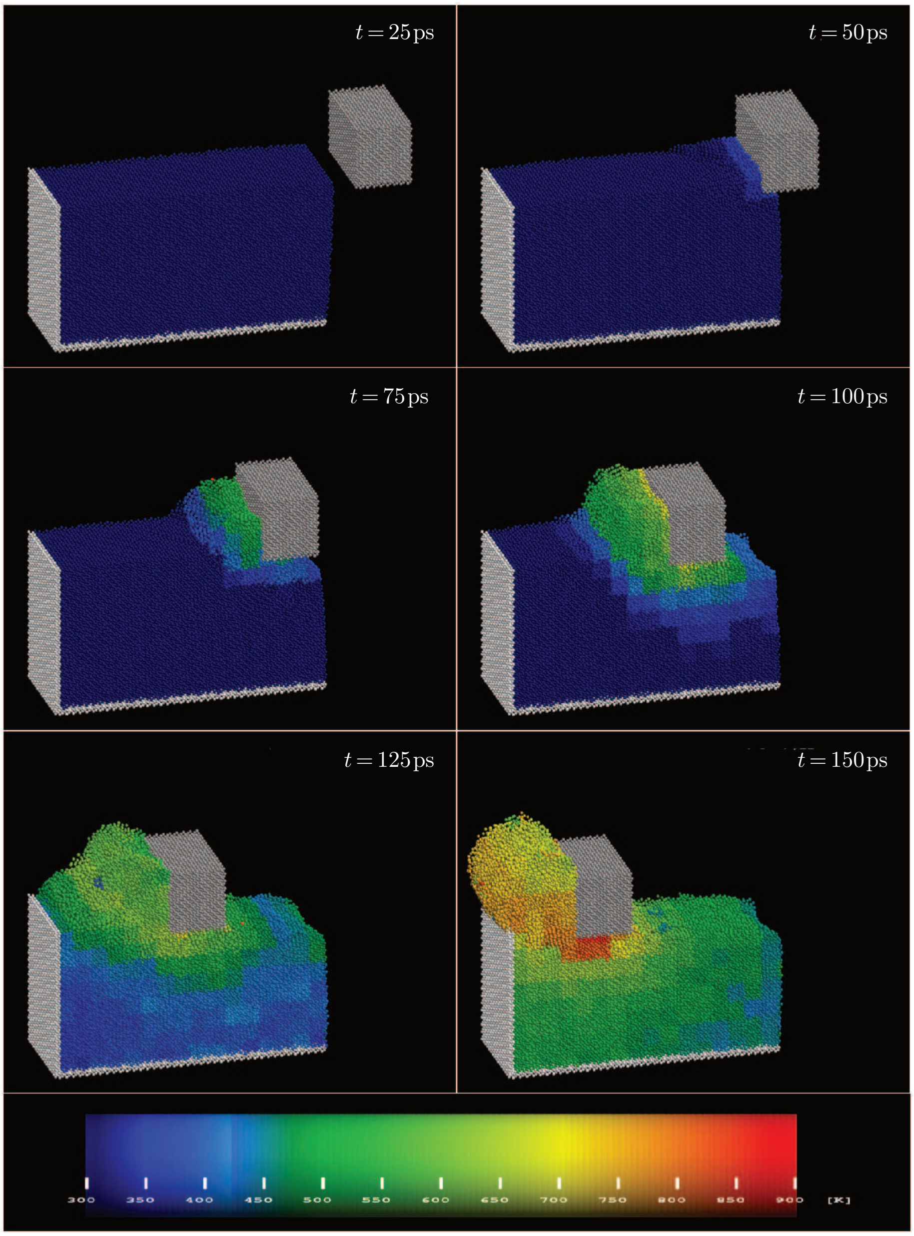

Figure 15. Temperature distributions at selected instants for the initial system temperature of $T = 100$K. Machining depth $2a$, machining speed $v = 100$m/s

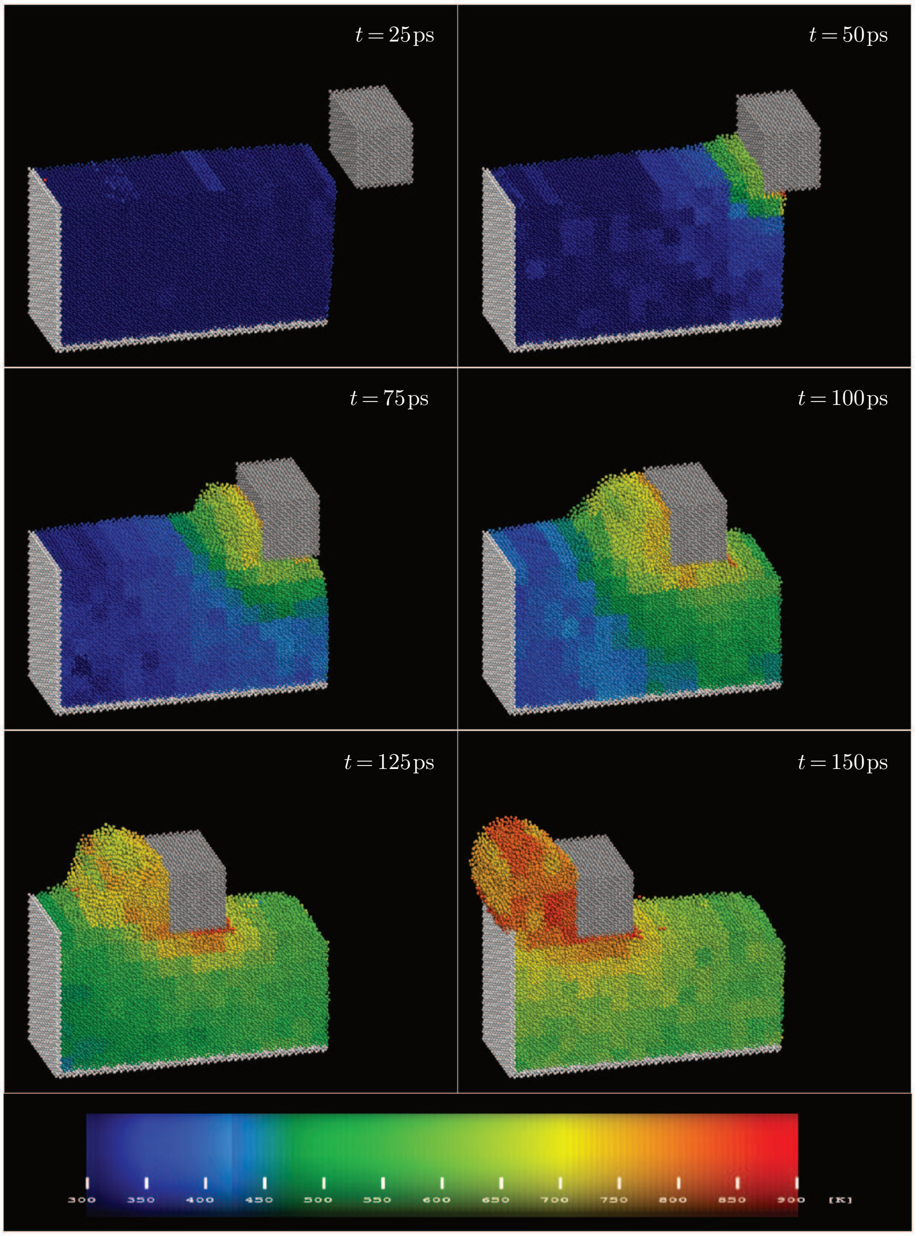

Figure 16. Temperature distributions at selected instants for the initial system temperature of $T = 300$K. Machining depth $2a$, machining speed $v = 100$m/s

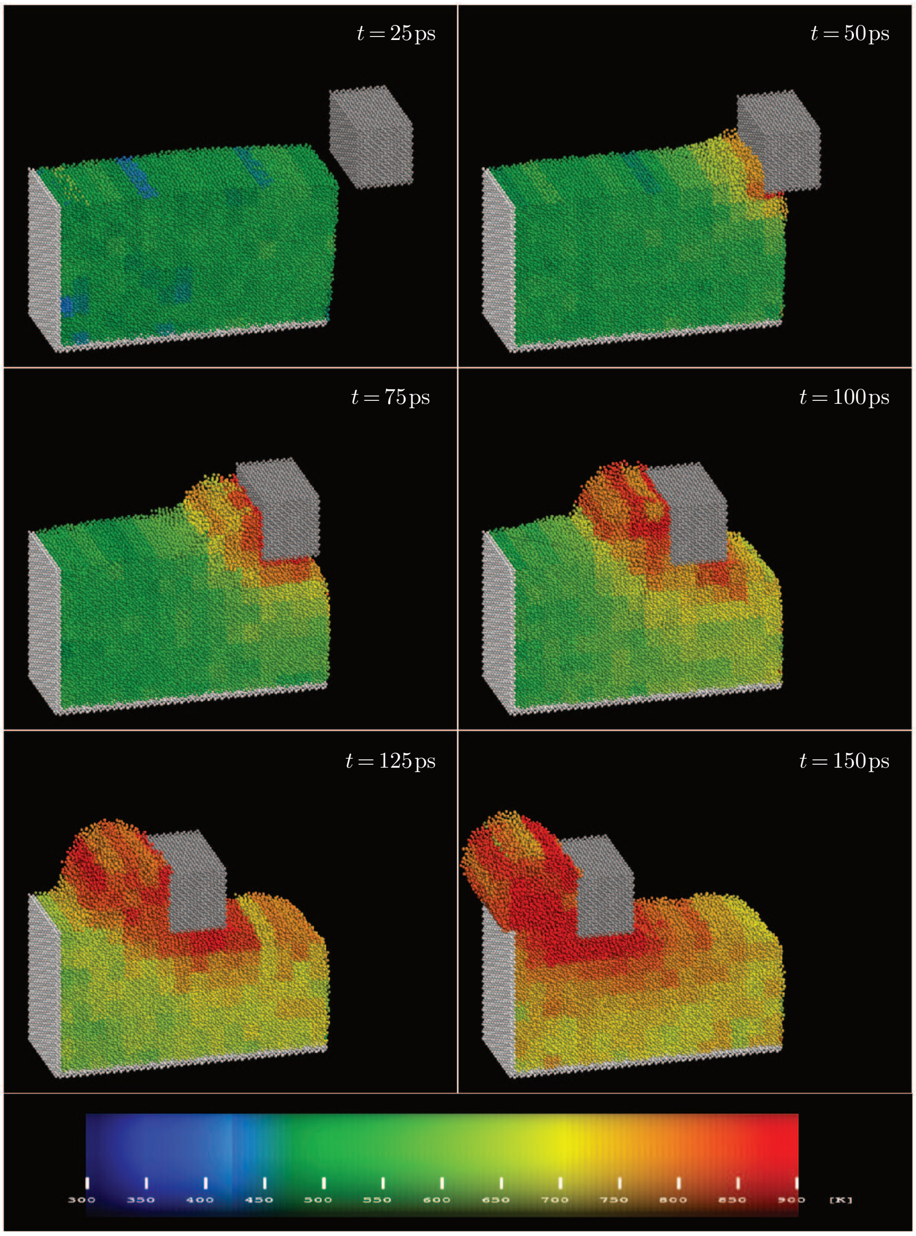

Figure 17. Temperature distributions at selected instants for the initial system temperature of $T = 500$K. Machining depth $2a$, machining speed $v = 100$m/s

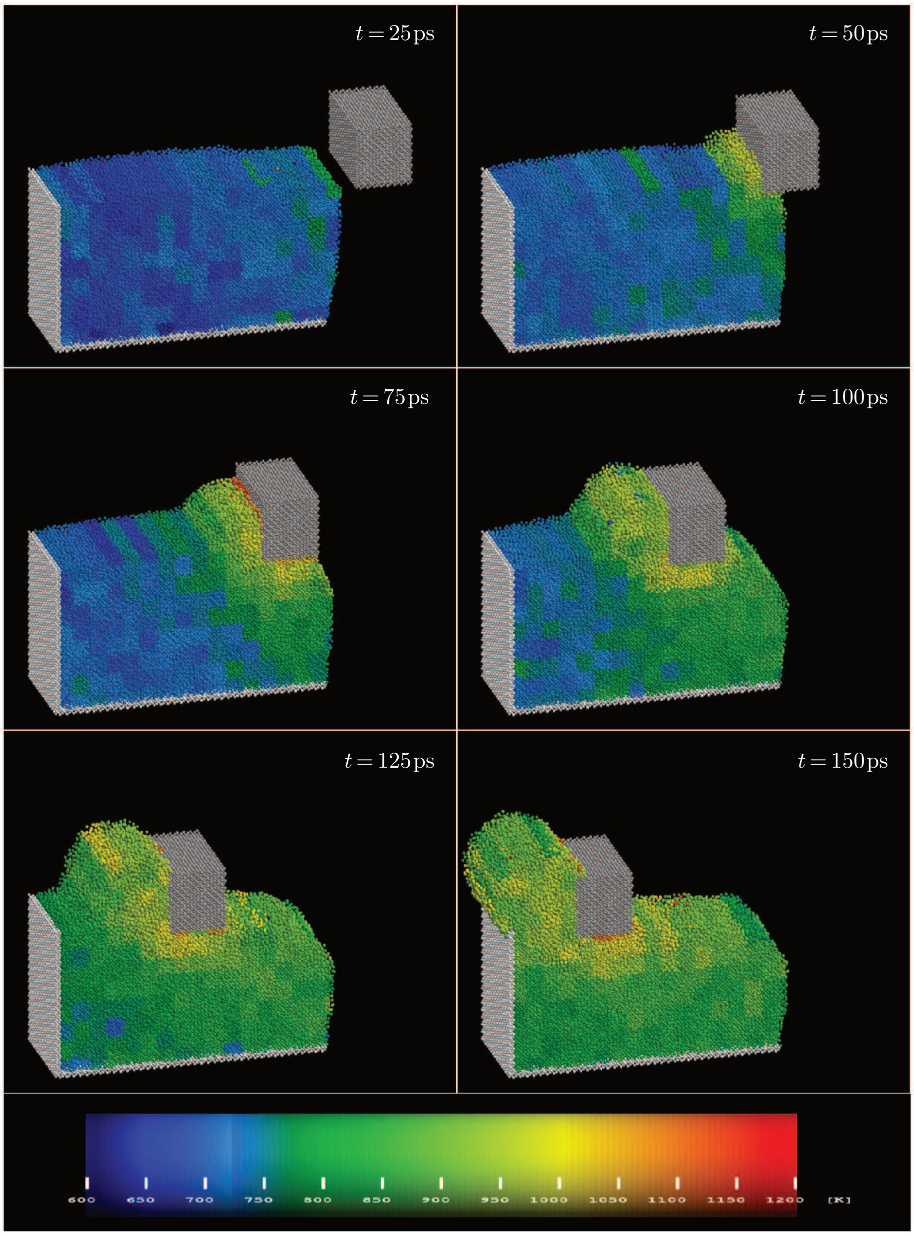

Figure 18. Temperature distributions at selected instants for the initial system temperature of $T = 700$K. Machining depth $2a$, machining speed $v = 100$m/s

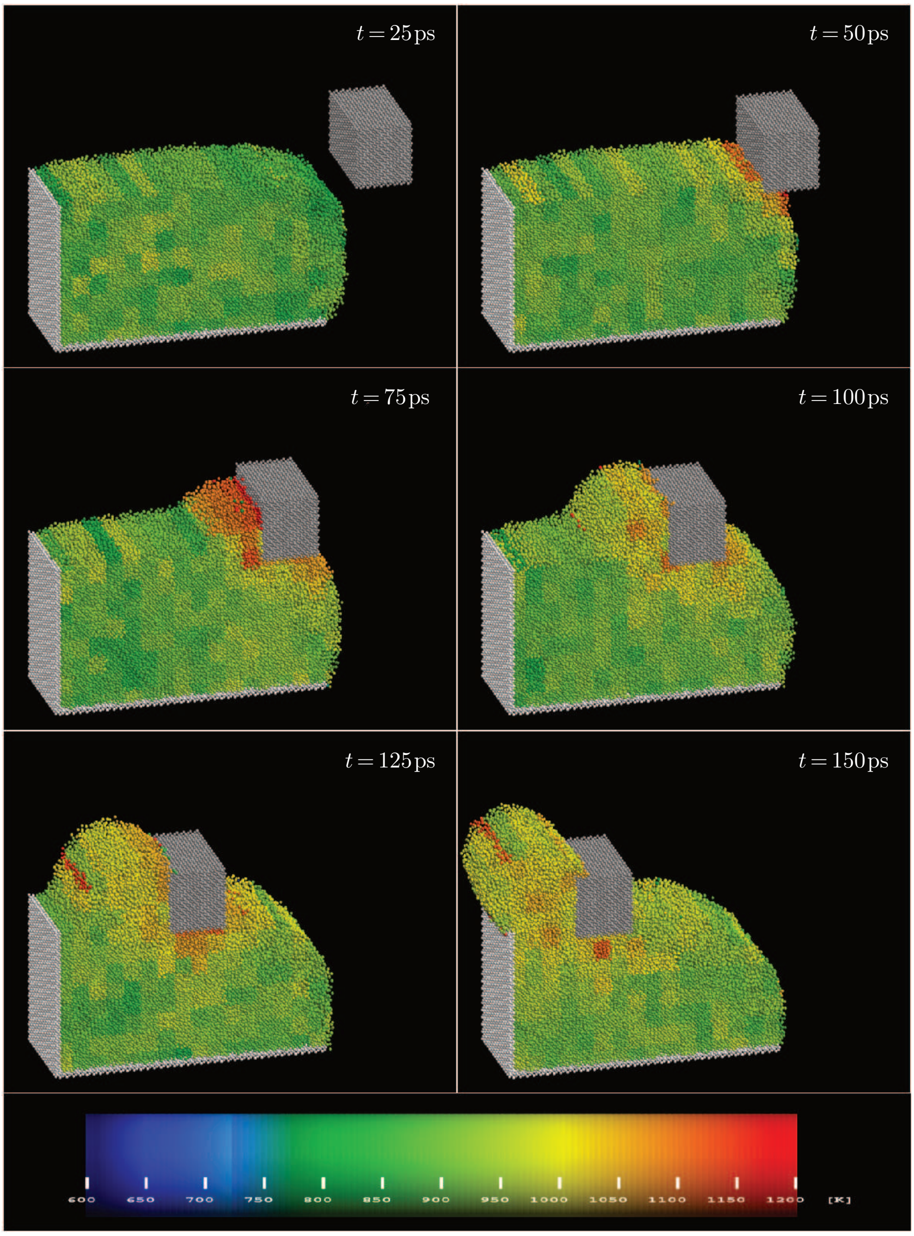

Figure 19. Temperature distributions at selected instants for the initial system temperature of $T=900$K. Machining depth $2a$, machining speed $v=100$m/s

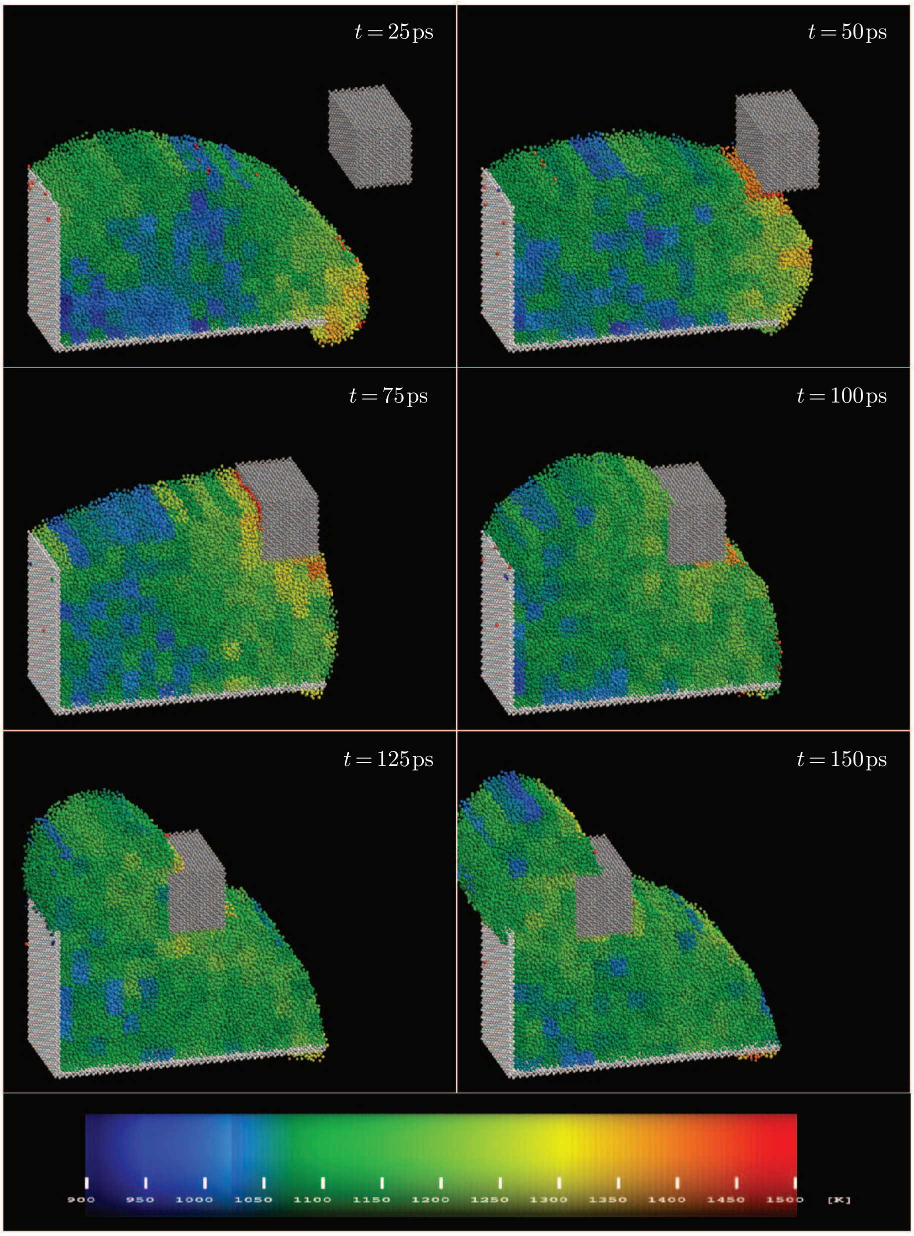

Figure 20. Temperature distributions at selected instants for the initial system temperature of $T = 1100$K. Machining depth $2a$, machining speed $v = 100$m/s

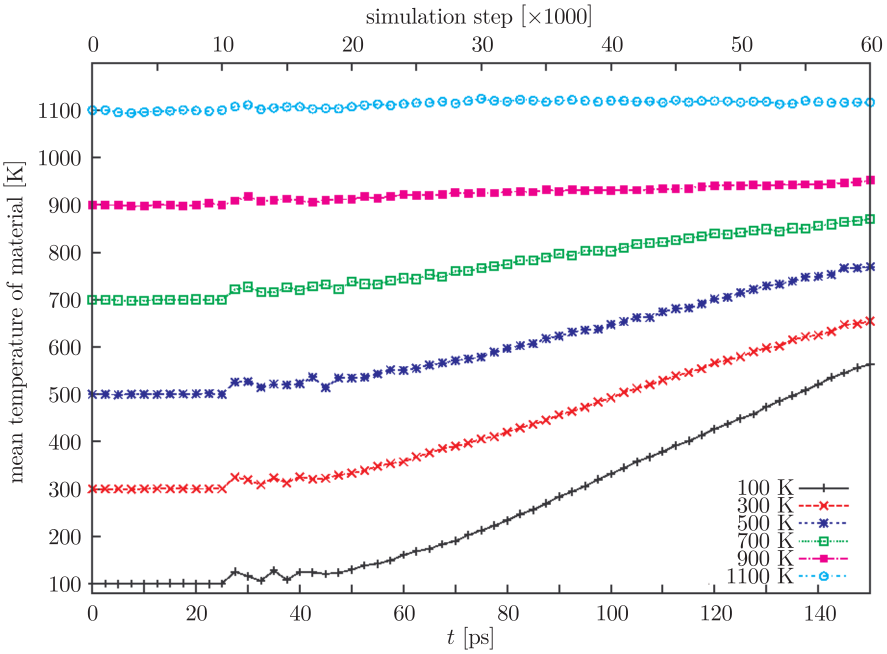

Figure 21. Mean temperature of machined material as function of time for various initial temperatures of the system. Machining depth $2a$, machining speed $v = 100\,\mathrm{m/s}$, tool type A

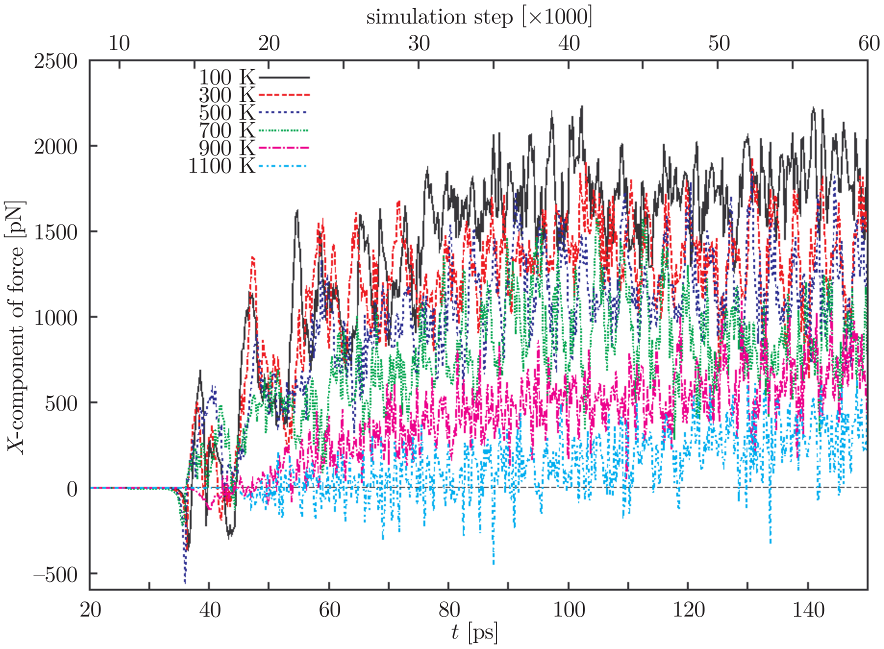

Figure 22. Force component X as a function of time for various initial temperatures of the system. Machining depth $2a$, machining speed $v = 100\,\mathrm{m/s}$, tool type A

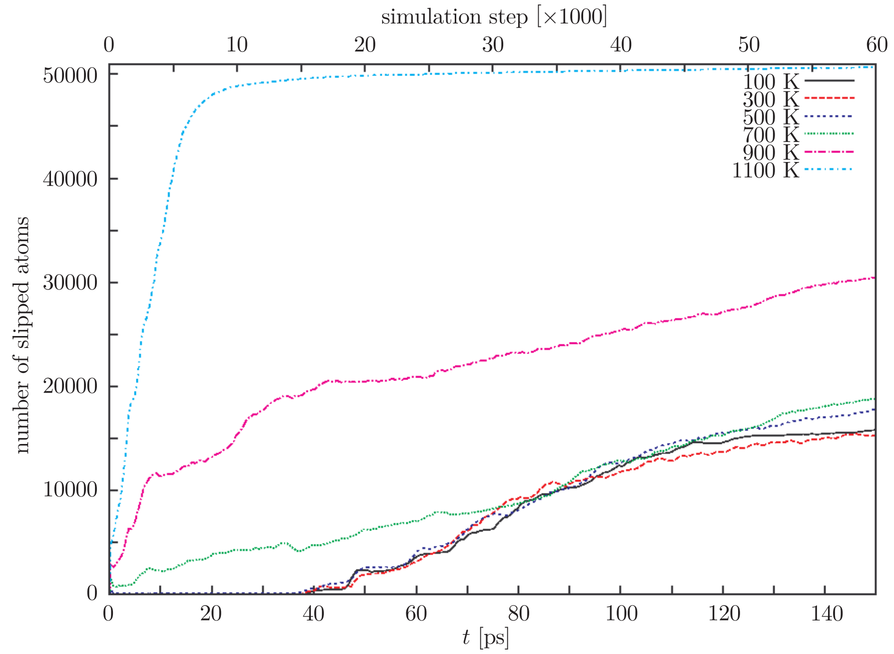

Figure 23. Number of slipped atoms as a function of time, for various initial temperatures of the system

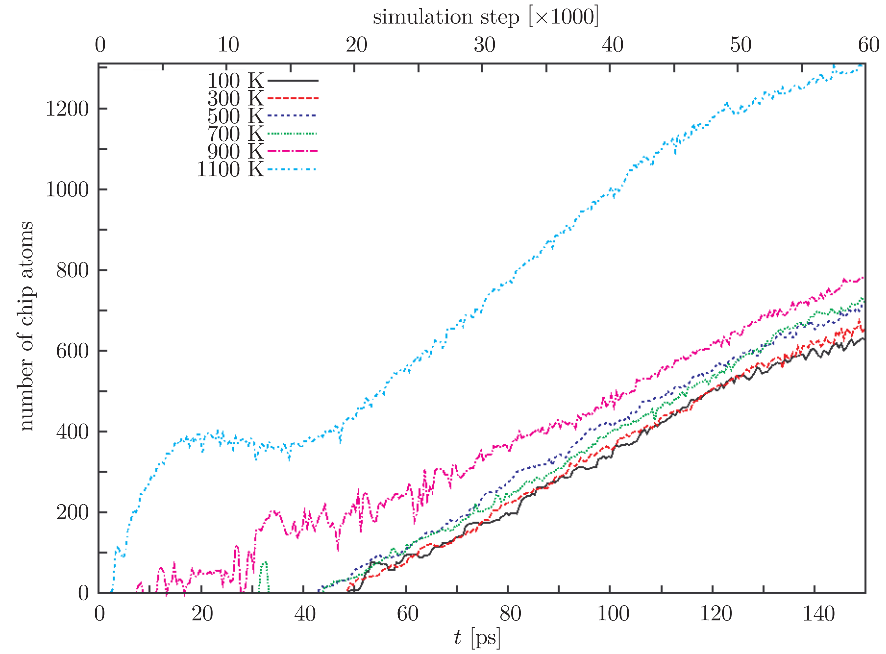

Figure 24. Accumulation of slipped atoms in the chip as function of time, for various initial temperatures of the system

whenever machined systems with smaller initial temperatures reach a temperature of 700K or higher locally in the machined area, the process should be interpreted taking into account spontaneous destabilization of the structure of the workmaterial.

4.2. Effect of tool speed on machining

Test simulations with various machining speeds, 10m/s, 50m/s, 100m/s, 200m/s, 500m/s and 1000m/s were performed, with the remaining parameters kept fixed. Figures 25–30 show the fields of total displacement of atoms for three selected machining speeds. It is apparent that the total number of atoms which have moved to large distances during machining clearly decreases with an increase in the machining speed (*cf.* Figures 31 and 32).

This is mainly related to the size of the chip: the total number of slipped atoms as a function of (the logarithm of the) machining speed for various points in time is shown in Figure 33, a similar plot for the number of slipped atoms constituting the chip only is shown in Figure 34.

It is apparent from Figure 33 that the number of displaced atoms at 25ps does not depend on the machining speed: it stays at a constant level and it is to a greater extent linked with a spontaneous relaxation of the workmaterial edge than with machining. However, for the times corresponding to the full penetration of the tool into the material there is a certain range of intermediate machining speeds for which the total number of relocated atoms is smaller than for the highest and lowest speeds. At the same time it can be seen (*cf.* Figure 34) that the number of atoms forming a chip is systematically smaller with an increase in speed. This means that more and more atoms of the workmaterial located below the tool are being displaced and the chip size decreases with an increase in the machining speed.

Figure 41 shows the Cartesian force component x for various machining speeds. It can be seen that the material becomes harder to machine at extreme speeds (the scale on the x-axis) is in reference to the nominal simulation, *i.e.* at $v = 100$m/s. For other speeds the simulation times and numbers of steps are different In the figure, the plots are scaled so as to present the runs on a comparable footing, *e.g.* x for a $v = 10$m/s axis has been compressed ten times). Greater resistance with very high machining speeds may be atributed to the fact that the workmaterial atoms do not manage to leap into new equilibrium positions in time, while performing thermal vibrations, when yielding to the approaching tool. Another manifestation of the hardening effect is the acute maximum of the force occurring for the highest speed considered (1000m/s) at the moment when the tool edges clash with the workmaterial.

Figures 35−40 illustrate the temperature distributions in the machined material for selected machining lengths for three different machining speeds: low, medium and high. For a speed of 10m/s, the material is heated at the expense of the work performed by the external force applied to it. At the same time, the process is slow to the extent that the temperature distribution in the machined

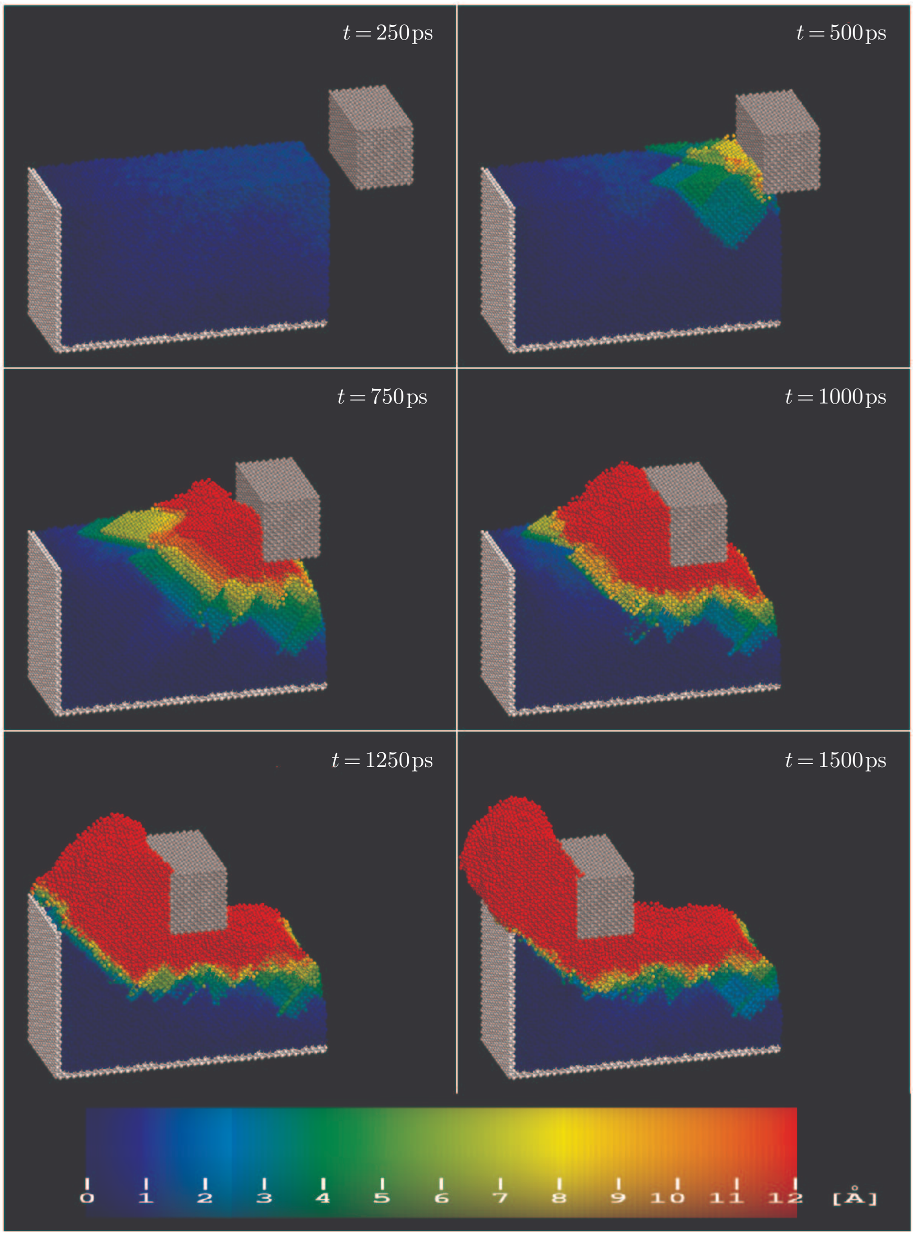

Figure 25. Atomic configurations at selected instants for the initial system temperature of $T = 300$K. Machining depth $2a$, machining speed $v = 10$m/s

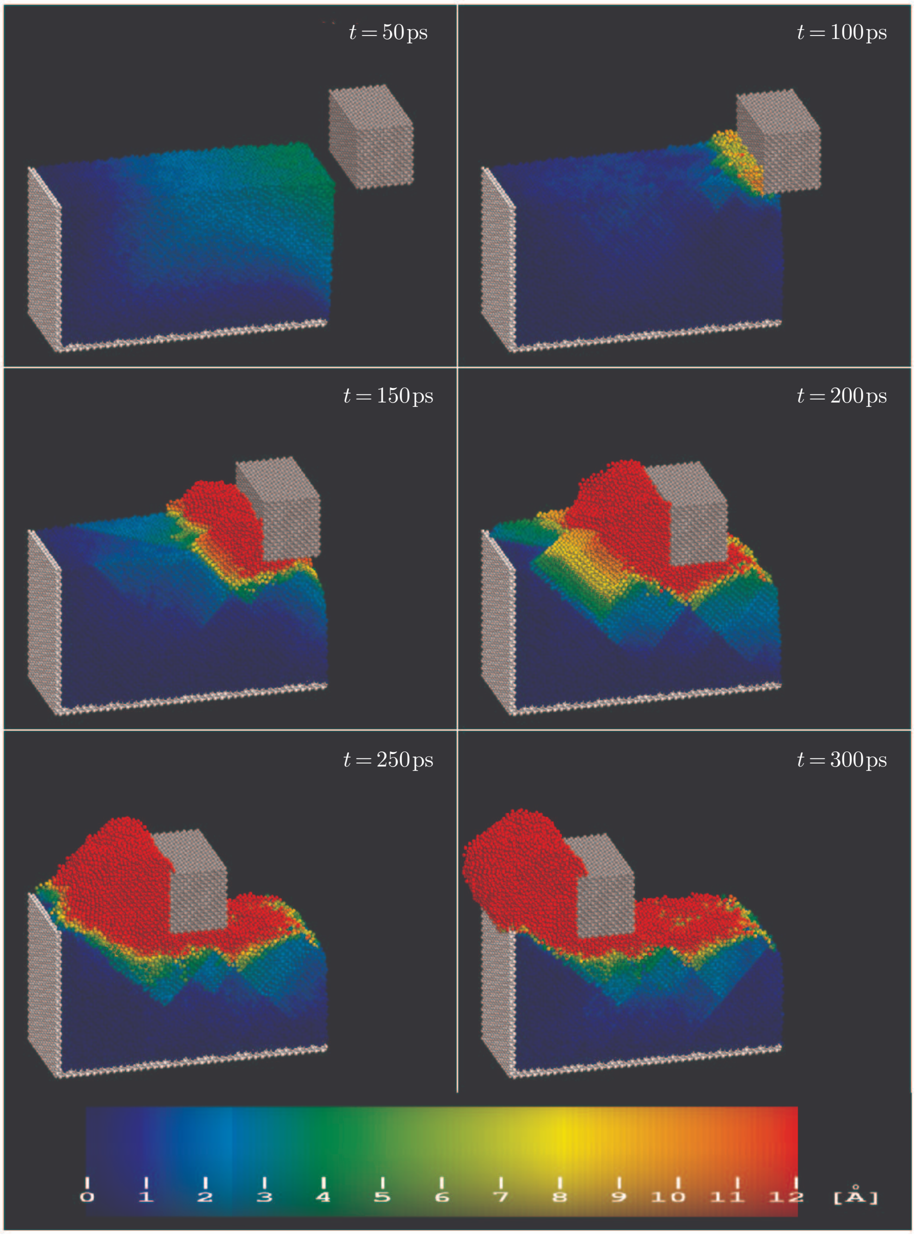

Figure 26. Atomic configurations at selected instants for the initial system temperature of $T=300$K. Machining depth $2a$, machining speed $v=50$m/s

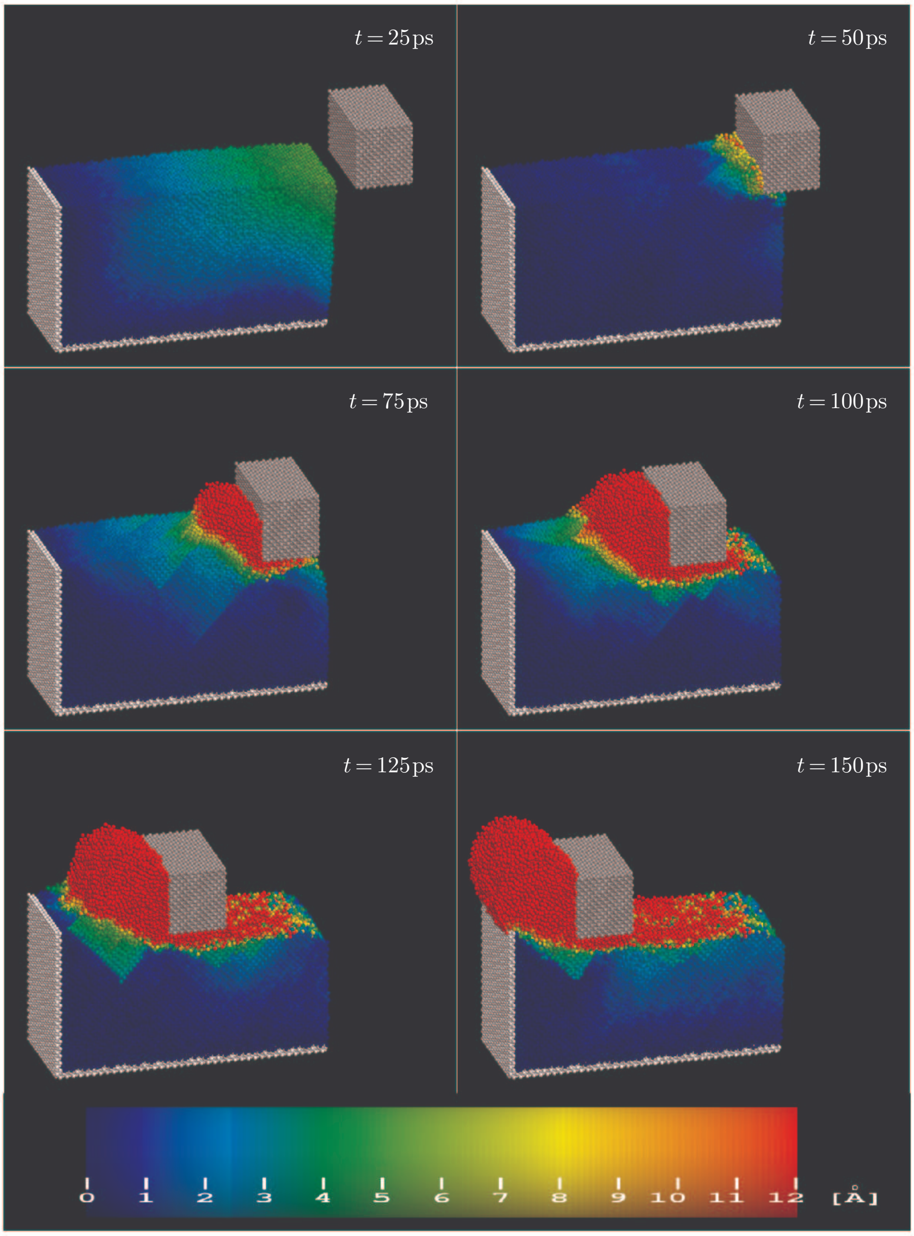

Figure 27. Atomic configurations at selected instants for the initial system temperature of $T = 300$K. Machining depth $2a$, machining speed $v = 100$m/s

