Tutorial 2: Molecular Dynamics
Lab 2. LAMMPS (Molecules)

The authors:

Prof. Dmitry Aksenov

PhD Arseniy Burov

November, 2023

Skoltech

Tutorial 2 agenda

1. **Essense of Molecular Dynamics**
   a. Force-fields
   b. Basic principles
   c. MD potentials
2. **Lab 2. LAMMPS**
   a. How to set-up your lab
   b. Molecules
   c. Basic commands and functions
   d. Solids

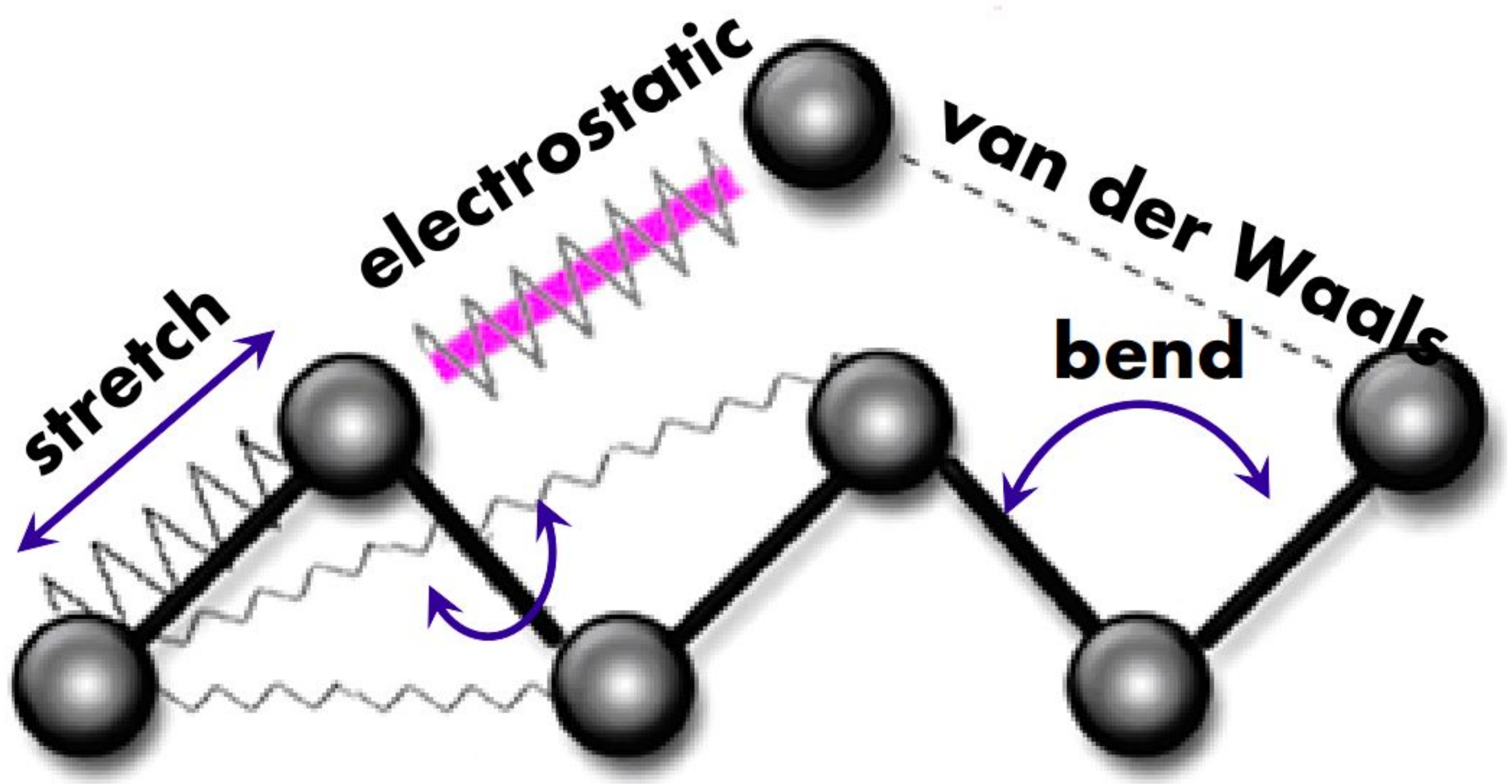
$\left.\begin{matrix} c \\ d \end{matrix}\right\}$ **Separate class**

Molecular mechanics and force-fields

Force Field is a computational method that is used to estimate the forces between atoms within molecules, between molecules, and between atoms in solids.

