

MiniMol – a minimal molecular dynamics/statics program

Ellad B. Tadmor* and Ronald E. Miller†

Version 2.01, Sept 2011

1 Overview

MiniMol¹ is a minimal molecular dynamics (MD) and molecular statics (MS) program provided with the book *Modeling Materials: Continuum, Atomistic and Multiscale Techniques* by Ellad B. Tadmor and Ronald E. Miller, Cambridge University Press, 2011.

MiniMol is available both as a nanoHUB tool (see <http://nanoHUB.org>) which features a convenient graphical user interface (GUI) and as a downloadable source code in both Fortran90 and C.² The source code is stripped of the GUI related commands and runs with a text input and output. These relatively simple, comments codes can be used by those seeking a better understanding of how a molecular simulation code is written. This document specifically describes the nanoHUB MiniMol tool and its GUI.

MiniMol has the following capabilities/features:

1. MS simulations, which involve energy minimization of static configurations using a conjugate gradient (CG) minimizer.
2. Constant energy MD.
3. Constant temperature MD with one of the following:
 - velocity rescaling
 - Nosé-Hoover thermostat
 - Langevin thermostat
4. Periodic boundary conditions (PBCs) for an orthogonal periodic box using the minimum image convention.³
5. A crystal generating tool that can be used to produce initial atomic configurations for the simulation of single crystals.

*Dept. of Aerospace Engineering and Mechanics, University of Minnesota, Minneapolis, MN, USA.
E-mail: tadmor@aem.umn.edu

†Dept. of Mechanical and Aerospace Engineering, Carleton University, Ottawa, Canada.
E-mail: ron_miller@carleton.ca

¹MiniMol is based on a program called “md3.f90”, originally written by Furio Ercolessi as an MD tutorial, which the authors have adapted and extended with his gracious permission.

²The MiniMol source code will be available for download on the book website (<http://modelingmaterials.org>) after the publication of the book in October 2011.

³The minimum image convention places limits on the size of the periodic box, see Appendix A.

6. MiniMol is “KIM-compliant” which means it will work with any interatomic model conforming to the Knowledgebase of Interatomic Models (KIM) application programming interface (API), for details see: <http://openKIM.org>

7. MiniMol uses the following units:

length	Å
time	ps
energy	eV
temperature	K
mass	$\text{eV} \cdot \text{ps}^2 / \text{Å}^2$

2 Running MiniMol

To run the MiniMol tool under nanoHUB, do the following:

1. Log in to nanoHUB (<http://nanoHUB.org>).
2. Click on the “My HUB” menu entry at the top of the screen.
3. In the “My Tools” pane, click on “All Tools”.
4. Scroll down to “MiniMol” and click on it.
5. Click on the “Launch Tool” button to the right of the tool name at the top of the screen.

If all is successful, you will see a window like that shown in Fig. 1.

The next step is to define the input fields necessary to perform the simulation.

3 Specifying MiniMol Input

3.1 Providing MiniMol with an initial atomic configuration

In either MD or MS mode, MiniMol requires an initial atomic configuration, consisting of the coordinates of each atom and the dimensions of the orthogonal periodic simulation box. For constant energy MD simulations, the initial velocities of all the atoms are also required.

The initial configuration can be provided in one of two ways:

Input Configuration File: By choosing “File” as the source for the atomic coordinates, you will find a pull-down menu from which you can either upload a file from your computer in the appropriate format (see Appendix A), or choose one of the example configuration files provided with MiniMol. (Make sure the atomic species matches the interatomic model selected above. Note that MiniMol is currently only set up to work with single species configurations.)

Crystal Generator: By choosing “Crystal” as the source for the atomic coordinates, you will find a number of crystal generator options. These are explained in Appendix B. Note that this option cannot be used for constant energy MD simulations because it does not provide velocities. However, it can be used for constant temperature MD simulations.