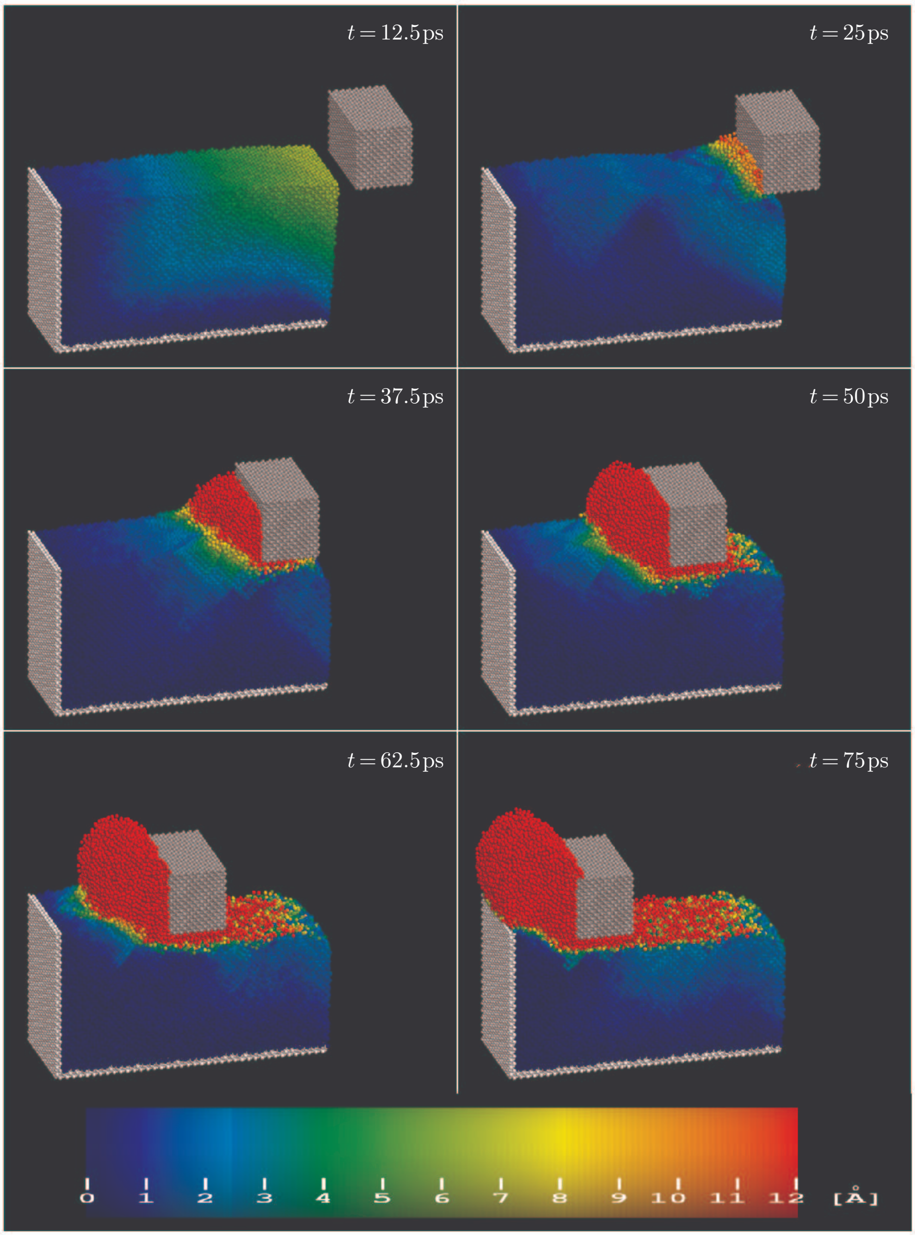

Figure 28. Atomic configurations at selected instants for the initial system temperature of $T = 300\,\mathrm{K}$. Machining depth $2a$, machining speed $v = 200\,\mathrm{m/s}$

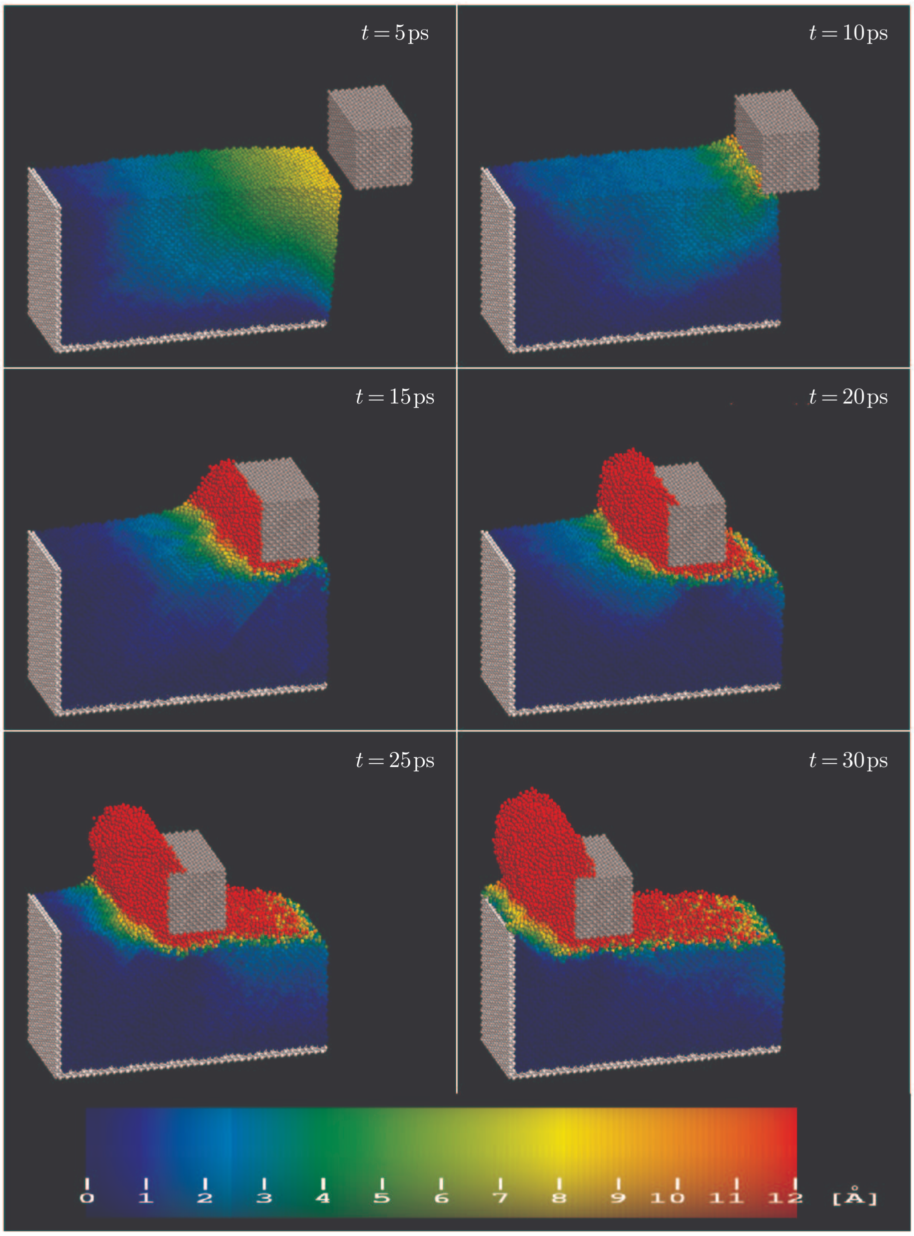

Figure 29. Atomic configurations at selected instants for the initial system temperature of $T = 300$K. Machining depth $2a$, machining speed $v = 500$m/s

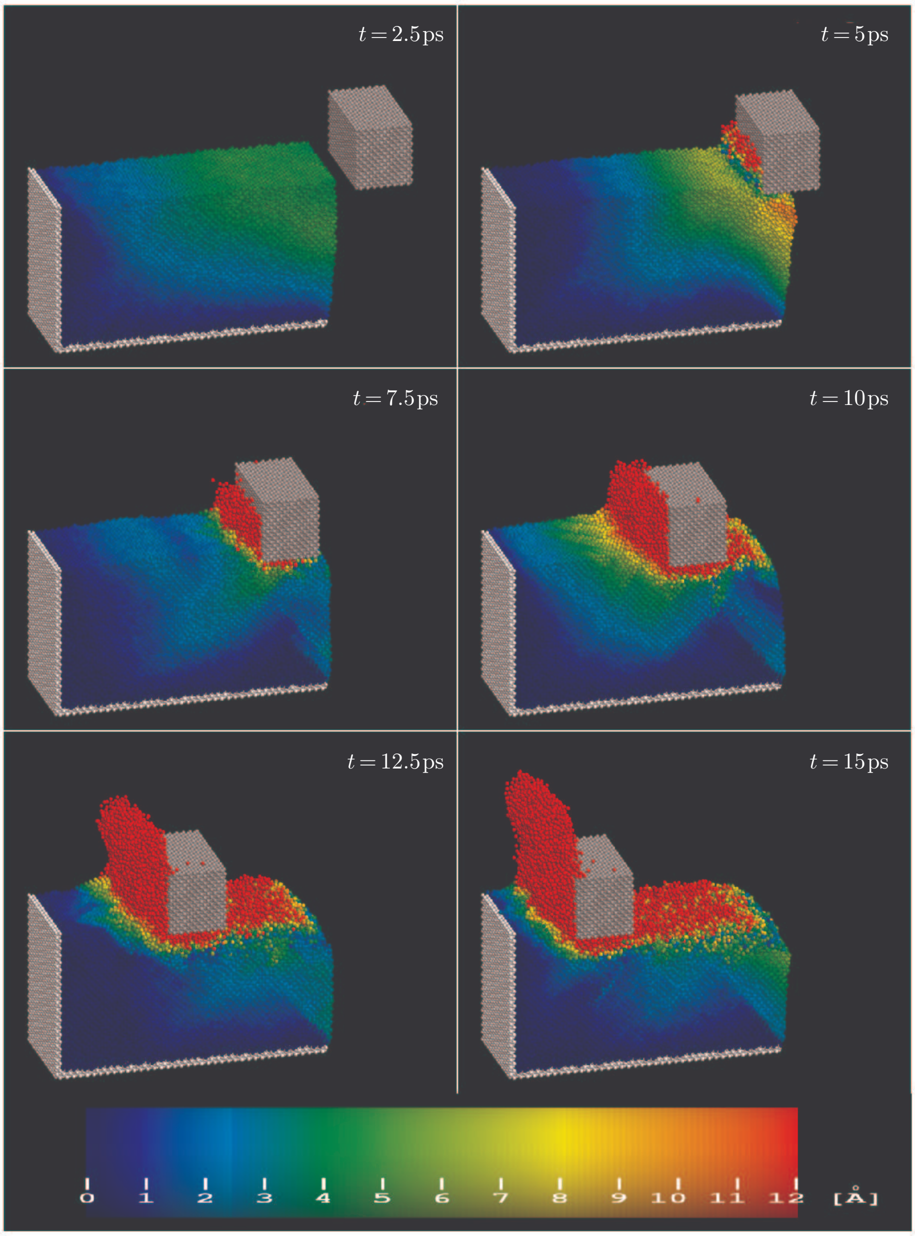

Figure 30. Atomic configurations at selected instants for the initial system temperature of $T = 300$K. Machining depth $2a$, machining speed $v = 1000$m/s

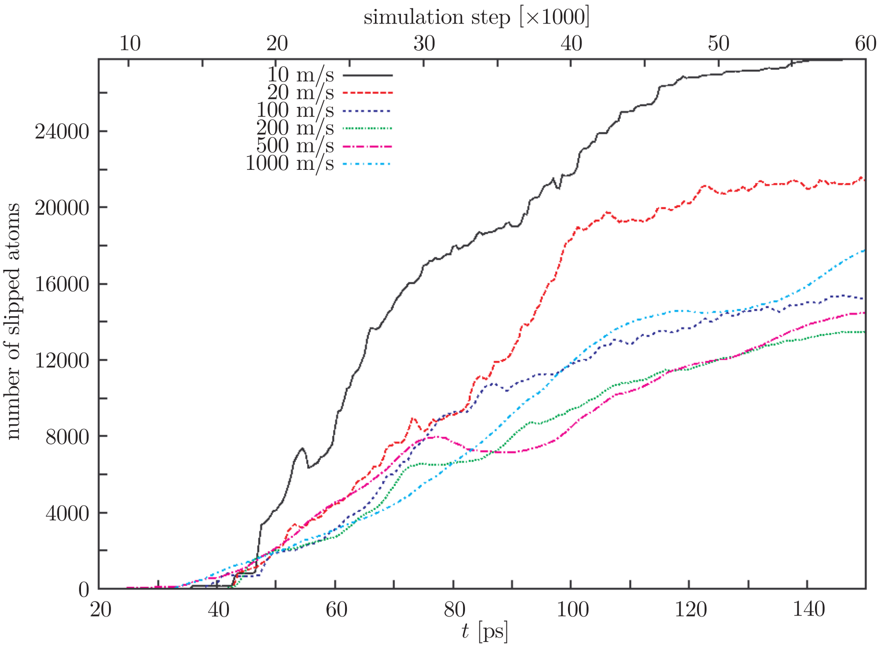

Figure 31. Total number of "slipped" atoms for various machining speeds

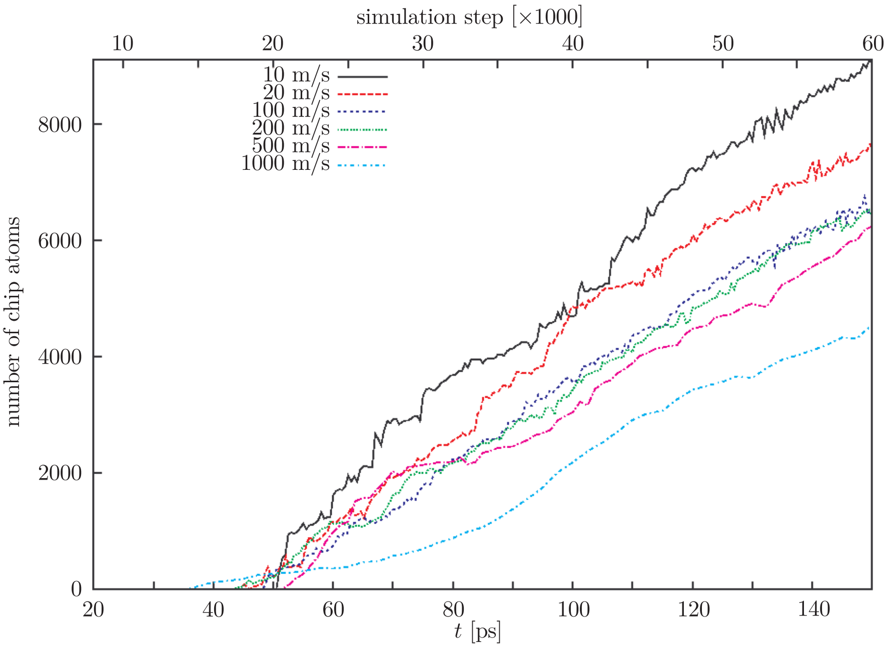

Figure 32. The size of the chip for various machining velocities

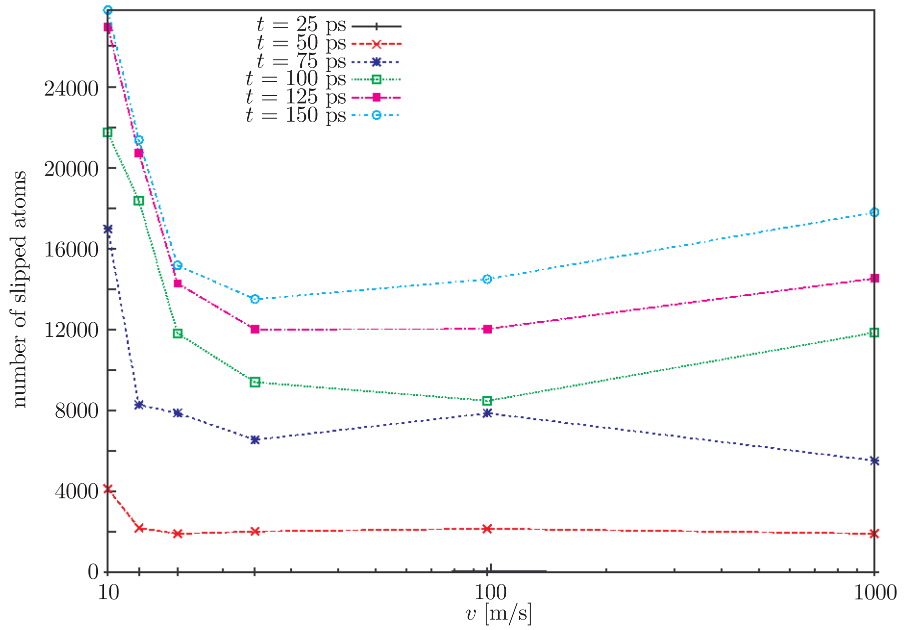

Figure 33. Total number of "slipped" atoms for various machining speeds

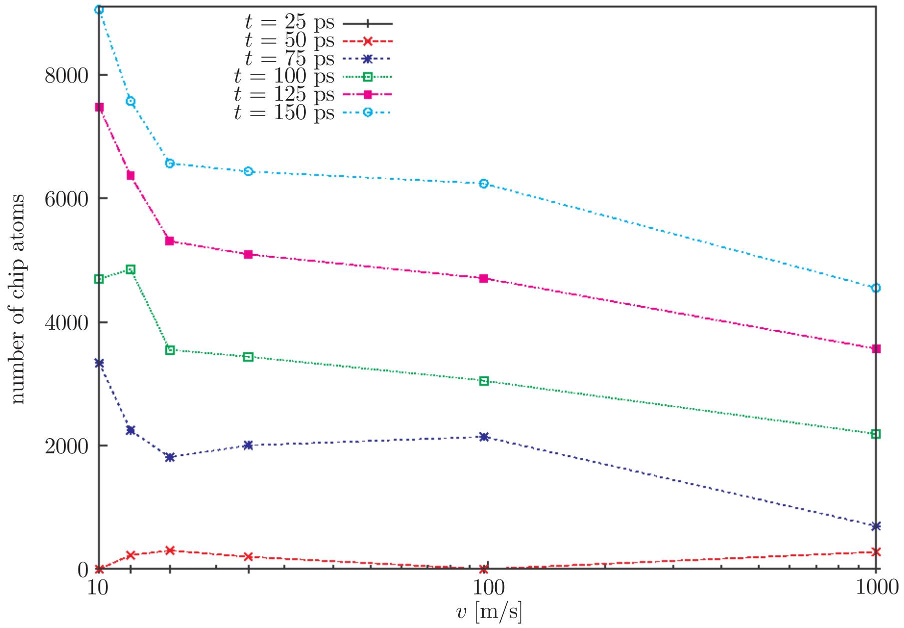

Figure 34. The size of the chip for various machining speeds

material is almost homogeneous at all times. The temperature of the chip is equal to the temperature of the machined material. With a high machining speed the temperature distribution is strongly heterogeneous, the material is strongly heated only in the machining area (under the tool and in front of the approaching edge). The temperature of the chip is also high.

Figure 42 shows the changes of the mean temperature of the machined material with time, for increasing machining speeds. The machined material heats up systematically, and the higher the machining speed, the more intense the heating (as the force needed to deform the machined material and the work performed thereon increase). The behaviour of the system for short times is very interesting which is best seen at the lowest machining speed. In this case, the tool approaches slowly enough so that there is time for a temporary quasi-equilibrium to develop between the workmaterial and the tool, which is by that time close enough to attract the upper-right corner of the material. The stretching of the bonds between the corner atoms slows their motion down which translates to a local lowering of kinetic energy which is significant to the extent that it has impact on the mean temperature of the whole machined cube over which averaging takes place.

Following the initial tests, the configuration with the initial temperature of $T_0 = 300\,\mathrm{K}$ and the machining speed of $v = 100\,\mathrm{m/s}$ were selected for further systematic simulations. The chosen T_0 is low enough as not to cause spontaneous melting of the workmaterial and high enough (higher than the Debye temperature) to neglect the quantum aspects of lattice vibrations. The machining speed is high enough for the calculation times to be "reasonable". Subsequent production runs will investigate the effects of the machining depth, tool shape and workmaterial orientation on the machining process.

5. Results of the production runs

The production runs of ultraprecision machining were performed for six tool shapes, three machining depths and three crystallographic orientations of the workmaterial. This section contains a detailed description of the results of the performed simulations.

First (Section 5.1), a general description of the simulation results will be presented, based mainly on the visual assessment of the obtained atomic configurations. The forces acting on the tool, pressure and shear stress distributions, spatial temperature distributions and their evolution in time, development of slips will subsequently be discussed (Sections 5.2–5.5). Finally (Section 5.6), details of the formation of the chip will be discussed.

5.1. General description of the deformation process

Figures 43–60 show the fields of displacement of atoms at certain instants of the nanomachining process performed with various depths (1, 2 and 5 of the doubled smallest distances between crystallographic planes parallel to the upper

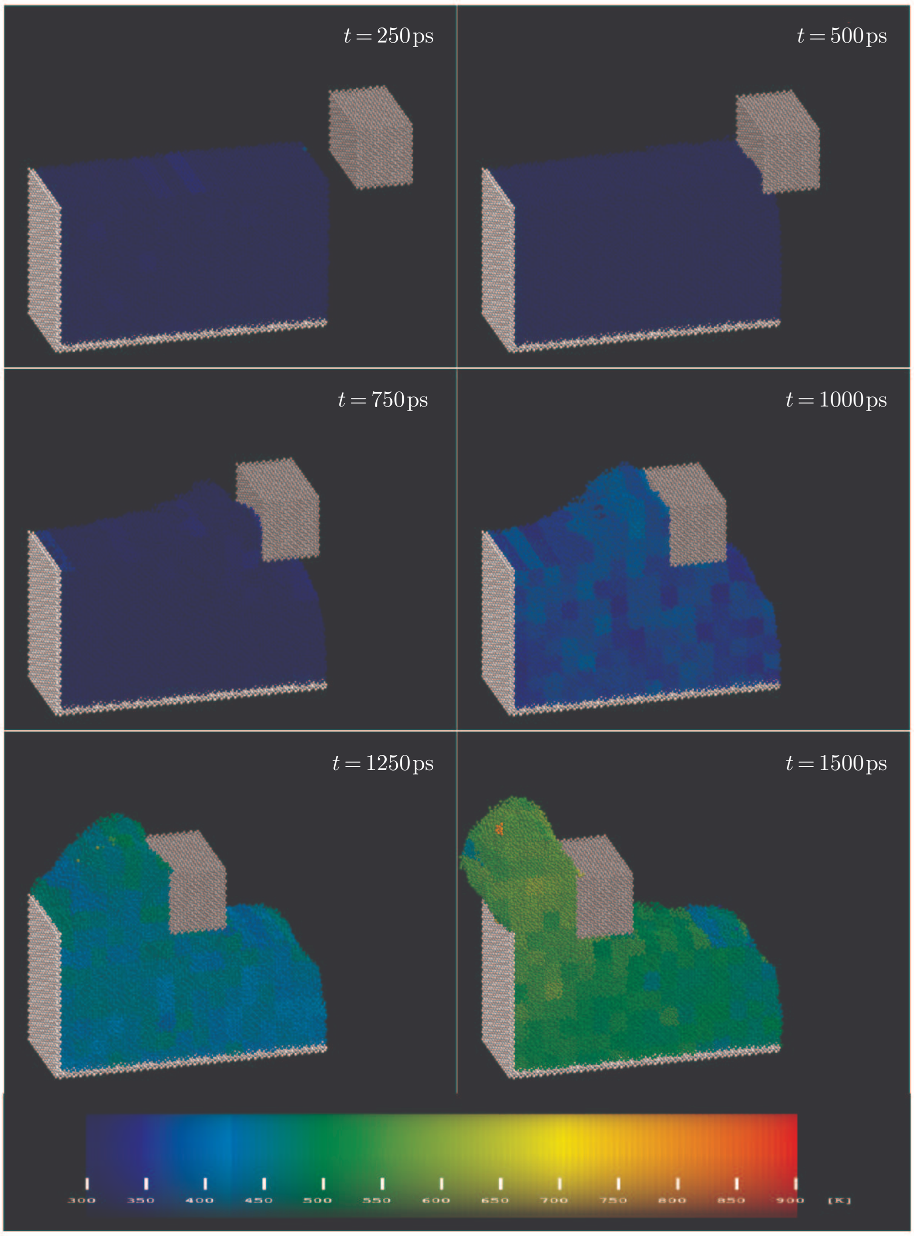

Figure 35. Temperature distributions at selected instants for the initial system temperature of $T = 300$ K. Machining depth $2a$, machining speed $v = 10$ m/s

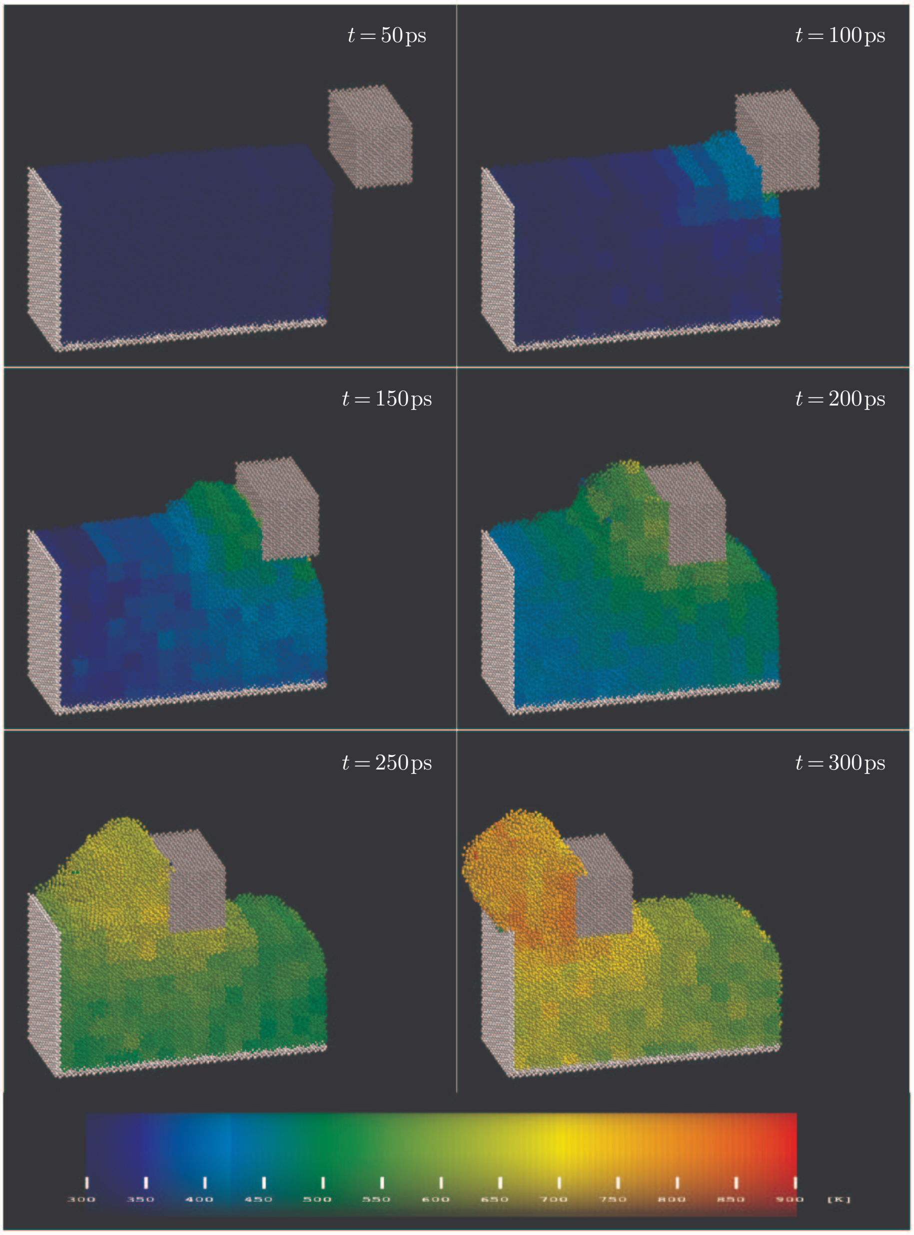

Figure 36. Temperature distributions at selected instants for the initial system temperature of $T = 300$K. Machining depth $2a$, machining speed $v = 50$m/s

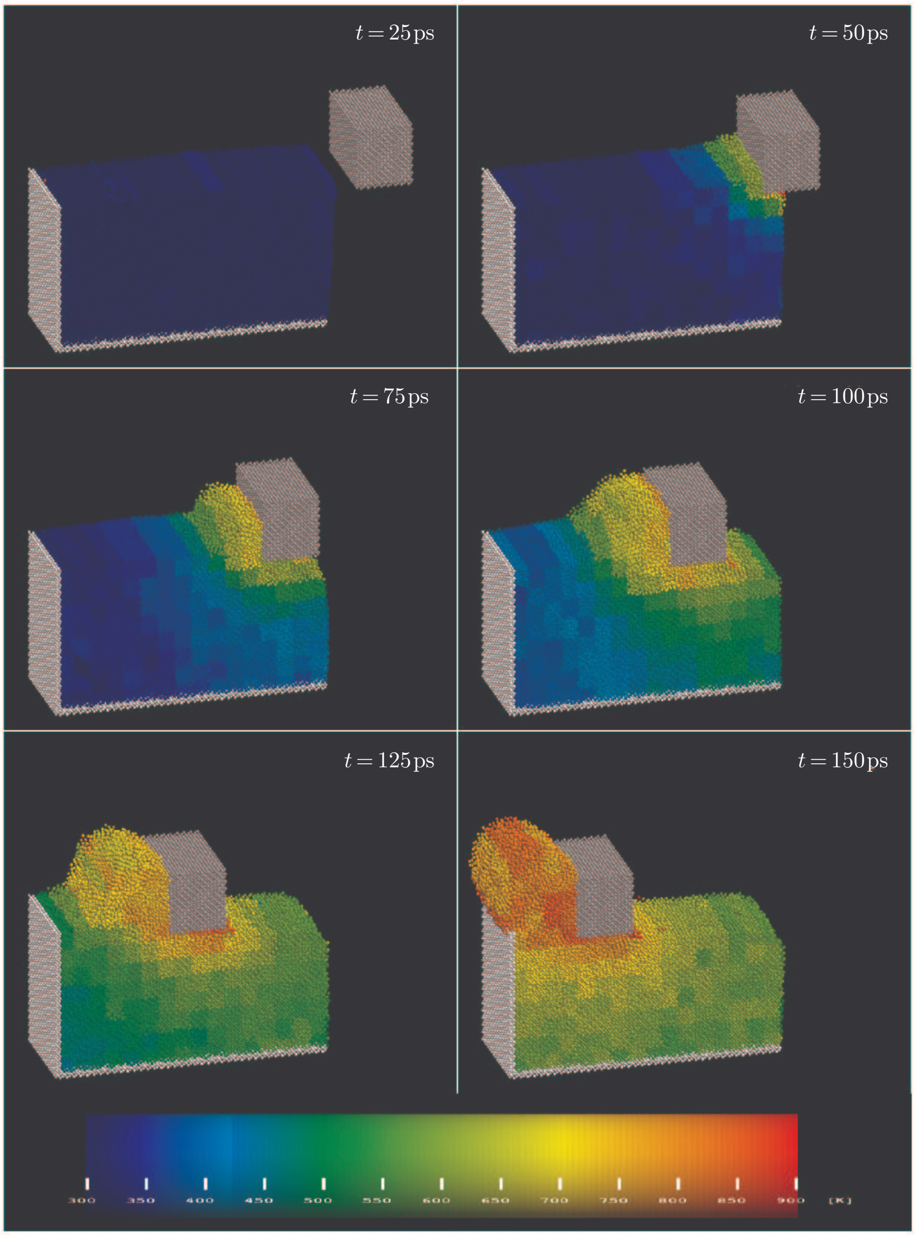

Figure 37. Temperature distributions at selected instants for the initial system temperature of $T = 300\,\mathrm{K}$. Machining depth $2a$, machining speed $v = 100\,\mathrm{m/s}$

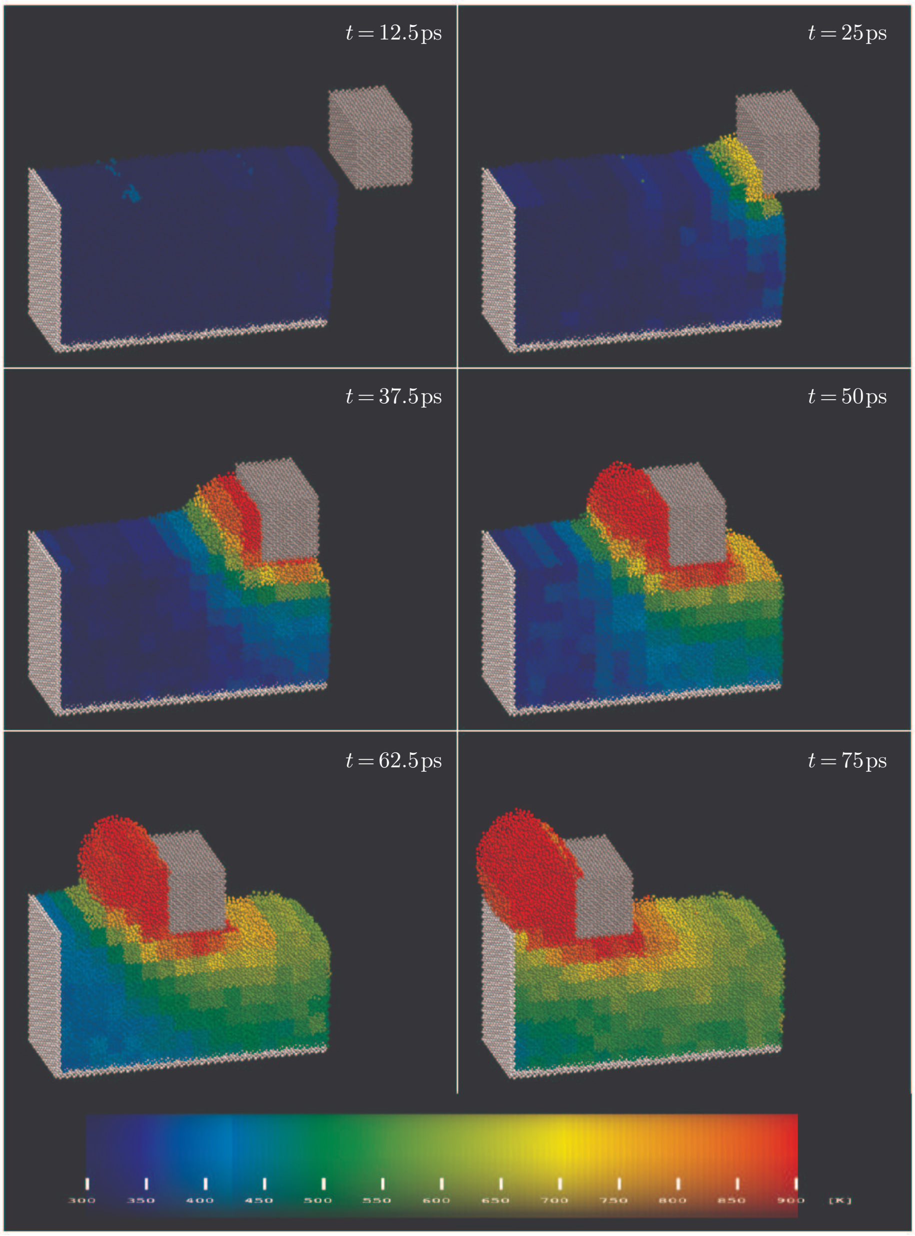

Figure 38. Temperature distributions at selected instants for the initial system temperature of $T = 300$K. Machining depth $2a$, machining speed $v = 200$m/s

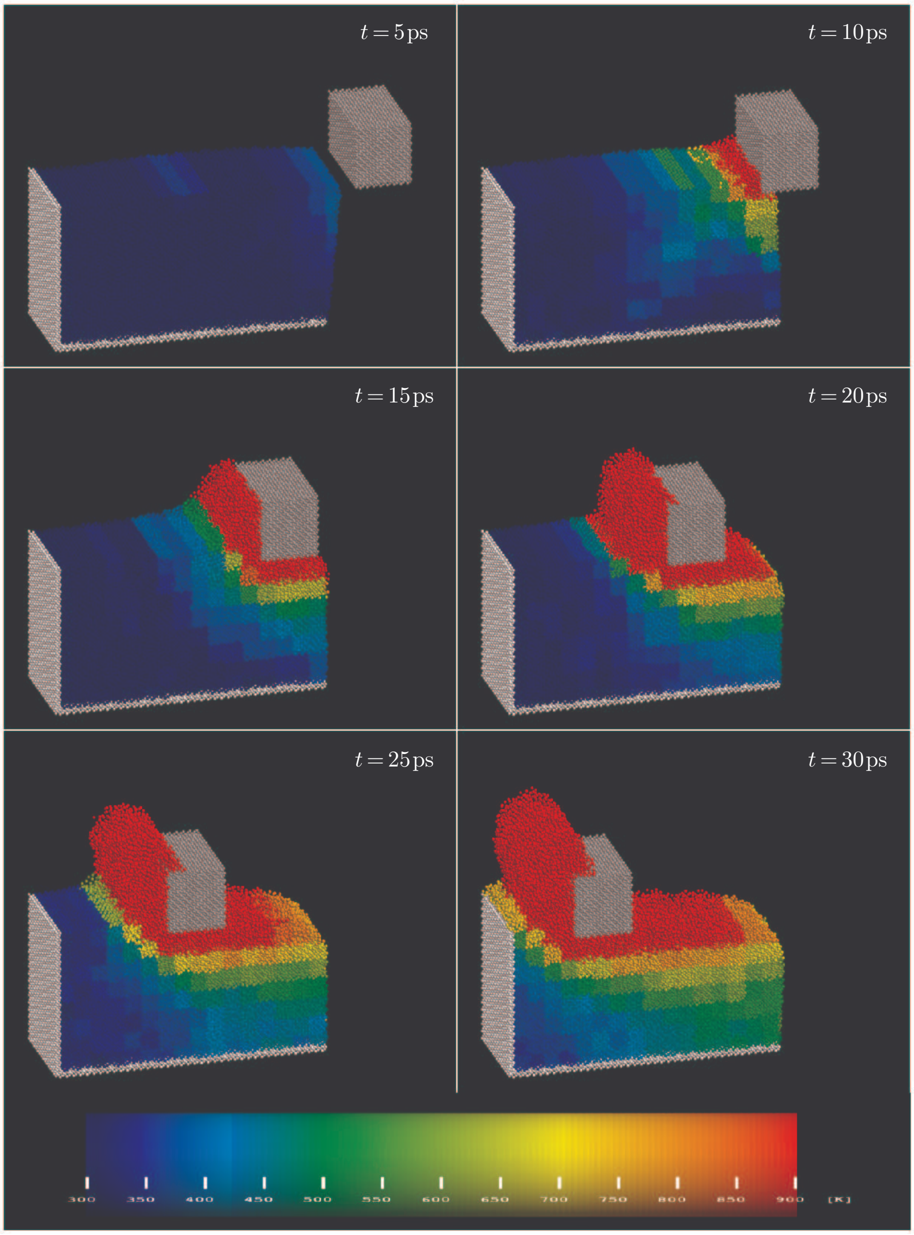

Figure 39. Temperature distributions at selected instants for the initial system temperature of $T = 300\,\mathrm{K}$. Machining depth $2a$, machining speed $v = 500\,\mathrm{m/s}$

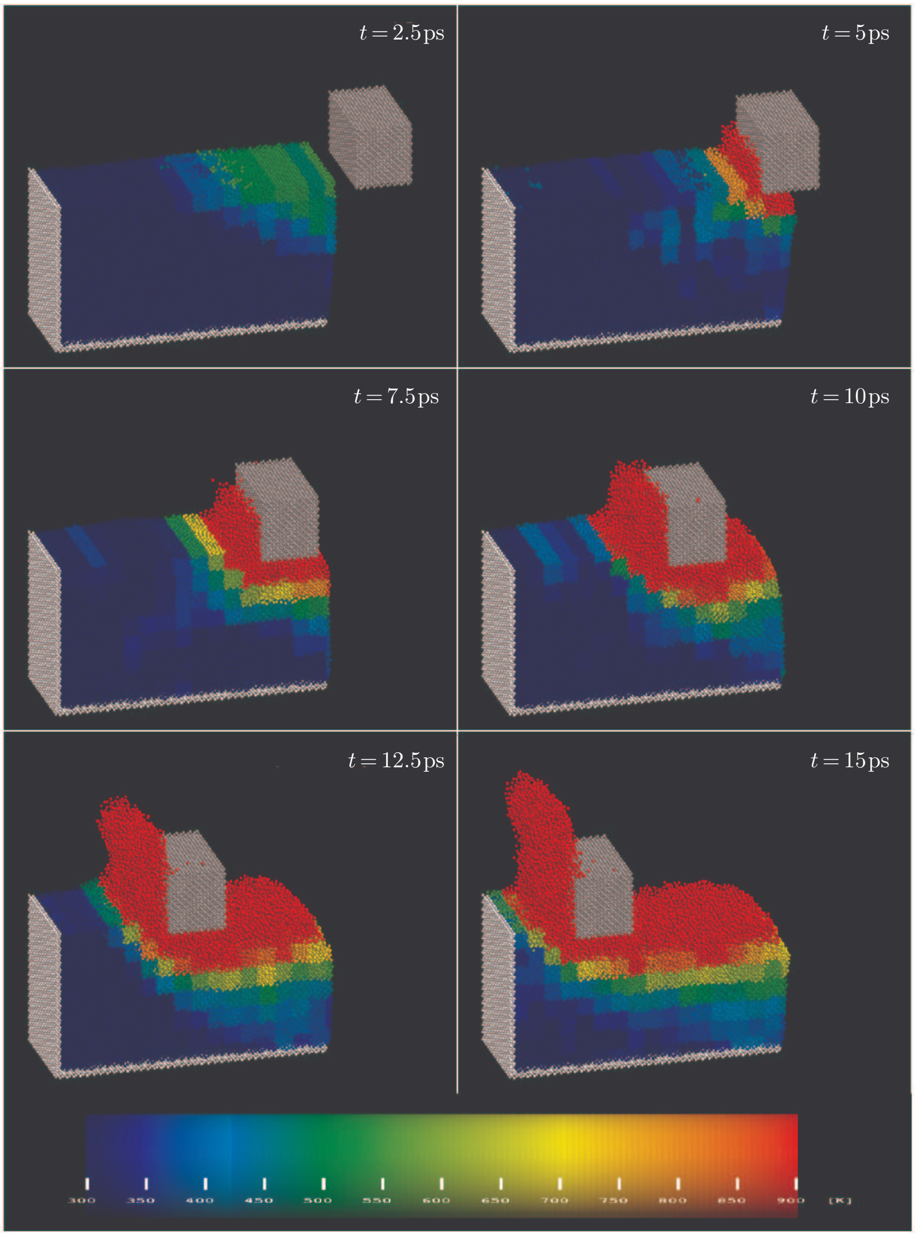

Figure 40. Temperature distributions at selected instants for the initial system temperature of $T = 300$K. Machining depth $2a$, machining speed $v = 1000$m/s

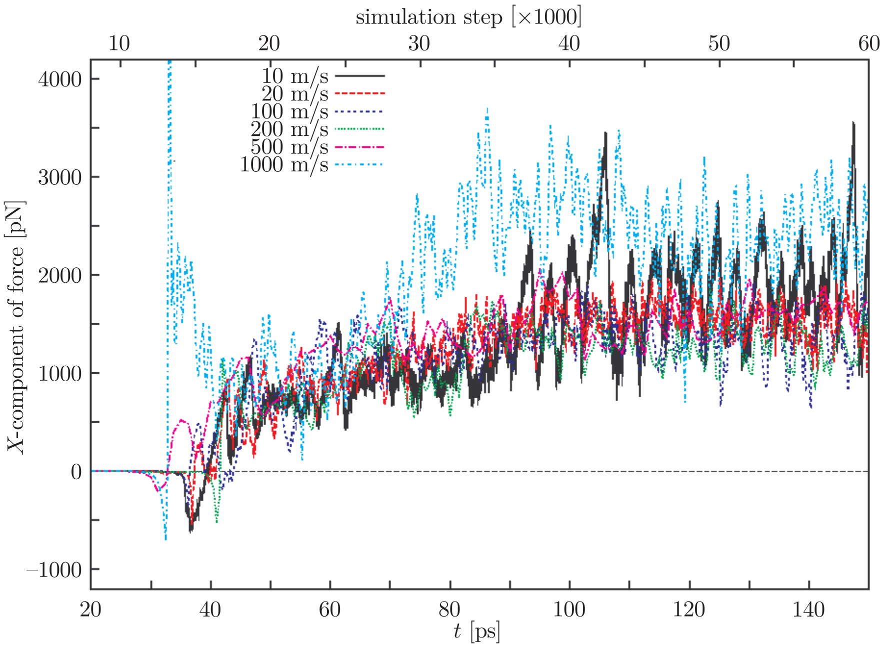

Figure 41. *X*-component of force vs. time for various machining speeds

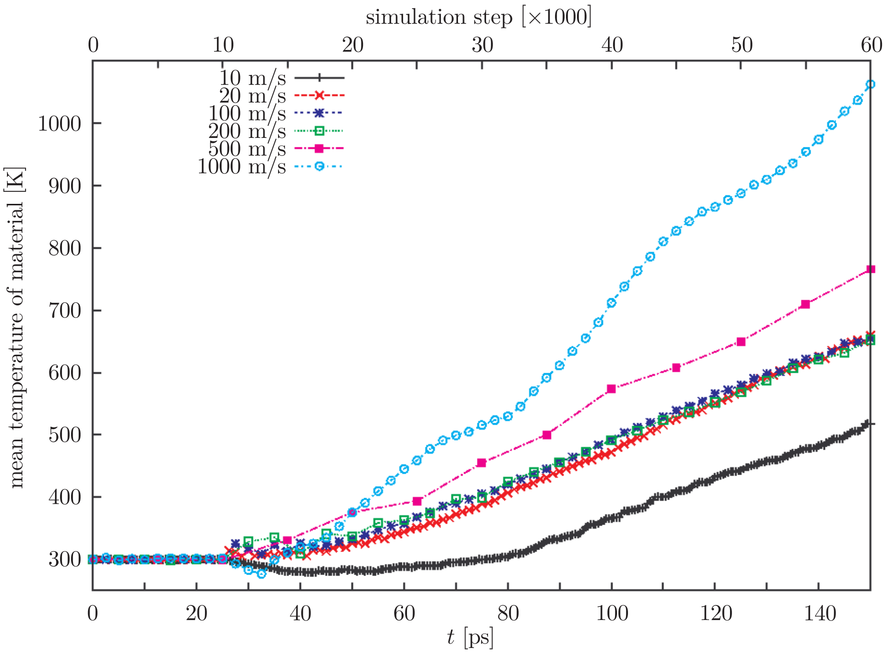

Figure 42. Average temperature of the workmaterial vs. time at various machining speeds

machined monocrystal surface, as described in Section 4) with rigid tools A–F of various shapes and orientations with respect to the workmaterial. Atoms marked in the figures in red have moved by more than 12Å with respect to their initial positions, which indicates they have "come from far away", from regions distant from their current positions. This is a trivial observation in case of the atoms forming the chip. A more interesting thing to observe is the "reddening" of the machined surface behind the tool. In case of tools with where the contact surface with the workmaterial was large (*i.e.* tool shapes A and C), the machined surface includes many atoms that have been displaced significantly. In case of tools with a small contact surface – atoms originating "from far away" are in significantly smaller numbers (tools B, D, E). Tool F represents an intermediate case. This effect is obvious: the larger the contact surface, the stronger and more numerous are the bonds of atoms with the workmaterial where the atoms move along with the tool after having been adsorbed. For tools with a small contact surface, the number of atoms "spread" on the machined surface of atoms grows with an increase in the machining depth.