Force-field energy:

$$E_{FF} = E_{str} + E_{\text{bend}} + E_{\text{tors}} + E_{vdw} + E_{\text{el}} + E_{\text{cross}}$$



$$E_{FF} = E(\text{stretch}) + E(\text{bending}) + E(\text{torsion}) + E(\text{vanderWaals}) + E(\text{electrostatic}) + E(\text{cross}-\text{terms})$$

Molecular mechanics and force-fields

Total energy:

$$E_{\text{total}} = E_{\text{bonded}} + E_{\text{nonbonded}}$$

Bonded (short-range) contribution:

$$E_{\text{bond}} = E_{\text{str}} + E_{\text{angle}} + E_{\text{dihedral}}$$

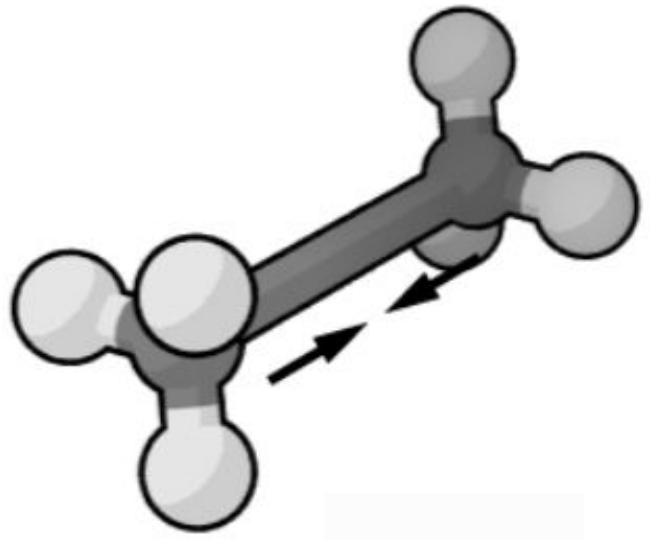
Nonbonded (long-range) contribution:

$$E_{\text{nonbonded}} = E_{\text{electrostatic}} + E_{\text{van der Waals}}$$

Skoltech

Molecular mechanics and force-fields

Bond stretching



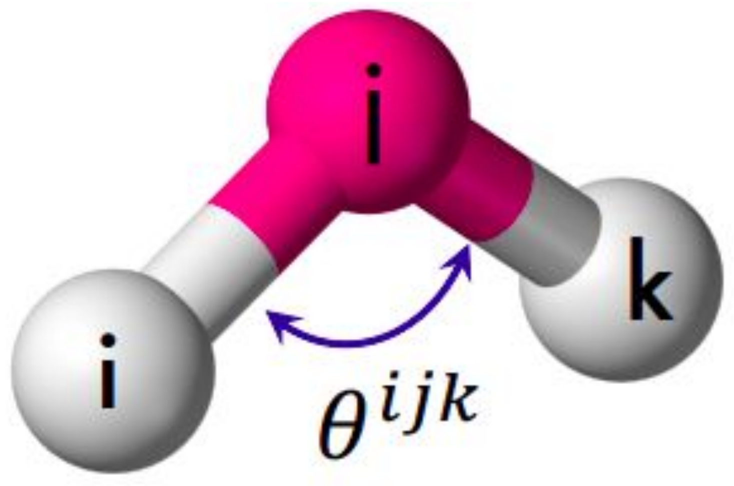
$$E_{\mathrm{str}} = \frac{k_{ij}}{2}\left(l_{ij} - l_{0,ij}\right)^2$$

Electrostatic Interactions

$$E_{\mathrm{Coulomb}} = \frac{1}{4\pi\varepsilon_0}\frac{q_i q_j}{r_{ij}}$$

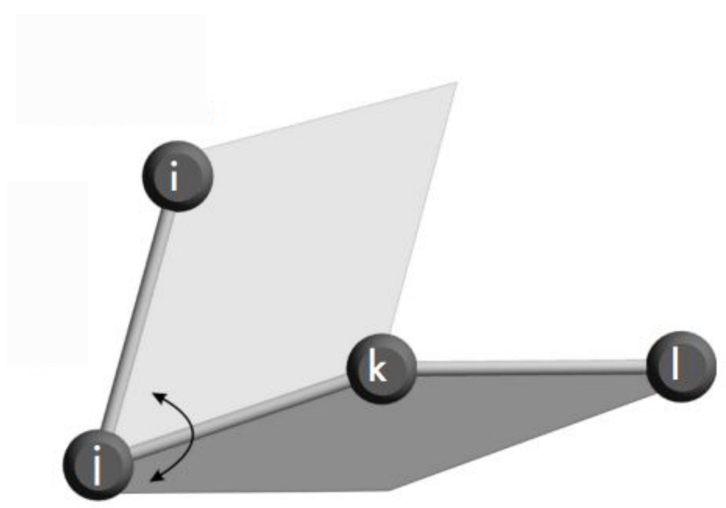
Molecular mechanics and force-fields

The bending energy



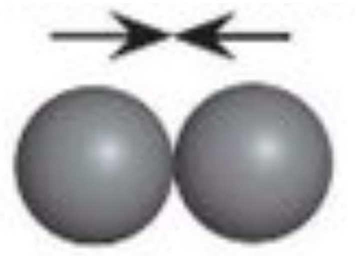
$$E_{\text{bend}}\left(\theta^{ijk}-\theta_0^{ijk}\right)=k_2^{ijk}\left(\theta^{ijk}-\theta_0^{ijk}\right)^2$$

Torsion energy



$$E_{tors}(\omega^{ijkl})=\frac{1}{2}V_1^{ijkl}(1+\cos(\omega^{ijkl}))+ \\ +\frac{1}{2}V_2^{ijkl}(1-\cos(2\omega^{ijkl}))+ \\ +\frac{1}{2}V_3^{ijkl}(1+\cos(3\omega^{ijkl}))$$

