

MOLOCH computer code for molecular-dynamics simulation of processes in condensed matter

F.A. Sapozhnikov^a, V.V. Dremov, G.V. Ionov, I.V. Derbenev, and N.E. Chizhkova

Russian Federal Nuclear Centre – Zababakhin Institute of Applied Physics (RFNC-VNIITF), Snezhinsk, Russia

Abstract. Theoretical and experimental investigation into properties of condensed matter is one of the mainstreams in RFNC-VNIITF scientific activity. The method of molecular dynamics (MD) is an innovative method of theoretical materials science. Modern supercomputers allow the direct simulation of collective effects in multibillion atom sample, making it possible to model physical processes on the atomistic level, including material response to dynamic load, radiation damage, influence of defects and alloying additions upon material mechanical properties, or aging of actinides. During past ten years, the computer code MOLOCH has been developed at RFNC-VNIITF. It is a parallel code suitable for massive parallel computing. Modern programming techniques were used to make the code almost 100% efficient. Practically all instruments required for modelling were implemented in the code: a potential builder for different materials, simulation of physical processes in arbitrary 3D geometry, and calculated data processing. A set of tests was developed to analyse algorithms efficiency. It can be used to compare codes with different MD implementation between each other.

1 Short-range molecular dynamics challenges

There are several challenges to deal with while implementing short-range MD, including neighbours search within a predefined radius and decomposition algorithms to parallelize between computer nodes. The better they are implemented, the more flexible the code is.

1.1 Neighbours search within predefined radius

For the fast calculation of forces on atoms, a cutoff radius is used in MD, i.e., an atom is assumed to interact only with atoms within the cutoff radius.

One of the approaches to MD computing is the creation of neighbours lists for each atom. For this end, atoms within the cutoff radius need to be found as fast as possible.

The Link Cell Method (LCM), which is easy-to-implement and quick, is commonly used for this purpose. MOLOCH implements an original algorithm called the Ordered Space Method (OSM).

The LCM divides space (Fig. 1(b)) into cubes with a constant rib length that is equal to or smaller than the cutoff radius, and links all atoms into these cubes (Fig. 1(a)). Atom's neighbours are located in 27 nearest cubes. The LCM is simple in coding and quick in use, but its main disadvantage is memory failure when the sample is expanding (Fig. 1(d)).

This LCM disadvantage originates from its incontestable advantage – the constant rib length. In our method, the length is not constant, but this makes atom neighbours finding much more difficult.

^a e-mail: f.a.sapozhnikov@vniitf.ru

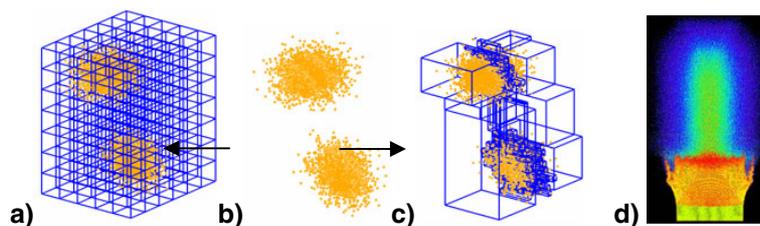


Fig. 1. a) Link Cell method; b) initial set of atoms; c) Ordered Space method; d) detonation of TATB-like explosive cylinder.

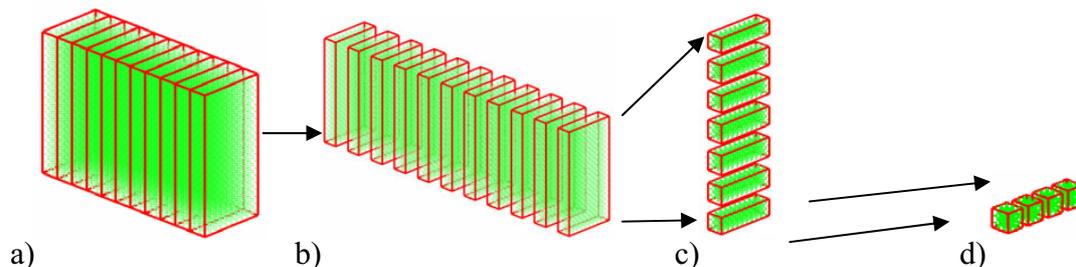


Fig. 2. Hierarchical spatial decomposition: a) initial sample; b) slices in X; c) bars in Y; d) zones in Z.