5.2. Forces acting on the tool

The force that must be applied to the tool to maintain a constant machining speed is directly related to the contact surface. The components of the reactive force acting on the tool are shown in Figures 61 and 62.

Figure 61 presents the time variations of the x-component of this force for the six tool shapes and three machining depths under study.

In practice, as mentioned earlier, the atoms constituting the tool were moved by an application of a preset speed, and the force on the tool, which is the reactive force of the workmaterial was measured. Direct comparisons of forces acting on different tools should be performed carefully, as the interaction force changes depending on the number of atoms forming the tool, and this number is different for tools C, D, E and A, B, F. On the other hand, it is hardly sensible to compare forces per tool atom as, with the adopted cut-off radius (12Å) some of the tool atoms (*viz.*, those that are outside of the interaction cut-off) do not interact with the material, and may interact only with the chip that ascends the tool during the final phases of machining at large depths.

The graphs in Figure 61 show that the contact surface between the tool and the workmaterial significantly affects the machining process. By referring to the magnitude of the mean x-component of the force, we conclude that the machining process is the easiest when tools D, B, E, and then C, A, F are used.

Tools D, B and E do not have a horizontal base and their contact area with the workmaterial is much smaller compared to the other tool shapes. Tool D attacks the material with a vertical side, thus merely pushing the chip horizontally, without forcing it into the workmaterial (unlike tools B, E, and F), offering the least machining resistance.

Machining with tools C, A and F necessitates the application of a greater force due to a larger contact surface of the tool and the material; shape C is

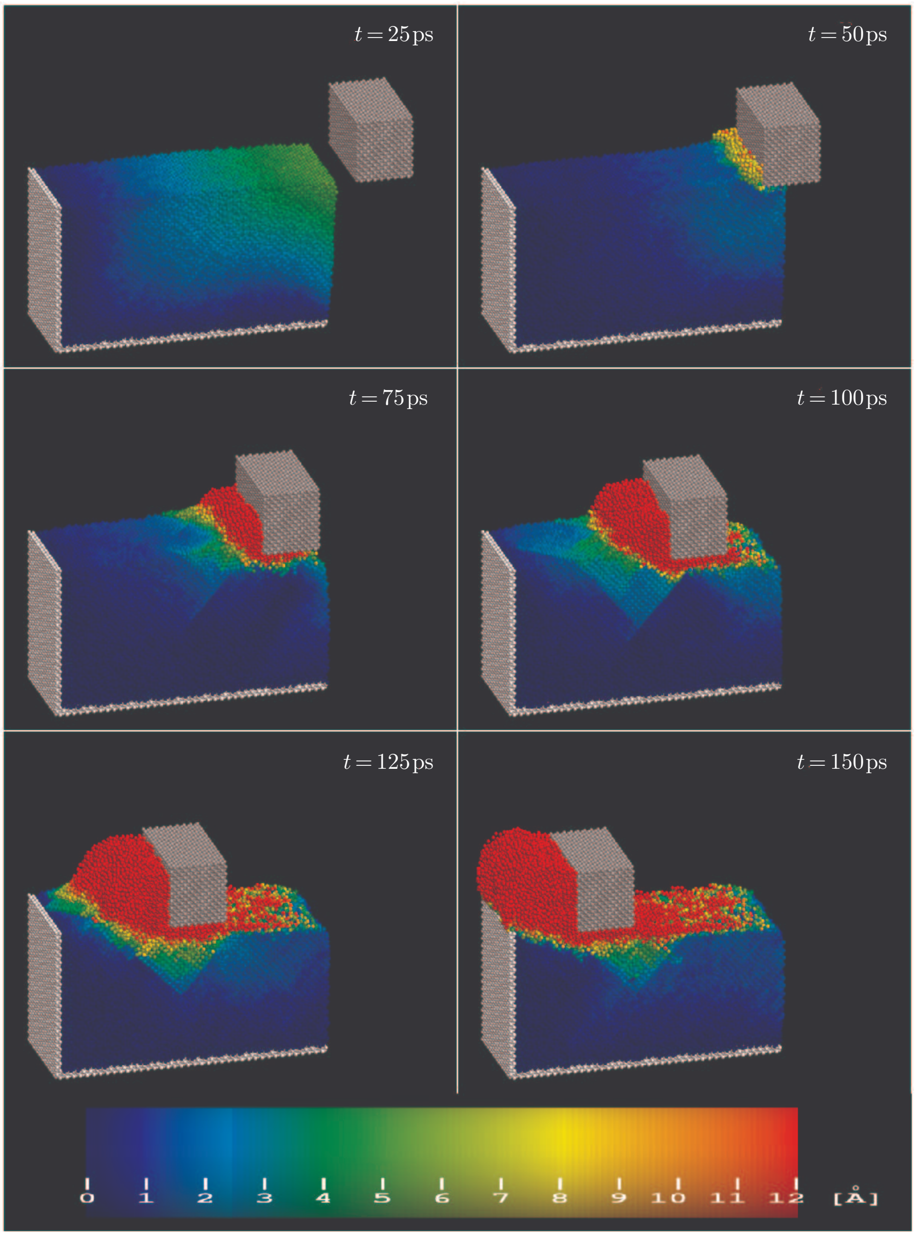

Figure 43. Atomic configurations at selected instants for workmaterial surface orientation (100). Tool A, machining depth $1a$

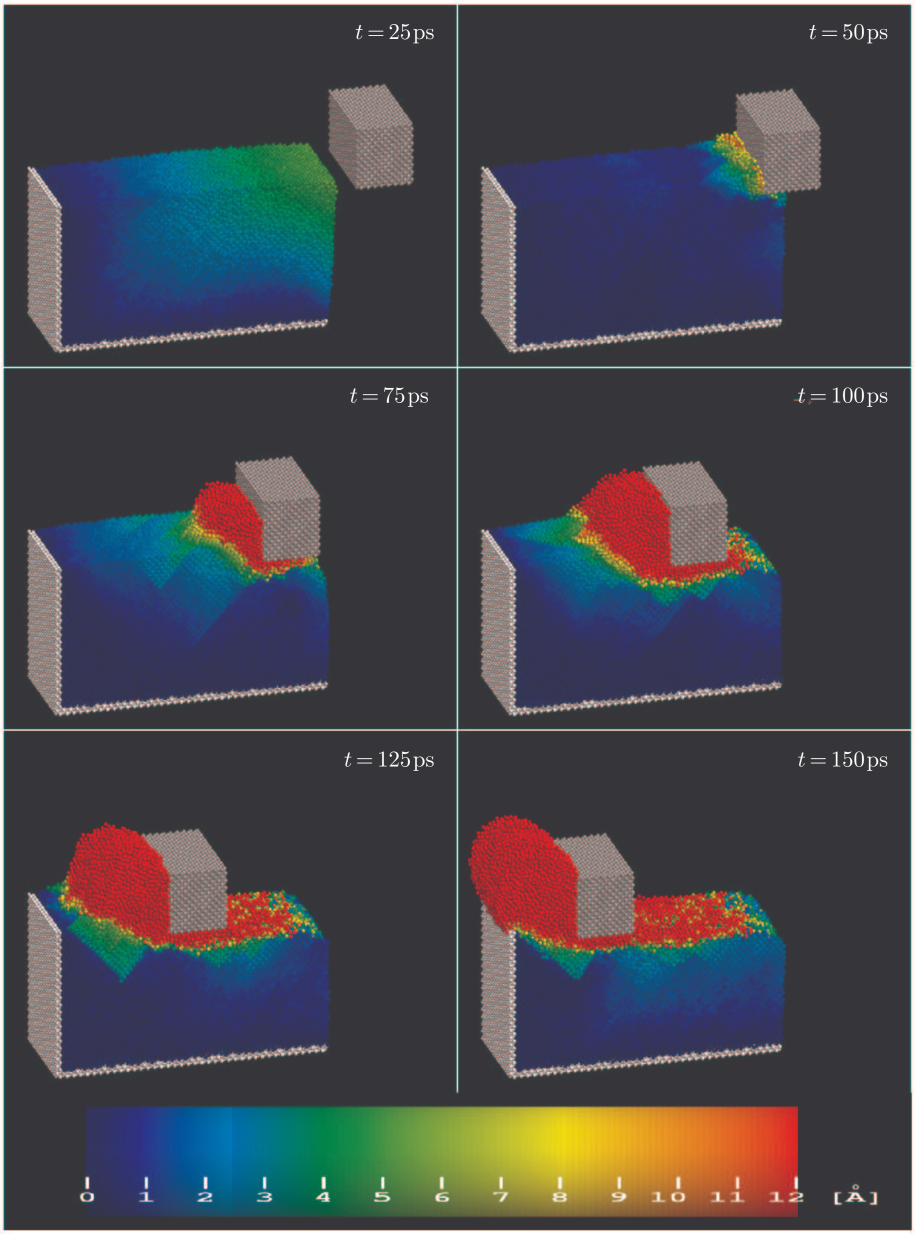

Figure 44. Atomic configurations at selected instants for workmaterial surface orientation (100). Tool A, machining depth $2a$

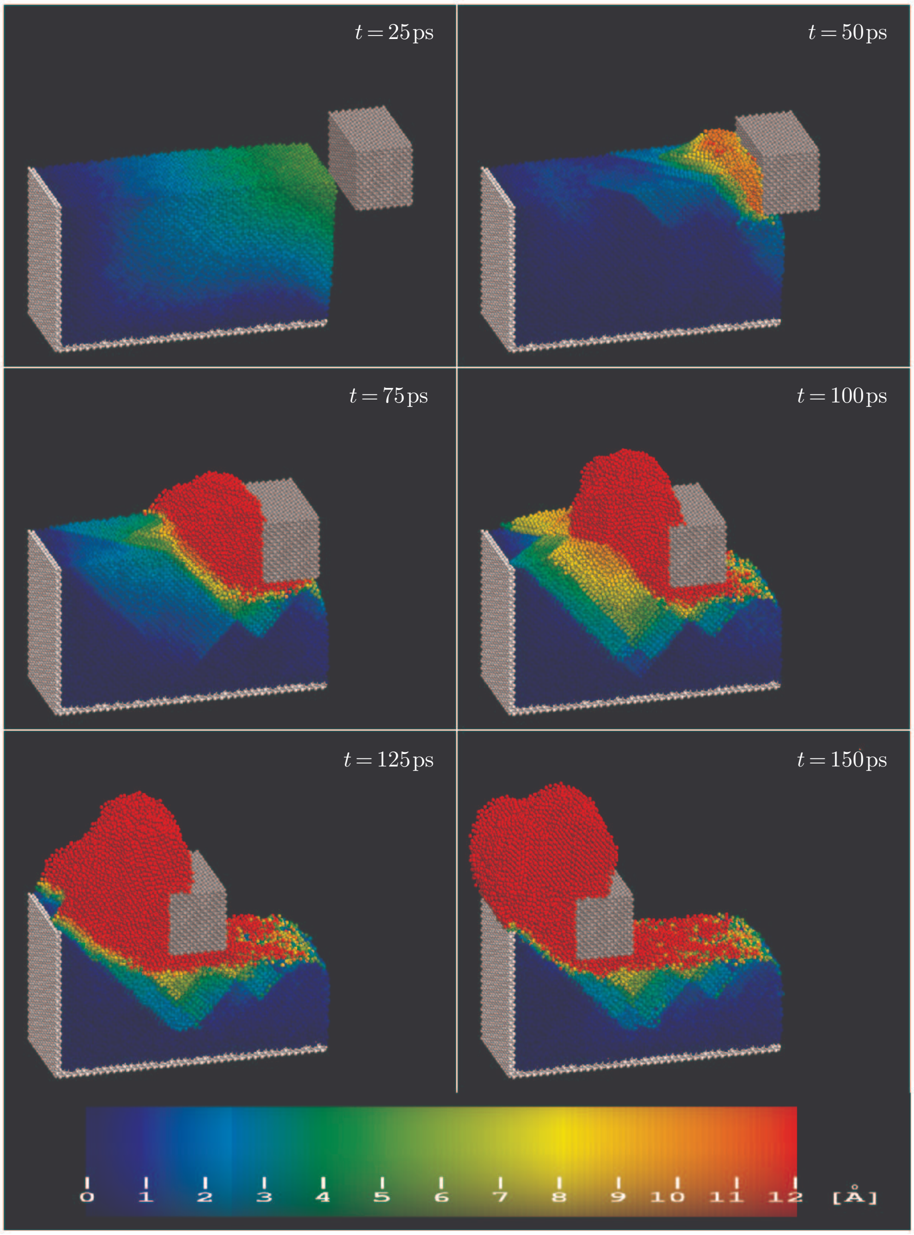

Figure 45. Atomic configurations at selected instants for workmaterial surface orientation (100). Tool A, machining depth $5a$

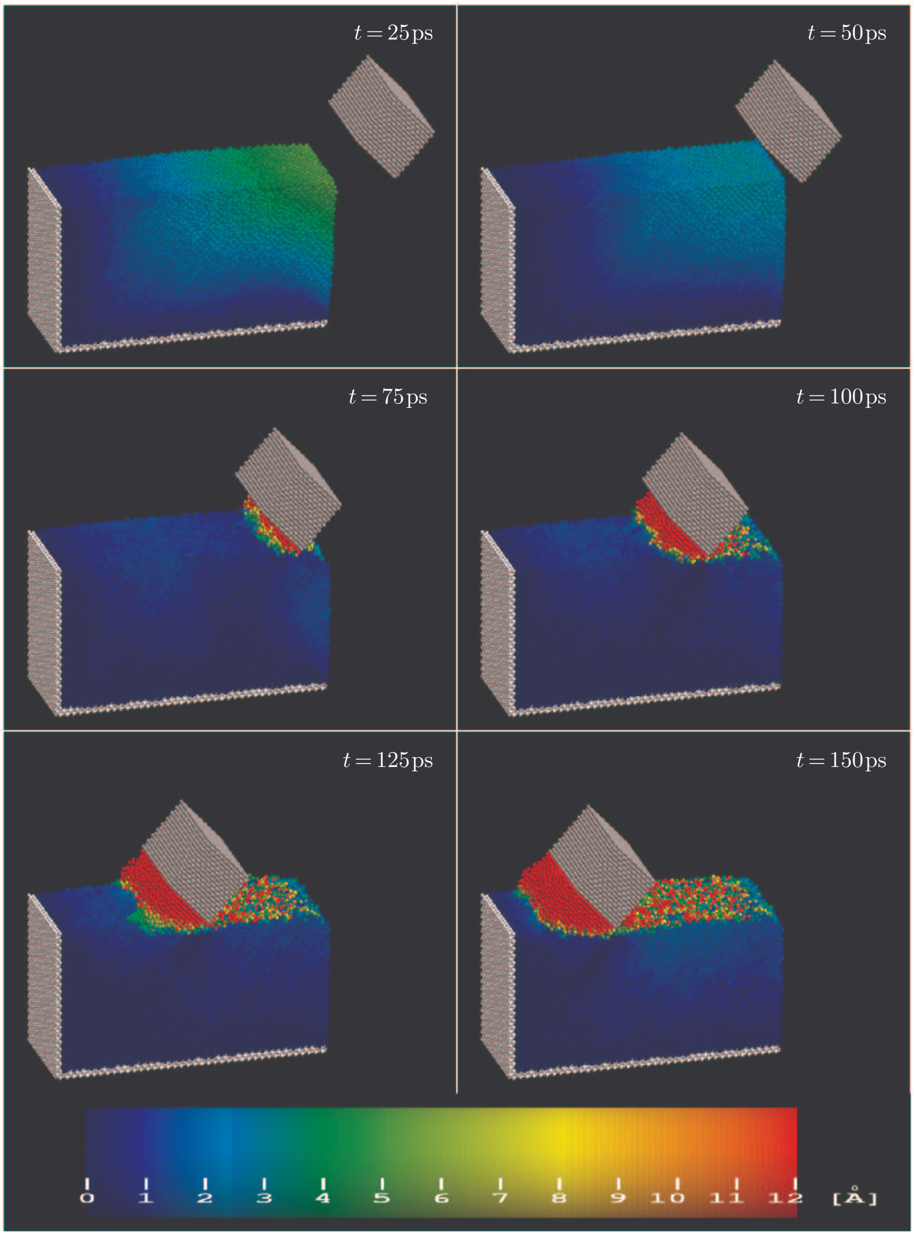

Figure 46. Atomic configurations at selected instants for workmaterial surface orientation (100). Tool B, machining depth $1a$

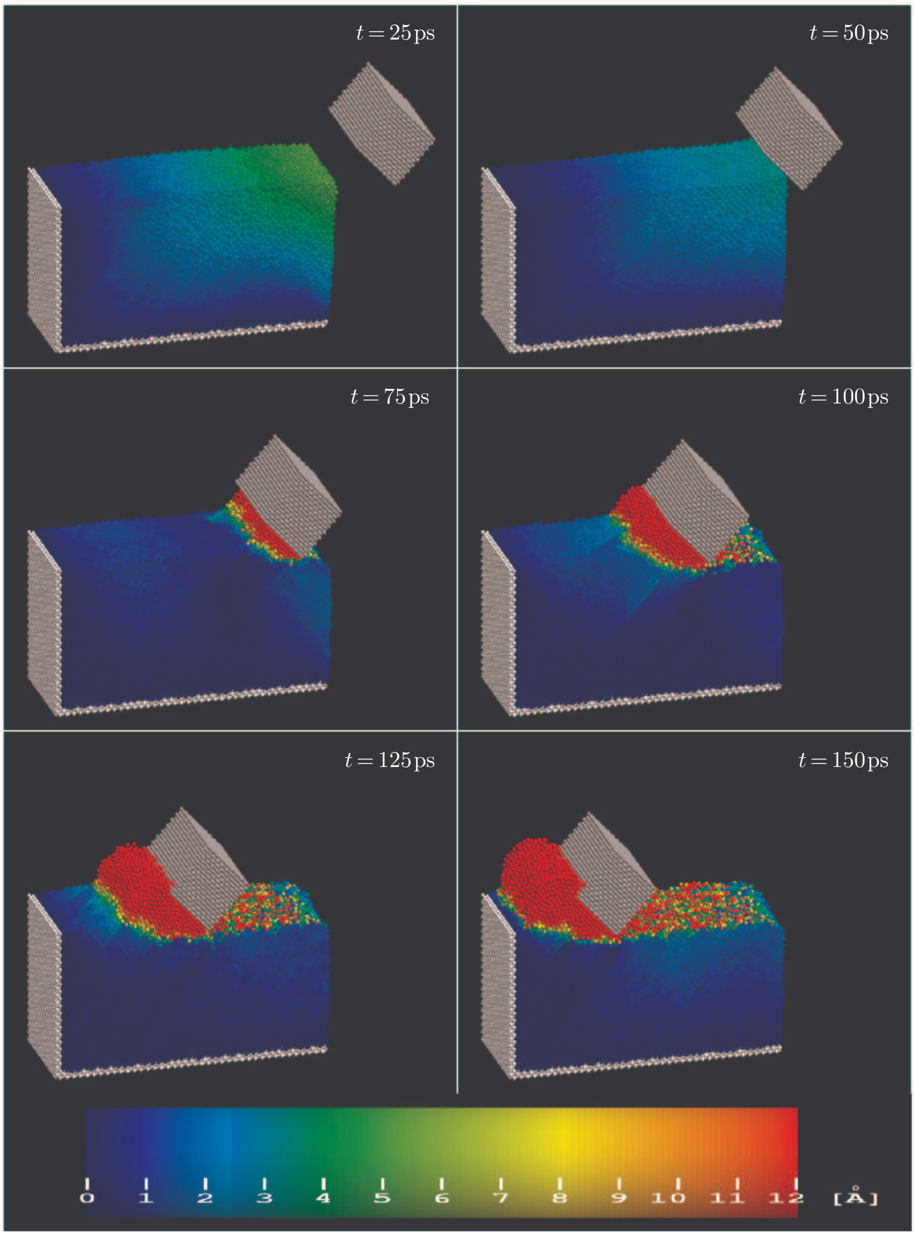

Figure 47. Atomic configurations at selected instants for workmaterial surface orientation (100). Machining tool B, machining depth $2a$

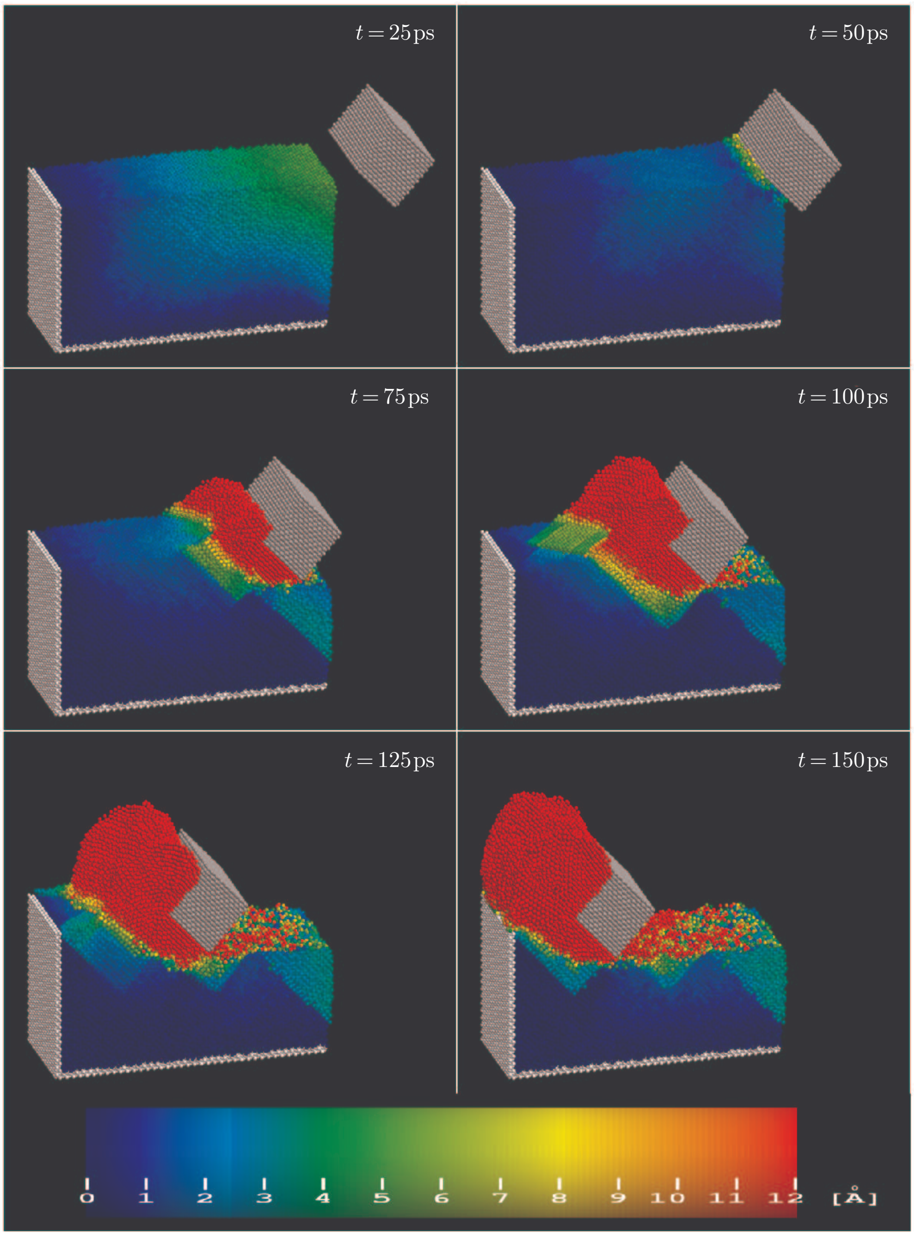

Figure 48. Atomic configurations at selected instants for workmaterial surface orientation (100). Tool B, machining depth $5a$

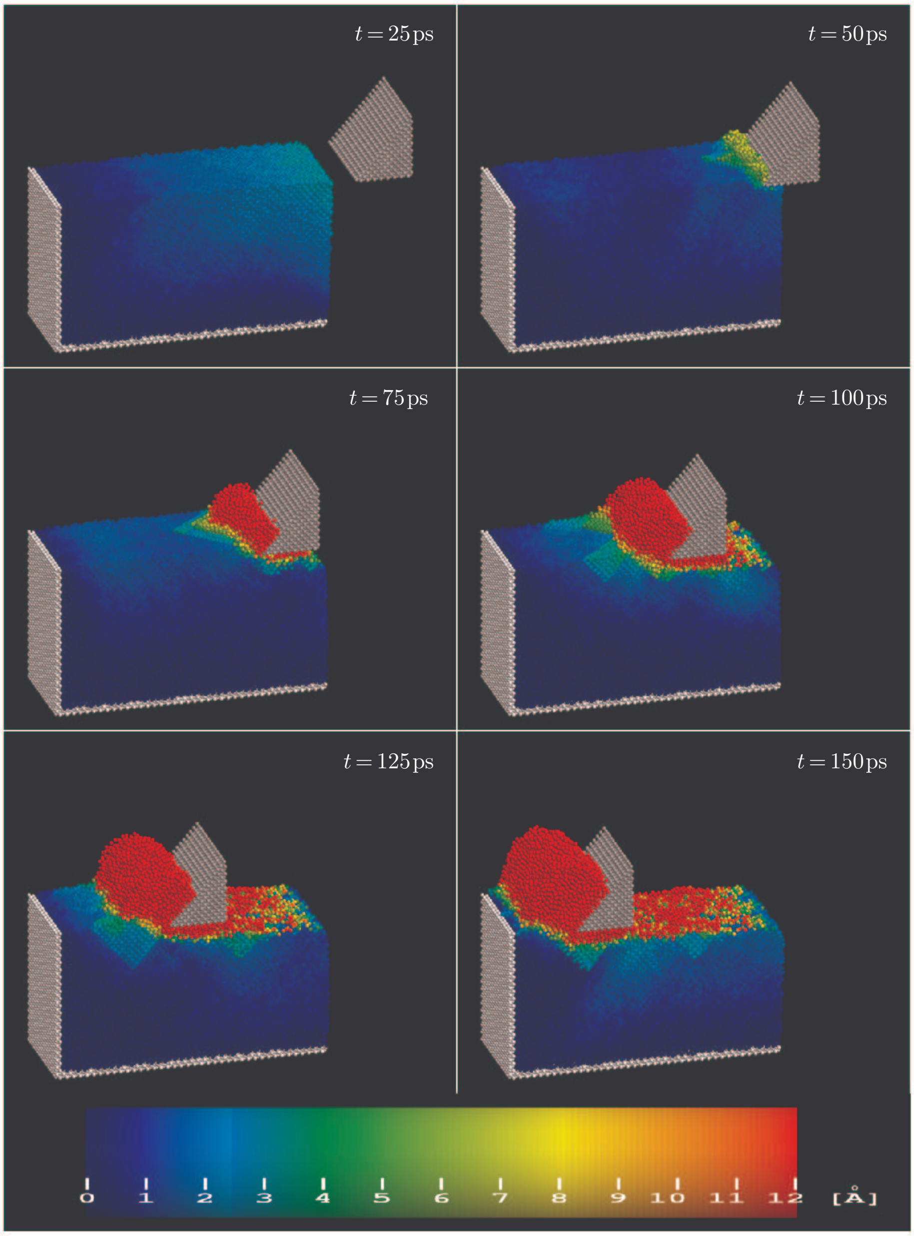

Figure 49. Atomic configurations at selected instants for workmaterial surface orientation (100). Tool C, machining depth $1a$

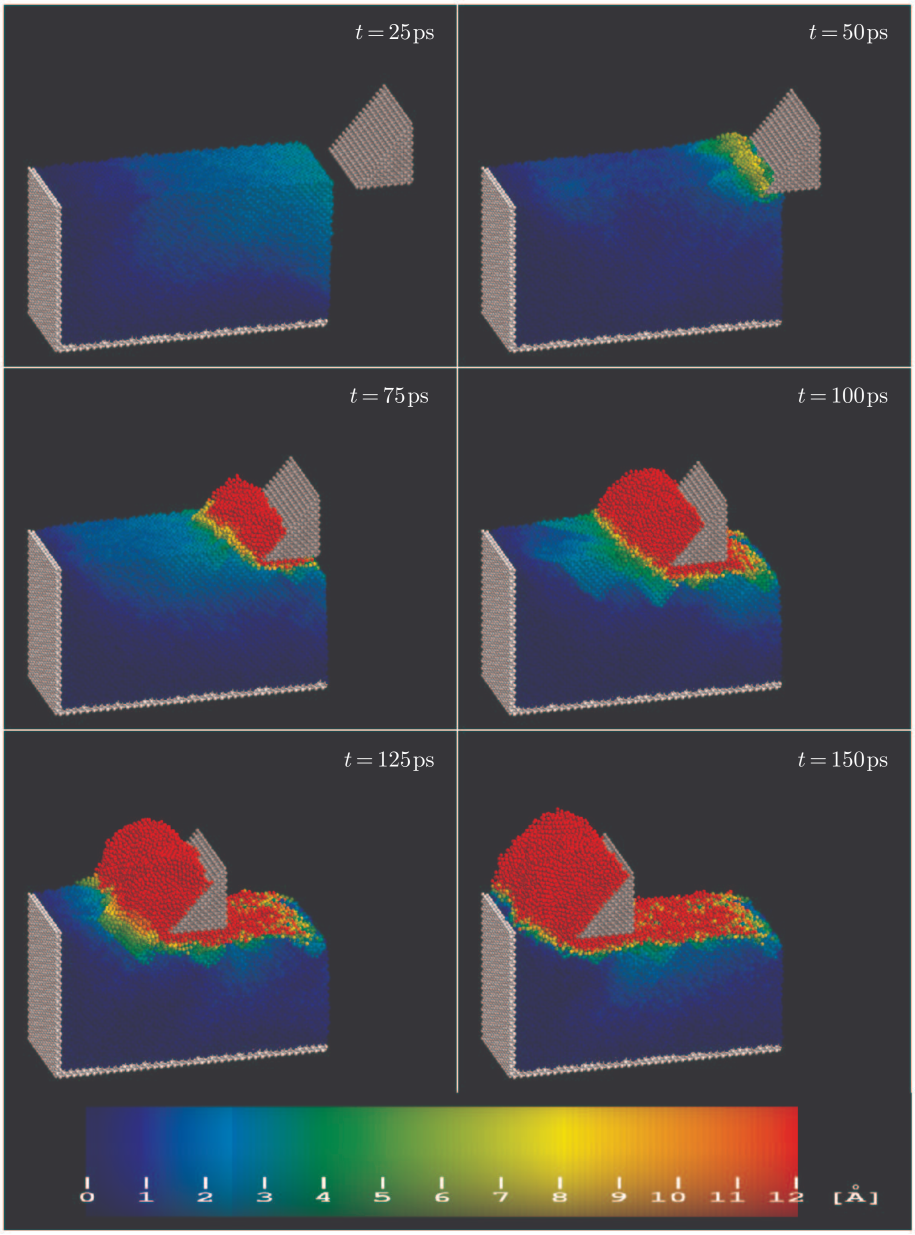

Figure 50. Atomic configurations at selected instants for workmaterial surface orientation (100). Tool C, machining depth $2a$

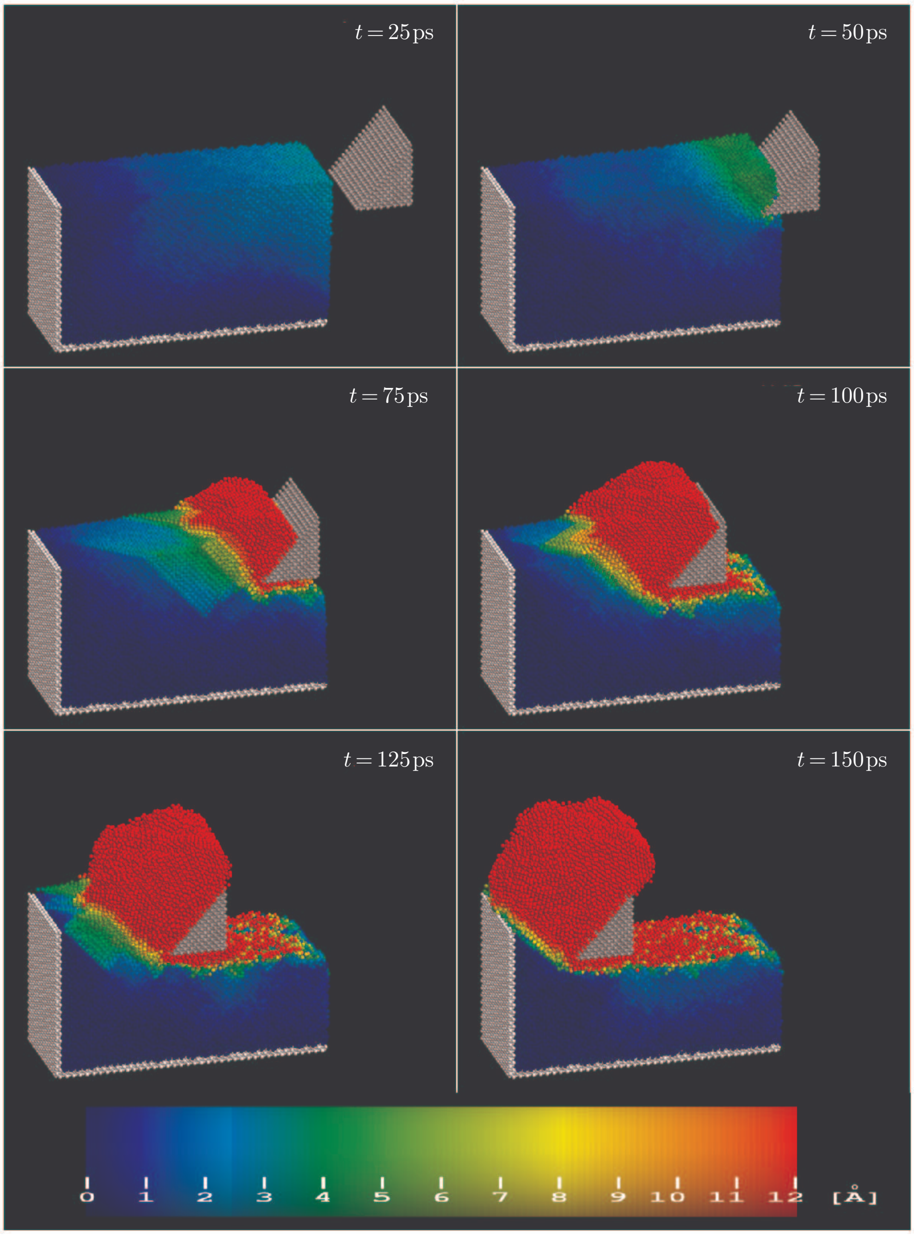

Figure 51. Atomic configurations at selected instants for workmaterial surface orientation (100). Tool C, machining depth $5a$

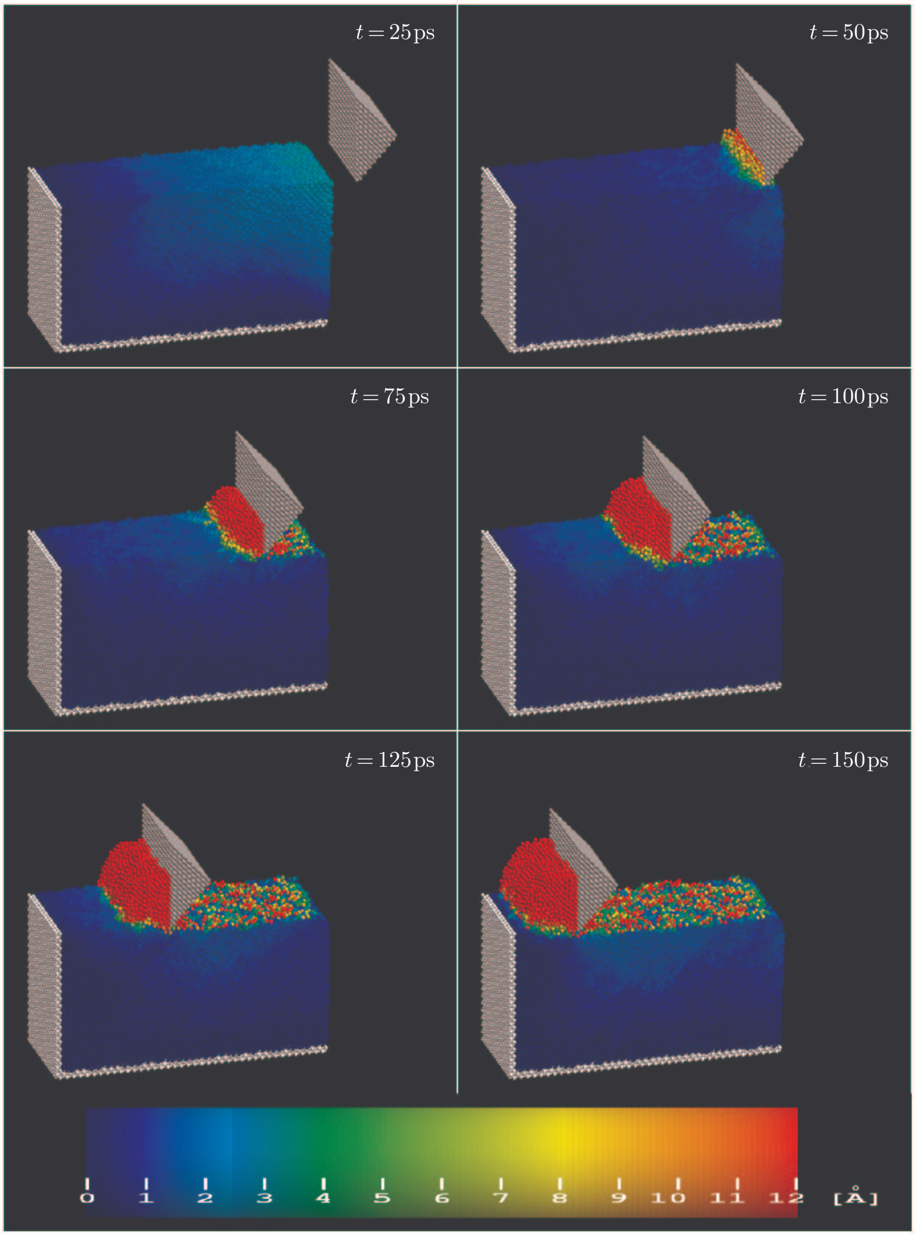

Figure 52. Atomic configurations at selected instants for workmaterial surface orientation (100). Tool D, machining depth $1a$

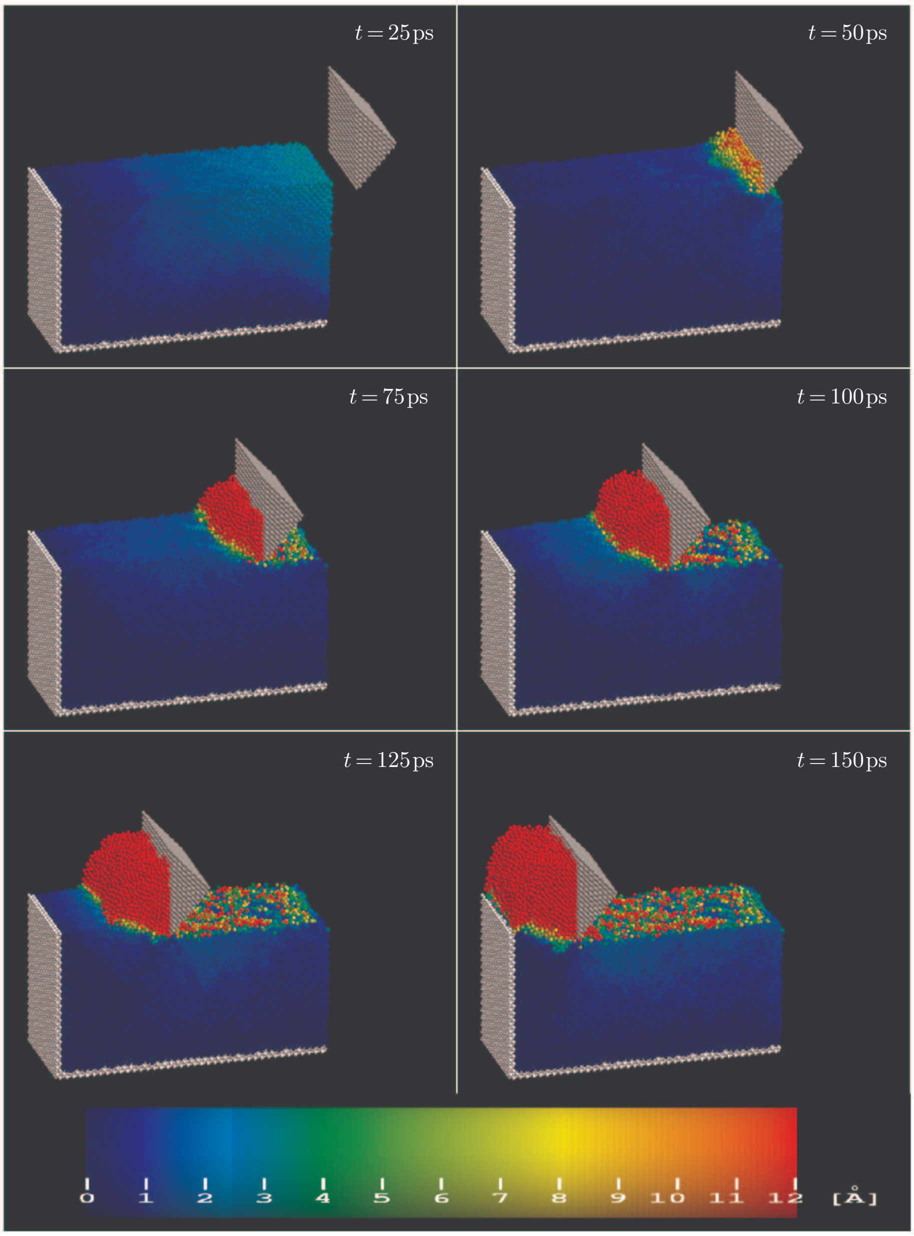

Figure 53. Atomic configurations at selected instants for workmaterial surface orientation (100). Tool D, machining depth $2a$