Van der Waals energy



$$E_{\text{vdW}}=-\frac{Ar_ir_j}{(r_i+r_j)6r_{ij}^2}$$

Skoltech

Potentials for Molecular Dynamics

Pair potentials are used for gases, liquids, closely (co)packed lattices (metals, ionic solids), but not for systems with covalent bonds (molecules).

$$U_{LJ,t}(r)=\begin{cases}4\varepsilon\left(\left(\frac{\sigma}{r}\right)^{12}-\left(\frac{\sigma}{r}\right)^{6}\right) & r\le r_c\\ 0 & r>r_c\end{cases}$$

For **solids** embedded atom/ion model (EAM/EIM) accounts for collective (non-pairwise) interactions:

$$E_i=\underbrace{\boxed{F_\alpha\left(\sum_{i\neq j}\underbrace{\rho_\beta}_{\text{Electron cloud contribution}}(r_{ij})\right)}}_{\text{Embedding energy}}+\frac{1}{2}\sum_{i\neq j}\underbrace{\phi_{\alpha\beta}}_{\text{Pair interaction}}(r_{ij})$$

Basic principles of Molecular Dynamics

- Solve Newton’s equation of motion for *N* classical particles (3*N* coupled equations).
- For now, let limit ourselves by natural NVE ensemble.
- The force depends on positions only (not velocities).
- Each particle is allowed to interact simultaneously with every other particle and can experience an additional external potential.
- A single point in a 6*N*-dimensional phase space (*p*,*r*) represents our dynamical system.

$$\vec{F} = m\vec{a}$$

$$\mathrm{E}_{tot} = \sum_{i=1}^{N} \frac{1}{2} m\vec{v}_i^2 + V(\vec{r})$$

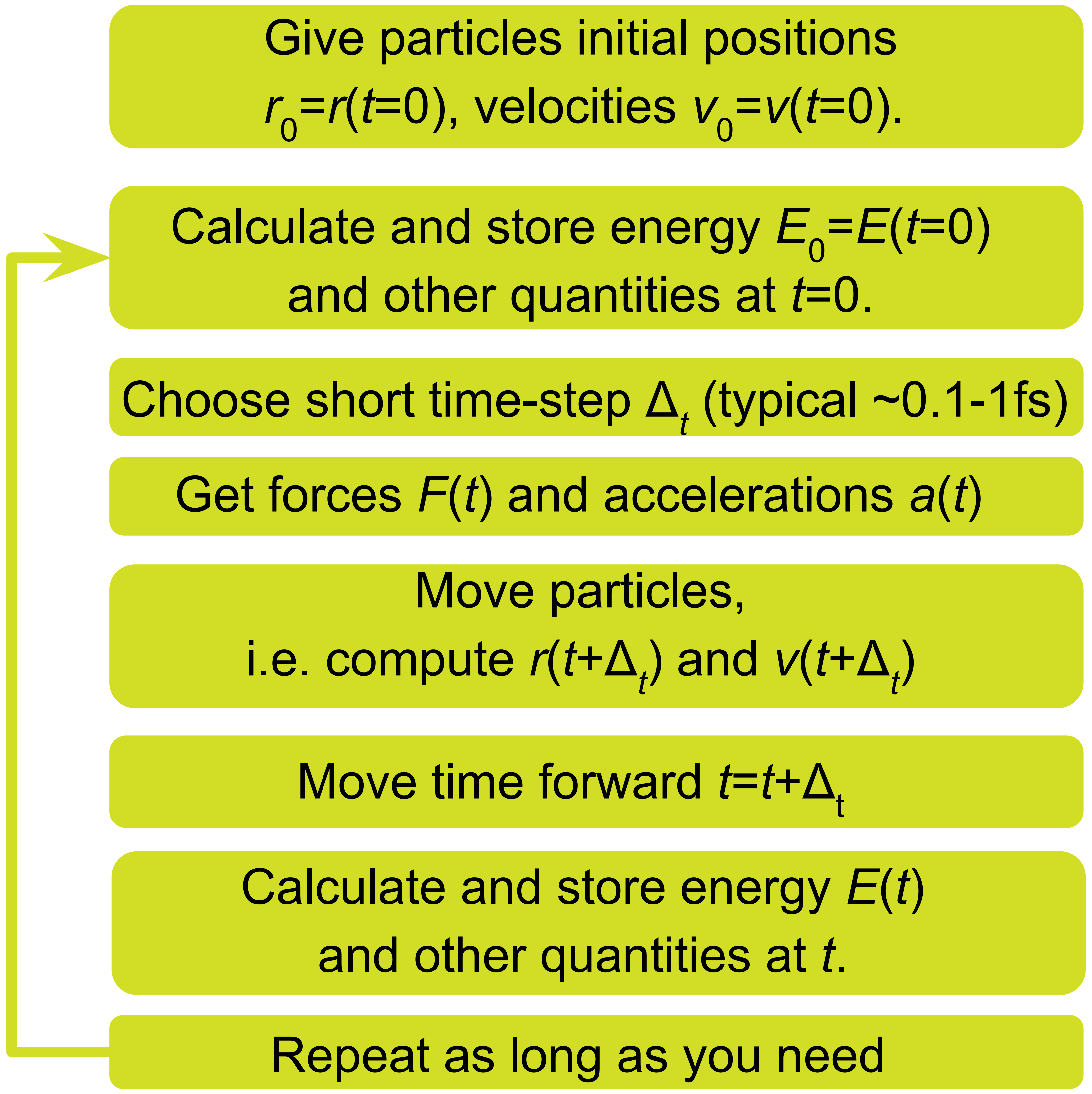
$$-\frac{dV}{d\vec{r}} = m\frac{d^2\vec{r}}{dt^2}$$