The OSM divides space into boxes with approximately equal atomic concentrations (Fig. 1(c)) to get rid of the dependence on sample size. It is slower than the LCM by 10% in LJ like potentials (a 10 Å cutoff radius and 140 neighbours) and by 20% in MEAM like potentials (a 5 Å cutoff and 14 neighbours), but works without memory failure.

1.2 Parallelization method

Parallel MD techniques use force, atom and spatial decompositions. As shown in [1], the spatial decomposition is most effective when the numbers of atoms and processors are large.

MOLOCH uses a hierarchical spatial decomposition (Fig. 2) which allows processor responsibility zones to be excellently determined with respect to local atomic concentrations. It also allows effective dynamic balancing with processor responsibility zones being locally changed simultaneously with the exchange of atoms either at each time step, or when imbalance threshold is exceeded.

The algorithm of zoning includes the following steps.

1. Cut a sample along the X axis into *slices* with approximately equal number of atoms (Fig. 2(b));
2. Cut each slice, independently of the others, along the Y axis into *bars* with approximately equal number of atoms (Fig. 2(c));
3. Cut each bar, independently of the others, along the Z axis into *zones* with approximately equal number of atoms (Fig. 2(d));
4. Each processor is assigned a zone of space with atoms (or without them if balancing simply does not have time to change the zones at a high rate of atomic flow).

2 MOLOCH code

An MD calculation implies three phases: sample generation, MD simulation and results analysis. Reliability of the entire calculation is questionable if at least one of the phases gives non-physical results. MOLOCH implements effective algorithms that ensure reliable results.

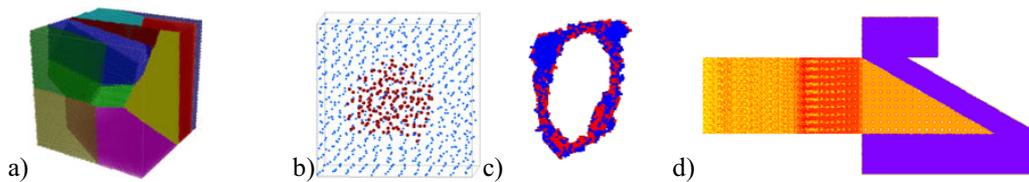


Fig. 3. Sample generation: a) polycrystalline; b) and c) defects; d) complicated geometry.

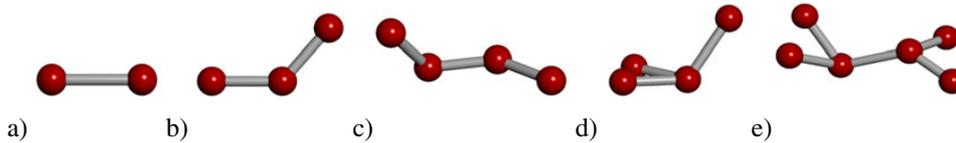


Fig. 4. Molecular Mechanics Force Field elements: a) covalent bond; b) valence angle; c) torsion angle; d) out of plane angle; e) out of plane–out of plane angle.

2.1 Sample generation

For getting physically true samples, MOLOCH implements a generator of single- and poly-crystals by the Voronoy cell method (Fig. 3(a)). It is possible to add dopes and defects. There is a special library of pre-calculated defects, for example, a helium bubble in plutonium lattice (Fig. 3(b)) or a loop dislocation (Fig. 3(c)). A superposition of geometrical constraints can be used: half plane, sphere, cylinder, box (Fig. 3(d)).

2.2 MD simulation

2.2.1 Potentials

MOLOCH implements such well known potentials as Lenard-Jones, EXP-6, EAM, MEAM, Brenner REBO 2002, and Molecular Mechanics Force Field (Fig. 4).

We developed and implemented in the code the following potentials: GEAM (Generalized Embedded Atom Model) for materials with complicated electronic structures and Modified REBO ABCD (a long-range modification of REBO) for a model explosive containing atoms of 4 types (parameterization for TATB) [2].

Parameters for the existing and new potentials are adjusted with a parallel genetic algorithm which minimizes error in the description of a given set of experimental or calculated data by a particular potential.