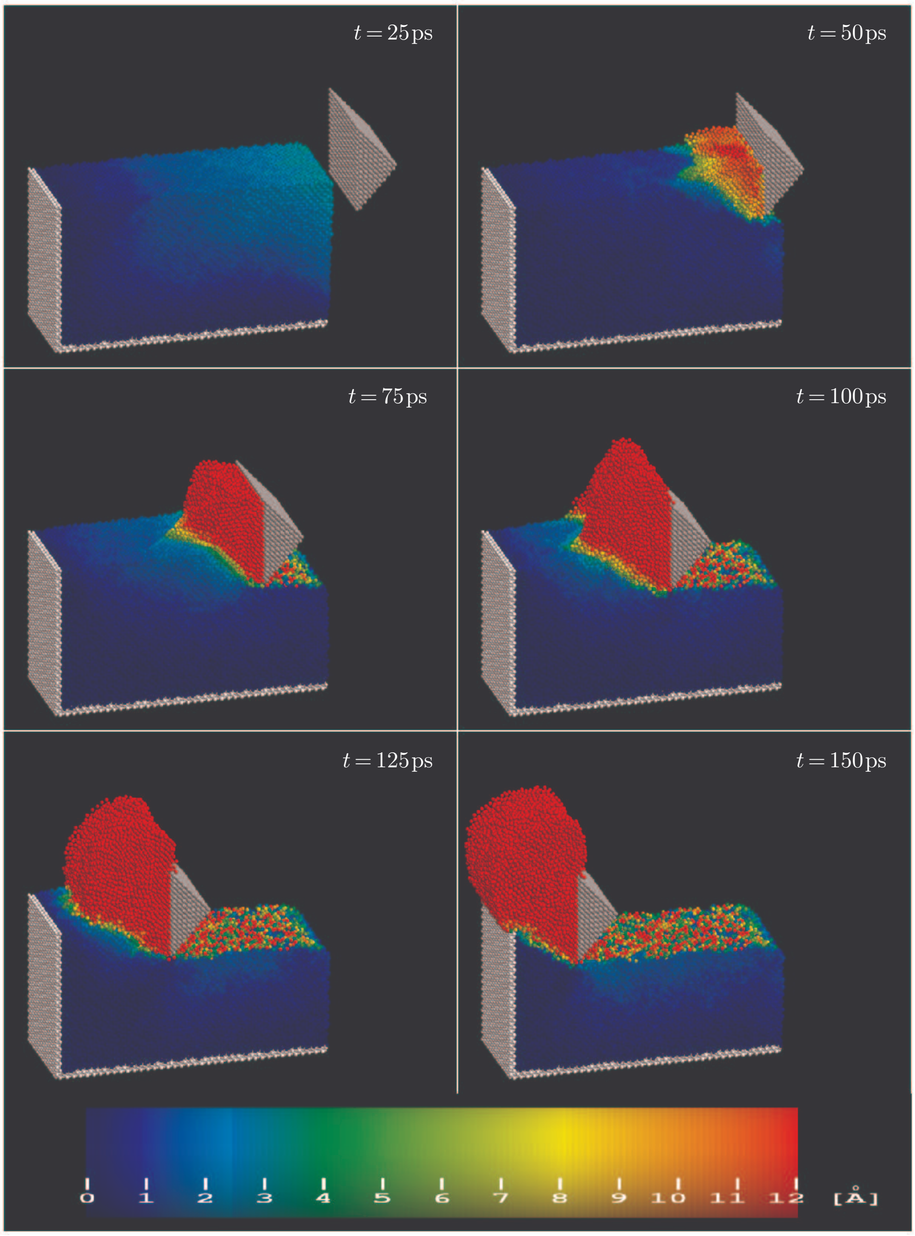

Figure 54. Atomic configurations at selected instants for workmaterial surface orientation (100). Tool D, machining depth $5a$

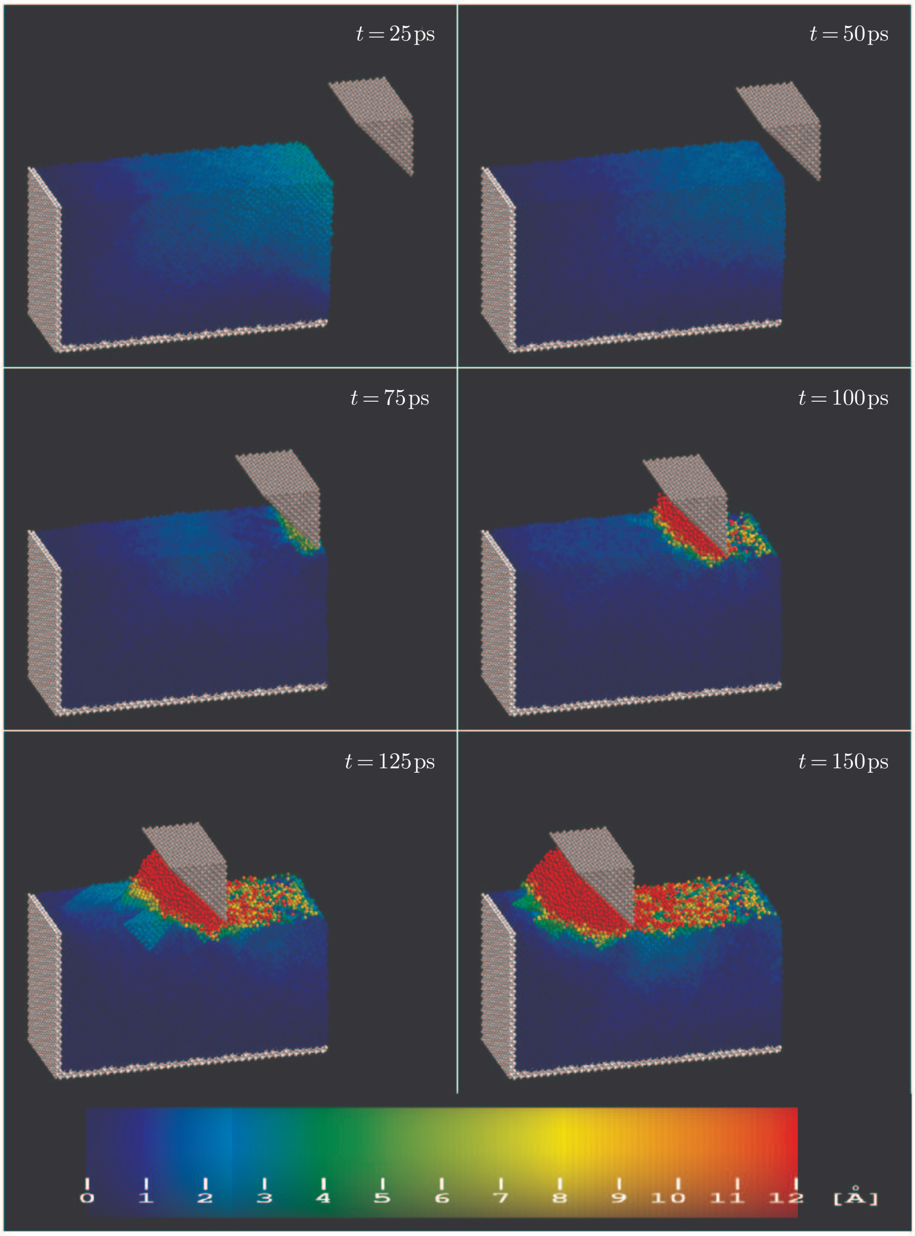

Figure 55. Atomic configurations at selected instants for workmaterial surface orientation (100). Tool E, machining depth $1a$

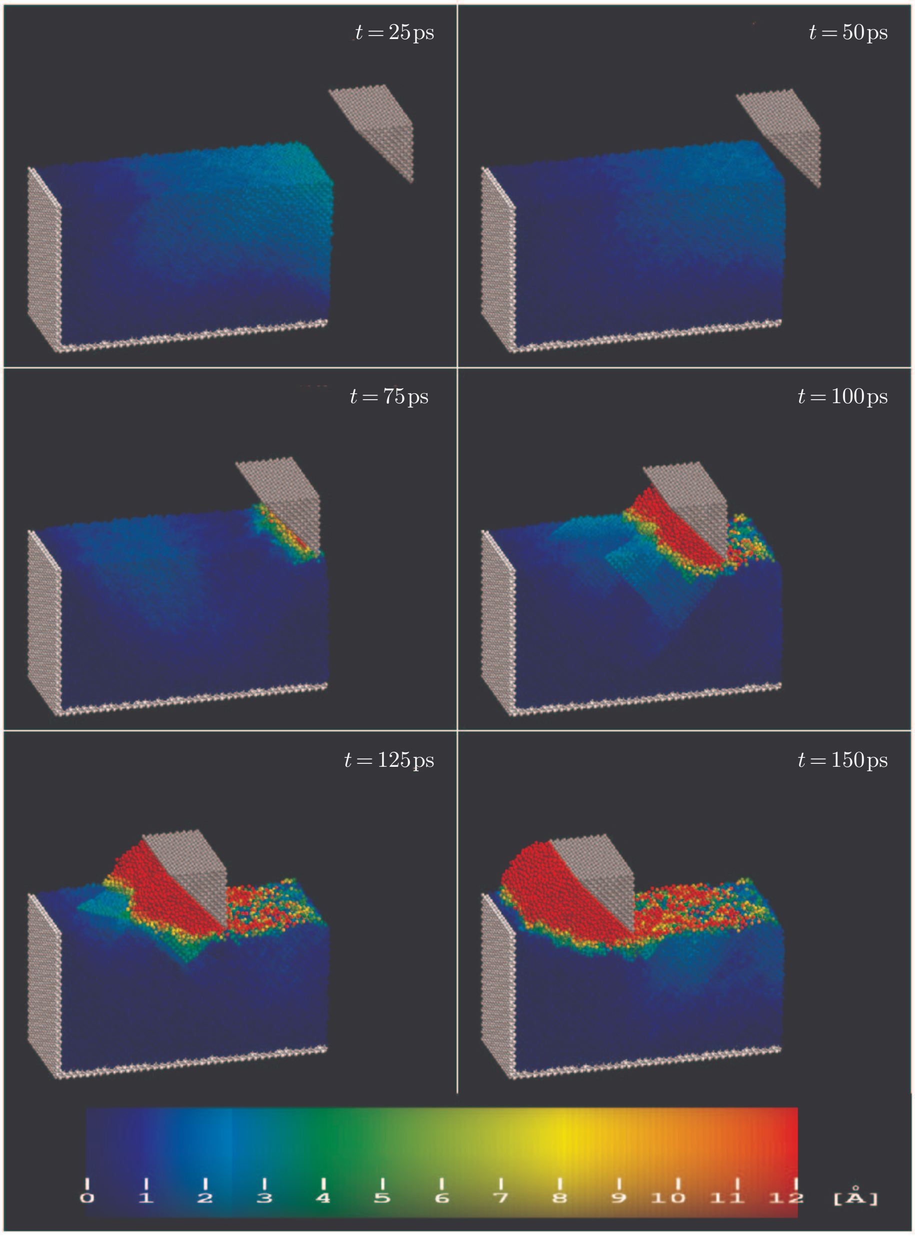

Figure 56. Atomic configurations at selected instants for workmaterial surface orientation (100). Tool E, machining depth $2a$

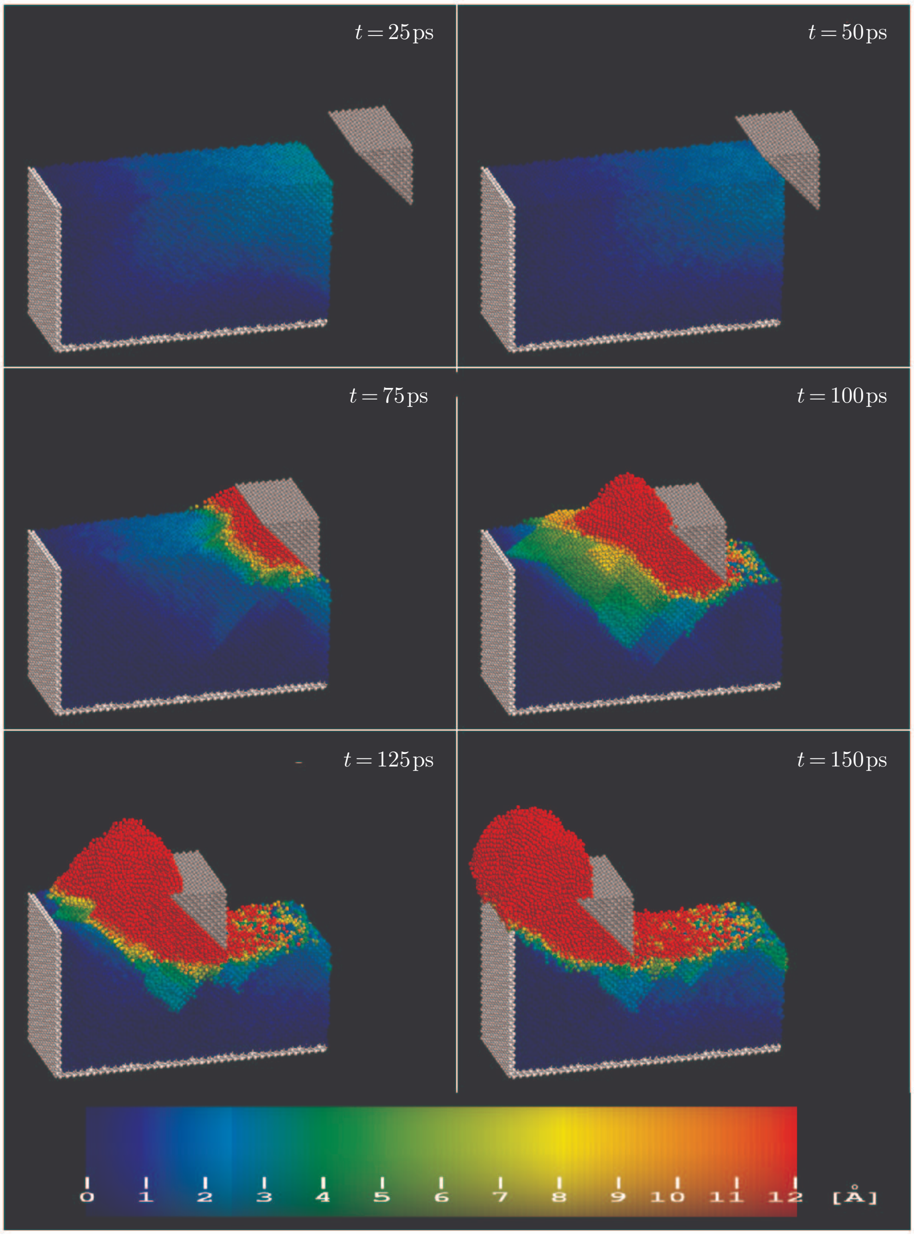

Figure 57. Atomic configurations at selected instants for workmaterial surface orientation (100). Tool E, machining depth $5a$

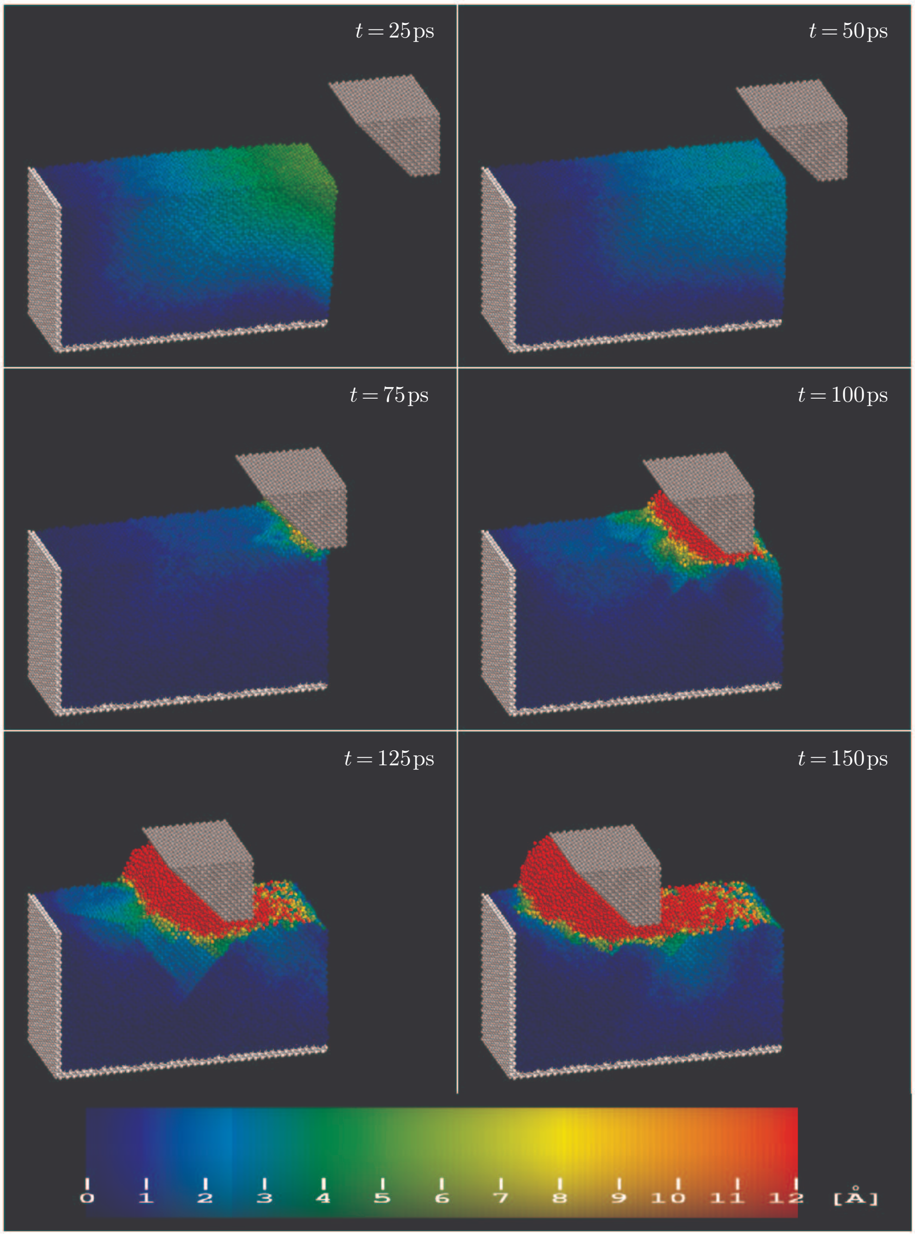

Figure 58. Atomic configurations at selected instants for workmaterial surface orientation (100). Tool F, machining depth $1a$

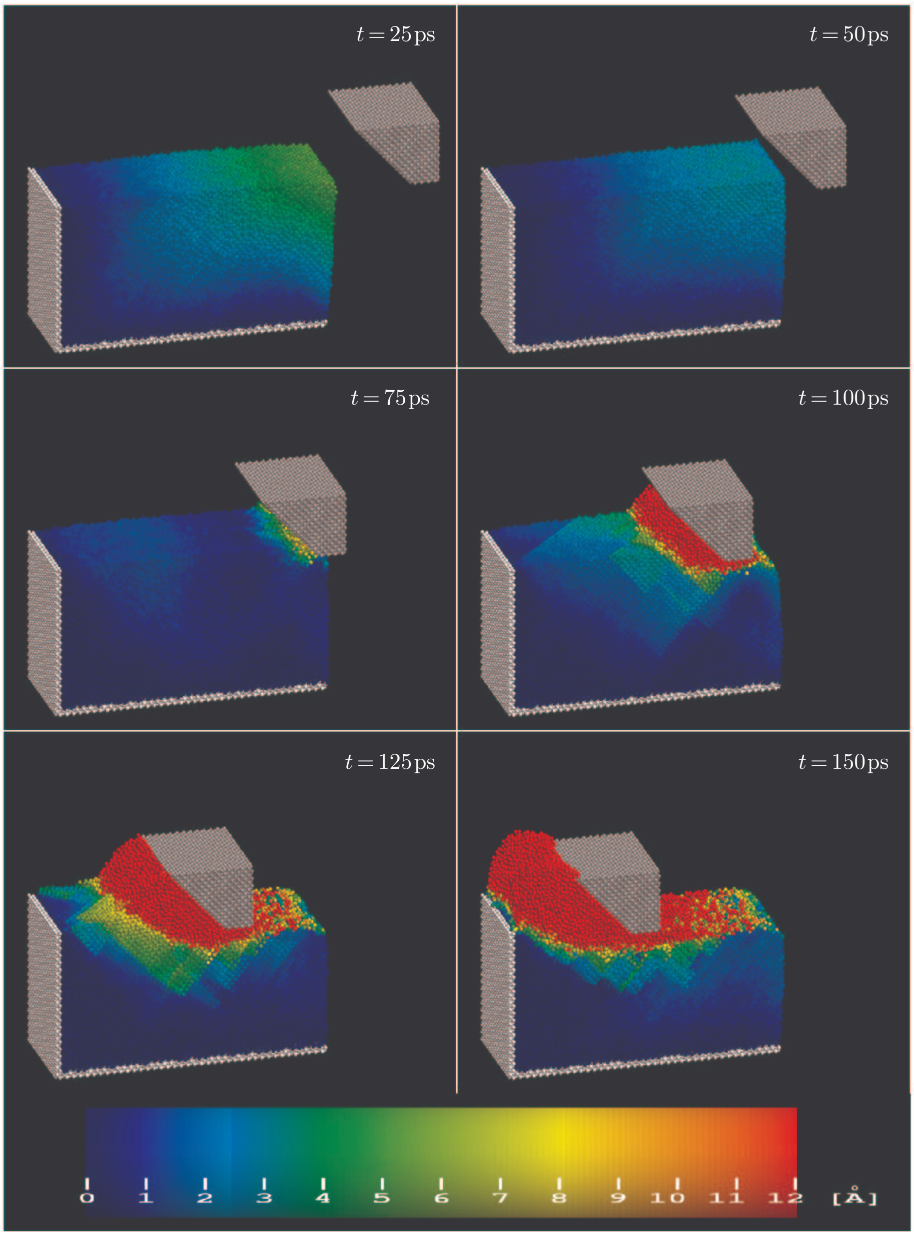

Figure 59. Atomic configurations at selected instants for workmaterial surface orientation (100). Tool F, machining depth $2a$

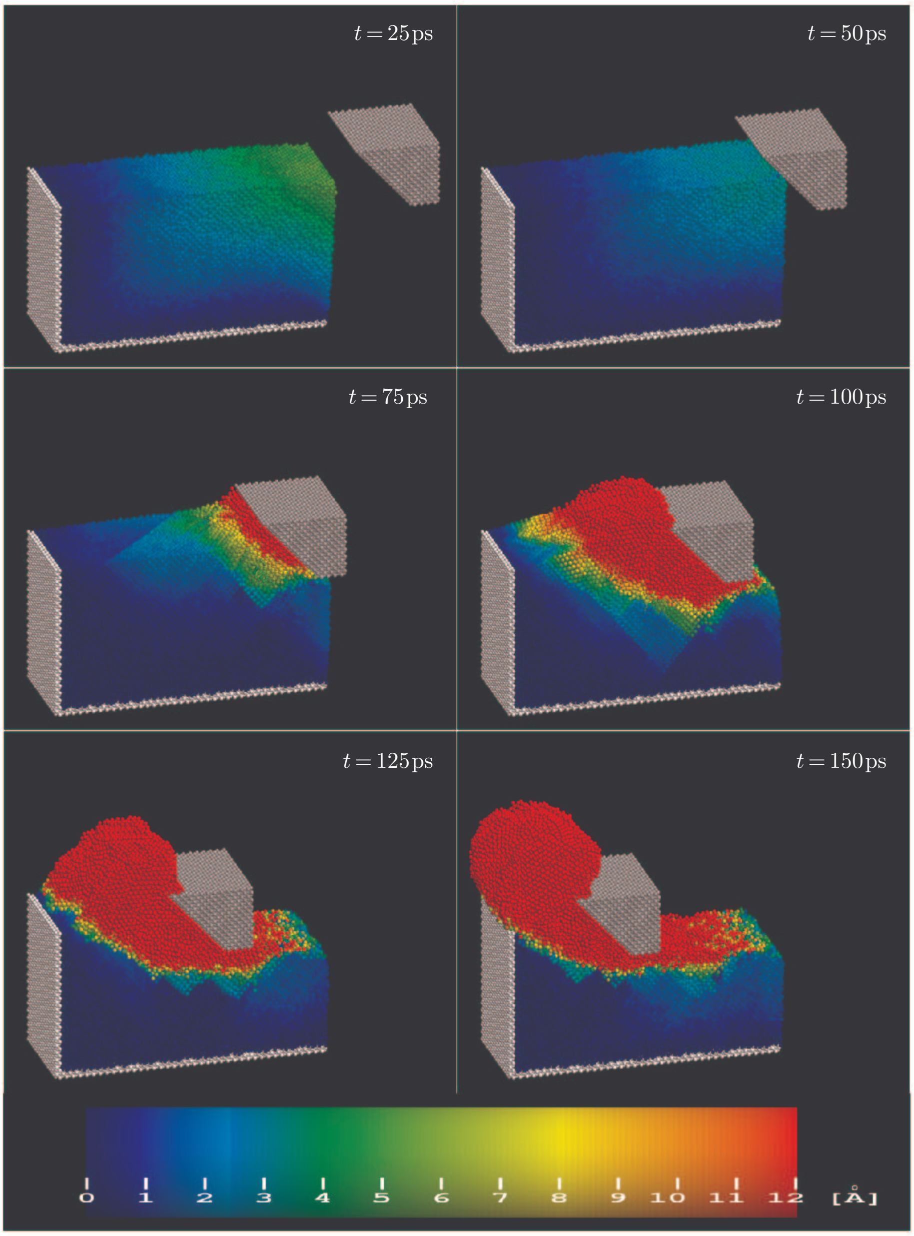

Figure 60. Atomic configurations at selected instants for workmaterial surface orientation (100). Tool F, machining depth $5a$

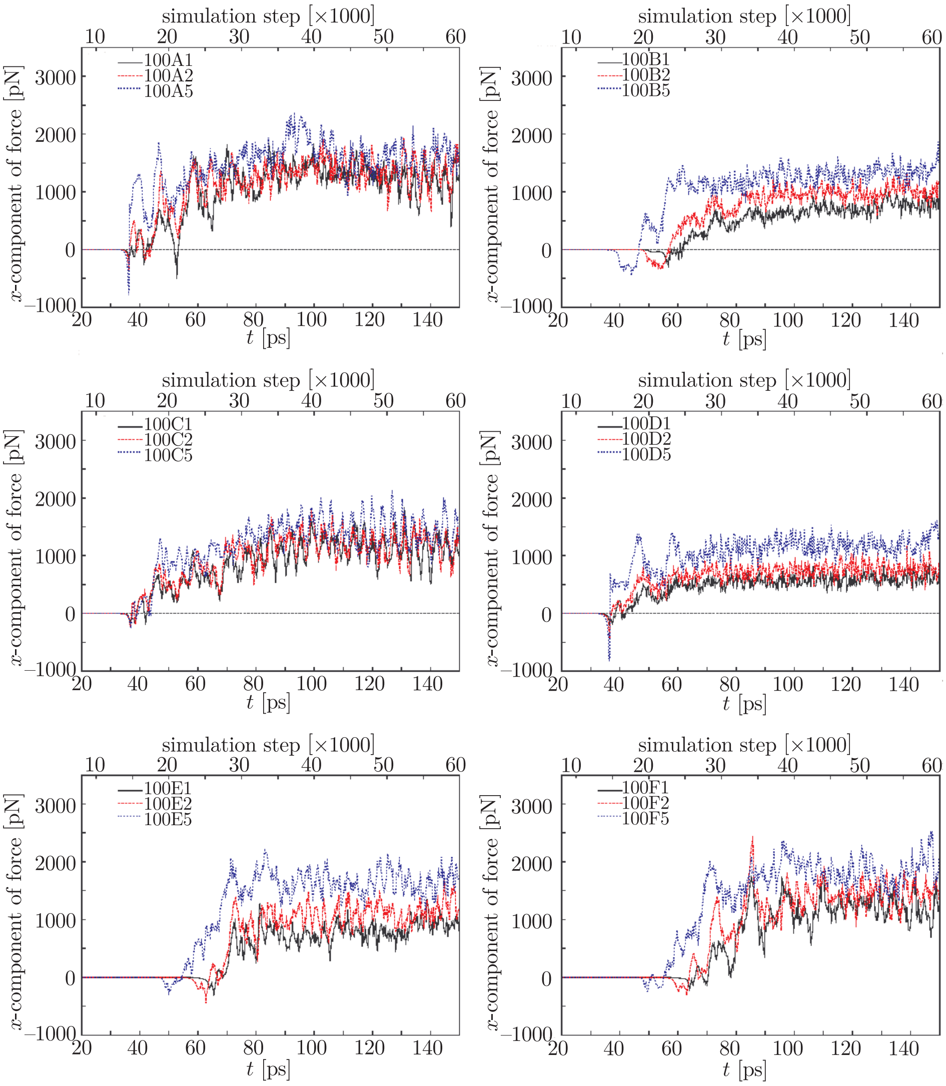

Figure 61. Horizontal components of the force for all six types of the machining tool studied, for various machining depths. Workmaterial surface orientation (100)

the most favourable as the inclined tool front easily moves the forming chip up, letting it "climb" the tool. Tool A pushes the chip with a vertical side, whereby the required force is slightly larger, and tool F seems to be the most unfavourable, as it not only contacts the material with the horizontal base but, in addition to that, the tool front forces the forming chip into the workmaterial directly in front of the tool.

The predominant influence of the horizontal base the tool on the x-component of the force needed for machining is confirmed by the observation that, for tools A and C (which have the largest contact with the material), this component

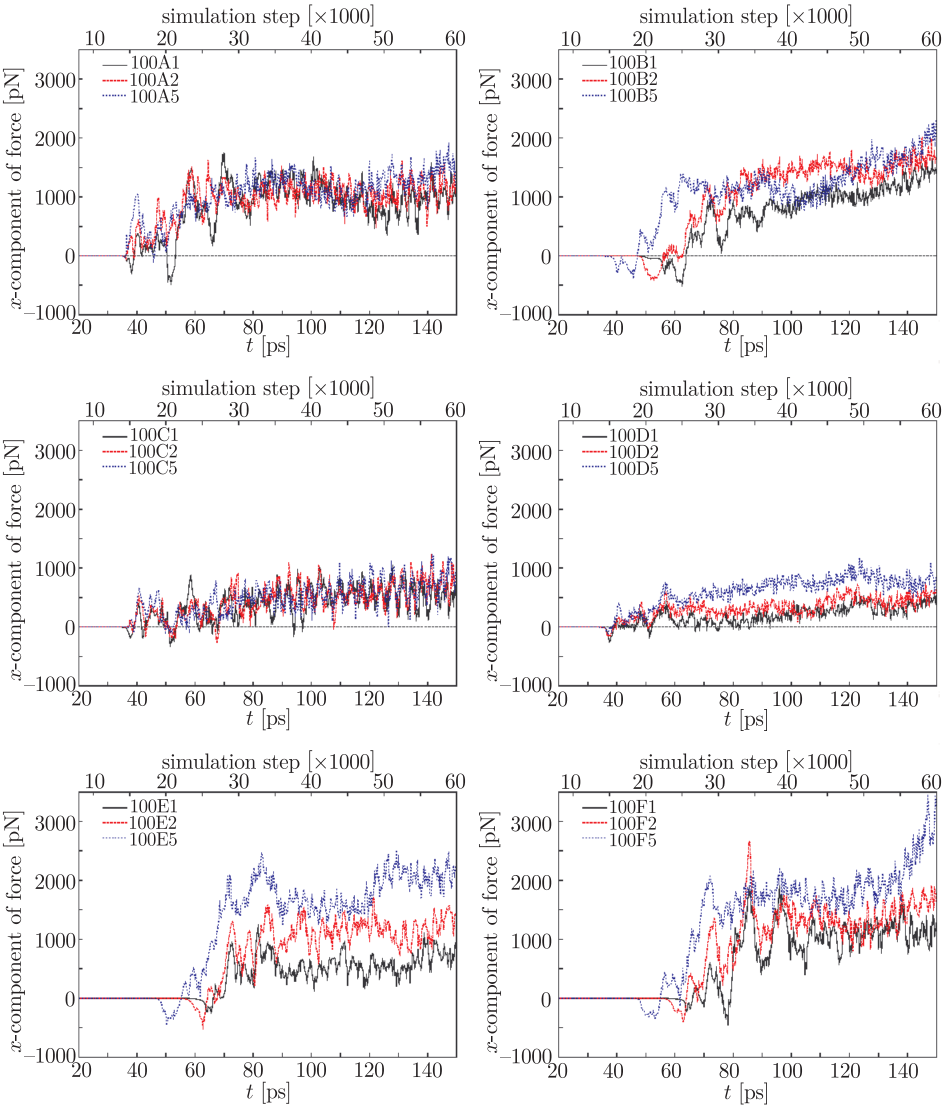

Figure 62. Vertical components of forces for all six types of machining tool for various machining depths. Work material surface orientation (100)

practically doesn't depend on the machining depth, while for shapes D, B, and E (*i.e.*, those with a small contact area with the material) the needed force clearly increases for larger machining depths due to the resistance of the machined workmaterial and the formation of chip.

For tool F (with a medium contact surface with the workmaterial) the situation is intermediate, *i.e.*, the effect of the machining depth on the x-component of the force is moderate. Interesting observations can be made when examining the variability of this force component. Going through the tool shapes from D, through B, E, A, F to C we observe a transition from slightly fluctuating forces to quasiperiodic variations which are evidently present for tool C. The distance between the the peaks in the force for tool C corresponds to the tool having moved by one lattice constant.

Hence, it may be postulated that the atomic granularity of the workmaterial over which the horizontal base of the tool moves is responsible for the sudden peaks and troughs of the force experienced by the tool.

Because of this, for tool C the significance of what exactly is happening in front of the tool is small, as the tool easily moves the chip upwards. In contrast, for tool D, the oscillations related to the granular nature of the workmaterial are not observed due to the fact that the tool does not contact the material with any horizontal surface. A similar situation occurs for tool B and, to a smaller degree, for tool E, however, in this case, the fact that the chip is pushed in front of the tool leads to leaps in the forces, which are not characterized by quasiperiodicity (as was the case for tool C), due to the fact that the pushed chip does not have a crystalline structure.

The time variations of the x-component of the force for other crystalline orientations of the machined material ((110), (111)) look similar, except for the fact that due to the incompatibility between the tool orientations and the workmaterial no quasiperiodic variations are observed, even for tools which have the greatest contact with the material. For tool E and tool F it was observed that the force component in direction x was slightly smaller when machining the material in orientations (110) and (111), yet the difference never exceeded 30%.

We will now turn our attention to the normal (y) component of the force acting on the tool. In Figure 62 it can be seen that it is the inclination of the tool front that has the predominant impact on the normal force, and the occurrence of the horizontal contact surface, or lack thereof, has only secondary significance.

The tools that push the chip into the workmaterial with an obliquely inclined front(F, E, and B) experience a strong reactive normal force as a result, with the magnitude clearly increasing with an increase in the machining depth (this is mostly apparent for tools F, E, and, to a smaller extent, for tool B). The tools which do not push the chip into the workmaterial (A, C, and D) do not experience such strong normal forces. The normal forces acting on tool D are almost nonexistent – which is explained by the fact that the tool has little contact with the workmaterial (no horizontal base), and lacks the oblique hang-off characteristic of tools F, E, B, under which the chip could gather). Tools A and C experience somewhat larger normal forces as a result of the action of the workmaterial on their horizontal base. For tool C this is balanced, to a certain extent, by the interaction with the atoms of the chip that forms easily atop the tool.

For other crystalline orientations of the machined material ((110), (111)) the time variations of the normal component of the force look similar, and the impact of the inclination of the tool front manifests itself in an indirect way here. For tools

A, C, and D, the typical magnitudes of the normal force are comparable, while for tools B, E, and F, this force is by several dozen percent larger for the (100) orientation than for the other two, which may be explained referring to Figure 112, from which it follows that the machining of (100)-oriented workmaterial results in the formation of the largest chip, which positions itself under the tool hang-off, pushing it upwards.

5.3. Distribution of pressure and shear stress

Figures 63–68 present hydrostatic pressure fields, whereas Figures 69–74 present the distribution of the xy component of the atomic stress tensor for the (100)-oriented workmaterial. Irrespective of the tool shape, the hydrostatic pressure reaches maximum values (in the order of 100–150 GPa) directly at the contact point between the tool and the workmaterial.

The region of increased pressure extends from immediately under the chip to several layers of the workmaterial below the tool level. For distances beyond 15–20 Å from the tool, the increase in pressure is no longer perceptible, with the exception of the final stages of machining, where the tool comes close to the artificially immobilised atoms on the right of the workmaterial.

As the machining process progresses, the increased pressure region moves together with the tool, and an accumulation of stresses can be seen in the machined material resulting in the buildup of pressure in the order of 70–100 GPa.

Irrespectively of the tool shape, the pressure is insignificant (not more than ±20 GPa) in the part of the workmaterial that has already been machined. Similarly, there is no buildup of hydrostatic pressure inside the chip, except in its lower part, which is being acted on by the front of the tool.

With increased machining depths, a modest increase of pressure during the final stage of the machining, as well as in the extent of the zone of increased pressure at the point of contact of the front of the tool with the workmaterial was observed.

The exact distribution of the pressure field at the contact area between the front of the tool and the workmaterial depends on the tool shape. For tool C, which attacks the material with a sloped front, moving the chip upwards, the pressure magnitudes were the smallest. For tools A and D, which attack the material with a vertical front the zone of pressure buildup was extended into the space directly in front of the tool (the bottom of the chip). Tools B, E, and F, which, owing to their shape, force atoms into the workmaterial directly in front of the tool, generate the highest pressure, with the stresses concentrated mainly under the inclined tool front.

The abovementioned features of the pressure fields are common for all orientations of the machined material, the differences resulting from a change of the orientation are of a quantitative nature only – the highest pressure (both locally, around the contact point; and in the whole workmaterial) was observed for the (110)-oriented workmaterial, and the lowest – for the (111)-oriented workmaterial.

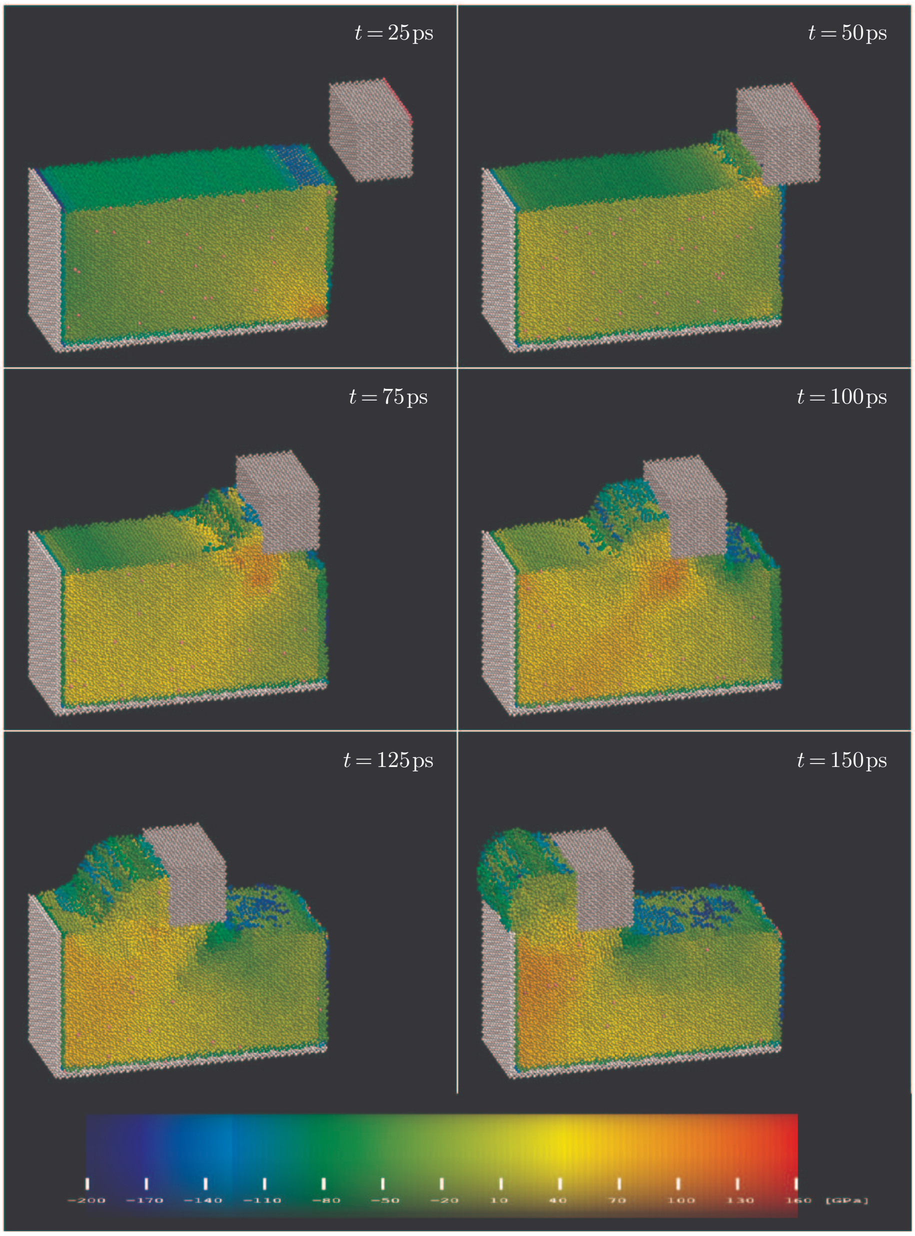

Figure 63. Pressure distribution at selected instants. Workmaterial surface orientation (100), machining depth $1a$

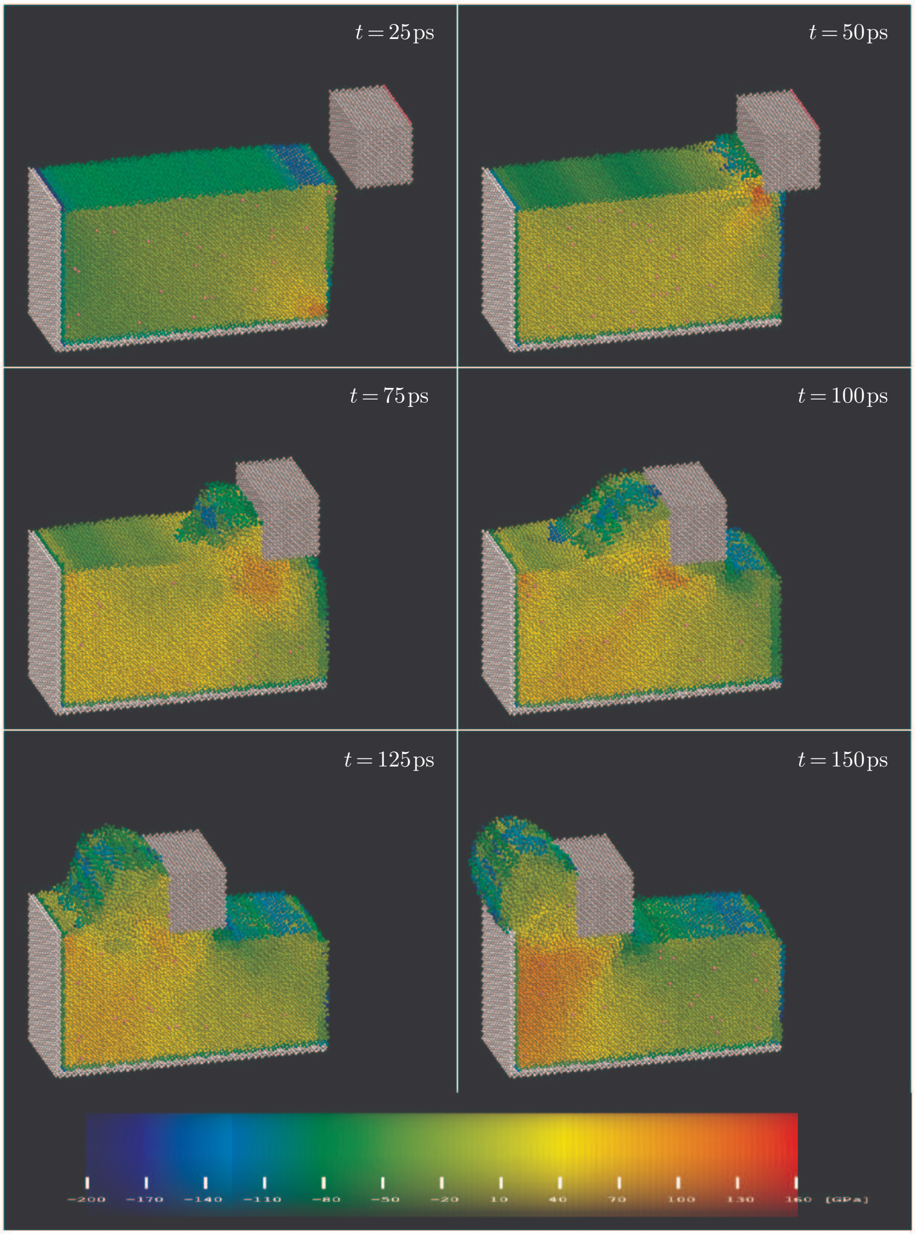

Figure 64. Pressure distribution at selected instants. Workmaterial surface orientation (100), machining depth $2a$