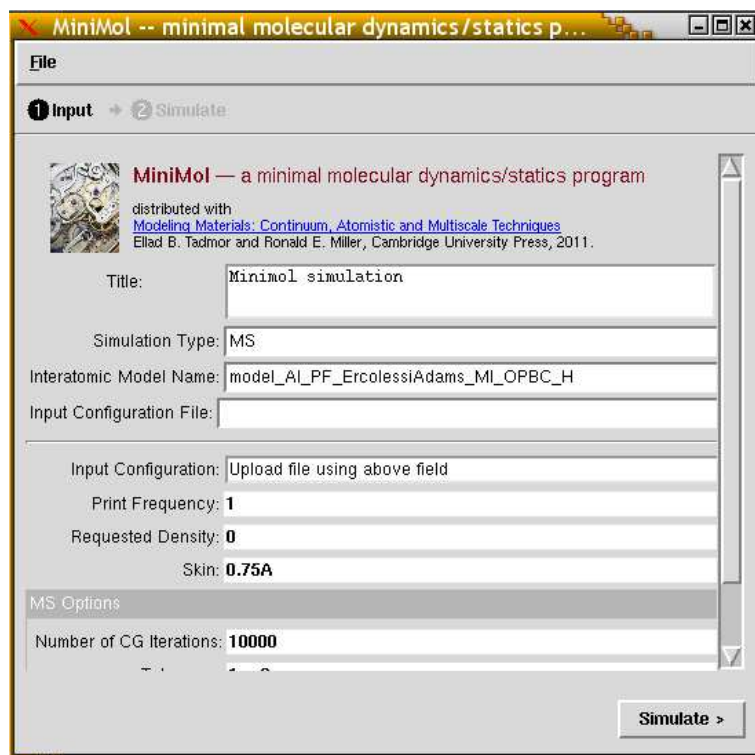


Figure 1: Screen shot of the nanohub MiniMol tool.

Some helpful tricks for generating input configurations:

Defected crystals: The crystal generator can only be used to create perfect infinite crystals. However, by running in MS mode with “Number of CG iterations” to 1 (see section 3.3), you can pass your initial configuration through MiniMol unchanged and then download the final atomic configuration. This file can then be modified as desired (either off-line or directly in the MiniMol input screen) and used as a new input file. For example, you can create free surfaces by changing the periodic lengths, vacancies by removing atoms, interstitials by modifying atomic coordinates, etc.)

Initializing velocities: In constant temperature MD simulations, MiniMol automatically initializes the velocities, but constant energy simulations require velocities to be provided. You can run a single time step of a constant temperature MD simulation to produce an output configuration file with velocities that can then be downloaded and used as input for a constant energy simulation. More commonly, a constant temperature simulation is run for some period of time to achieve equilibrium before the constant energy simulation is started.

3.2 Input Common to both MS and MD

In both MS and MD modes, the following fields must be defined:

Title: An arbitrary string which is the title given to the calculation.

Simulation Type: Select either “MS” or “MD” from the pull-down menu that opens when you click on the field.

Interatomic Model Name: Select from the list of available KIM-compliant interatomic models in the pull-down menu.

Print Frequency: This parameter controls that amount of output generated. A value of “1” means that output will be generated at every MS iteration or MD time step. A value of “2” means output will be generated only every second iteration/time step, and so on. (This provides a convenient way to limit the size of the output file.)

Requested Density: If you set “Change Atomic Density” to “Yes”, then you need to enter a value here in units of atoms per \AA^3 . The periodic box volume and atomic coordinates are rescaled at the beginning of the simulation to match your requested density. This can be handy for using the same input sample at different temperatures.

Skin: A parameter setting the amount of padding used when generating Verlet neighbor lists. “Skin” changes the CPU time but has no effect on trajectories. If the value is too small, neighbor lists will have to be repeatedly regenerated. If the value is too large, a large number of neighbors will be unnecessarily examined at each force calculation. Typically, a value of about 0.75\AA is a good choice. Note that setting “Skin” to an overly large value can cause the program stop because the maximum allowed size for the neighbor lists is exceeded.

3.3 Additional Input in MS mode

MS simulations require the following additional input fields:

Number of CG iterations: Maximum number of CG iterations allowed. The program will terminate when this limit is reached even if the simulation is not converged (see tolerance below).

Tolerance: Convergence tolerance for the CG routine. The minimization will terminate when during a line minimization step, the changes to the components of all atom positions are less than “Tolerance” (given in units of \AA).