Skoltech

Basic principles of Molecular Dynamics

- **Initialize:** select positions and velocities.
- **Propagate:** compute all forces, and then determine new positions.
- **Equilibrate:** let the system reach equilibrium for a given thermodynamic ensemble (e.g. NVE) and ‘forget’ about initial conditions.
- **Sample (average):** accumulate long enough trajectory and calculate quantities of interest.

Algorithm:



Give particles initial positions $r_0=r(t=0)$, velocities $v_0=v(t=0)$.

Calculate and store energy $E_0=E(t=0)$ and other quantities at $t=0$.

Choose short time-step Δ_t (typical ~0.1-1fs)

Get forces $F(t)$ and accelerations $a(t)$

Move particles, i.e. compute $r(t+\Delta_t)$ and $v(t+\Delta_t)$

Move time forward $t=t+\Delta_t$

Calculate and store energy $E(t)$ and other quantities at t.

Repeat as long as you need

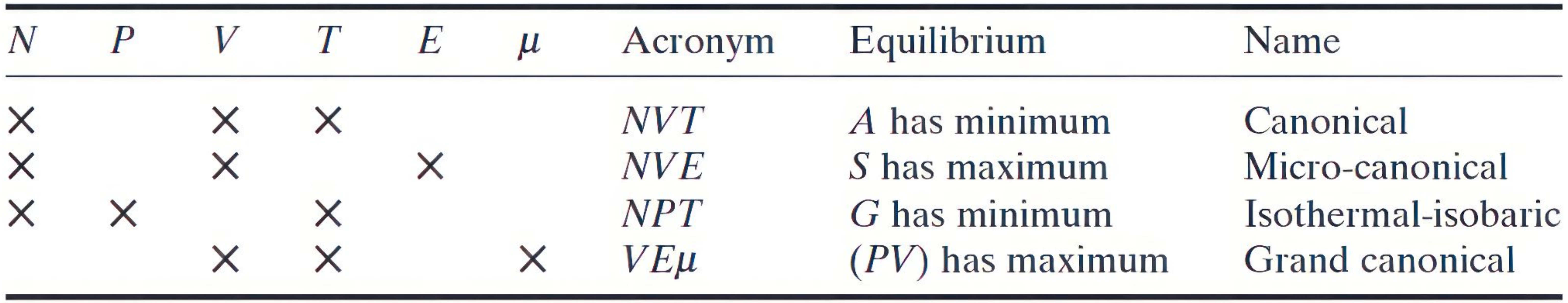
Skoltech

Ensemble and thermostat

Ensemble: collection of microscopic states consistent with thermodynamic boundary conditions; defined by 3 variables (NVT) or (NVE) or (NPT) or (mVT).

Microcanonical (NVE): Newtonian system (*N*=const) in box (*V*=const) with elastic walls (or periodic boundary conditions).
Canonical (NVT): Newtonian system (*N*=const) in box (*V*=const) with non-elastic walls (walls are equilibrated with *T*=const - thermostat).
Isothermal-isobaric (NPT): Newtonian system (*N*=const) in box with varying volume (keeping *P*=const - barostat) and non-elastic walls (keeping *T*=const - thermostat)
Grand-canonical (μVT): Open system (number of particles is not conserved but their energy in the reservoir is fixed at *μ*).

Table. Constants in different ensembles, and corresponding equilibrium states. (From Jensen book)