MOLOCH is written in C++ which allows new potentials to be easily implemented at minimal changes in the code.

2.2.2 Simulation

The NVE, NVT, and NPT ensembles can be modeled with the simultaneous use of several potentials and a variable time step. Complicated geometry constraints are implemented (e.g., a moving collapsing spherical piston).

2.2.3 Automations

MOLOCH automatically calculates phonon dispersion curves and DOS at desired temperature (Fig. 5(a)), elastic moduli (Fig. 5(b)), equation of state, Hugoniot (using Hugoniotstat method), isotherms, cold curve.

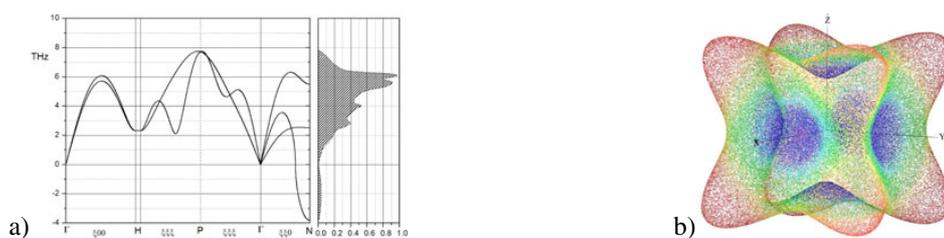


Fig. 5. a) Phonon dispersion curves and DOS, $T = 0\text{K}$; b) angular dependence of Young's modulus for FCC Cu.

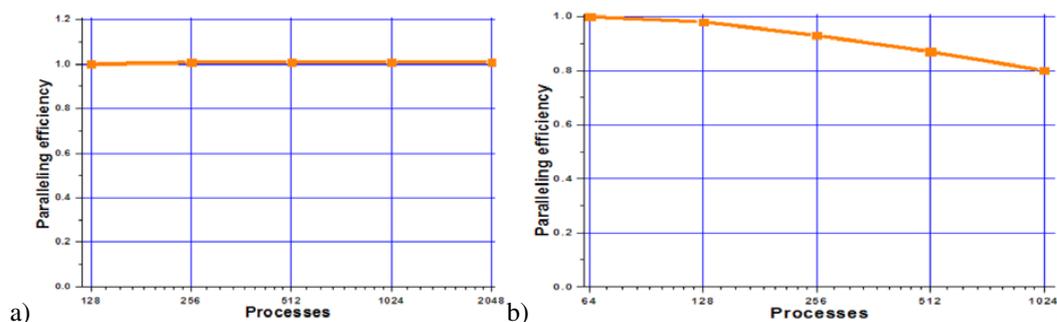


Fig. 6. a) Weak scaling test; b) strong scaling test.

2.3 Results analysis

MOLOCH creates 3D meshes for macroscopic characteristics (temperature, pressure and density), detects bonds and calculates chemical compositions (e.g., during carbon nanocluster evaporation modeling).

2.3.1 Automatic structure recognition

We developed a method called Adaptive Template Analysis (ATA) to recognize crystal structures and defects [3]. ATA recognizes BCC, FCC, HCP, Diamond and Graphen lattices (other ones can be easily added), and reconstructs vacancies, identifies interstitials and split interstitials, dislocations and stacking faults. ATA doesn't have any adjustable parameters, has low sensitivity to temperature disordering and local density fluctuations.

2.4 Paralleling efficiency

Dynamic balancing (usually taken when imbalance is above 5%) allows attaining high parallel efficiency even in non-equilibrium MD.

Figure 6(a) shows results of a weak scaling test for 64 processors single-crystal Cu $240 \times 25 \times 25$ u.c. (600'000 atoms) in full periodic $1/3$ to $2/3$ collision at $V = 1000\text{m/s}$. Multiplications of a sample were made in Y and Z dimensions ($240 \times 25 \times 50$, $240 \times 50 \times 50$, $240 \times 50 \times 100$, $240 \times 100 \times 100$).

Figure 6(b) shows results of a strong scaling test for single-crystal Cu $120 \times 70 \times 70$ u.c. (2'352'000 atoms) in full periodic $1/3$ to $2/3$ collision at $V = 1000\text{m/s}$.