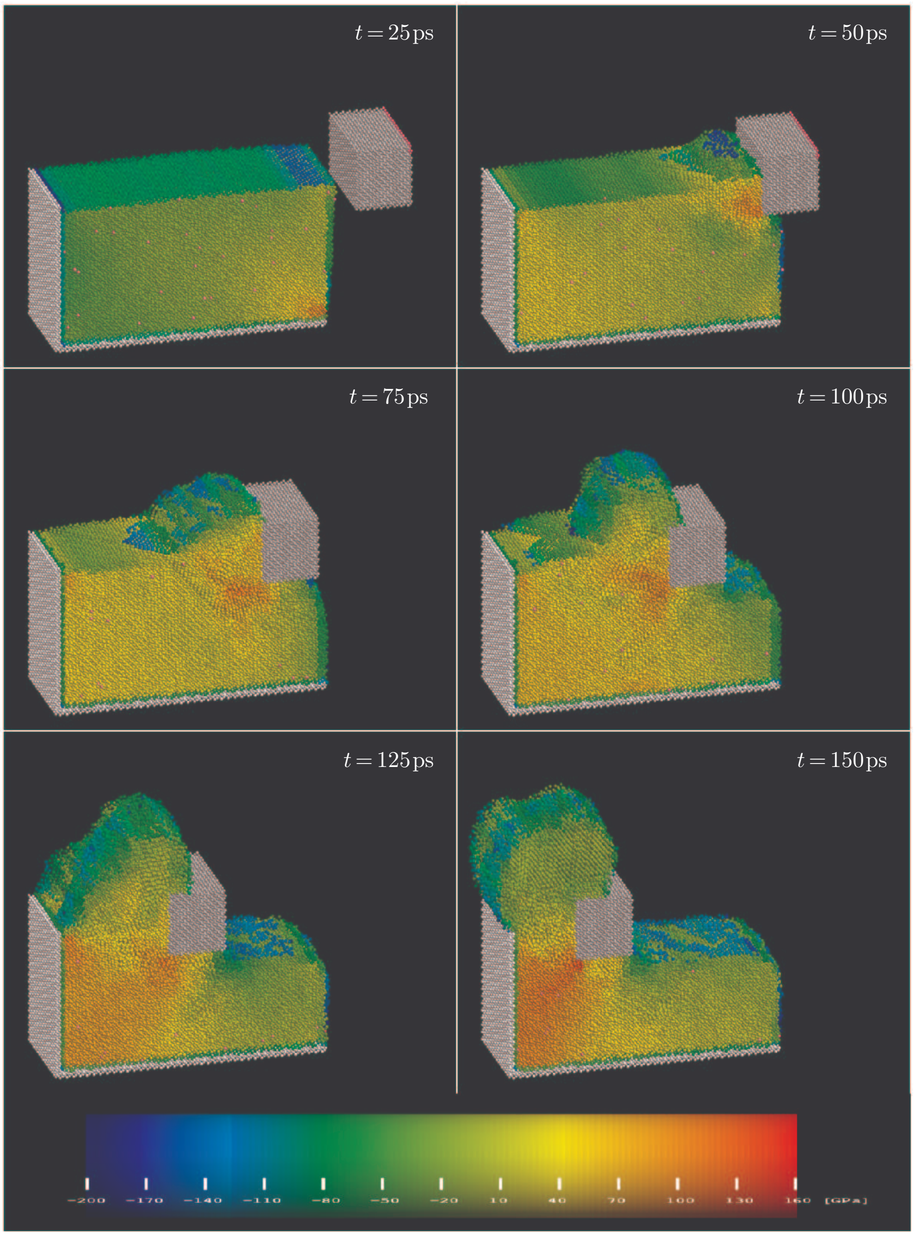

Figure 65. Pressure distribution at selected instants. Workmaterial surface orientation (100), machining depth $5a$

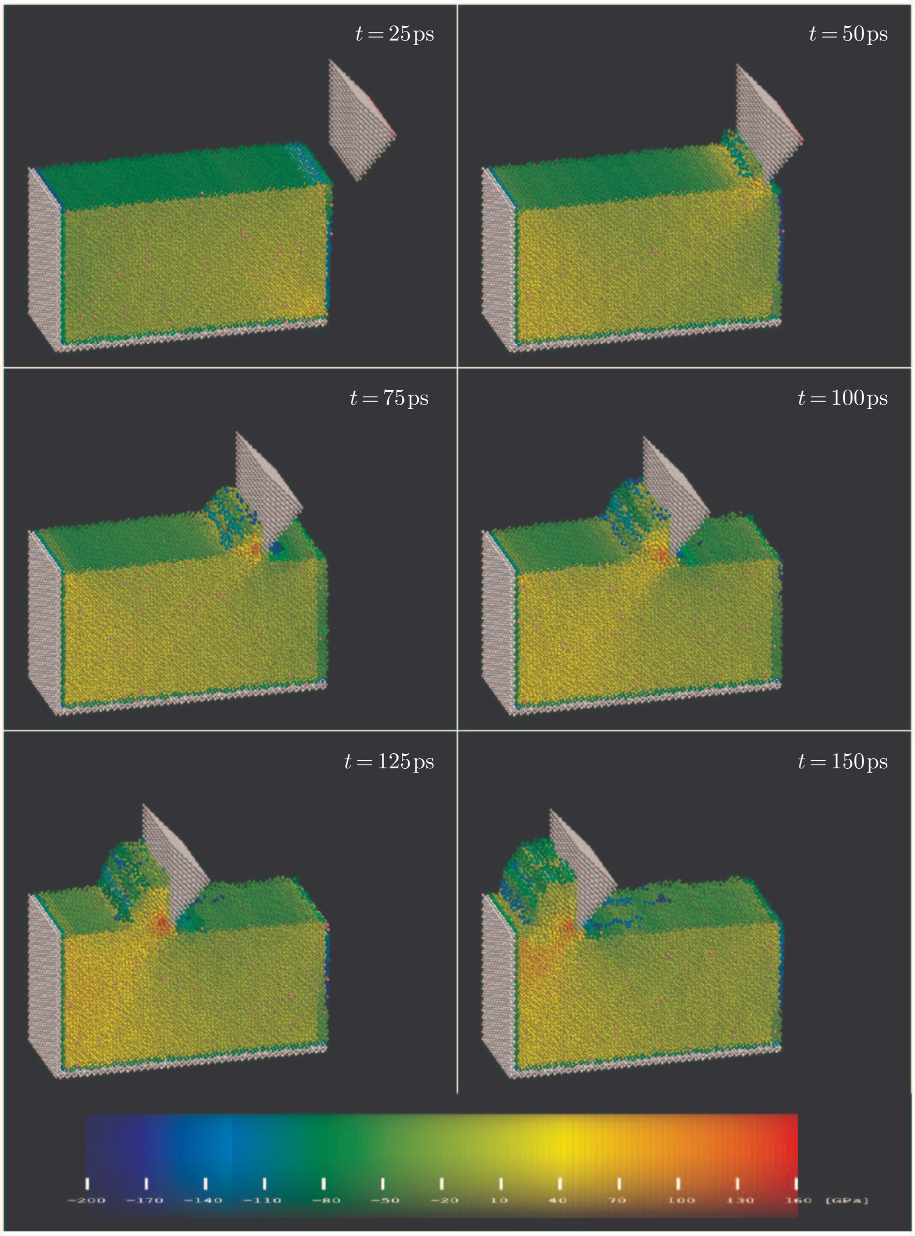

Figure 66. Pressure distribution at selected instants. Workmaterial surface orientation (100), machining depth $1a$

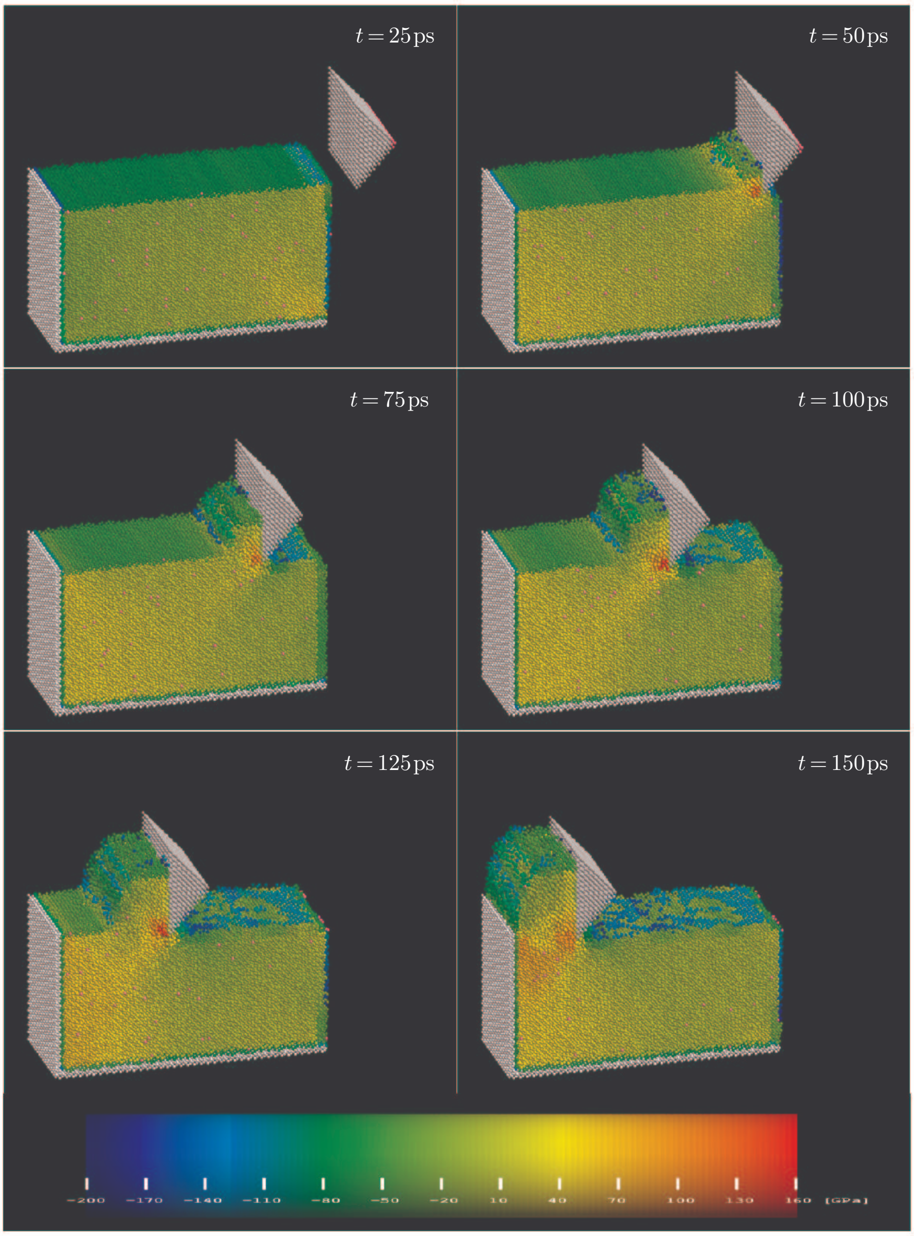

Figure 67. Pressure distribution at selected instants. Workmaterial surface orientation (100), machining depth $2a$

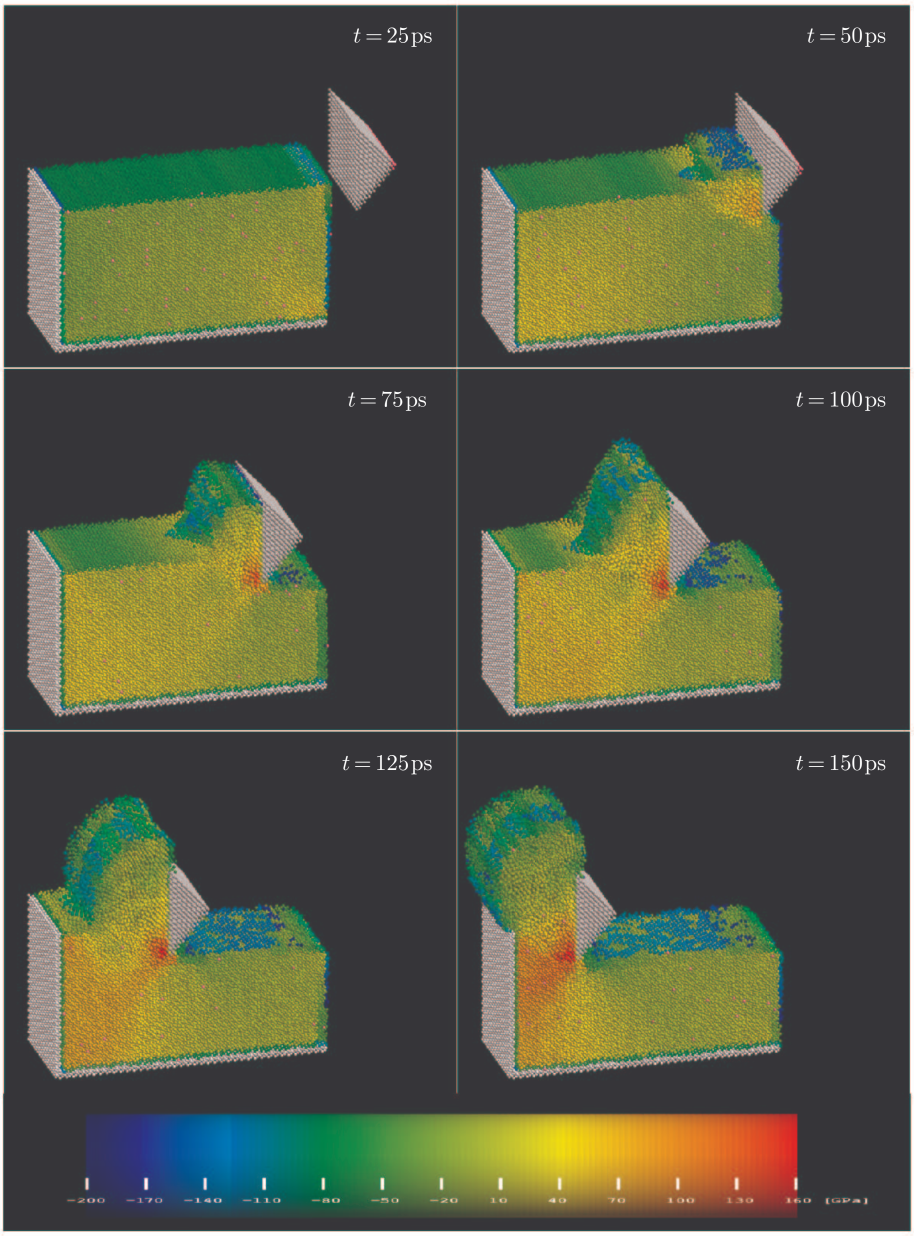

Figure 68. Pressure distribution at selected instants. Workmaterial surface orientation (100), machining depth $5a$

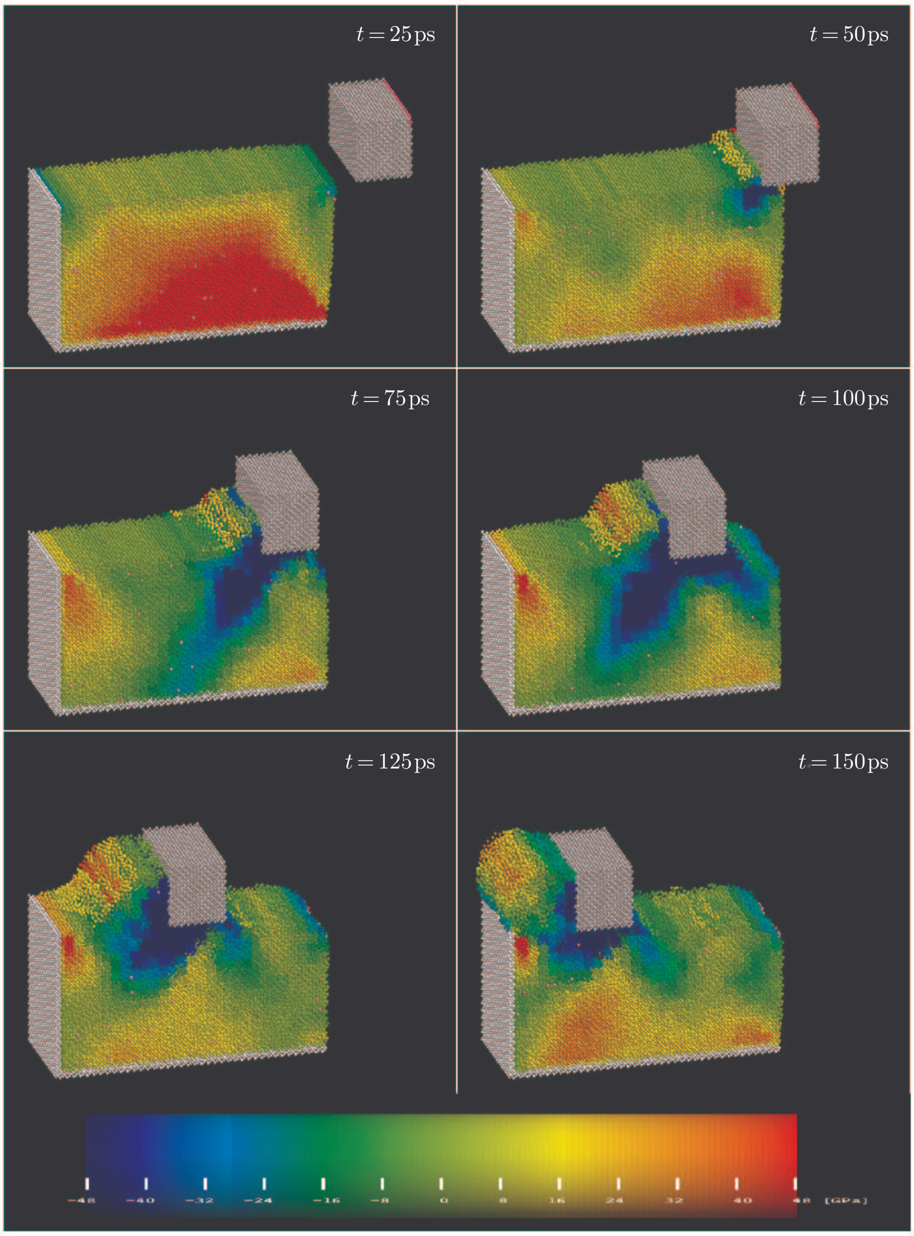

Figure 69. Distribution of the xy component of the stress tensor at selected instants. Workmaterial surface orientation (100), machining depth $1a$

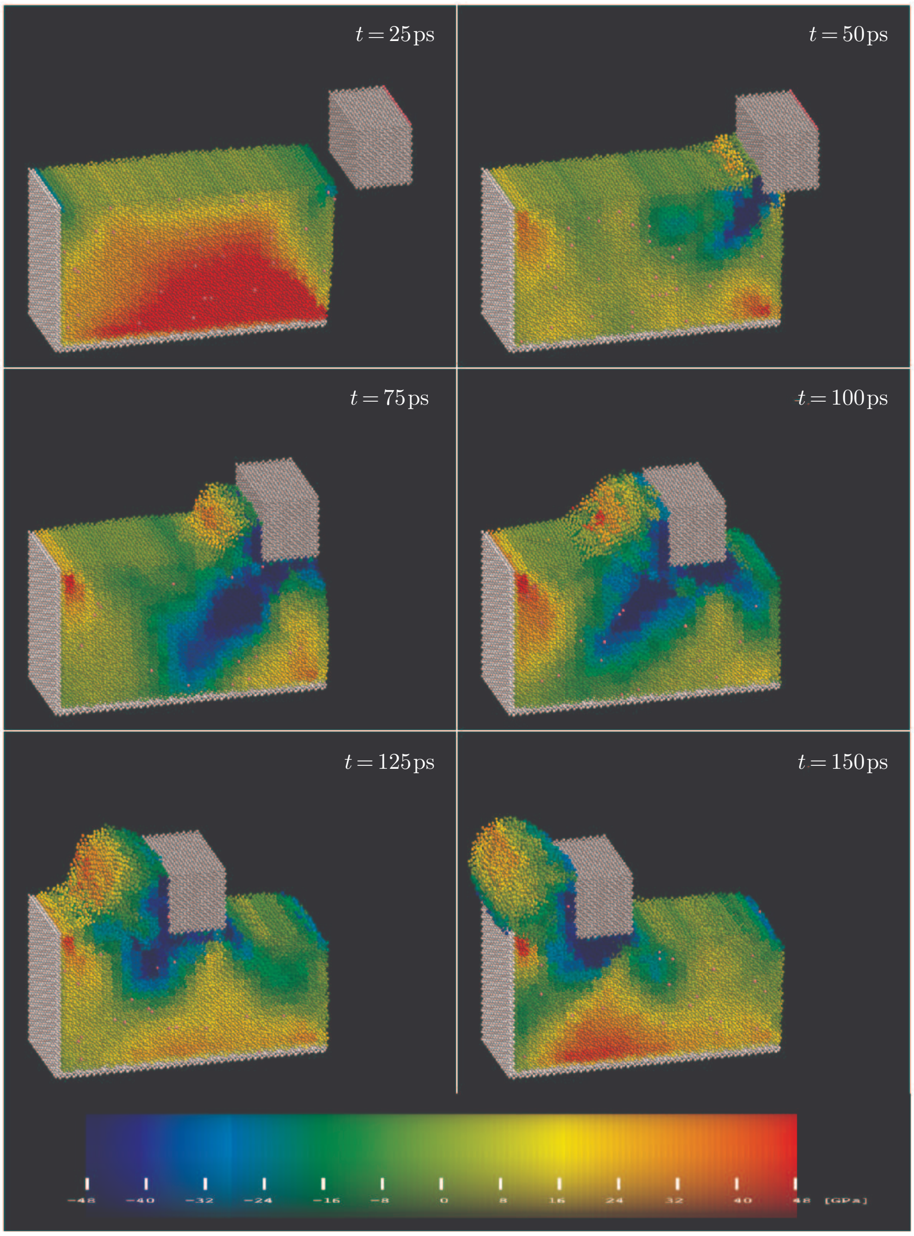

Figure 70. Distribution of the xy component of the stress tensor at selected instants. Workmaterial surface orientation (100), machining depth $2a$

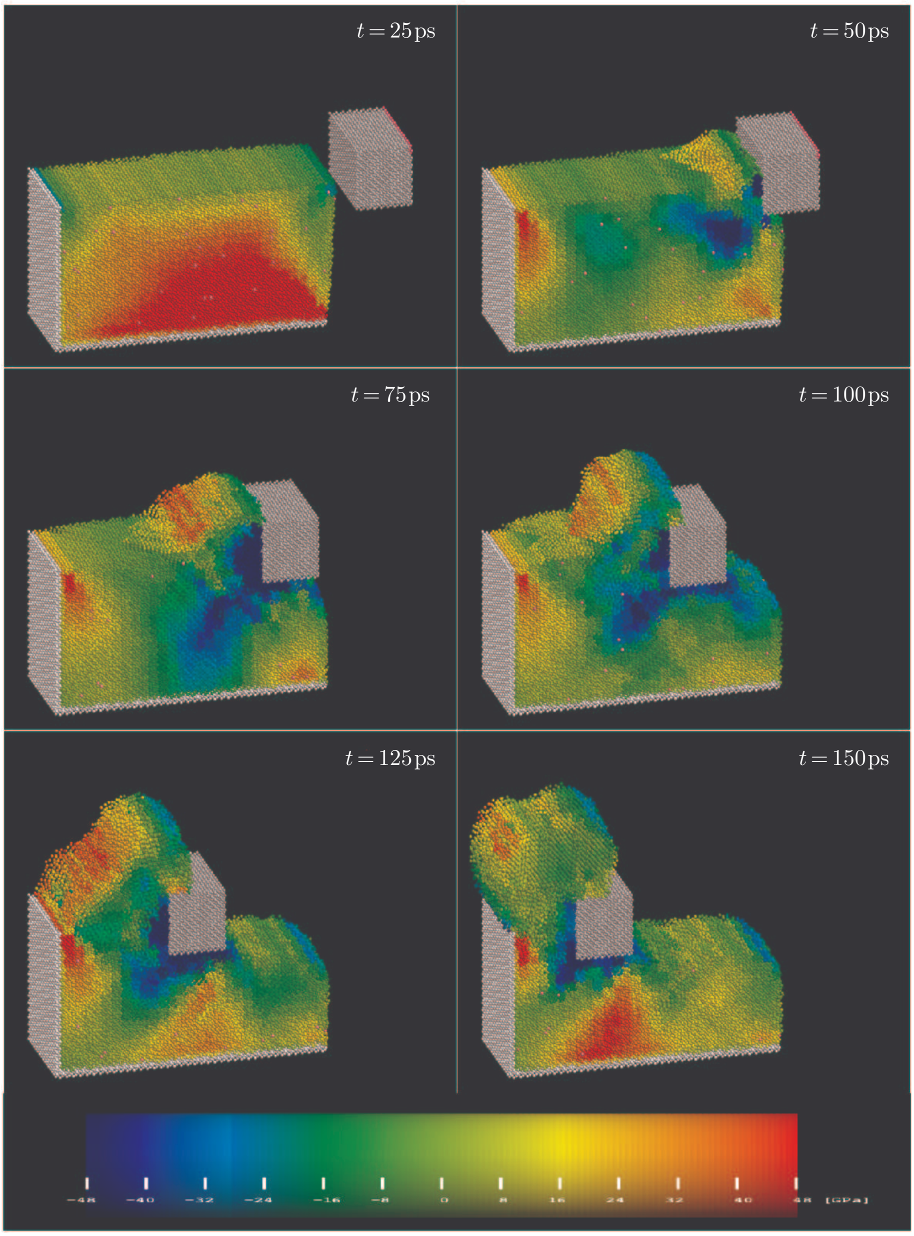

Figure 71. Distribution of the xy component of the stress tensor at selected instants. Workmaterial surface orientation (100), machining depth $5a$

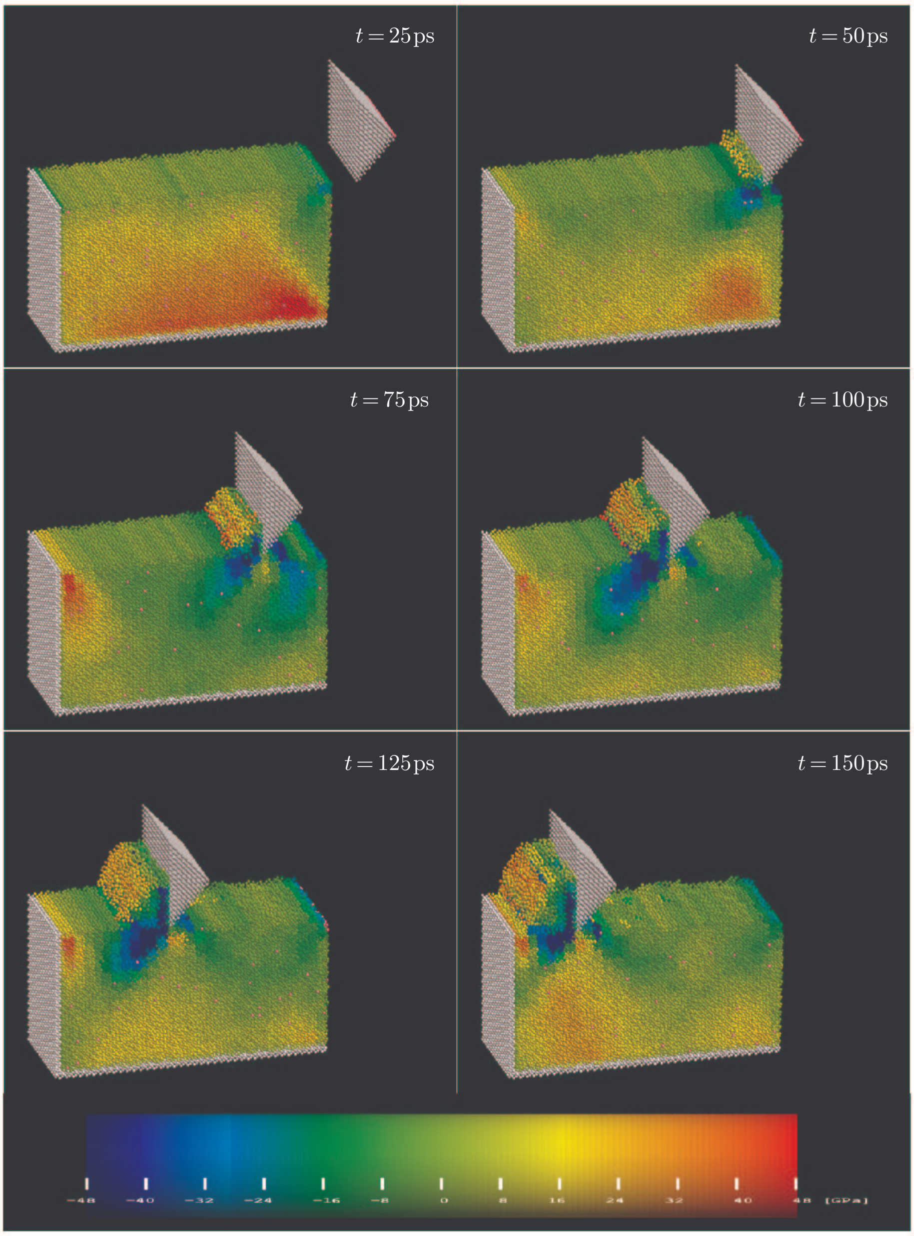

Figure 72. Distribution of the xy component of the stress tensor at selected instants. Workmaterial surface orientation (100), machining depth 1a

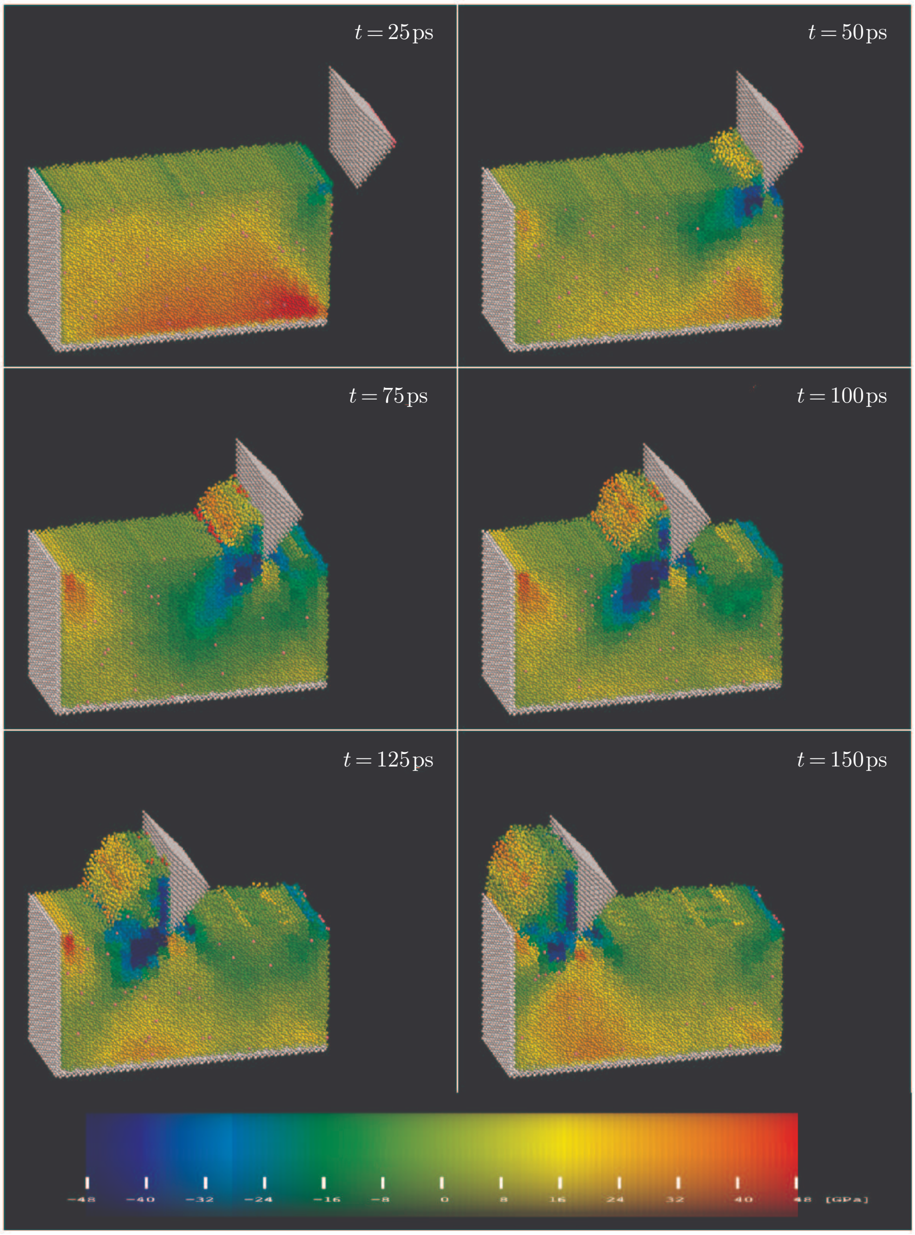

Figure 73. Distributions of the xy component of the stress tensor at selected instants. Workmaterial surface orientation (100), machining depth $2a$

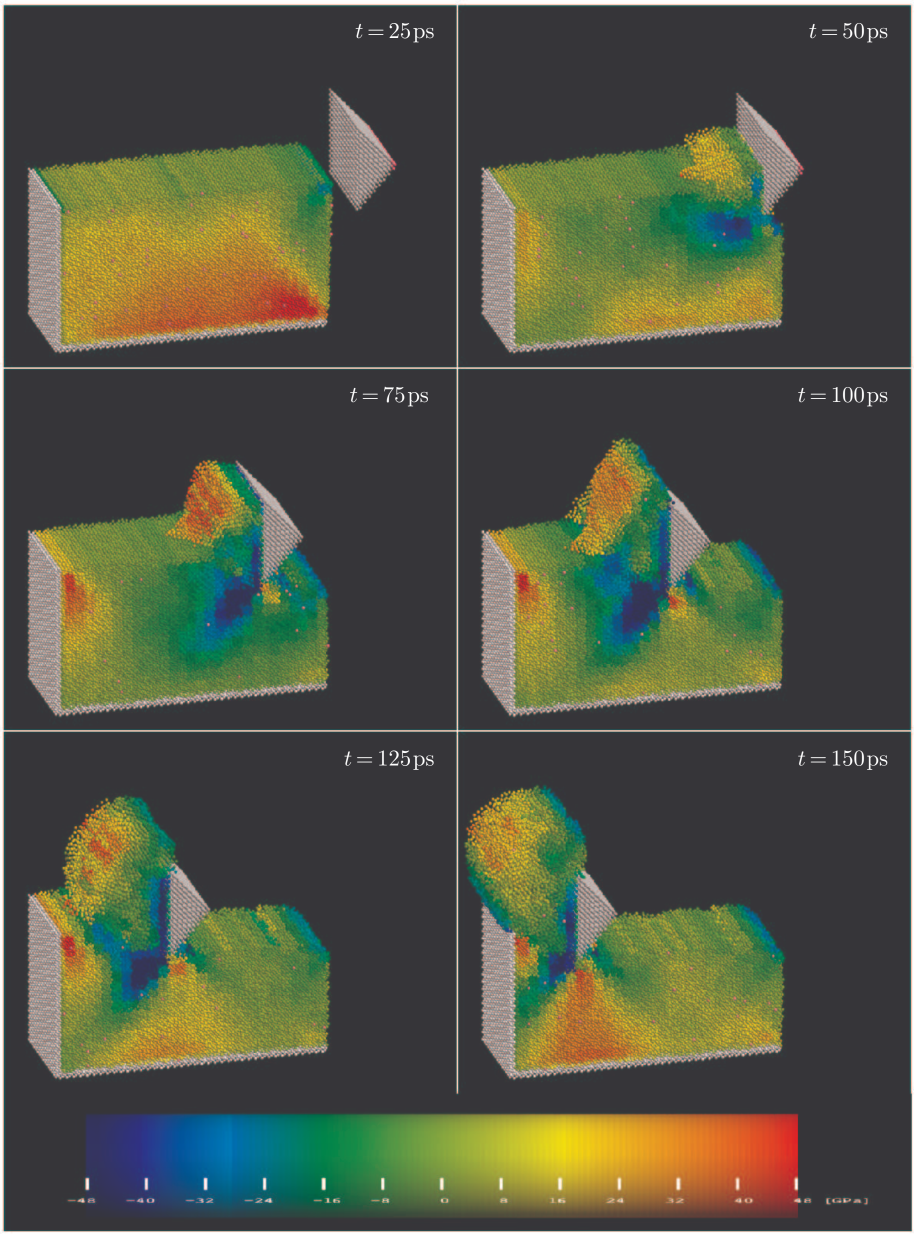

Figure 74. Distributions of the xy component of the stress tensor at selected instants. Workmaterial surface orientation (100), machining depth $5a$

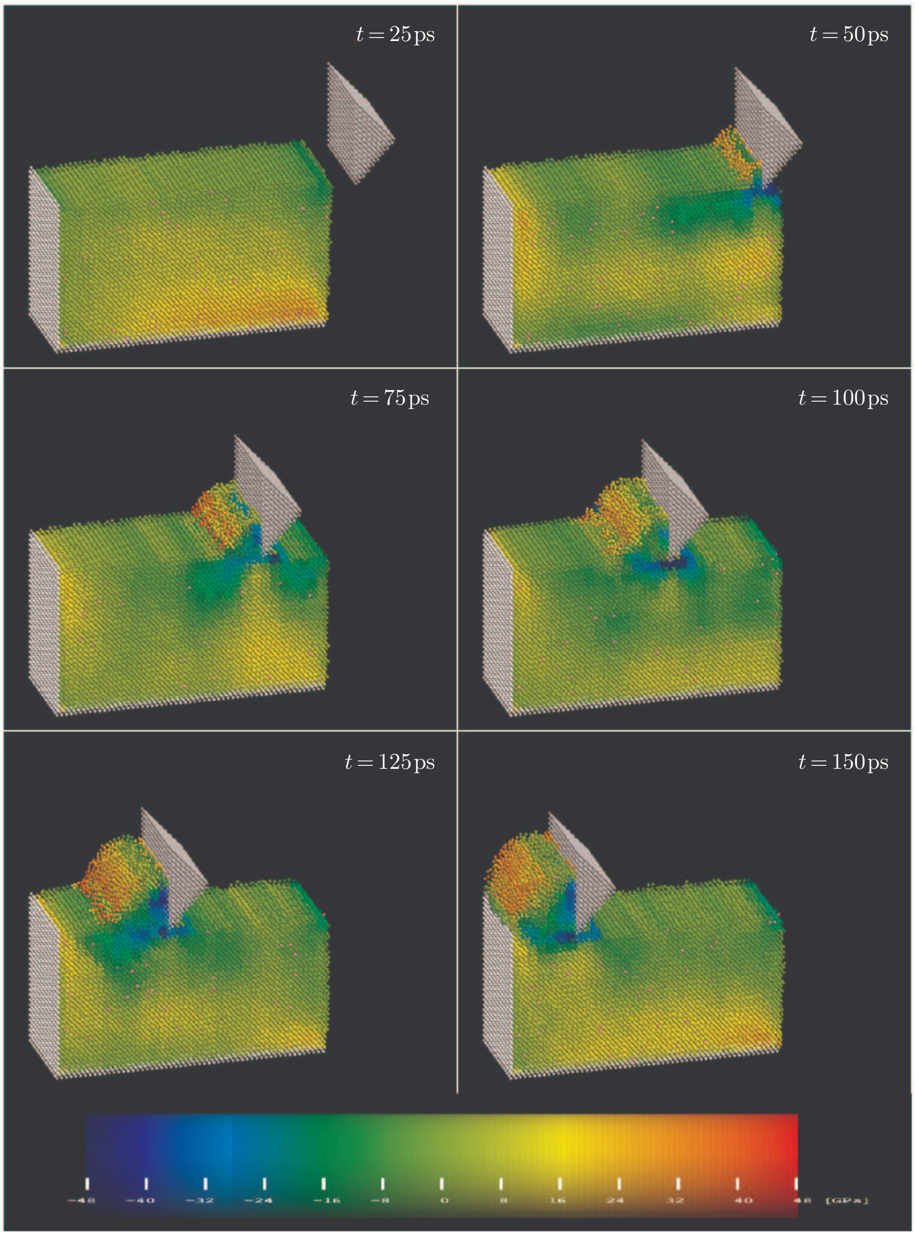

Figure 75. Distributions of the xy component of the stress tensor at selected instants. Workmaterial surface orientation (110), machining depth $2a$

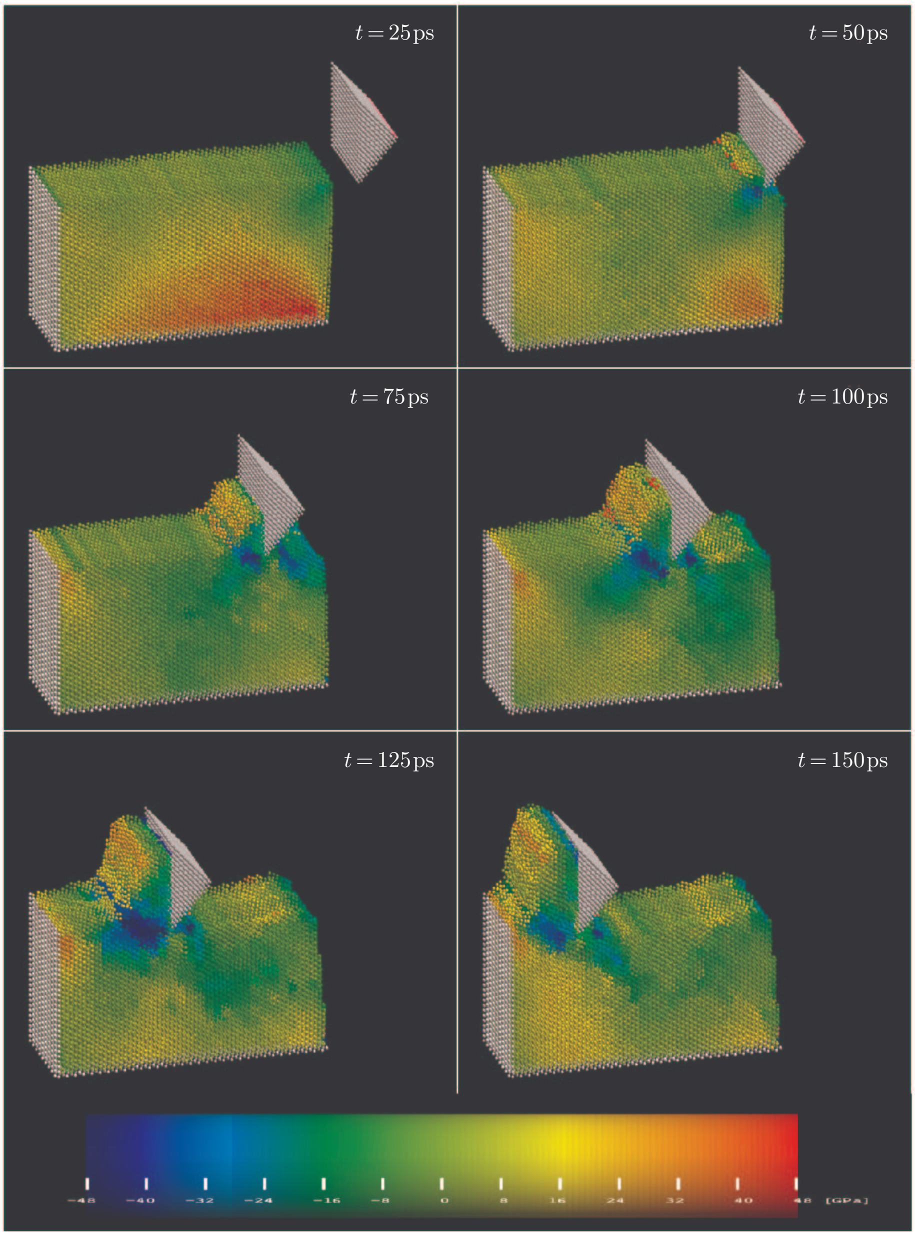

Figure 76. Distributions of the xy component of the stress tensor at selected instants. Workmaterial surface orientation (111), machining depth $2a$

In order to gain insight into the shear stresses building up in the machined workmaterial, the fields of the xy component of the stress tensor (S_{xy}) were analysed. In the first stage of the simulation, before the tool reached the machined material, a relaxation of the workmaterial, followed by its delicate bending towards the tool was observed. Even this small rightward bending yielded a xy-component of the stress tensor in the order of 40–60GPa for a considerable part of the workmaterial.

A more interesting issue is definitely the analysis of the distribution of S_{xy} after the contact of the tool with the workmaterial. Large negative values of S_{xy} (below -50GPa) were observed over the whole contact surface between the horizontal base of tool and the workmaterial (for tools A, C, and F), which again demonstrates that this horizontal contact surface of the tool strongly binds the atoms of the workmaterial, "capturing" them to follow the tool. Slightly smaller shearing was observed for the workmaterial with the (110) orientation (in the order of -30 to -40GPa). A similar region of large negative shears spreads from the tool-workmaterial point of contact at an angle of approximately 45 degrees downwards in the direction in which the tool is moving. This was observed for all tool shapes. Interestingly, the size of this region decreases with increased machining depths, which can be explained by the fact that the immobilized atoms, resist the formation of shear stresses. This is particularly well seen at the final stage of the machining – irrespective of the tool shape, an area, triangular in cross-section, can be seen forming under the tool, where S_{xy} reaches large positive values (above 30GPa).