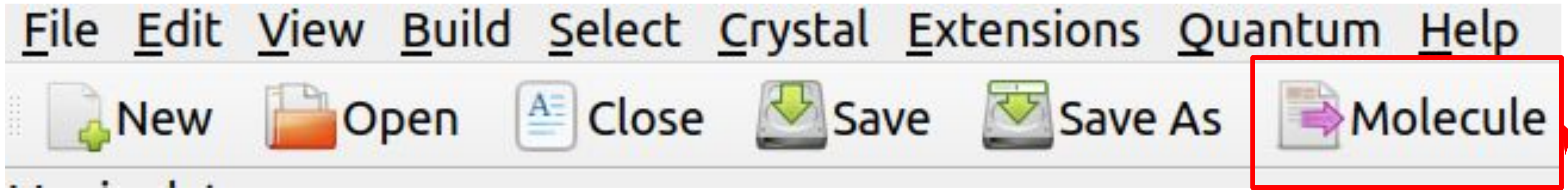


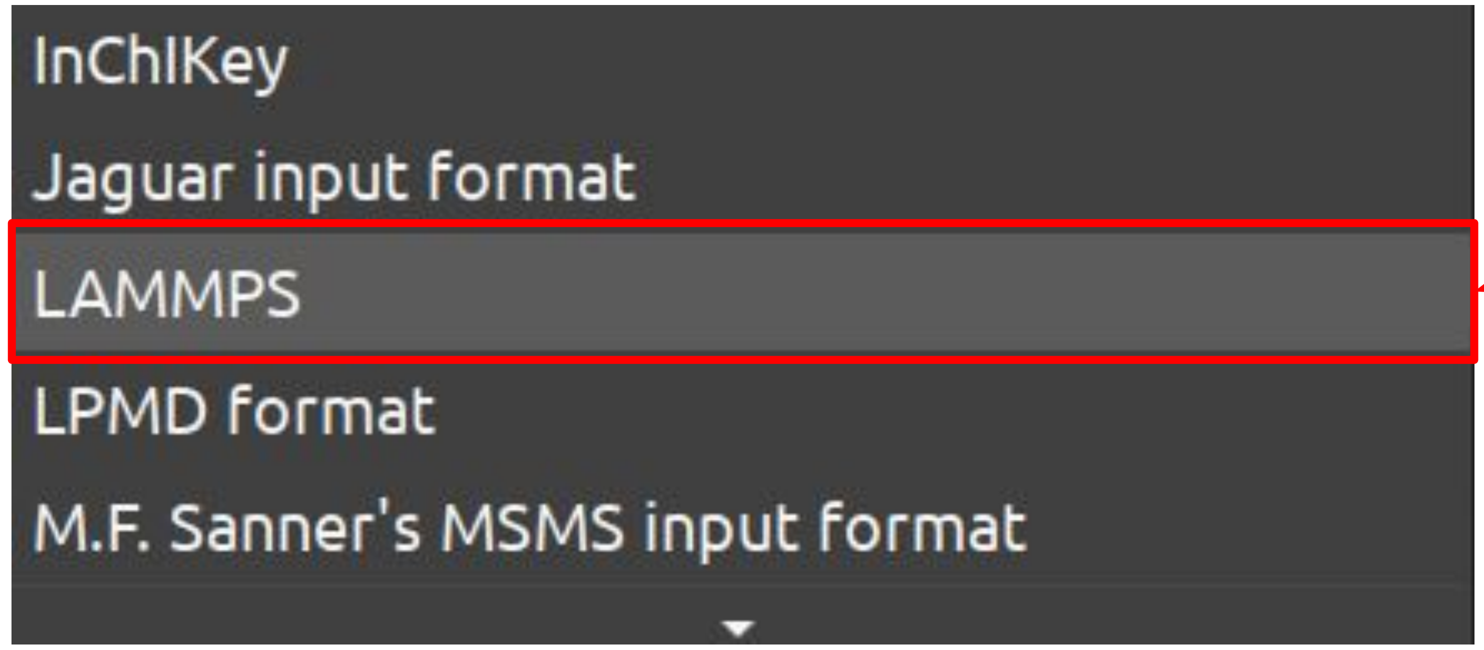
N	P	V	T	E	μ	Acronym	Equilibrium	Name
×		×	×			NVT	A has minimum	Canonical
×		×		×		NVE	S has maximum	Micro-canonical
×	×		×			NPT	G has minimum	Isothermal-isobaric
		×	×		×	$VE\mu$	(PV) has maximum	Grand canonical

Construction of molecules for LAMMPS

Avogadro editor

Avogadro is an advanced molecule editor and visualizer designed for cross-platform use in computational chemistry, molecular modeling, bioinformatics, materials science, and related areas. It offers flexible high quality rendering and a powerful plugin architecture.