Maximum Step Size: Limit on the allowed displacements of atoms during a line minimization step. No component of an atom’s position may change by more than this amount. (A value of 0.1\AA is typical.) This prevents overly large steps that could result in a system escaping from the basin of attraction of its local minimum.

3.4 Additional Input in MD mode

MD simulations require the following additional input fields:

Number of MD Time Steps: The number of MD time steps that will be performed.

Time Step: Size of a time step in ps. (Typical value: 0.001 ps). Larger values could lead to numerical instability.

MD Ensemble: Choose “NVE” to run a constant energy simulation, “NVT” to run a thermostat-controlled constant temperature simulation.

Temperature Requested If you have chosen the “NVT” (constant temperature) ensemble, then a zero or positive number here corresponds to the set temperature T (in units of K).

The next lines are only required if the constant temperature (NVT) ensemble is used.

Thermostat Type: The thermostat used to control the temperature. There are three possibilities: “VelocityRescaling”, “NoseHoover”, and “Langevin”. Note that velocity rescaling does not produce a correct canonical ensemble and is generally not recommended.

Thermostat Control Parameter: The control parameter for the thermostat. Setting this to a negative value uses the built-in defaults. This line is ignored for velocity rescaling.

Random Number Seed: An integer seed for the random number generator (used only by the Langevin thermostat and not required otherwise).

4 Executing the Simulation

Once all input has been specified, the simulation is executed by clicking on the “Simulate” button on the bottom of the MiniMol GUI window. The window will go gray for awhile as the simulation is performed and will then switch to the “Simulate” pane with the results when the simulation is completed. Different results are available depending on the simulation mode.

4.1 Results in MS Mode

The desired result can be selected from the “Result” pull-down menu at the top of the window:

MiniMol Output: This is the output sent to the screen by the MiniMol program as it runs. The input to the simulation is echoed at the top followed by a table showing the progress of the CG minimization. The columns in this table are:

“icyc” The iteration count in the current cycle. Occasionally when the minimization appears to be stalling, the CG minimizer will initiate a new cycle by restarting from the current gradient.

“itn” The CG iteration counter. The minimization will continue until convergence is achieved or the maximum number of iterations is reached (see Section 3.3).

“en” The total energy. This should be continually decreasing. Note that due to the limited precision shown, no change will be observed in the number from a certain point on.

“rn” The norm of the gradient of the energy. This should be close to zero for a converged simulation.

“imax” The number of the degree of freedom⁴ that moved the most during the CG iteration.

“smax” The magnitude of the displacement of degree of freedom “imax” (see above). The CG minimization converges when “smax” is less than the specified tolerance (see Section 3.3).

After the table, general results are printed including the total potential energy and average potential energy per atom, the final out-of-balance form norm (same as “rn” mentioned above), and the pressure computed using the virial expression.

Plot of total energy versus iteration count: This shows a graph of the total energy during the minimization process.

List of final atomic coordinates: The atomic coordinates after the relaxation, in a text format suitable to download.

View of final atomic positions: This shows a visual representation of the final atomic coordinates that can be rotated and queried using a simple molecular viewing tool.

4.2 Results in MD Mode

The desired result can be selected from the “Result” pull-down menu at the top of the window:

MiniMol Output: This is the output sent to the screen by the MiniMol program as it runs. The input to the simulation is echoed at the top followed by a table showing the progress of the MD time evolution. The columns in this table are:

“Time” The physical time simulated.

“Temperature” The temperature of the system. In a temperature-controlled simulation, this value should fluctuate about the requested temperature (see Section 3.4).

“Kinetic” The total kinetic energy of the system (i.e. sum of the kinetic energies of the atoms).

“Potential” The total potential energy of the system.

“Total Energy” The total energy of the system (= kinetic energy + potential energy).

“Pressure” The pressure computed using the virial expression.

At the end of the table mean, values are printed (i.e. averages over all time steps).

Plot of total energy versus MD time: This shows a graph of the total energy during the time evolution.

Plot of the temperature versus MD time: This shows a graph of the system temperature during the time evolution.

⁴In three-dimensional space, the degrees of freedom are numbered from 1 to $3N$, where N is the number of atoms. Degree of freedom 1 is the x coordinate of atom 1 and degree of freedom $3N$ is the z coordinate of atom N .