The shape of the region in which strong shearing was observed depends also on the crystalline orientation of the material. The abovementioned shape of the region extending in the south-west direction is characteristic for the material with the (100) orientation and the (111)-oriented material, for a tool with a horizontal base. With the (111) workmaterial orientation and a tool with a smaller contact surface (B, D, and E) the region of signifcant shearing will move in front of the tool, to a lower depth. For the (110)-oriented material, significant shearing occurred directly under the tool and, for tools that attacked the material with a vertical front (A, and D), directly in front of the tool. Typical region shapes for material orientations other than (100) are shown in Figures 75–76.

5.4. Temperature fields

Figures 77–82 show the temperature fields of the machined material at selected moments for the three machining depths for two selected tool shapes: A – with a large contact surface with the material and the chip and D – with the smallest contact surface.

From the figures it can be easily seen that even with a machining speed of 100m/s, the workmaterial heats up significantly at the point of contact with the tool (by approximately 200–300K, compared to the rest of the workmaterial). For tools that attack the material with a vertical front (which was the case for both tools shown in the figures), the most heated fragment of the system is the

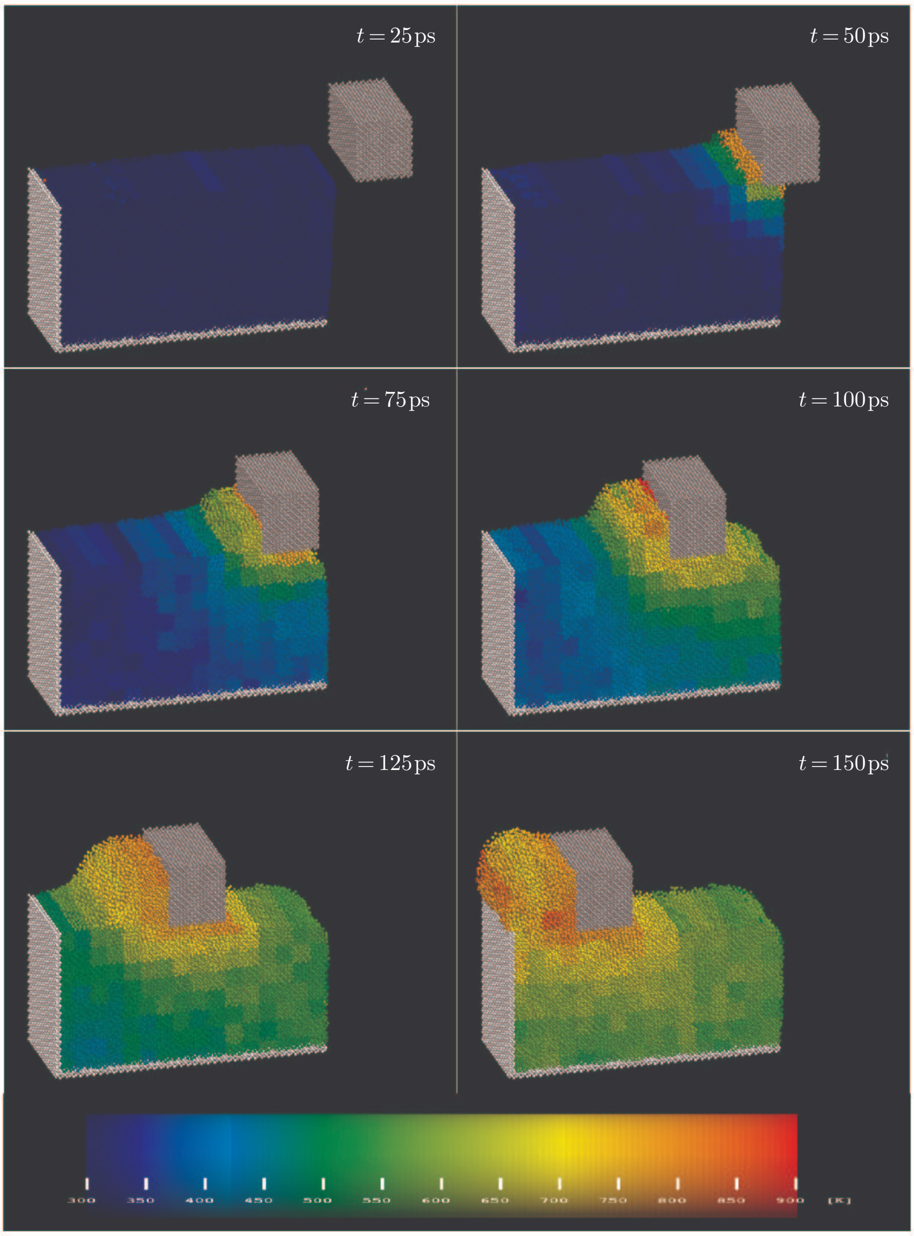

Figure 77. Temperature distribution at six different instants of the machining process for the (100)-oriented workmaterial with tool A. Machining depth $1a$

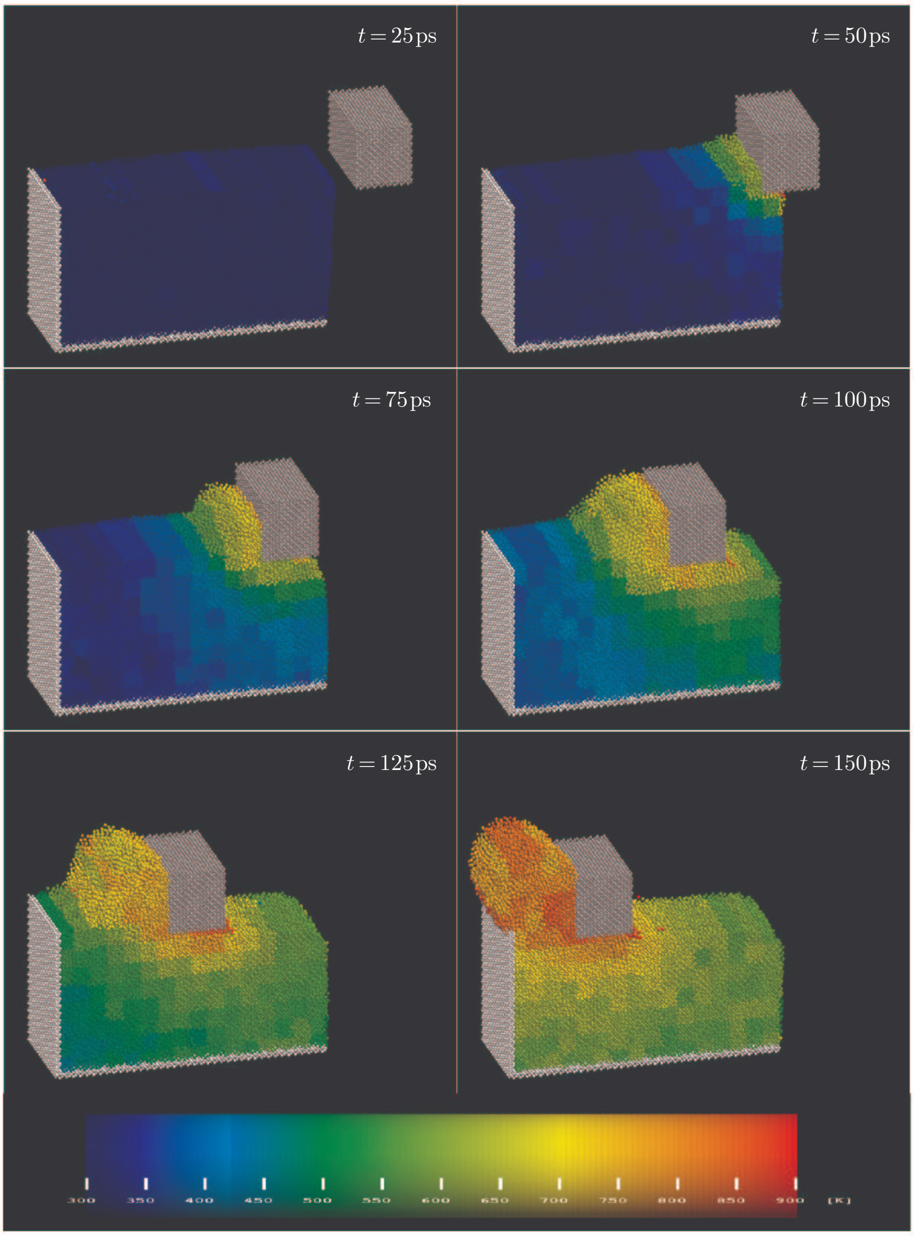

Figure 78. Temperature distribution at six different instants of the machining process for the (100)-oriented workmaterial with tool A. Machining depth $2a$

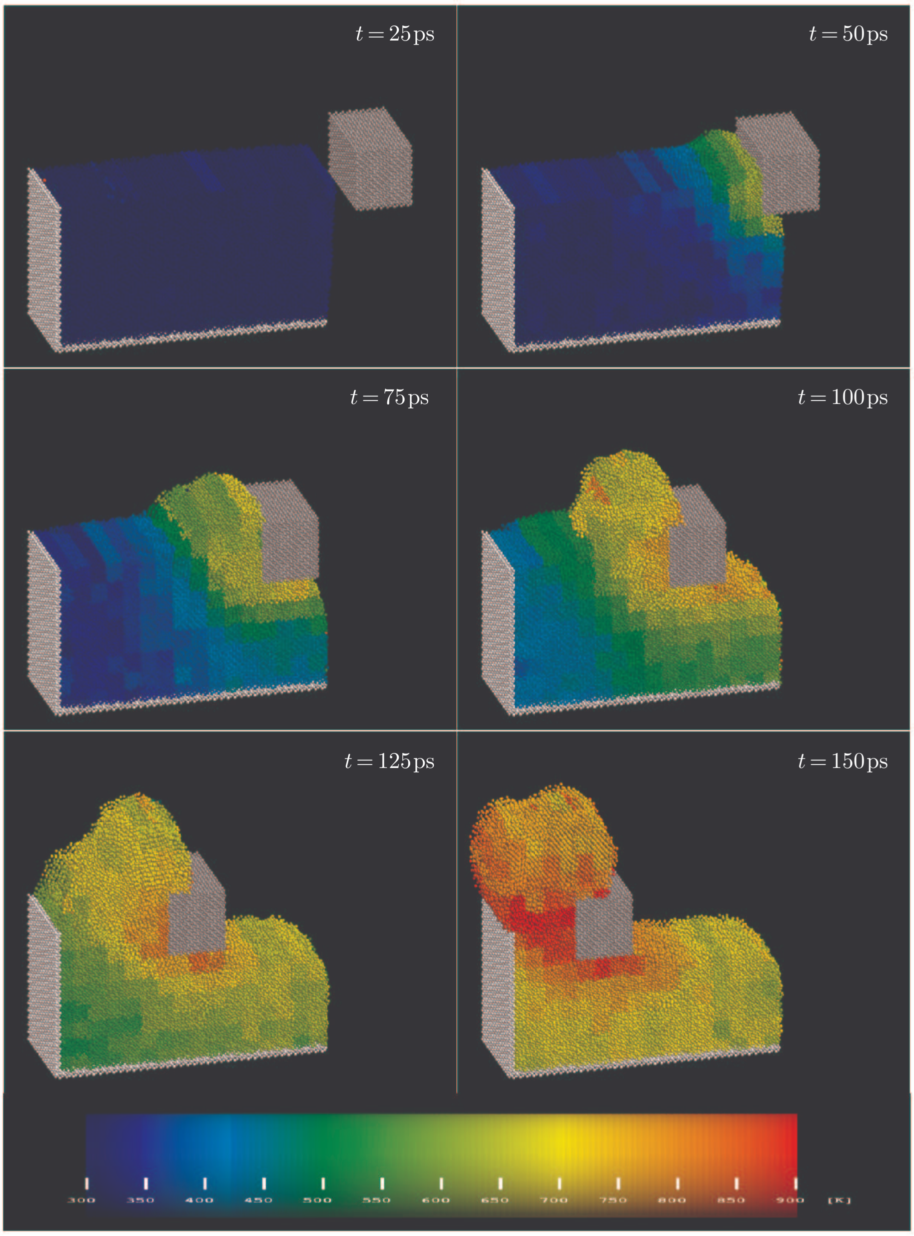

Figure 79. Temperature distribution at six different instants of the machining process for the (100)-oriented workmaterial with tool A. Machining depth $5a$

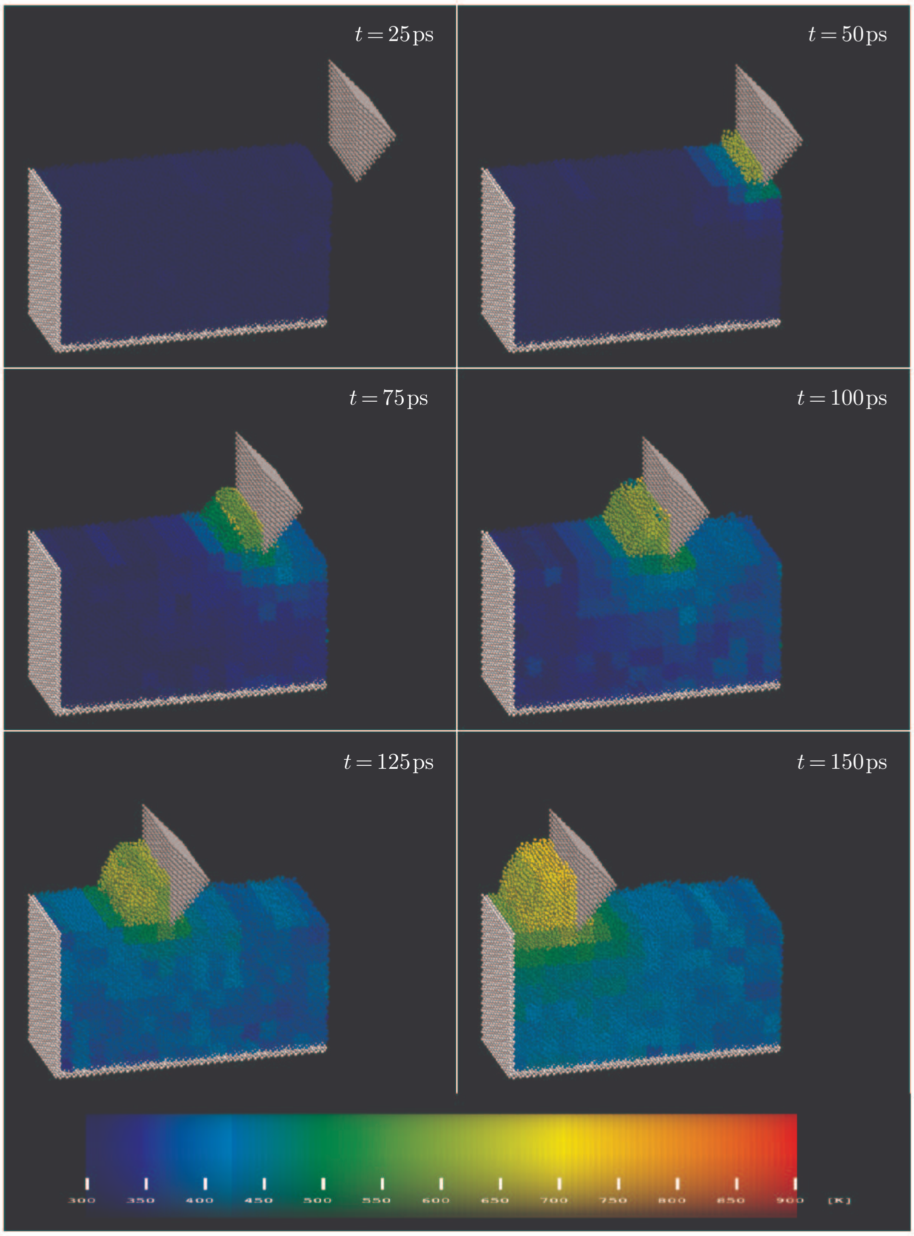

Figure 80. Temperature distribution at six different instants of the machining process for the (100)-oriented workmaterial with tool D. Machining depth 1*a*

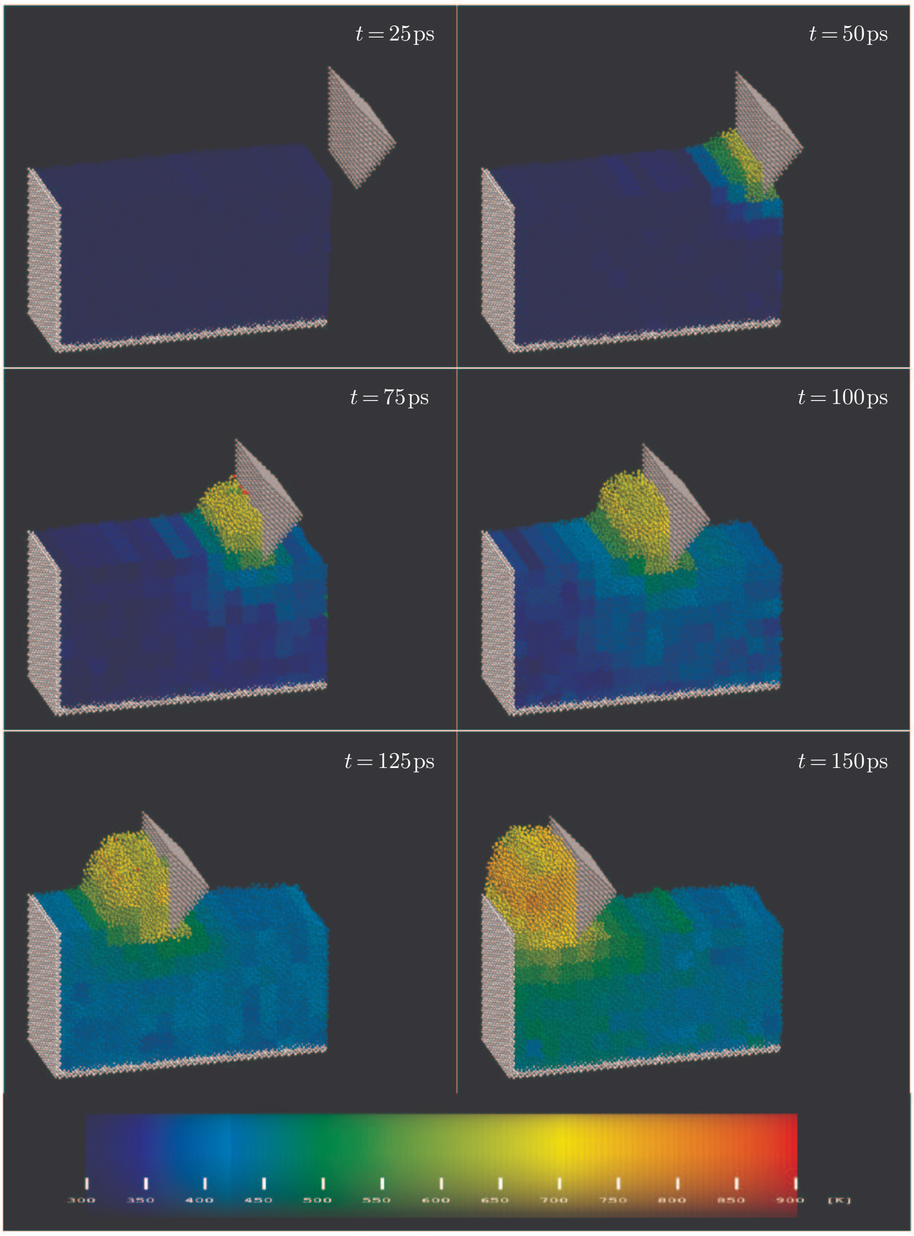

Figure 81. Temperature distribution at six different instants of the machining process for the (100)-oriented workmaterial with tool D. Machining depth $2a$

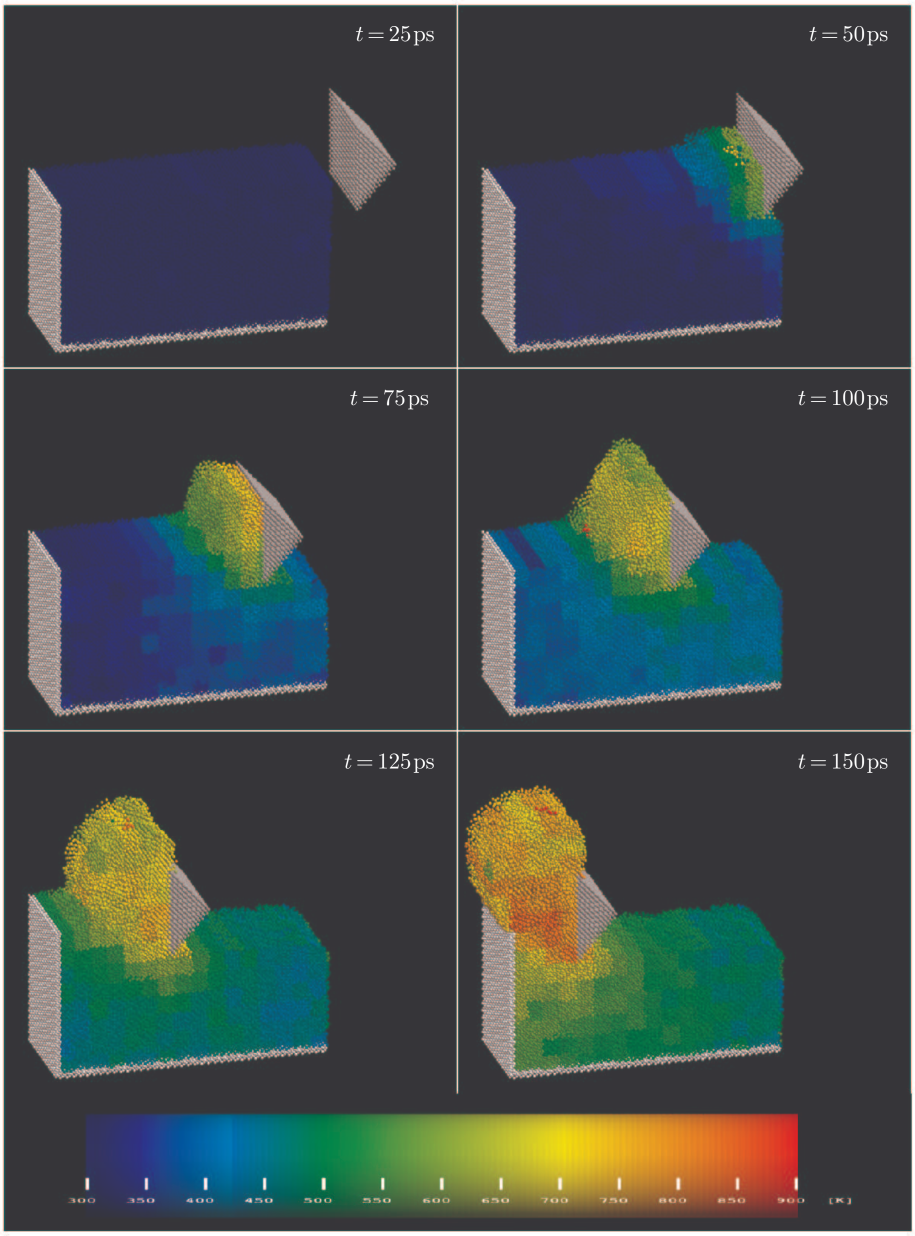

Figure 82. Temperature distribution at six different instants of the machining process for the (100)-oriented workmaterial with tool D. Machining depth $5a$

chip, and more precisely speaking – its part located directly in front of the tool. An important difference in temperature distributions occurs between the tools whose contact with the working material is large (A, C and F), and tools with a small contact surface with the material (B, D and E). For the former, significant heating up of the workmaterial was observed in the vicinity of the base of the tool which resulted in an equally high increase in temperature, both directly under the tool and in front of it, as well as in local heating up of the material directly behind the tool. Tools with small contact area with the material did not significantly heat up the material behind the tool. A larger friction surface between the tool and the material (tools A, C and F) resulted in a global increase in the workmaterial temperature by 100–150K compared to tools B, D and E. The presented figures also show clearly that the temperature distributions are maintained even after 100ps.

Figure 83 shows the dependence of the mean temperature of the machined material on the machining depth. As can be seen, from the moment when the whole length of the tool reaches the machined material, the temperature increases linearly, with the rate of the increase proportional to the machining depth.

Figure 84 shows how the mean temperature of the machined material increases during machining, for tools of varying shapes. The moment where the temperature begins to increase is different for different tool shapes. This is a consequence of the fact that differently-shaped tools take different lengths of time to contact the workmaterial. Therefore, attention should rather be directed to the slope of the curves in the area of linear increase in temperature. For tools with a similar contact surface, the slopes are roughly identical. The effect of the orientation of the workmaterial on its heating up during machining is minor. More significant heating up was observed for tools A, C, E and F during the machining of (100)-oriented material, however, the effect was not very pronounced – the final mean temperature of the machined material for that orientation was higher by only 50–70K compared to orientations (110) and (111), for which the temperature variations were almost identical. When machining was performed with tool D, the (110)-oriented material heated up the most, reaching a mean temperature that was higher by 35K than for (100) and by 60K than for (111). An analogous trend was observed for tool B, where the differences in the mean final temperature were less pronounced.

In order to better illustrate the temporal and spatial evolution of the temperature of the machined material, several regions were designated in which local temperature was measured. Three of them (A, B and C) are localized at the surface of the machined material, and two others (D and E) – in its middle section at successively greater depths. The exact location of these regions is shown in the upper panel of Figures 85–90.

The regions in question are cuboid-shaped with dimensions $2.5a\times2.5a\times e$, where a is the lattice constant and e – the workmaterial width along the z-direction.

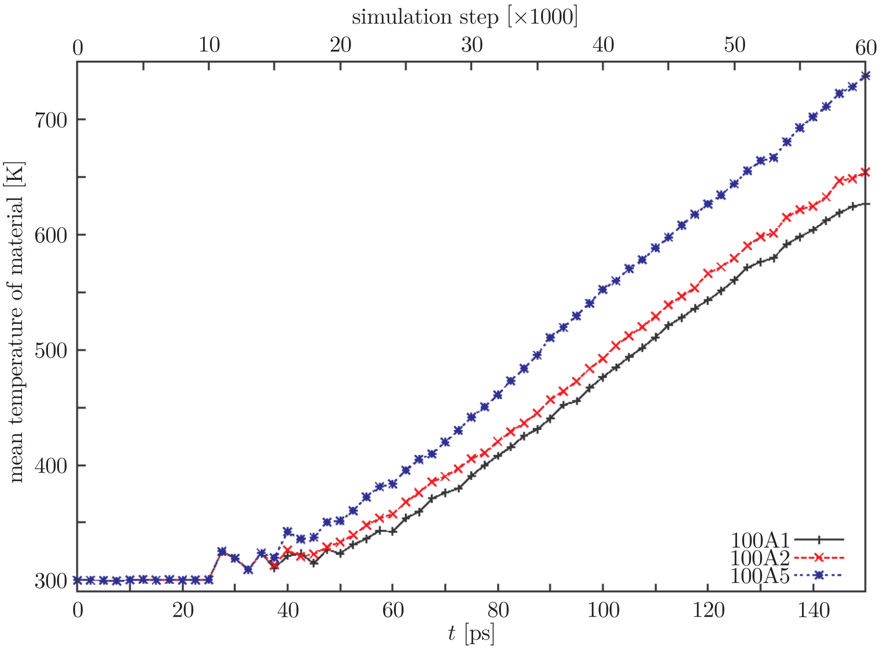

Figure 83. Mean temperature of the machined material in time, for varying machining depths. Machining speed $v = 100$ m/s, tool type A

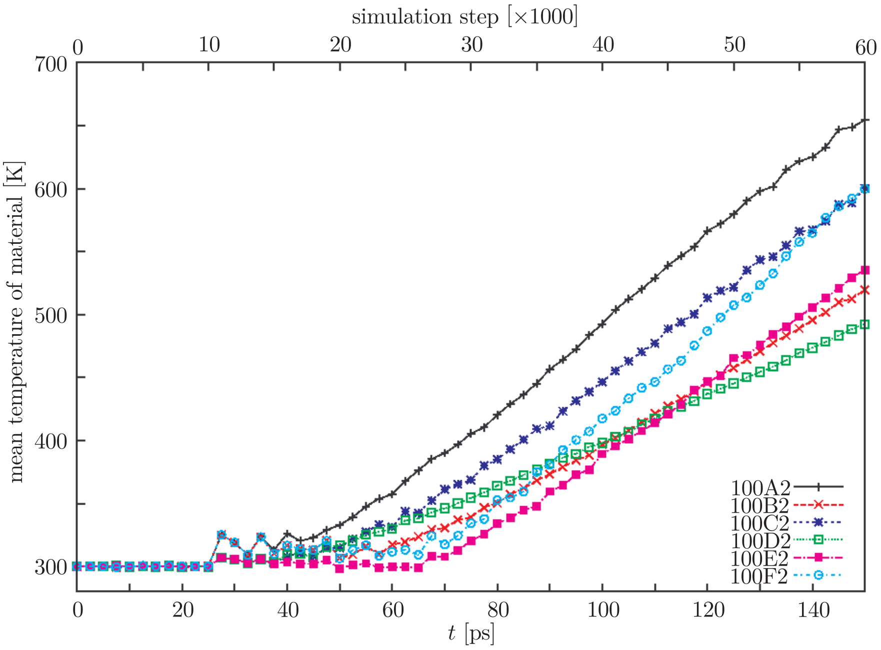

Figure 84. Mean temperature of the machined material in time, for varying tool shapes. Machining depth $2a$, machining speed $v = 100$ m/s

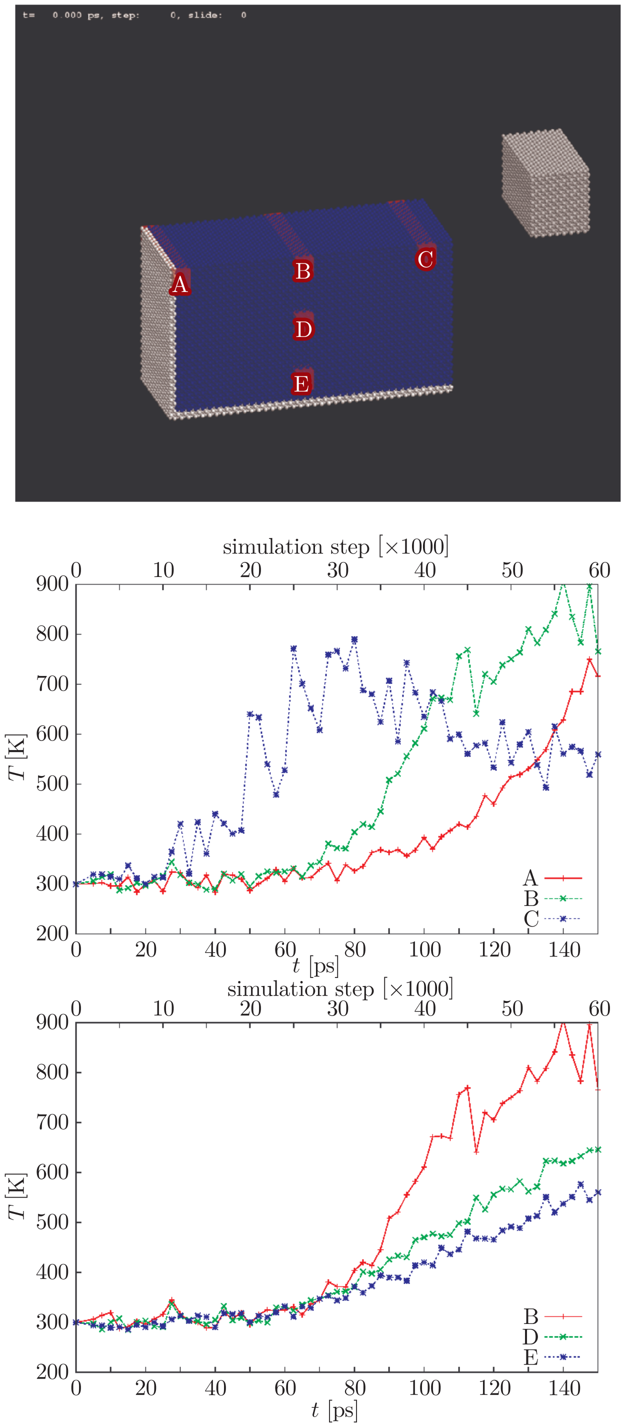

Figure 85. Changes of temperature with time, as measured in selected regions of the machined material. Machining depth $1a$, machining speed $v = 100$m/s

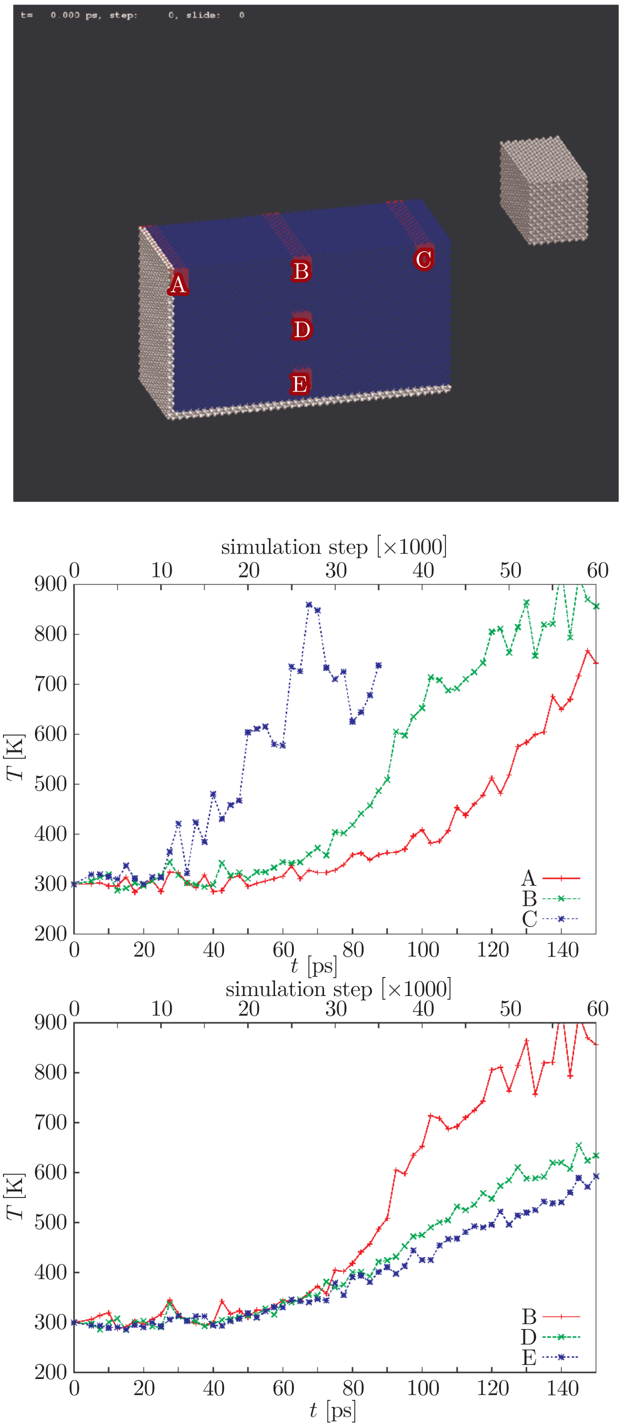

Figure 86. Changes of temperature with time, as measured in selected regions of the machined material. Machining depth $2a$, machining speed $v = 100\,\mathrm{m/s}$

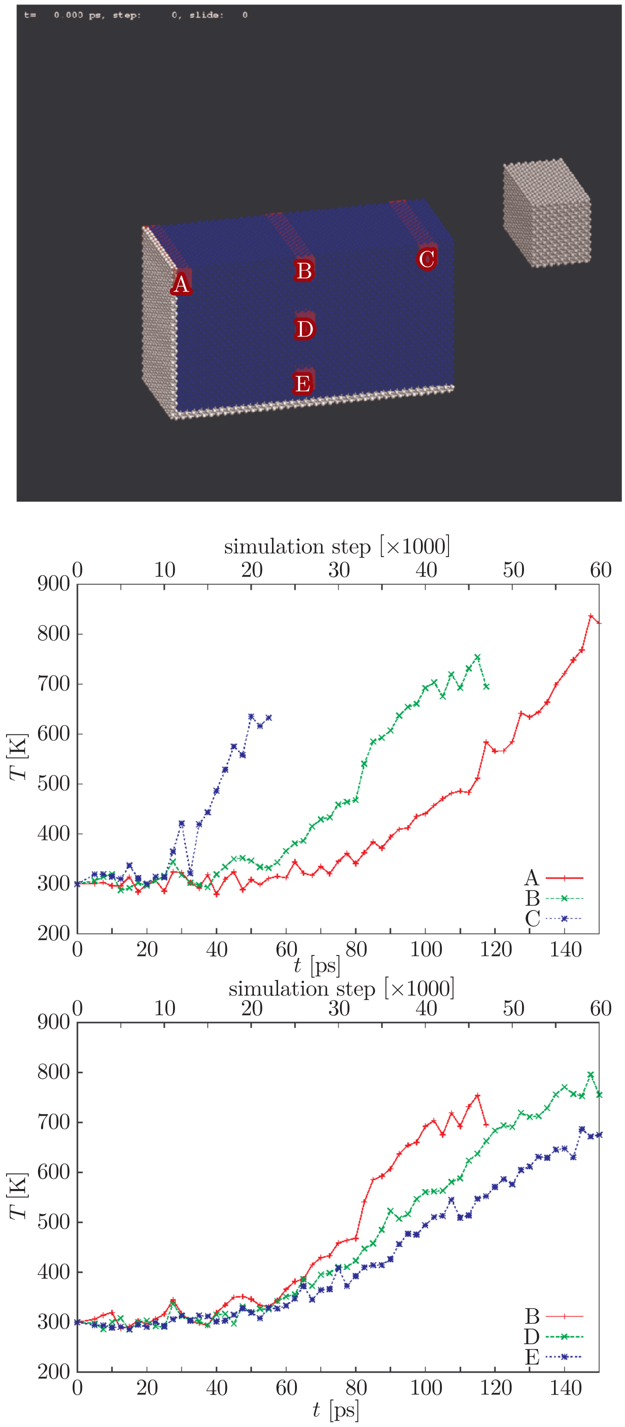

Figure 87. Changes of temperature with time, as measured in selected regions of the machined material. Machining depth $5a$, machining speed $v = 100\,\mathrm{m/s}$

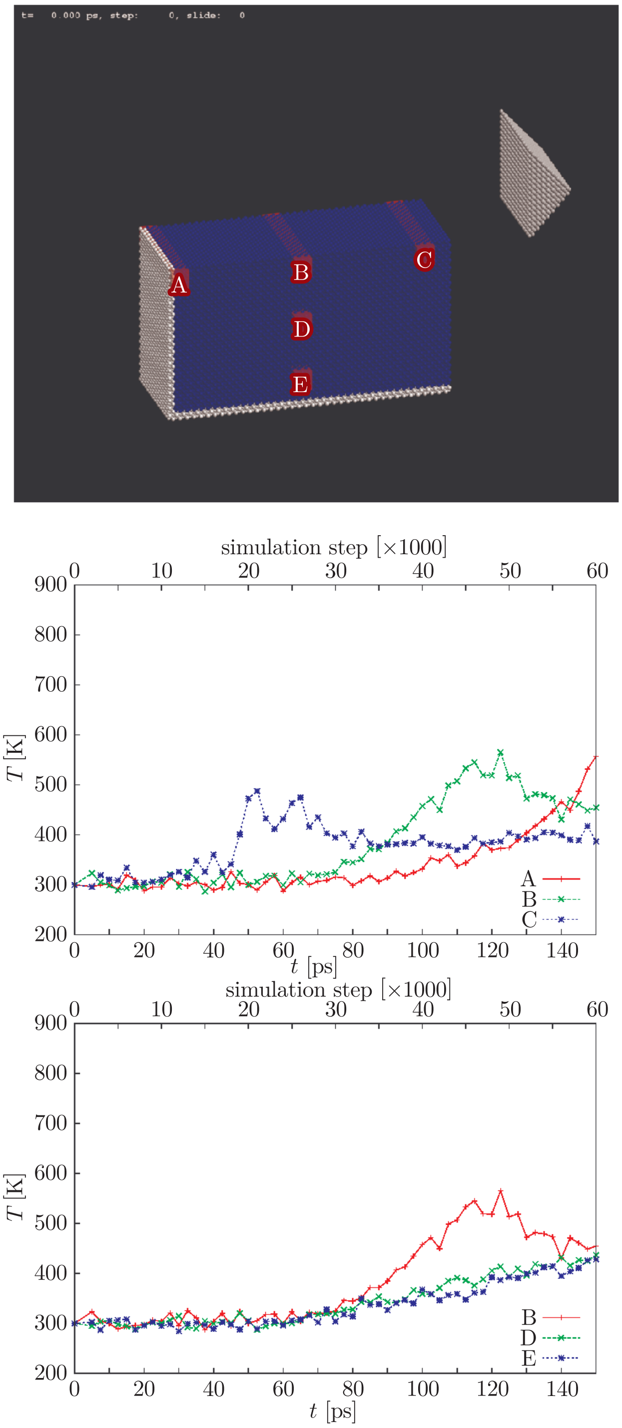

Figure 88. Changes of temperature with time, as measured in selected regions of the machined material. Machining depth $1a$, machining speed $v = 100\,\mathrm{m/s}$

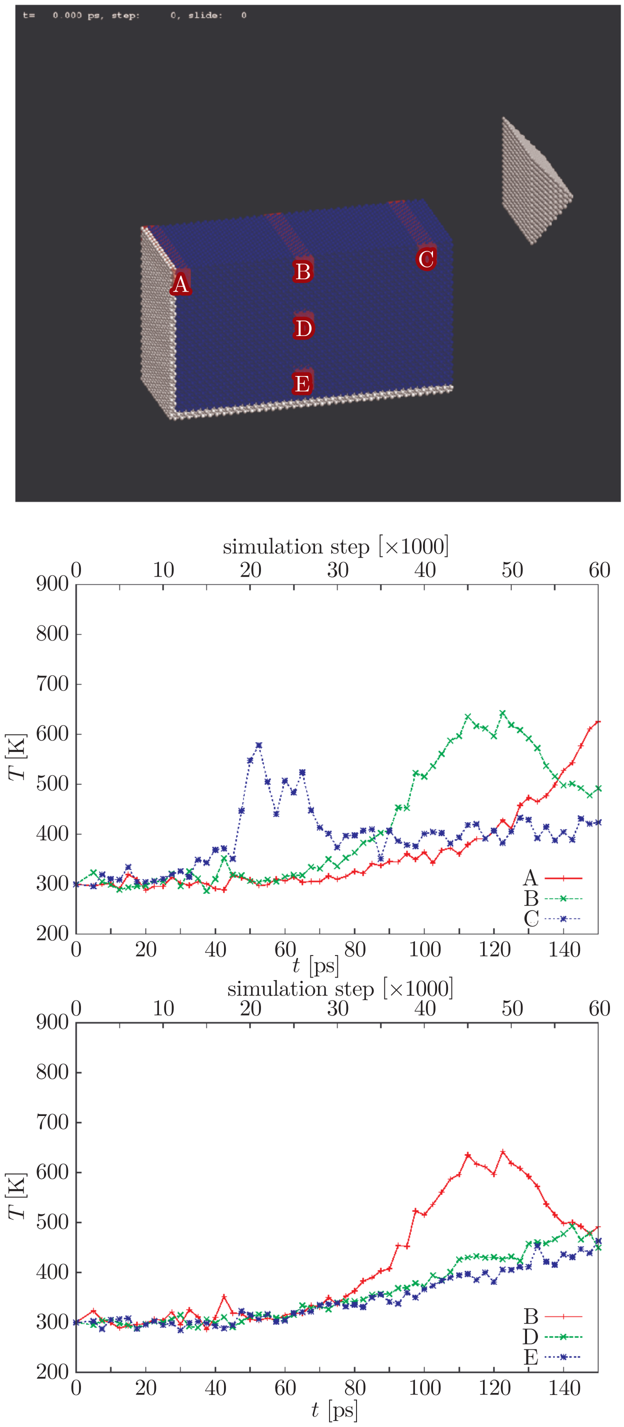

Figure 89. Changes of temperature with time, as measured in selected regions of the machined material. Machining depth $2a$, machining speed $v = 100\,\mathrm{m/s}$

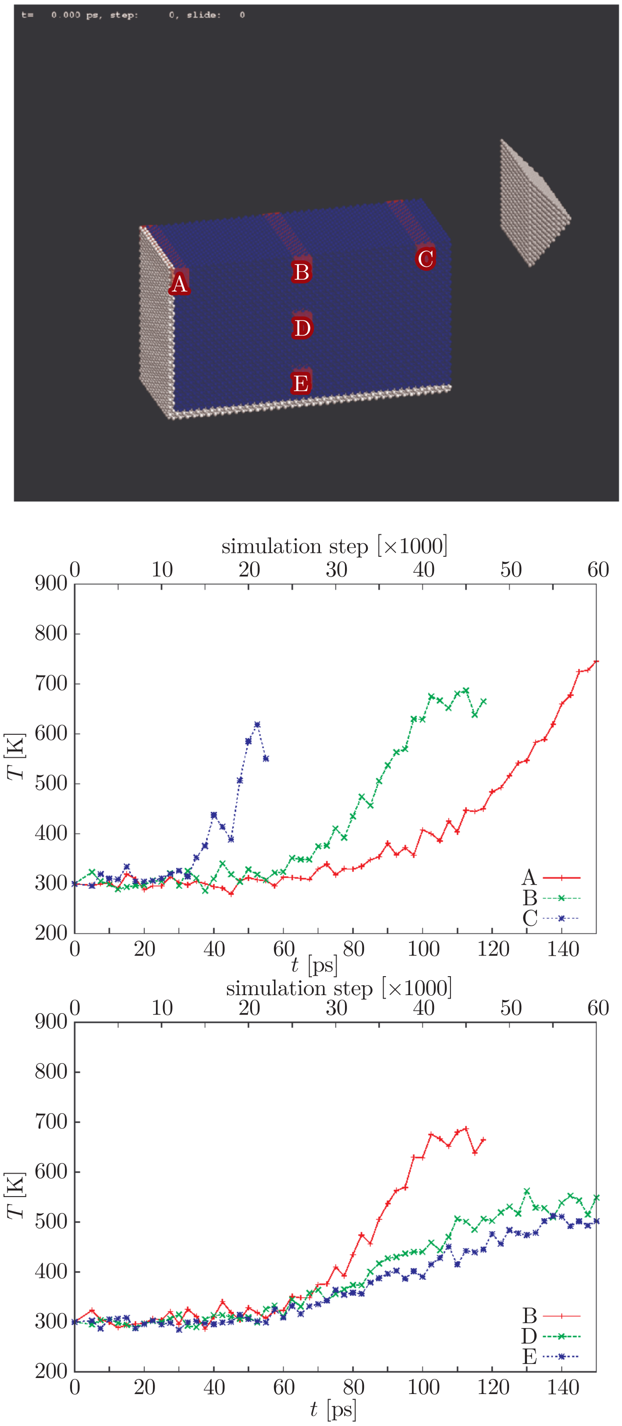

Figure 90. Changes of temperature with time, as measured in selected regions of the machined material. Machining depth 5a, machining speed $v = 100$m/s

The graphs present the temperature averaged over all atoms inside each of the mentioned regions, and over 1000 steps (2.5 ps). With the larger machining depths, the tool may have machined all atoms contained previously in areas B and C, and thus the graphs for those regions are truncated when they contain fewer than 50 atoms, to avoid representing the temperature of a region by noisy averages over a small number of atoms. Due to the fact that their velocities are undefined, tool atoms were not considered when calculating the temperature inside the regions. The obtained graphs show clearly that the temperature of the machined material is of a strongly local nature.