Export structure in the format of LAMMPS

Skoltech

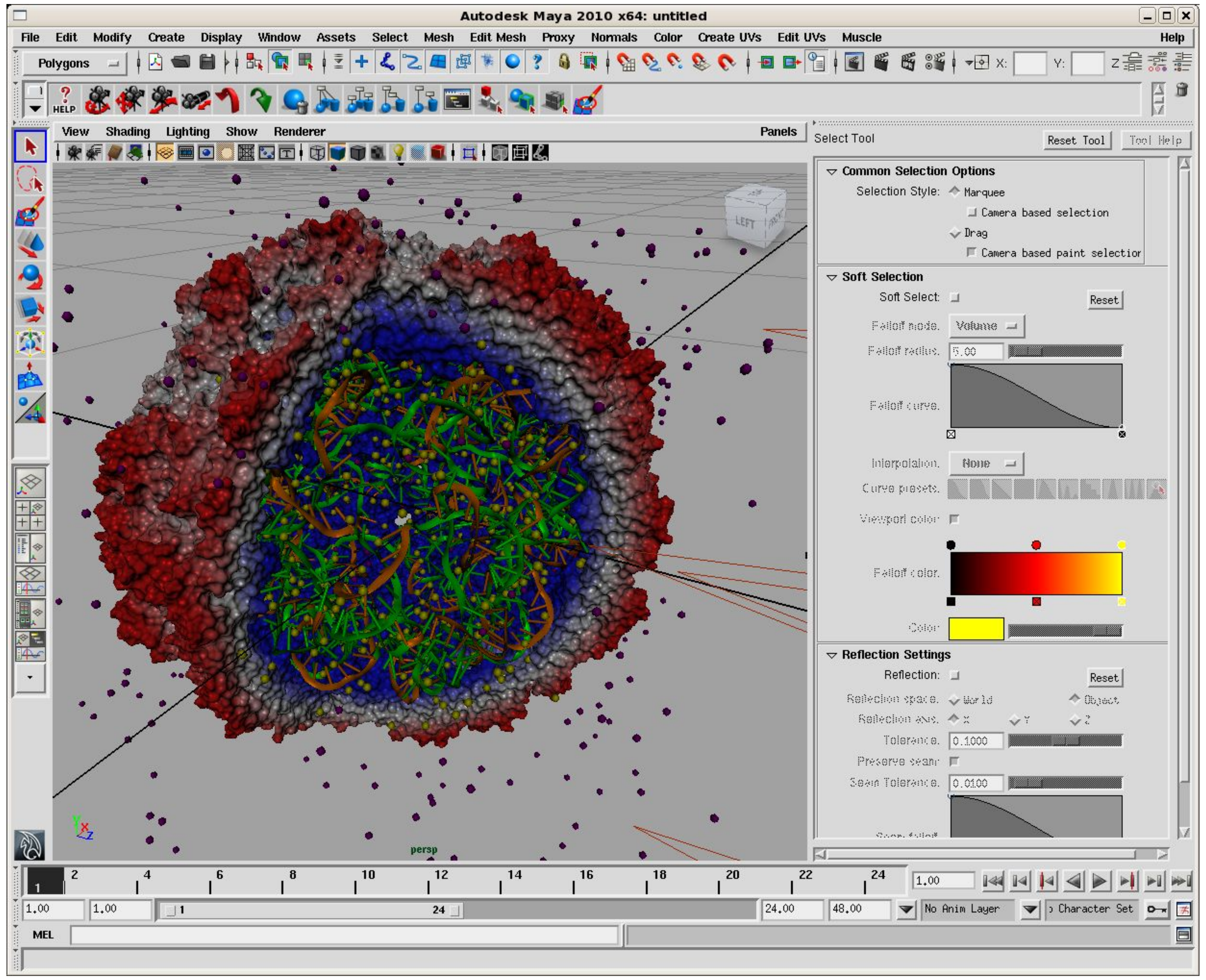
Visualization of MD calculations

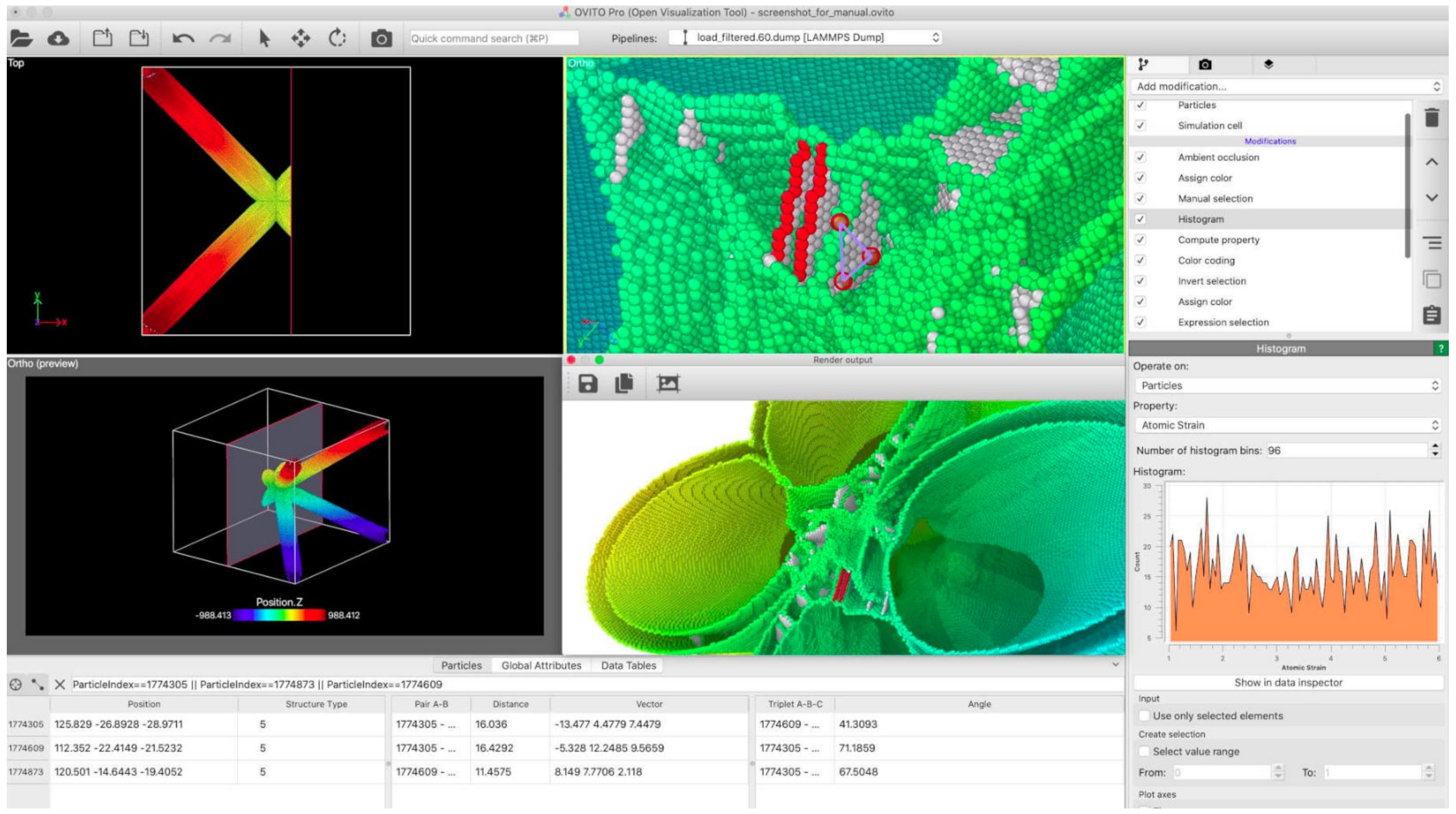
Visualization of MD calculations

Molecular visualization program for displaying, animating, and analyzing systems with large number of atoms.

VMD – better for molecules

Ovito – better for crystals


Autodesk Maya 2010 x64: untitled
File Edit Modify Create Display Window Assets Select Mesh Edit Mesh Proxy Normals Color Create UVs Edit UVs Muscle Help
Polygons
View Shading Lighting Show Renderer Panels
Select Tool
Reset Tool
Tool Help
Common Selection Options
Soft Selection
Reflection Settings
persp


OVITO Pro (Open Visualization Tool) - screenshot_for_manual.ovito
Pipelines: load_filtered.60.dump [LAMMPS Dump]
Top
Ortho
Ortho (preview)
Render output
Position.Z
-988.413
988.412
Histogram
Operate on: Particles
Property: Atomic Strain
Number of histogram bins: 96
Particles Global Attributes Data Tables

ParticleIndex==1774305 || ParticleIndex==1774873 || ParticleIndex==1774609

	Position	Structure Type	Pair A-B	Distance	Vector	Triplet A-B-C	Angle
1774305	125.829 -26.8928 -28.9711	5	1774305 - ...	16.036	-13.477 4.4779 7.4479	1774609 - ...	41.3093