Paralleling efficiency is high in these non-equilibrium MD tests (imbalance was less than 5%).

3 Test set for MD codes algorithms

There are many tests that can be used to verify if MD results are physically correct but we also need tests that could help measure the efficiency of each particular algorithm and the entire MD implementation. The test setups provided below are devised to test such important algorithms as dynamic balancing, multi-potential modelling and neighbours list building.

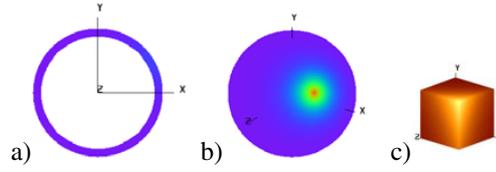


Fig. 7. a) A test sample cut; b) spherical layer coloured with respect to kinetic energies of atoms; c) control sample – cube with almost the same number of atoms as in spherical layer.

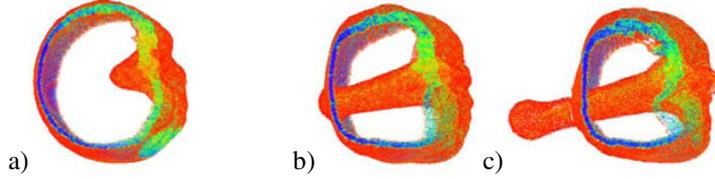


Fig. 8. Cuts of the test sample with $R_{big}=722 \text{ \AA}$ at different times: a) 15 ps; b) 30 ps; c) 45 ps.

MOLOCH uses sum of neighbours list building time and energy and force calculation time to balance processors loading. Let's calculate *imbalance* (D) using following formula:

$$D = \frac{t_{bal}^{max} - t_{bal}^{min}}{t_{bal}^{average}} \cdot 100\%, \quad \text{here } t_{bal} = t_{NeighboursList} + t_{Energy,Forces}$$

3.1 “Death Star” test

The test aim is to measure the efficiency of dynamic balancing (if implemented) which must be quick (to balance on every MD step if high imbalance) and independent of the axial direction of atomic flows. The problem implements a fully 3D deformation of a thin spherical layer (Fig. 8).

Physical analogue: Fast Ignition.

Problems: Considerable 3D deformations produce intensive atomic flows.

Test sample: A zero-centred spherical layer is cut from single crystal FCC copper (EAM [4]) with outer and inner radii R_{big} and $R_{small} = 0.95R_{big}$, respectively (Fig. 7(a)). Velocities of all atoms are pointed to the centre (Fig. 7(b)) and defined as:

$$v = V_{max} e^{-\arccos(\cos \theta \cos \theta_0 + \sin \theta \sin \theta_0 \cos(\phi - \phi_0))}, \quad \text{here } V_{max} = 3000 \text{ m/s is maximal velocity.} \quad (1)$$

The spherical coordinates of a specified direction of loading are $\phi_0 = \pi/6$ and $\theta_0 = \pi/3$.

The spherical coordinates of an atom are $\phi = \text{atan2}(y/R_{big}, x/R_{big})$ and $\theta = \text{acos}(z/R_{big})$ where x, y, z are its Cartesian coordinates.

Control sample: A single crystal FCC copper cube with $L = 0.995 \cdot \sqrt[3]{(R_{big}^3 - R_{small}^3)} \cdot 4\pi/3$ (Fig. 7(c)); initial temperature 300 K.

Simulation: Free boundary conditions; timestep 1.5 fs, 30'000 timesteps.

Outcome: The ratio μ of test sample calculation time to the control sample calculation time is to be close to 1. If it is much greater, then the balancing efficiency is low.

MOLOCH outcome: 1) $\mu = 0.95$ for $R_{big} = 361 \text{ \AA}$ (100 u.c., 2.39 millions of atoms) on 128 processors. Average imbalance is 4.6% for the test sample and 4.3% for the control one.

2) $\mu = 1.1$ for $R_{big} = 722 \text{ \AA}$ (200 u.c., 19 million atoms) on 1024 processors. Average imbalance is 15% for the test sample and 7% for the control sample.

3.2 “Cocktail B-52” test

The test shows the efficiency of MD simulation with several potentials with strongly different cutoff radii, specifically MEAM (3.5 Å) and LJ (10 Å).