At the moment when the tool reaches the first of the regions (C), increasing its temperature to approximately 700 K (for the case of tool shapes with larger contact areas with the workmaterial), the local temperature in other regions still does not exceed 400 K. The time needed for the kinetic energy of particles to be transferred across distances of interest here (50–100 Å) is in the order of 100 ps which can be estimated by looking at the rate of cooling of region C when the tool has moved farther (only for the smallest machining depth, for larger depths this region is emptied from atoms). The temperature variations in subsequent regions (B, then A) are similar, *i.e.* their significant heating up is observed as the tool approaches, and there is no cooling off of these regions, because the simulation terminates before the tool can move past region B. The tool never reaches region A, despite this, the temperature of this region still does increase significantly due to the fact that it is directly in front of the tool (and as a result of the slow transport of kinetic energy from the rest of workmaterial).

As work is constantly being performed on the system when machining with a constant speed, the mean workmaterial temperature increases, and in consequence subsequent regions (C, B and A) reach higher temperatures at the moment when the tool approaches them.

The fact that the machined material heats up locally is also demonstrated by referring to the graphs illustrating the temperature of subsequently deeper regions (B, D and E). As the tool passes above it, region B reaches a temperature that is higher by 200–300 K from that in the regions located deeper within the material, whereby the differences between the subsequent regions decrease, due to that fact that the temperature gradients decrease with depth.

A comparison of the graphs for the two tool shapes under study shows the great role that is played, again, by the contact surface between the tool and the workmaterial in the heating of the latter. In the case of tool A, the final temperatures in region D were 646 K, 634 K and 755 K (for increasing machining depths), whereas for tool D the same region heated to 436 K, 450 K and 549 K, respectively.

This observation, this time in terms of the mean temperature of the workmaterial, is confirmed by Figure 84.

5.5. The nature of the plastic deformation of the machined material

In this section, the plastic deformation of the machined material will be discussed. Figures 91–108, where the workmaterial atoms are colored in accordance with their magnitude S_i of the slip vector, make it possible to gain an insight into the nature of the slips that take place in the material. The atoms whose slip vector is zero are shown only as dark gray outlines, making it easy to distinguish the regions of interest. It is only those atoms that have not moved away from any of their original neighbours by more than a certain percentage of the lattice constant (here 25%) that have a zero slip vector, hence each atom that "loses" at least one neighbour will be shown incolor in these figures. When an entire plane is colored in blue (which is the leftmost color in the colorscale), it is an indication that all atoms lying in this plane have been removed from their original neighbourhood, and (likely) moved into new lattice positions. The situation is similar for other colors, and the specific color ascribed to an atom makes it possible to conclude how much the distances of a given atom from its neighbours have changed. Dislocations propagating in the material will be characterized by small (smaller than one lattice constant) non-zero slip vector magnitudes. For atoms forming the chip, the magnitude of the slip vector will initially be small (as some neighbours have not been picked up by the tool, remaining in the workmaterial, while some are moving together with the atom of interest), whereupon its magnitude will subsequently increase (as the original neighbours are left farther and farther away). However, as the adopted cut-off radius, within which neighbours are sought, is relatively small (2.6Å), most of the original neighbours of the atom now belonging to the chip generally form the chip together with it, and their relative positions change, due to the fact that the chip does not have a crystalline structure. As a result, the typical slip vector lengths for chip atoms are larger (5–15Å), however, they are significantly smaller than the distances by which the atoms forming the chip have moved. Atoms directly on the machined surface that is produced after the tool has passed will have large magnitudes of S_i (above 10Å). Not only have those atoms been dislocated from their nodal positions, but most of their neighbours (practically all those positioned above) have been translated away by the tool.

The adopted threshold value λ (25% of the lattice constant), which determines the relative distance above which the slipped atoms are counted is relatively small, as a result, it is possible to observe the reversing of "near-slips" comprising atom planes of small but non-zero magnitudes of S_i, which have moved relatively far, however, not far enough to reach an energetically stable new location.

The high sensitivity resulting from the abovementioned low threshold makes it possible to trace not only the actual dislocations occurring in the material, but also regions where they may potentially appear (it is possible to differentiate the "near-dislocations" from dislocations on animations from the simulation where it is seen, snapshot by snapshot, whether the colored areas are stable, or whether

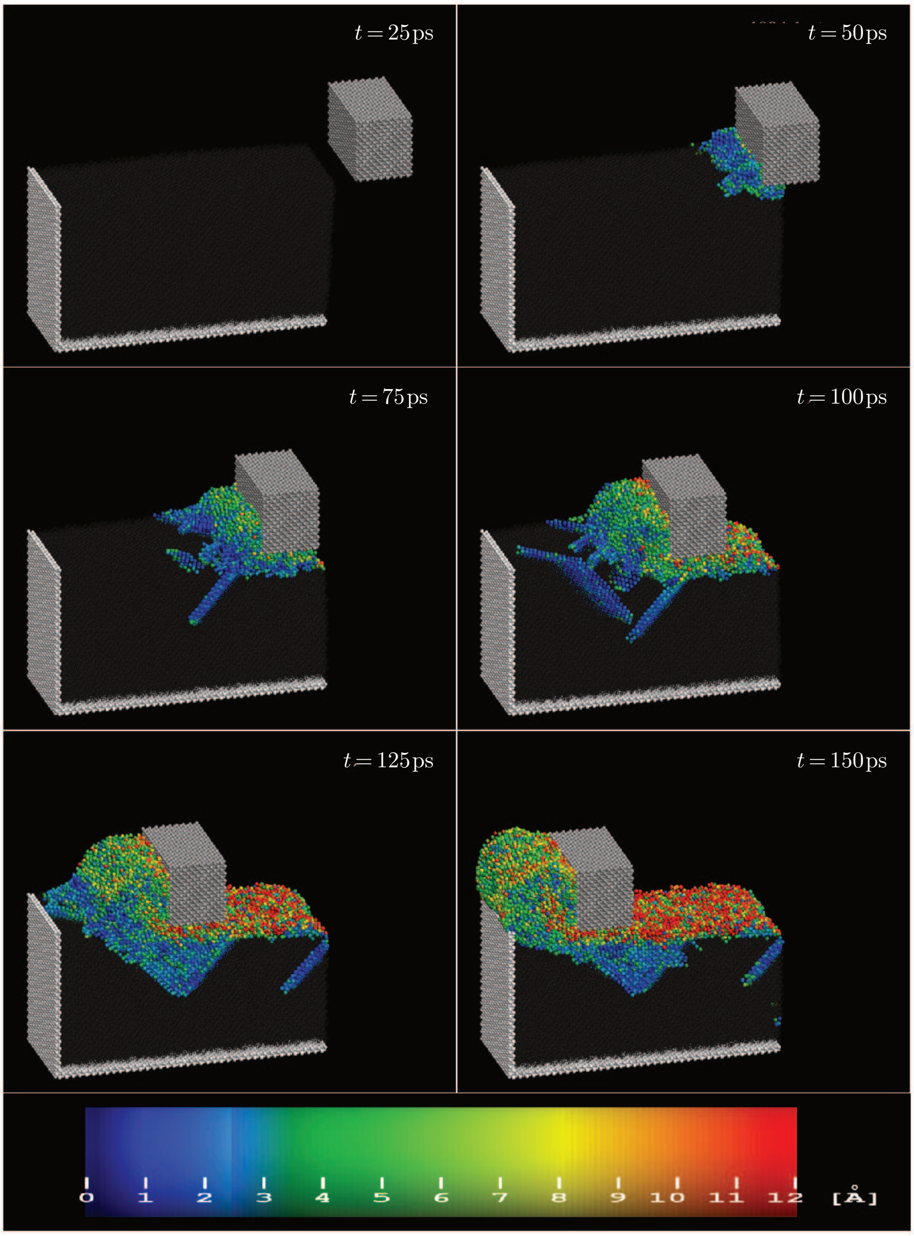

Figure 91. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool A and a machining depth of $1a$

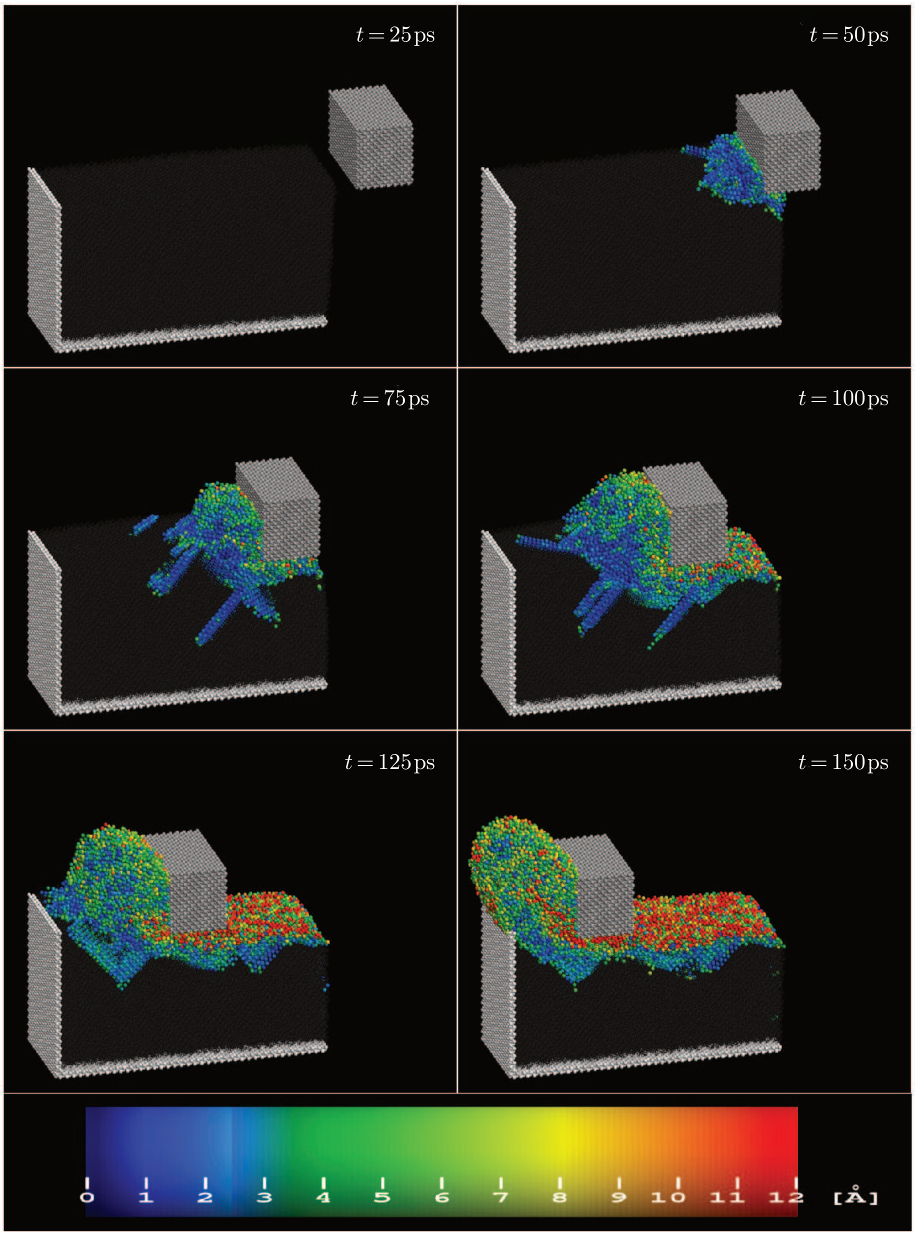

Figure 92. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool A and a machining depth of $2a$

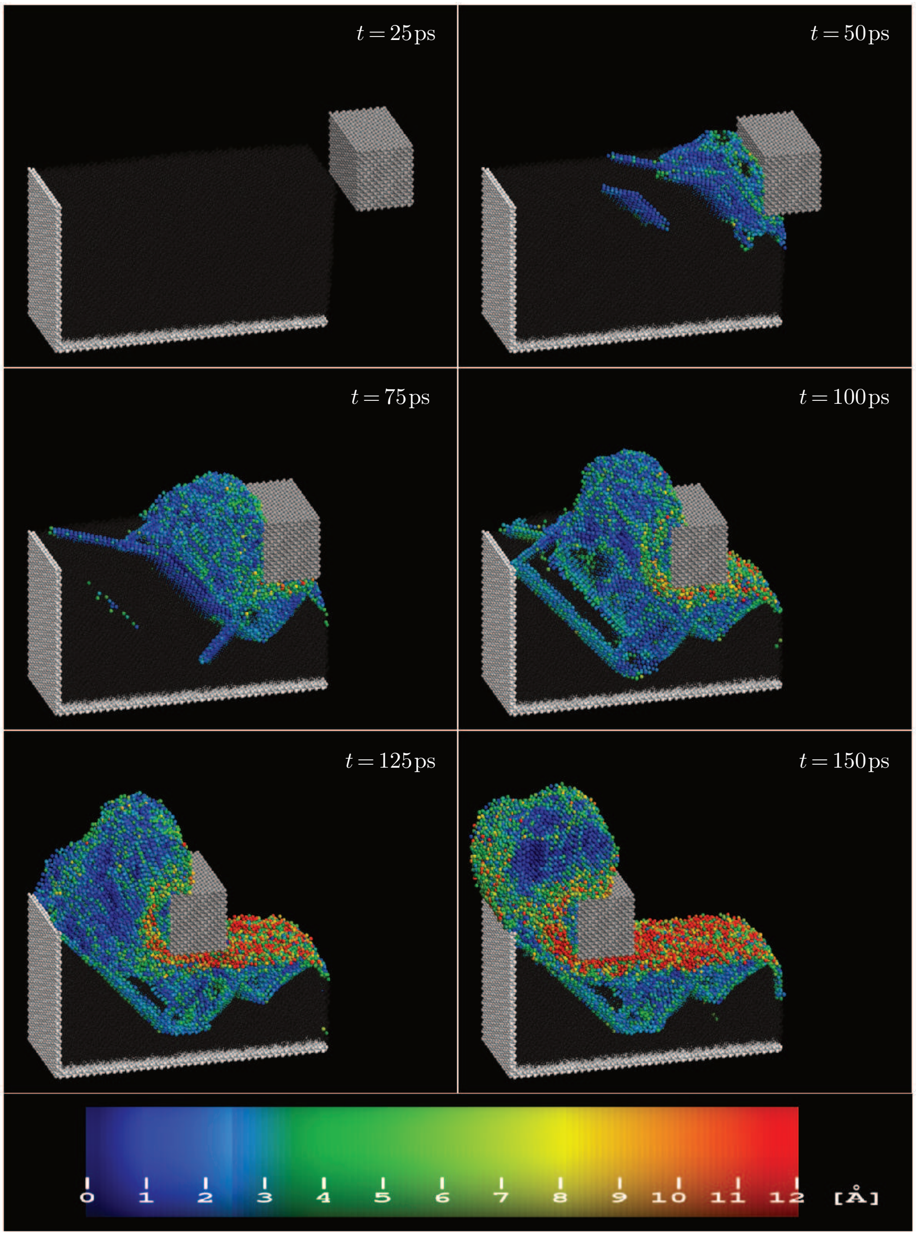

Figure 93. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool A and a machining depth of $5a$

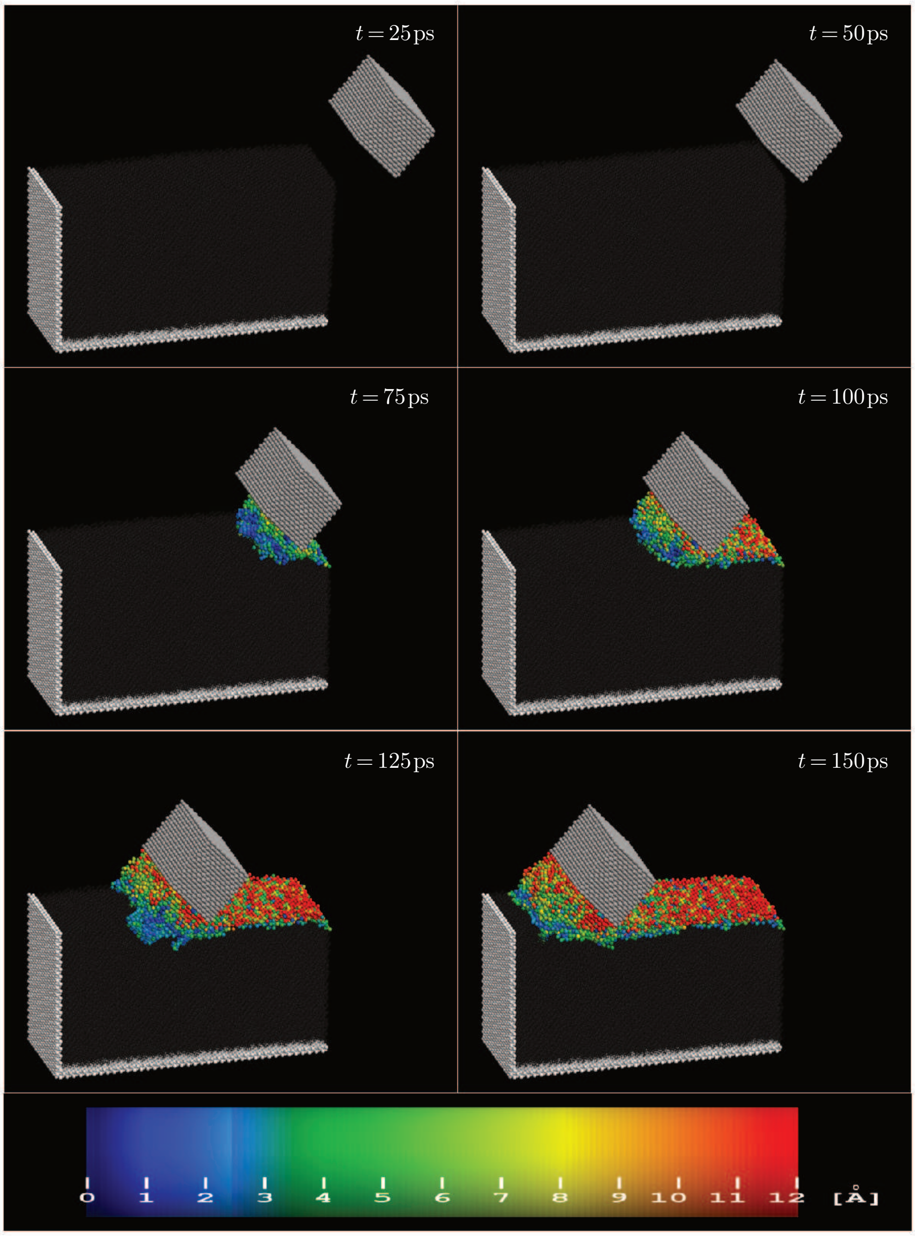

Figure 94. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool B and a machining depth of $1a$

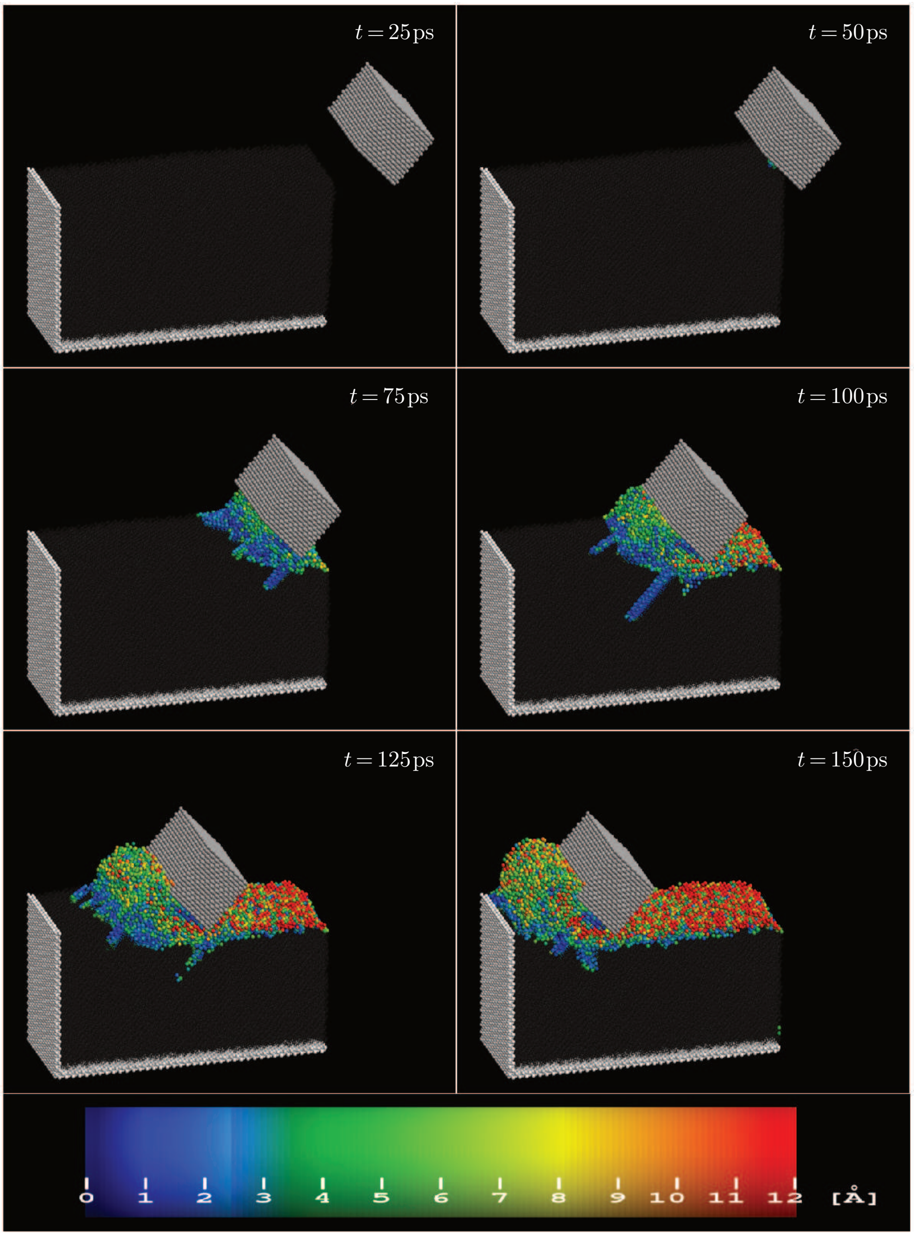

Figure 95. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool B and a machining depth of $2a$

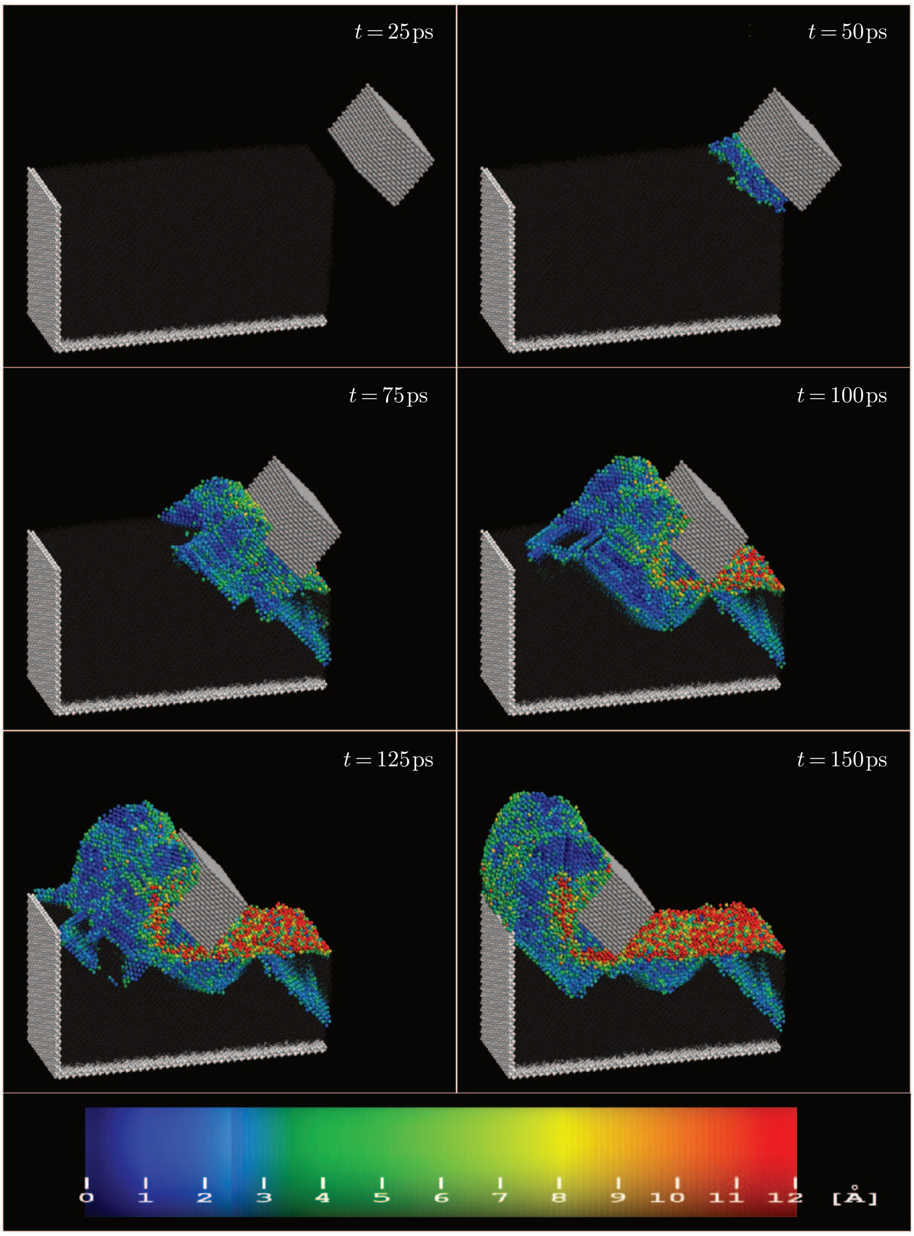

Figure 96. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool B and a machining depth of $5a$

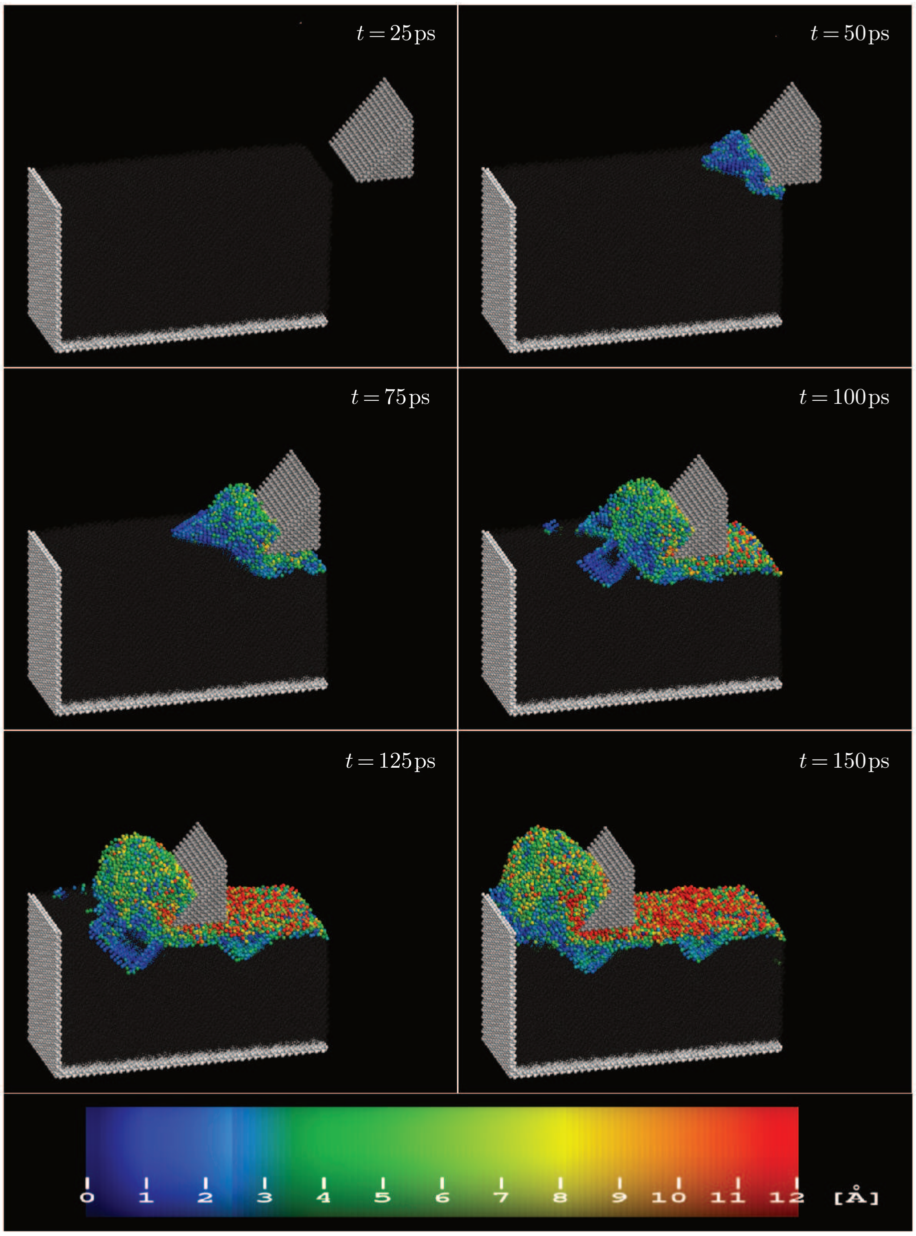

Figure 97. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool C and a machining depth of $1a$

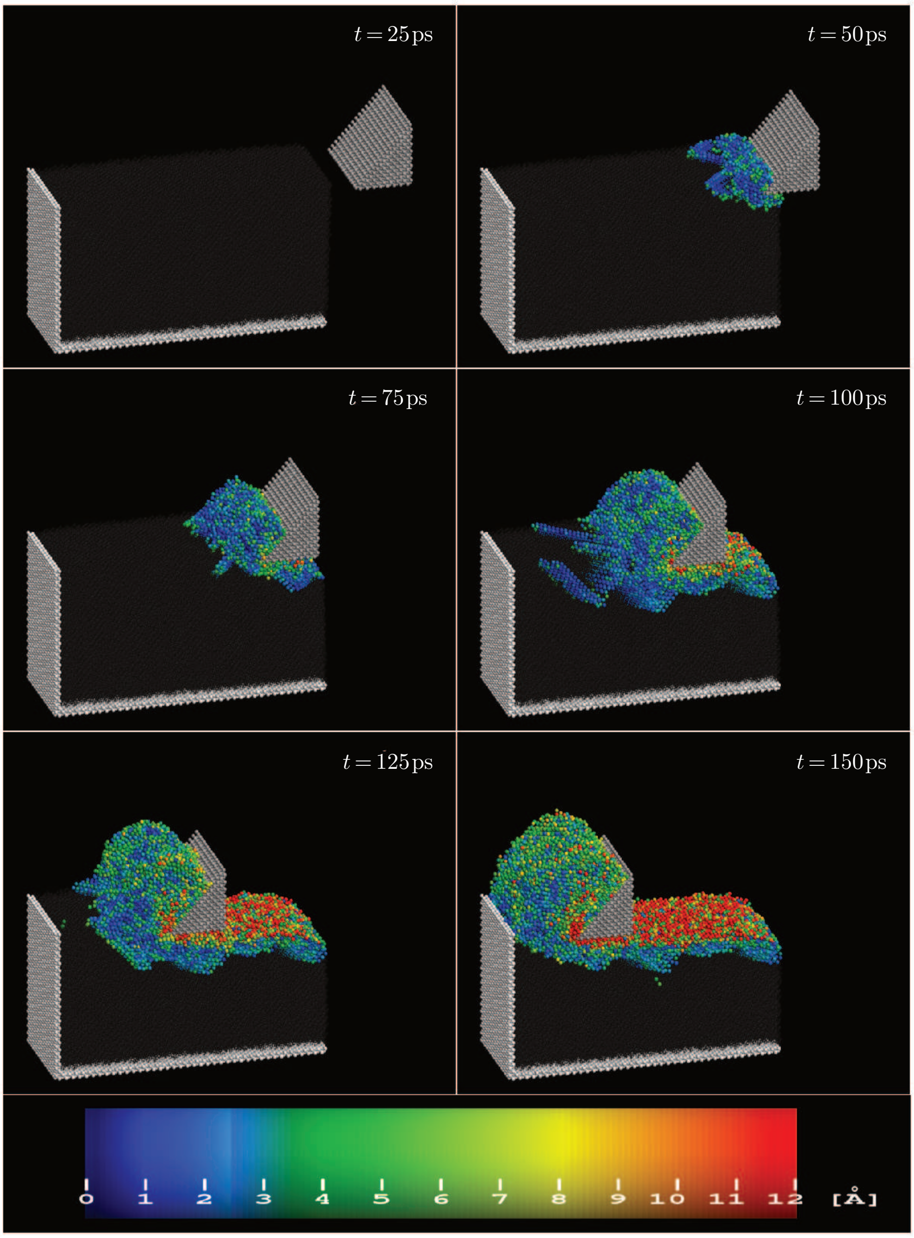

Figure 98. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool C and a machining depth of $2a$

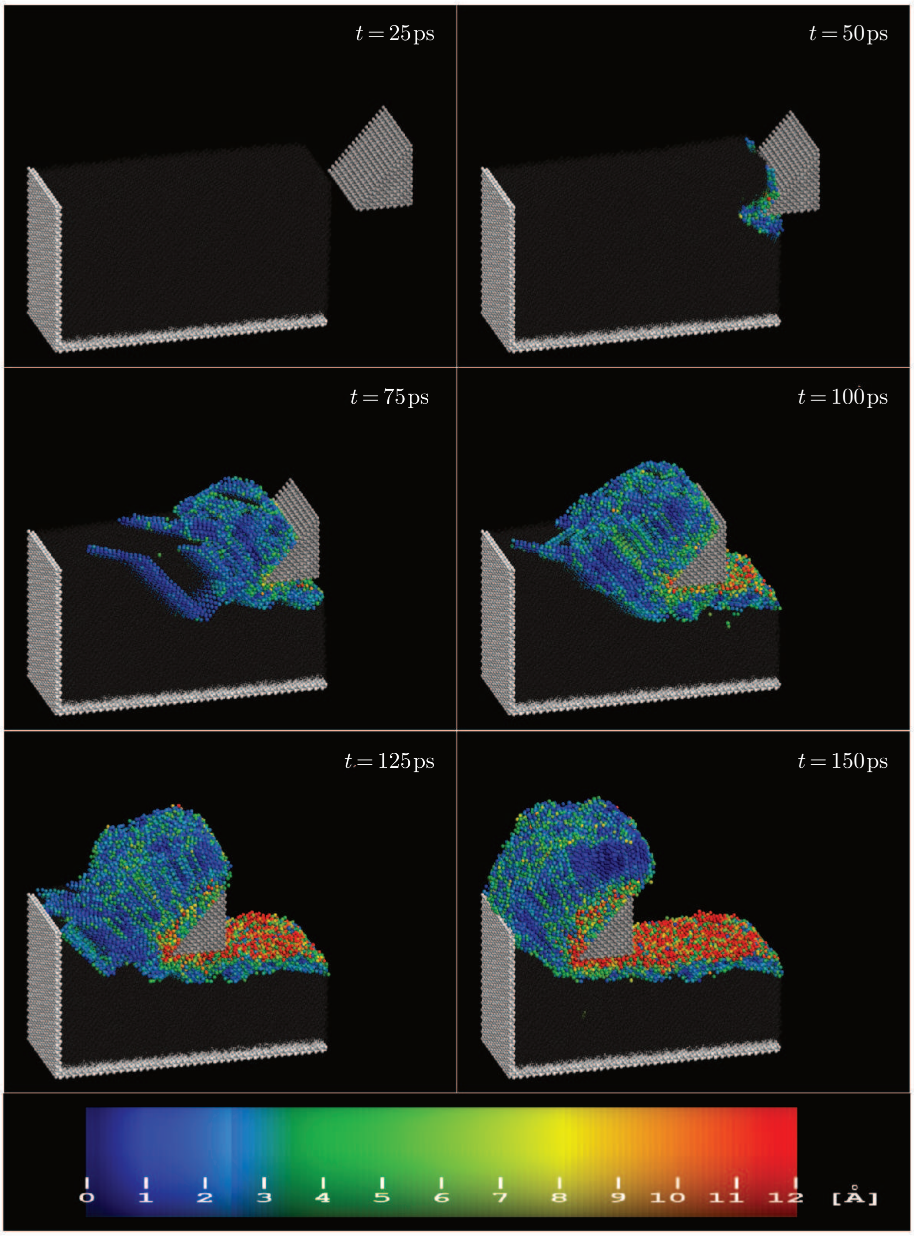

Figure 99. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool C and a machining depth of $5a$

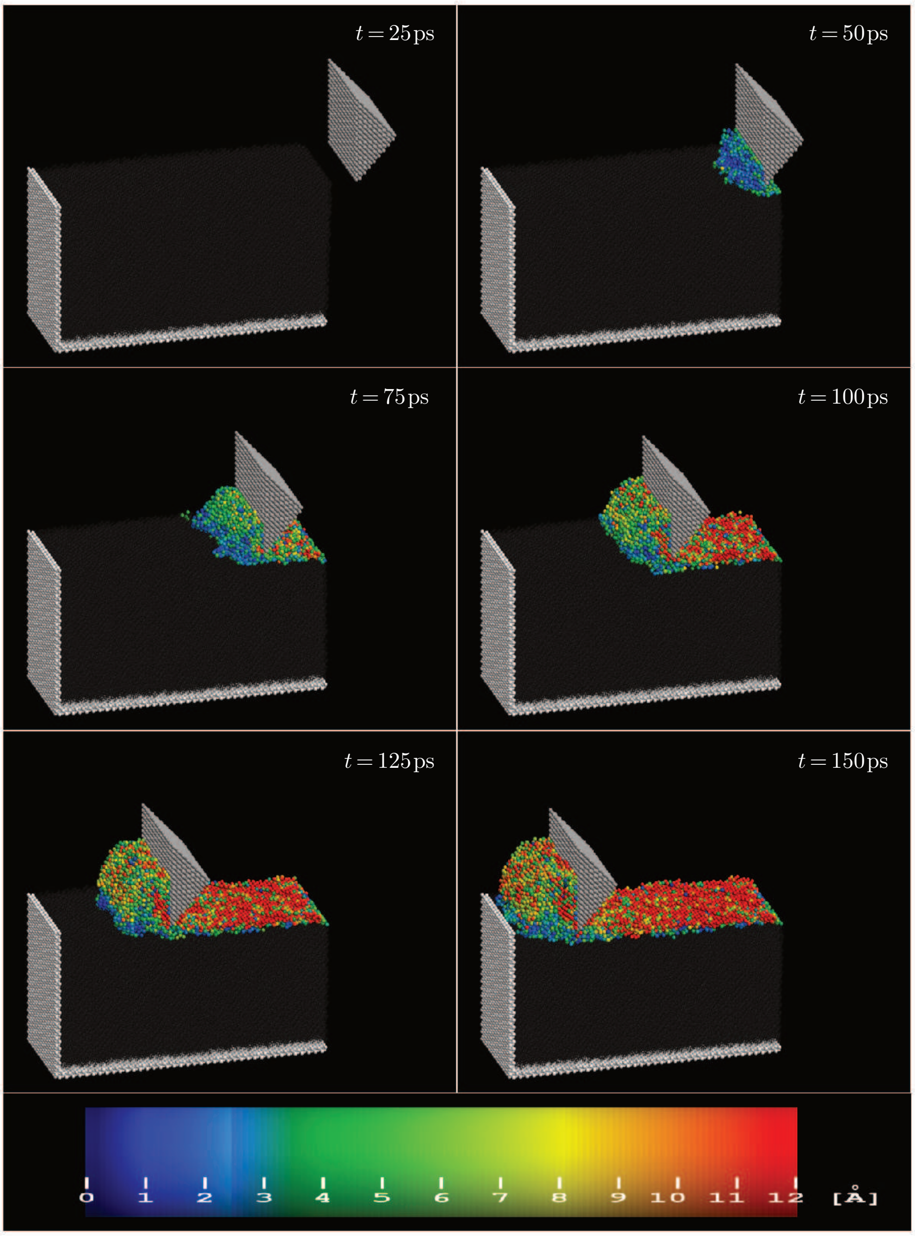

Figure 100. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool D and a machining depth of $1a$