1774609	112.352 -22.4149 -21.5232	5	1774305 - ...	16.4292	-5.328 12.2485 9.5659	1774305 - ...	71.1859
1774873	120.501 -14.6443 -19.4052	5	1774609 - ...	11.4575	8.149 7.7706 2.118	1774305 - ...	67.5048

List with other software

Skoltech

Lab 2. Set up LAMMPS and files

LAMMPS

Large-scale Atomic/Molecular
Massively Parallel Simulator

How to set up LAMMPS locally

Advantages: no need to transfer files between your computer and the remote VIrtual Machine.
No need to wait until tasks of other people finish.

Windows

The Windows version, can be downloaded here: **[Windows LAMMPS]**.

The name should have the following formats:

LAMMPS-64bit-GUI-stable.exe

LAMMPS-64bit-stable.exe

Linux

The Linux version for your distributive, can be downloaded here: **Linux LAMMPS.**

MacOS

The macOS version can be downloaded here: **MacOS LAMMPS.**

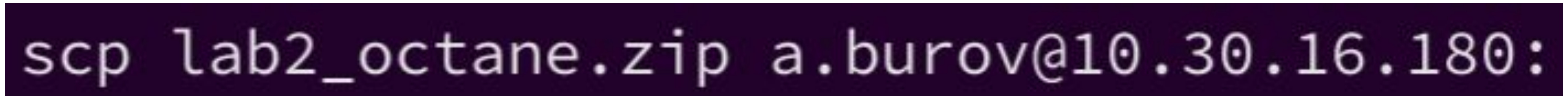
Download archive with **Lab files for octane** and unzip it.

Settings the Lab on the Virtual Machine

Credentials for the Virtual Machine:
‘your_login’@10.30.16.180
‘your_password’

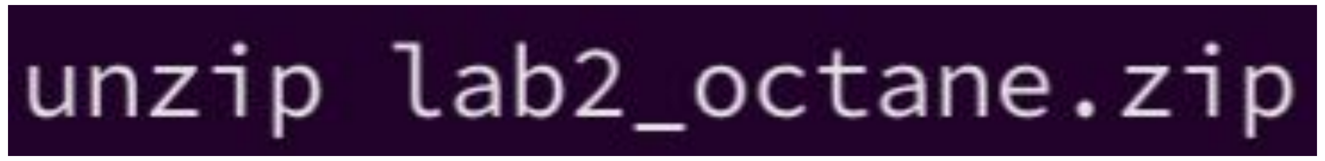
Download archive with [Lab files for octane](#).