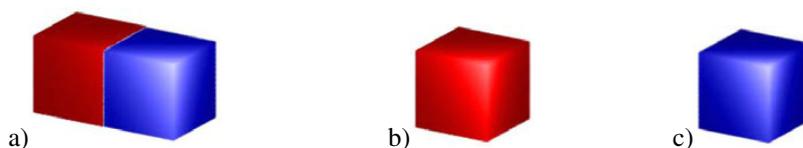


Fig. 9. a) Test sample; b) control HCP Be cube; c) control FCC Ar cube.

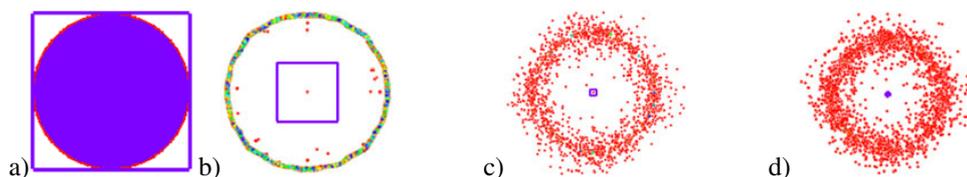


Fig. 10. Sphere longitudinal cut (different scales) at different times: a) 0 ps; b) 15 ps; c) 150 ps, d) 300 ps. Square – initial system size.

Physical analogue: Modelling of interface instability.

Problem: Time of MEAM triple interactions is extremely sensitive to the cutoff value, so it may slow down several times on looking for interaction within ‘useless’ for this potential large cutoff.

Test sample: A bar of HCP Be (MEAM [5]) and FCC Ar (LJ: $\sigma = 3.4 \cdot 10^{-10}$, $\varepsilon = 1.7 \cdot 10^{-21}$) cubes (Fig. 9(a)) with initial temperature 300K; Be-Ar interaction is neglected.

Control samples: Independent calculations of HCP Be and FCC Ar cubes (Fig. 9(b) and 9(c)).

Simulation: Free boundary conditions, timestep 1.5fs, 1000 timesteps.

Outcome: The ratio μ of test sample calculation time to the sum of control samples calculation times is to be close to 1. If it is much greater, then the efficiency of multi-potential modelling is low.

MOLOCH outcome: 1) $\mu = 1.8$ for $L = 200 \text{ \AA}$ (988’130 Be atoms, 376’785 Ar atoms) on 128 processors. Average imbalance is 5% for the test sample, and 6.8% and 10% for Be and Ar cubes.

2) $\mu = 1.7$ for $L = 400 \text{ \AA}$ (7’901’960 Be atoms, 2’997’722 Ar atoms) on 1024 processors. Average imbalance is 7.8% for the test sample and 9% and 16% for Be and Ar cubes, respectively.

3.3 “Supernova” test

The test demonstrates feasibility of MD simulation for expanding samples.

Physical analogue: Supernova explosion.

Problems: During neighbours list building some Link Cell like algorithms with constant rib length (usually equal or less than cutoff radius) will produce memory failure on big porous samples.

Test sample: A FCC Cu (EAM [4]) sphere with atoms moving outward the centre at 3000m/s.

Simulation: Free boundary conditions, timestep 1.5fs, 200’000 timesteps.

Outcome: Completed calculation with satisfying energy conservation.

MOLOCH outcome: Test was successfully passed for $R = 200 \text{ \AA}$ (2.8e6 atoms) on 128 processors. Average imbalance is 10%. Volume increased 224’140 times (Fig. 10). Energy was conserved.

References

1. S. Plimpton, J. Comp. Phys., **117**, 1–19 (1995)
2. F. A. Sapozhnikov, V. V. Dremov, I. V. Derbenev, A. V. Karavaev, and L. Soulard, AIP Conference Proceedings, Volume **955**, pp. 463-466
3. F.A. Sapozhnikov, G.V. Ionov and V.V. Dremov, Russian Journal of Physical Chemistry B, 2008, Vol. **2**, No. 2, pp. 238–245
4. Y. Mishin, D. Farkas, M. J. Mehl, and D. A. Papaconstantopoulos, Phys. Rev. B **59**, 3393, 1999
5. V. Dremov, A. Karavaev, F. Sapozhnikov, L. Soulard et al., DYMAT 2009 (2009) 1277–1283