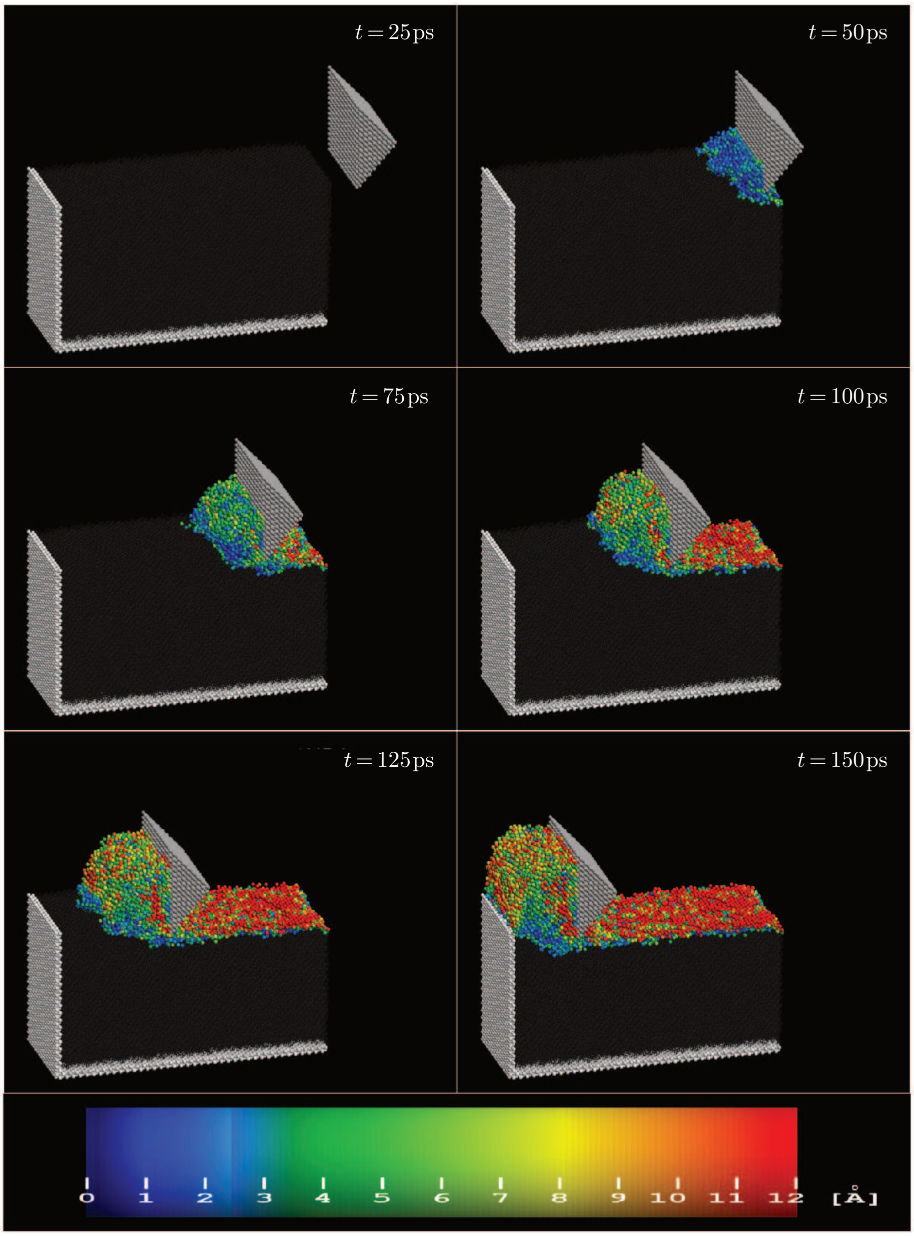

Figure 101. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool D and a machining depth of $2a$

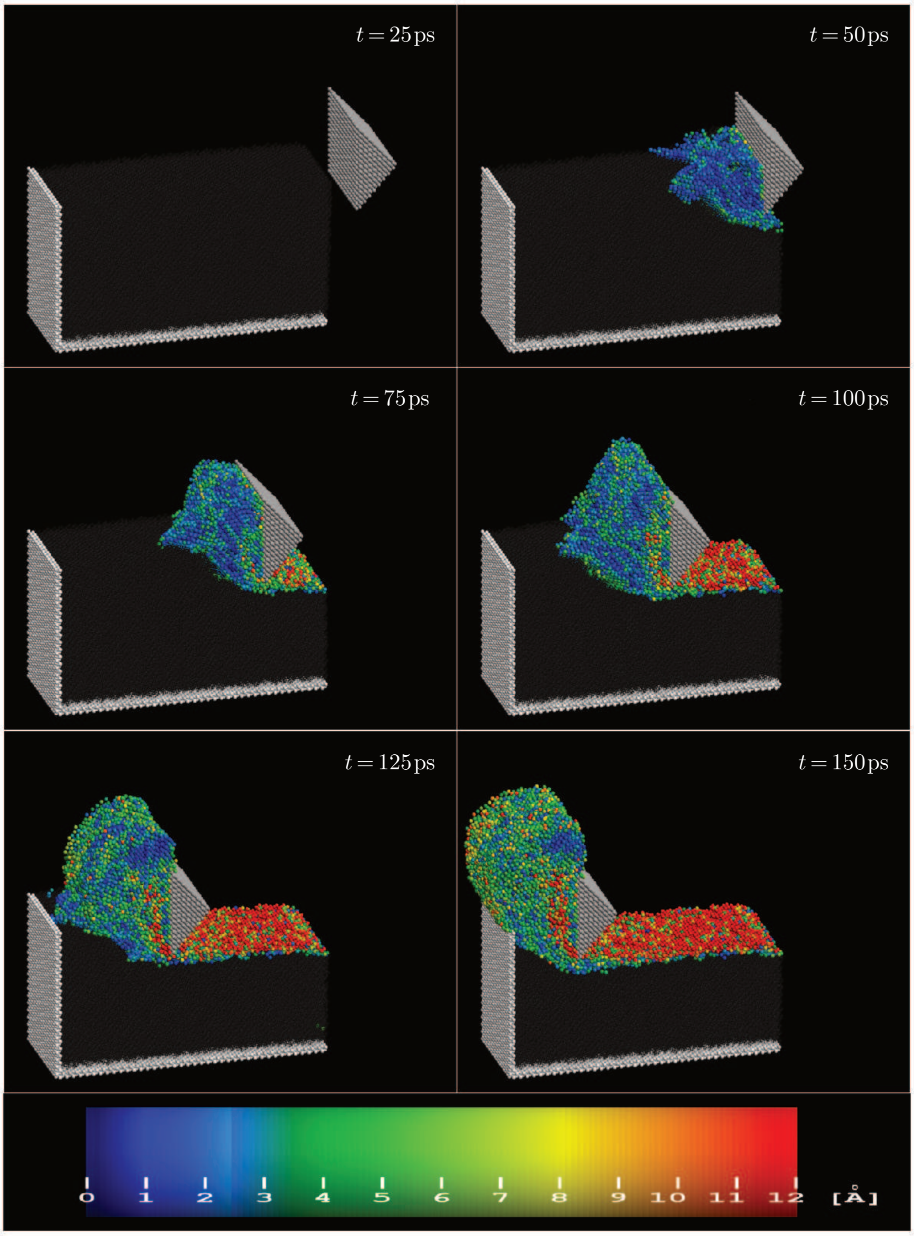

Figure 102. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool D and a machining depth of $5a$

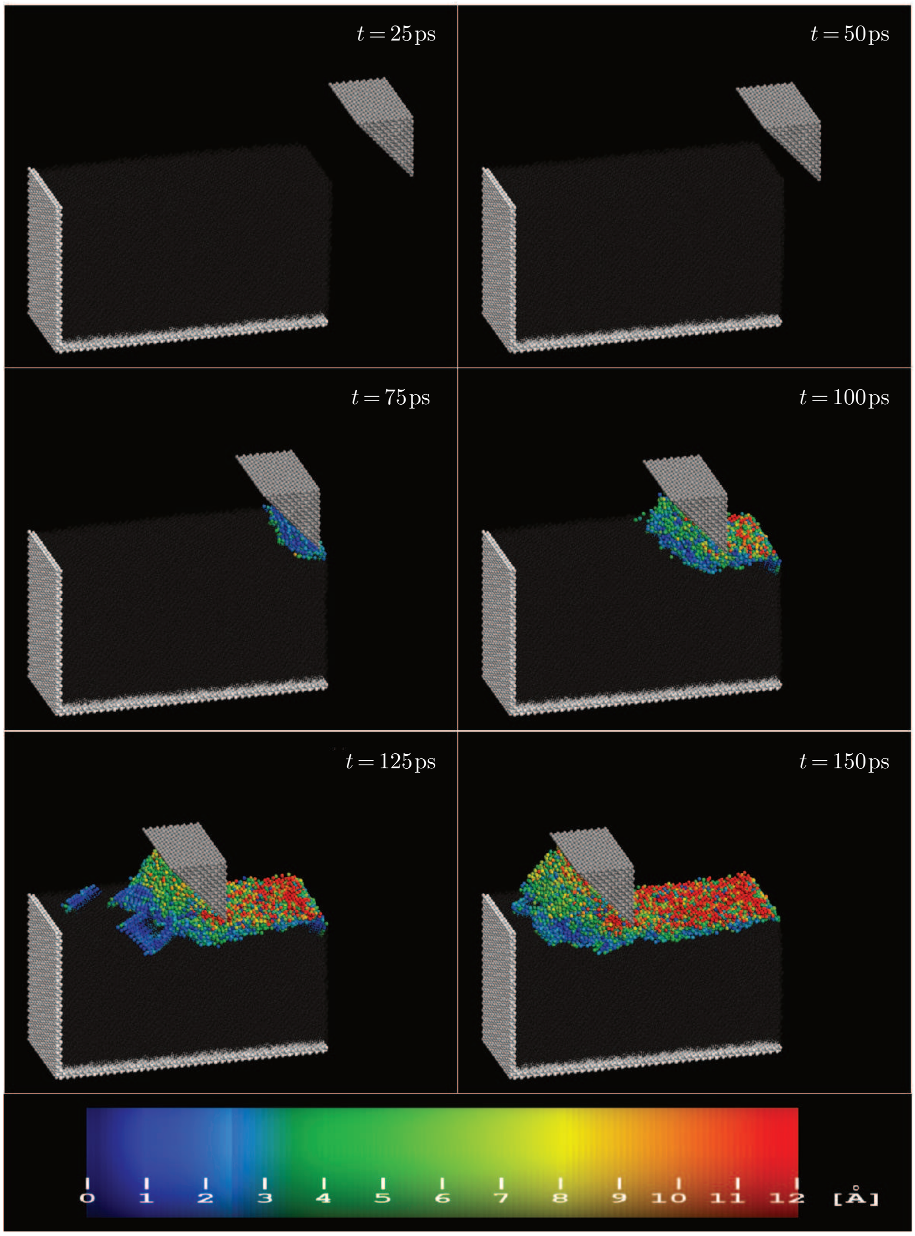

Figure 103. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool E and a machining depth of $1a$

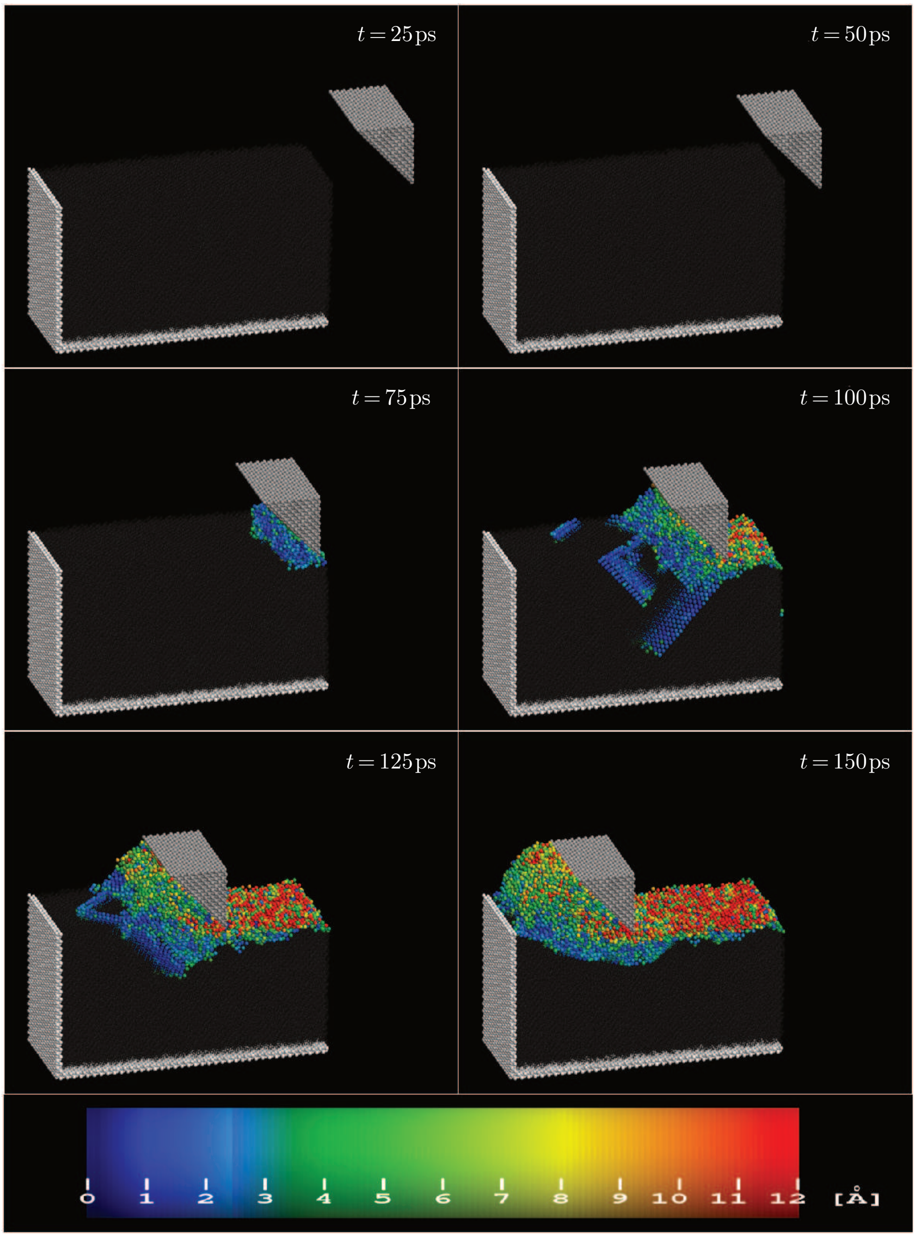

Figure 104. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool E and a machining depth of $2a$

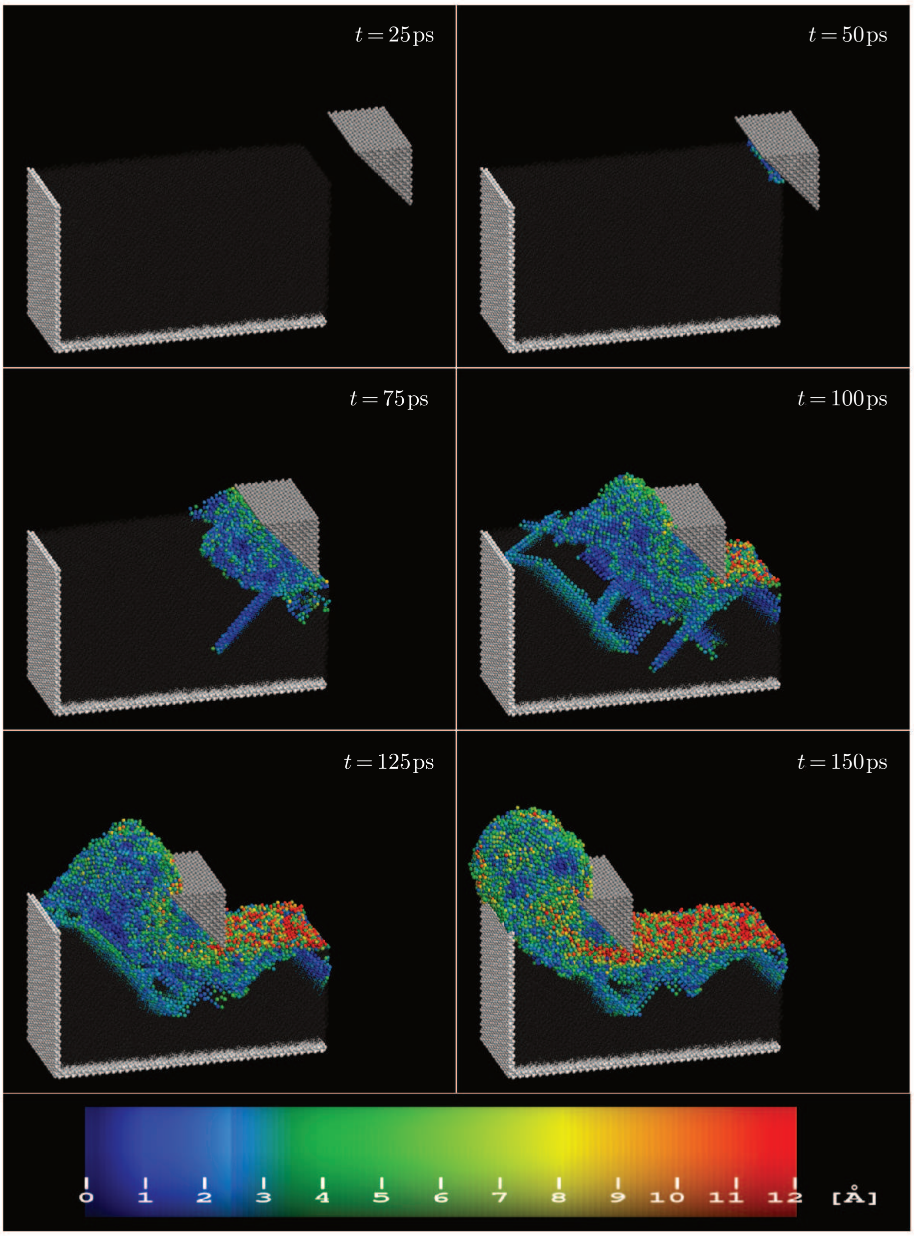

Figure 105. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool E and a machining depth of $5a$

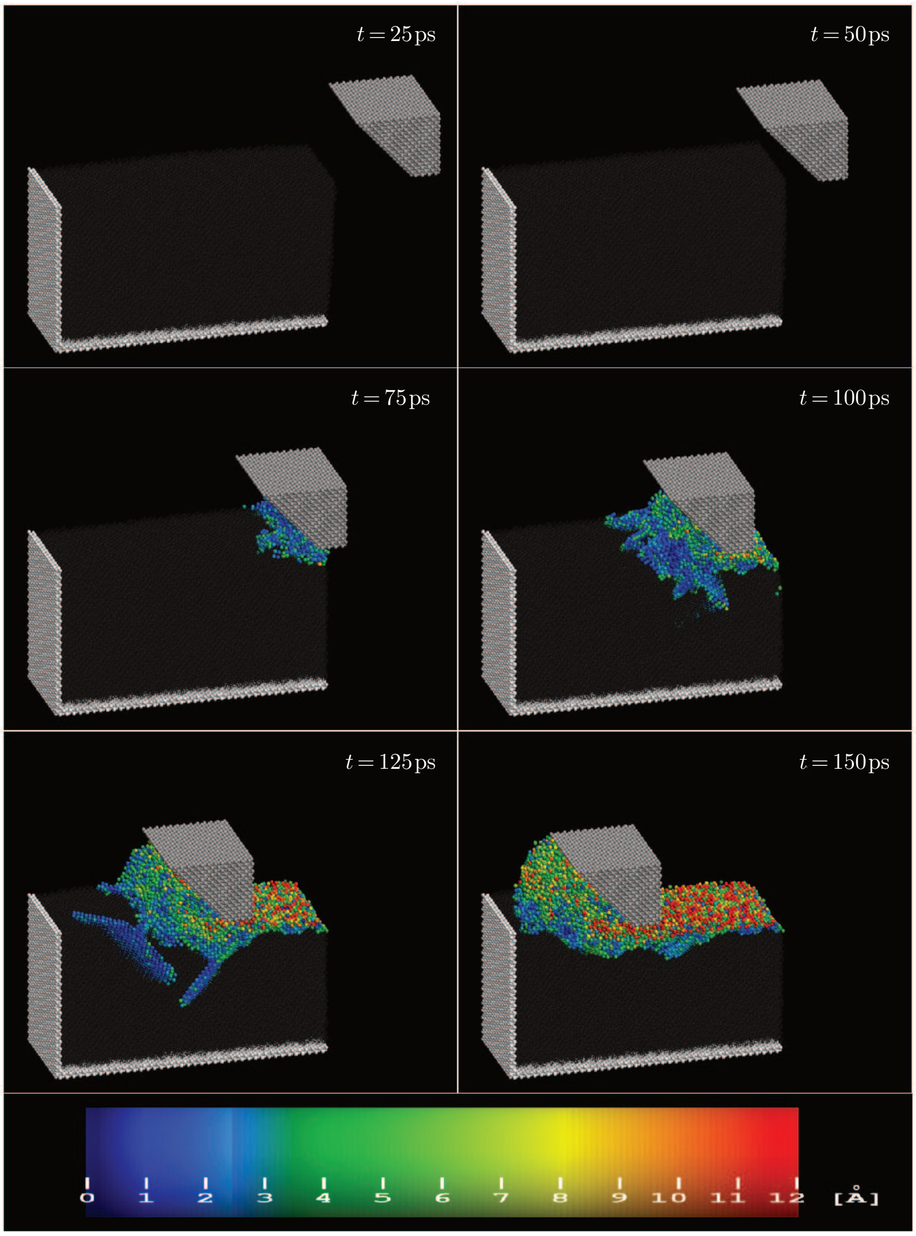

Figure 106. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool F and a machining depth of $1a$

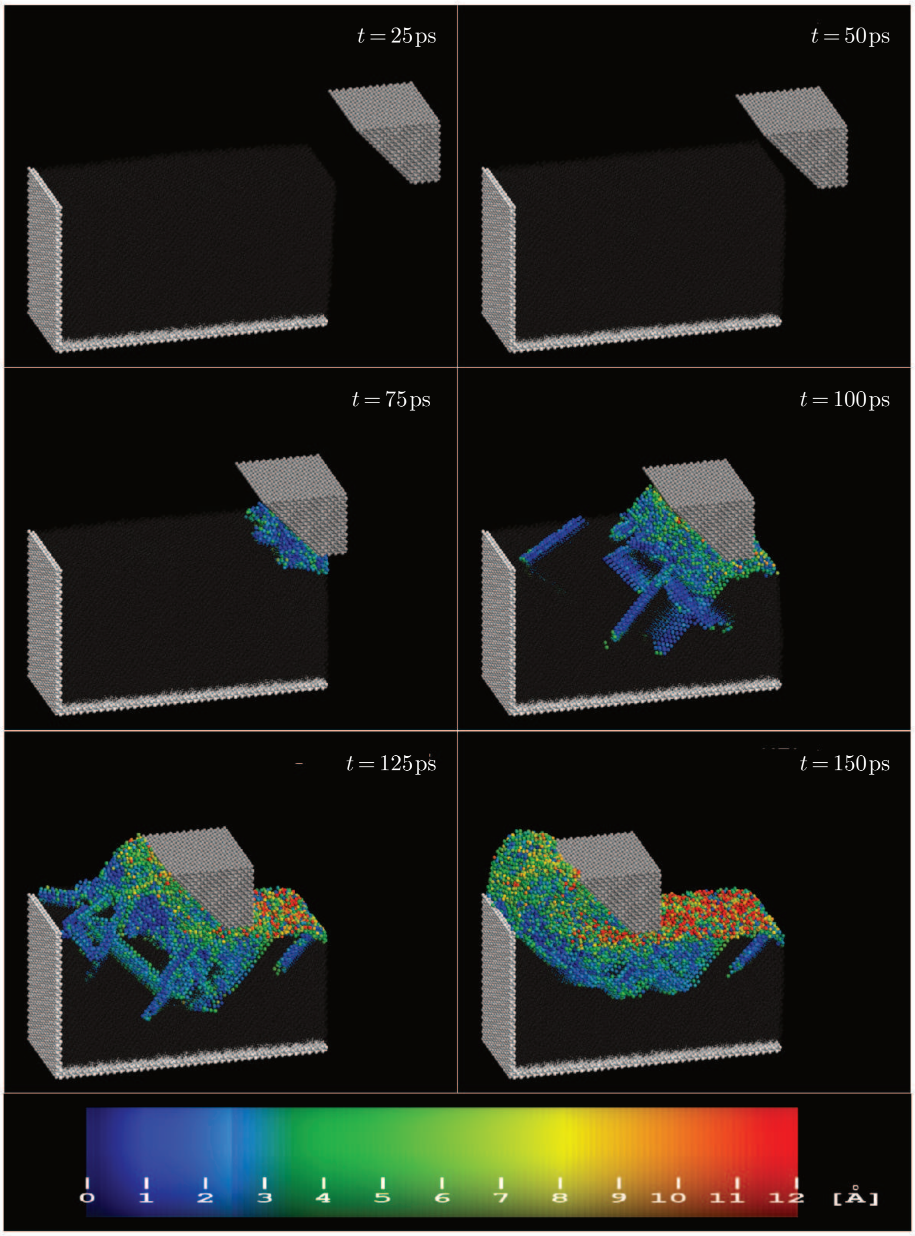

Figure 107. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool F and a machining depth of $2a$

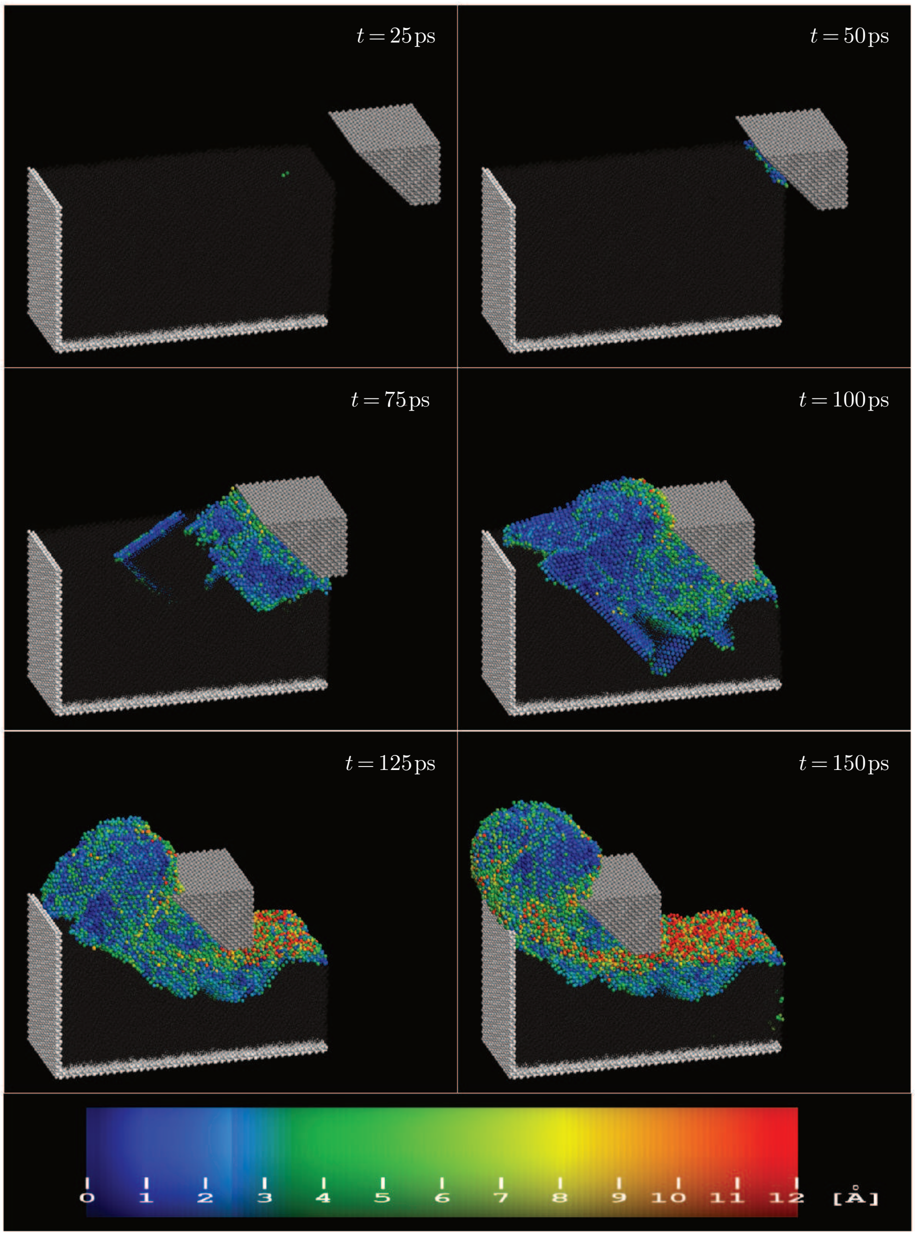

Figure 108. Distribution of the magnitudes of the slip vector at six instants, for (100)-oriented workmaterial, tool F and a machining depth of $5a$

they periodically appear and retreat. Similarly, the plots showing the number of atoms with a non-zero slip that make it possible to differentiate between those two situations – a decrease in the number of slipped atoms means a retreating "near-dislocation").

From the presented figures it can be seen that the machining depth has a significant effect on the way the machined workmaterial degrades. Let us consider the process of machining with tool B. For a depth of $1a$, there are practically no dislocations observed propagating in the workmaterial. All atoms of a non-zero slip vector either belong to the small chip pushed in front of the tool or form a shallow trace of the tool's passage. For the machining depth of $2a$, it may be observed how close the material is to yielding, for $t = 100\,\mathrm{ps}$ an appearing slip plane can be seen which is not eventually formed (*i.e.* the energetic barrier needed for the slip has not been overcome) and the final configuration resembles that which was achieved at the depth of $1a$, except for a small region directly under the tool front. At the machining depth of $5a$, the slip planes are well visible already for $t = 75\,\mathrm{ps}$, both under and in front of the tool. At subsequent machining stages it can be seen that the slip plane which appeared at the beginning and is located at the front of the workmaterial has not retreated, and the whole workmaterial is deformed up to a depth of several lattice constants below the base of the tool.

The slip vector analysis also offers insight into the effect of the tool shape on the slips taking place in the workmaterial. The most favorable shape seems to be shape D – even for a machining depth of $5a$ no propagating dislocations are observed, and all atoms with a non-zero slip vector are part of the chip or the trace after the passage of the tool (*cf.* Figures 100–102). For tool C, slips are observed to appear in front of the chip, with their number increasing with the machining depth, however they are shallow to the extent that the material is not found to be significantly degraded upon the completion of machining, and no slipped atoms are observed below the depth of five lattice constants (Figures 97–99).

For the remaining tool shapes, machining results in greater degradation of the workmaterial. For tool B, as has been mentioned before, permanent damage to the machined material occurs only with the machining depth of $5a$ (Figures 94–96.

For tool shapes A, E and F, dislocations occur, in great numbers, already for smaller machining depths, they are located deeper, and upon the completion of machining, the resultant workmaterial contains atoms with a non-zero slip vector which are found as deep as 45 Å below the machining depth (*cf.* Figure 93, last panel).

5.6. Chip formation

A preliminary view of the chip formation process is given in Figures 43–60, however, for a more detailed analysis, reference should be made to the graphs presenting the number of atoms forming the chip, depending on the shape of the tool used and the machining depth, shown in Figures 109–111.

The manner in which the atoms belonging to the chip are detected was described earlier in Section 3.8. An analysis of the abovementioned graphs leads

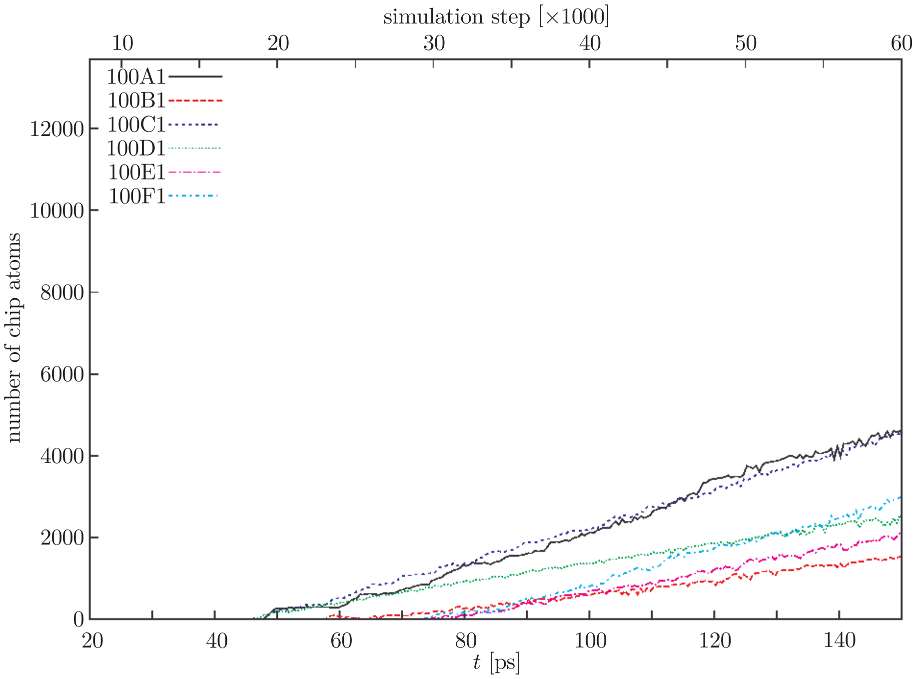

Figure 109. Chip size, as a function of time for varied tool shapes and a machining depth of $1a$

to an observation that the largest chip is formed when machining with tools A and C, then with tools D and F, and the smallest chip is formed when machining with tools B and E.

There are many reasons for the observed situation. The most trivial reason is understood by referring to Figure 6, from which it can be easily noticed that the adopted definition of the tool-workmaterial distance (where the distance is measured to the foremost tool atom) automatically causes faster contact of the workmaterial and tools A, C and D, as the mentioned foremost atom contacts the workmaterial (C) or the tool attacks the workmaterial with a vertical side (A, D).

For tools B, E and F, where the foremost atoms are at the upper part of the tool, reaching a tool-workmaterial distance of zero does not mean physical contact between the tool and the workmaterial, particularly for the smaller machining depths. A confirmation for this effect may be found in the graphs illustrating the force acting on the tool (Figure 61) – tool A contacts the material at $t = 36$ps, regardless of the machining depth, while tool E contacts the material only at $t = 54$ps to $t = 65$ps, depending on the machining depth. In view of this, during further analysis one should rather focus on the slopes of the graphs corresponding to specific tools, rather than on the values themselves.

In so doing, we may conclude that the tools may be divided into two groups – tools A, C and F, which lead to the formation of larger chips, and

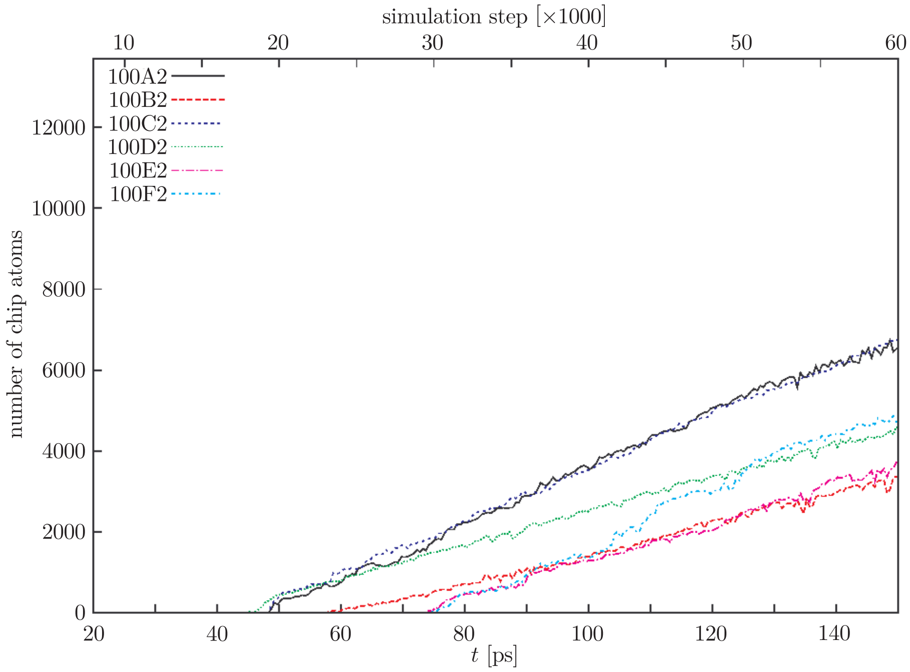

Figure 110. Chip size, as a function of time for varied tool shapes and a machining depth of $2a$

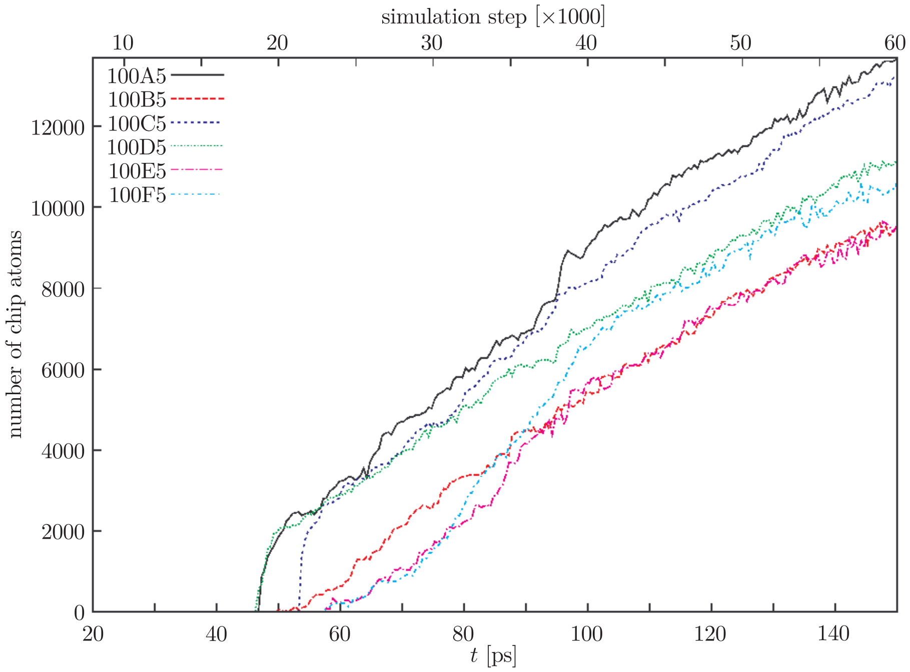

Figure 111. Chip size, as a function of time for varied tool shapes and a machining depth of $5a$

E, D, and B, which generate smaller chips. The reason for this becomes clear after viewing, frame by frame, the animations showing the machining process for each tool. The animations demonstrate that the lack of a horizontal base in tools E, D and B enables the transport of atoms under the tool to the back of the tool, thus allowing some of the atoms, which would otherwise contribute to the chip, to come to the machined material's surface behind the tool. In order to verify this, the number of atoms with a non-zero slip vector belonging to the machined area, immediately behind the tool and up to a depth of a (with respect to the tool) was determined. Upon the completion of machining, the number of "foreign" atoms located behind the tool for tools A, C and F was 139, 164 and 322, respectively and for tools E, D, B as many as 1765, 2219 and 2528 atoms were found, respectively.

Having in mind, the fact that the size of the chip upon the completion of machining was in the range of 2000 to 6000 atoms, the atoms that did move under the tool ideed consitute a significant loss of material which would otherwise become a part of the chip.

The values quoted in the preceding paragraph refer to machining the (100)-oriented workmaterial, however the described trends may be observed also for the other two workmaterial orientations. The only observed difference in the chip formation process consists in that fact that for orientation (100) the forming chip is slightly larger than for other orientations which, for the example of tool F, is illustrated in Figure 112.

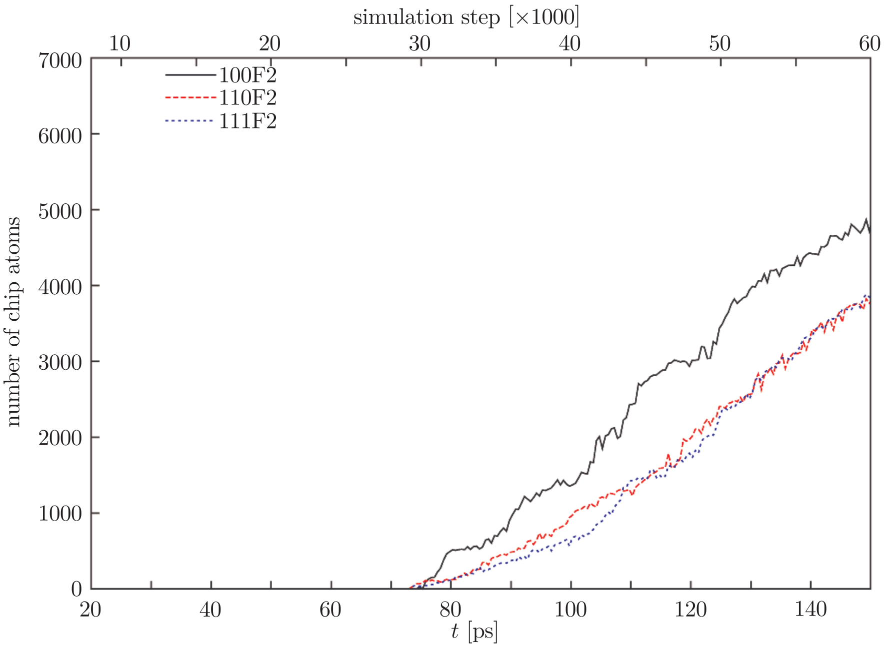

Figure 112. Numbers of chip atoms for orientation (100), (110) and (111), tool F, machining depth of $2a$

6. Summary and conclusions

The *nanoMD* package is a carefully parallelized and optimized suite of computer codes for the simulation, monitoring and visualization of the monocrystal machining process. The machined material may be freely oriented and fixed, the machining tool may be of any shape, and may be infinitely stiff or deformable. Algorithms for local averaging of physical quantities were developed and implemented.

The package developed in this work is optimized to the extent that on contemporary computers it permits running simulations of ultra-precision machining for monocrystals containing hundreds of thousands of particles with the machining speeds in the order of 1 m/s. It is a significant achievement, enabling researchers to perform nanomechanical simulations at the highest technical level.

A large number of nanomachining calculations for a copper monocrystal and several simulations for aluminum and lead monocrystals were conducted using the package.

The results of simulations for copper have been presented in this work since the results for other *fcc* metals are qualitatively similar, and the work described here was aimed at developing the simulation tools and analysis methods rather than at comparing the properties of specific metals in a detailed fashion.

Three sets of nanomachining simulations were performed for three Cu monocrystal surface orientations: (100), (110) and (111). Three machining depths with six nondeformable machining tools were used in each of those cases.

The 54 production simulations in question were performed for the initial temperature set for the system (300K) and for the a machining speed of 100 m/s. These parameters were selected as optimal on the basis of test simulations performed at varied speeds at varied temperatures.

The temperature distribution, hydrostatic pressure, xy-component of the stress tensor, the scalar field of absolute displacement, the field of the magnitude of slip vectors, the Falk-Langer coefficient and the von Mises invariant were monitored in the course of machining.

The local sampling and averaging of observables (*e.g.* temperature and pressure) constitues a novelty in the area of nanomachining simulations.

The developed simulation package makes it possible to realistically model the process of monocrystal machining and similar processes (nanocutting, nanoindentation, nanoscratching).

The developed tools for temporal monitoring of structure changes of the deformed material are sensitive enough for the differences in the physical parameters for the machining of crystals of various orientations and composed of different elements to be observed.

The last comment is a trivial requirement with respect to any physical test. However, it should be emphasized that previous nanomachining simulations reported in the literature were of a model nature, and the emphasis was laid on simulation algorithms alone. In general, the authors of the earlier works did not attempt to connect the results of their simulations to any real material (using only 2D systems, the Lennard-Jones or Morse potential, very small systems, unrealistically high machining speeds, *etc.*).

Several issues that appeared in the course of performing the simulations described in this work still need addressing, *e.g.* the problem of a physically sound way to fasten the machined crystal.

A stiff confinement of the crystal by artificially immobilizing the leftmost vertical surface of atoms was used in the simulations.

Such a geometry of the system was adopted so as to avoid the easy slipping of the material to the left, eliminating an additional effect superimposed on the machining and chip formation. However, in the final moments of the simulation, when the tool comes close to the left wall, the artifacts introduced by this approach become very clear. A makeshift solution to this problem would consist in performing simulations for a considerably longer monocrystal without immobilizing the left vertical surface. A more elegant solution would be to combine the micro and macro scales in one calculation model combining molecular dynamics with continuum mechanics.

Another problem is the fact that the the thermal electronic conduction, significant in the case of metals, was disregarded in the model used. As a result, the temperature field evolves exclusively by means of lattice conductivity which is at least controversial in the case of metal. This problem requires further careful studies.

References

[1] Hoover W G, De Groot A J, Hoover C G, Stowers I F, Kawai T, Holian B L, Boku T, Ihara S and Belak J 1990 *Phys. Rev.* **A 42** 5844

[2] Maekawa K and Itoh A 1995 *Wear* **188** 15

[3] Chandrasekaran N, Noori Khajavi A, Raff L M and Komanduri R 1998 *Phil. Mag.* **B 77** 7

[4] Sutton A P and Chen J 1990 *Phys. Rev. Lett.* **61** 139

[5] Rafii-Tabar H 2000 *Phys. Rep.* **325** 239

[6] *http://www.povray.org*

[7] *http:/www.imagemagick.org*

[8] Hull D 1982 *Dislocations*, PWN, Warsaw (in Polish)

[9] Nabarro F R N 1987 *Theory of Crystal Dislocations*, Dover, New York

[10] Zimmerman J A, Kelchner C L, Klein P A, Hamilton J C and Foiles S M 2001 *Phys. Rev. Lett.* **87** 165507

[11] Rodriguez de la Fuente O, Zimmerman J A, Gonzalez M A, de la Figuera J, Hamilton J C, Woei Wu Pai and Rojo J M 2002 *Phys. Rev. Lett.* **88** 36101

[12] Falk M L and Langer J S 1998 *Phys. Rev.* **E 57** 7192