Lists of final and mean atomic coordinates: The atomic coordinates are provided in a text format suitable to download. The user can choose either the final coordinates at the end of the simulation or the mean positions averaged over all time steps.

View of final and mean atomic positions: This shows a visual representation of either the final or mean atomic coordinates that can be rotated and queried using a simple molecular viewing tool.

A Configuration file format

The file containing the atom species and coordinates is written in a format that is compatible with the `.xyz` format used by molecular viewers such as Jmol.⁵ In short, MiniMol configuration files can be read directly by `.xyz` compatible molecular viewing software, but not all `.xyz` files can be used by MiniMol. In order to be compatible with MiniMol, the file must have the following format:

- The first line:

“N”

“N” The number of atoms.

- The second line:

“% L BoxSize_x BoxSize_y BoxSize_z [ViewShift_x ViewShift_y ViewShift_z]”

“%” Must be in column 1.

“L” Must be in column 3. L is either T or F and must be in column 3. It tells the program to expect velocities and accelerations of the atoms in the file (if set to T). Otherwise, the file only contains positions.

“BoxSize” The x , y and z lengths of the orthogonal periodic simulation box. Since MiniMol uses the minimum image convention when applying PBCs, the simulation box length along each direction must be at least twice the cutoff radius of the interatomic model.

For non-periodic boundary conditions in a given direction, increase the periodic length such that it is at least greater than the maximum distance between atoms along that direction plus the cutoff radius of the potential.

“ViewShift” This is optional, and will be treated as zero if missing. Otherwise, these are the x , y and z components of a rigid translation that will be applied to all of the atoms. It has no effect on the calculation results, but is sometimes helpful for viewing the output, since it can be used to move the atoms so that none are lying near the edges of the simulation box. This avoids some atoms “spilling” to the other side of the box in the output.

- The next N lines are the species and positions (three coordinates per line):

“spec x-coor y-coor z-coor”

⁵See http://en.wikipedia.org/wiki/XYZ_file_format for more on the file format and jmol.sourceforge.net for more about Jmol.

where `spec` is an element name (e.g. “Al”, “Ar”, etc.). Note that MiniMol is currently limited to atomic configurations containing a single atomic species. Atomic mass is determined from a look-up table inside the program based on standard values from the periodic table.

- Following the coordinate (optionally) are velocities on the next `N` lines and accelerations on the next `N` lines. These can be included to allow an MD simulation to restart from where a previous simulation left off. Each line containing a velocity or acceleration is prefaced with a “%” so that these lines are ignored by molecular viewer software.

B Crystal Generator Options

You can build a simulation box using a few simple input lines and options that will appear when “Crystal” mode is selected. The crystal generator options that will appear are:

Lattice Constant: For cubic crystals, it is helpful to think of this value as the cube edge length. More generally, though, it is just a scaling parameter that magnifies all of the Bravais lattice vectors described below. Units are Angstroms.

The Bravais Cell: This input consists of three vectors (the Bravais lattice vectors \mathbf{a} , \mathbf{b} and \mathbf{c}), an integer specifying the number of basis atoms, and then a line for each basis atom. These lines each consist of an atomic species (using standard chemical symbols) and the fractional coordinates of the atom within the Bravais cell. Thus, an aluminum atom may be specified as:

Al 0.0 0.5 0.25.

This means that the Al atom is located at the position $0.0\mathbf{a} + 0.5\mathbf{b} + 0.25\mathbf{c}$.

Note that the Bravais vectors \mathbf{a} , \mathbf{b} and \mathbf{c} are dimensionless. They will be scaled by the lattice constant, which will give them units of Angstroms.

X and Y: These are two vectors specifying crystal directions that will be oriented along the global X and Y axes in the simulation cell. X and Y must be orthogonal to each other and will be used to determine the third orthogonal axis, Z . These will be used to compute a rotation matrix that will be applied before generating the crystal.

NX, NY, NZ: Once the Bravais cell is rotated, the code determines the minimum orthogonal unit cell with sides parallel to the X , Y and Z directions. The values NX , NY and NZ are the number of times this cell is repeated in each direction to generate the desired crystal.