Upload the archive to the Virtual Machine:



```
scp lab2_octane.zip a.burov@10.30.16.180:
```

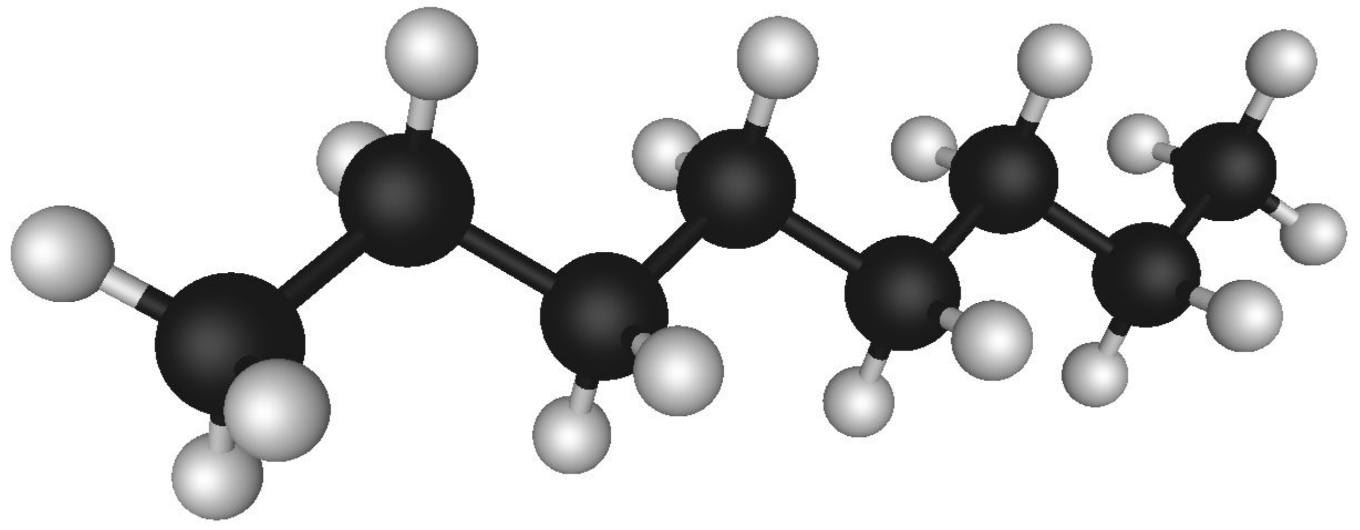
Login to the Virtual Machine and unzip the archive:



```
unzip lab2_octane.zip
```

Skoltech

Lab 2. LAMMPS. Molecules



Skoltech

LAMMPS units

Different units are **available.**

units real

Mass	–	grams/mole
Temp	–	Kelvin
Energy	–	kcal/mole, 43 meV
Distance	–	Angstrom
Time	–	femtoseconds
Force	–	(kcal/mole)/Angstrom, 43 meV/Å
Pressure	–	atmospheres, 0.0001 GPa
Charge	–	multiple of electron charge (1.0 is a proton)
Dipole	–	charge*Angstroms

How to use Lammps

To find input files for your calculations, you need go to *lab2_octane/sampleinputs*

3 input files are required:

anyname.inp (.lmpdat) - contains atomic structure

anyname.prm - parameters for force field

anyname.lam - control parameters including path to **anyname.inp anyname.prm**

To run:

lammps -in <*input_file.lam*>

Input file execution

LAMMPS executes by reading commands from a input script (text file), one line at a time. When the input script ends, LAMMPS exits. Each command causes LAMMPS to take some action. It may set an internal variable, read in a file, or run a simulation. Most commands have default settings, which means you only need to use the command if you wish to change the default.

LAMMPS does not read your entire input script and then perform a simulation with all the settings. Rather, the input script is read one line at a time and each command takes effect when it is read. Thus this sequence of commands.

```
timestep 0.5
run      100
run      100
```

200 steps with 0.5 fs timestep

```
run      100
timestep 0.5
run      100
```

100 steps with 1.0 fs (default) timesteps
100 steps with 0.5 fs timestep

Skoltech

Rules

LAMMPS commands are case sensitive. Command names are lower-case, as are specified command arguments. Upper case letters may be used in file names or user-chosen ID strings.

1. If the last printable character on the line is a “&” character, the command is assumed to continue on the next line.
2. All characters from the first “#” character onward are treated as comment and discarded. The exception to this rule is described in 6.
3. The line is searched repeatedly for $ characters, which indicate variables that are replaced with a text string. The exception to this rule is described in 6. ${myTemp} and $x refer to variables named “myTemp” and “x”, while “$xx” will be interpreted as a variable named “x” followed by an “x” character.
4. The line is broken into “words” separated by white-space (tabs, spaces). Note that words can thus contain letters, digits, underscores, or punctuation characters.
5. The first word is the command name. All successive words in the line are arguments.
6. If you want text with spaces to be treated as a single argument, it can be enclosed in either single or double or triple quotes.

Skoltech

Structure of input file

A LAMMPS input script typically has 4 parts:

1. Initialization
2. System definition
3. Simulation settings
4. Run a simulation

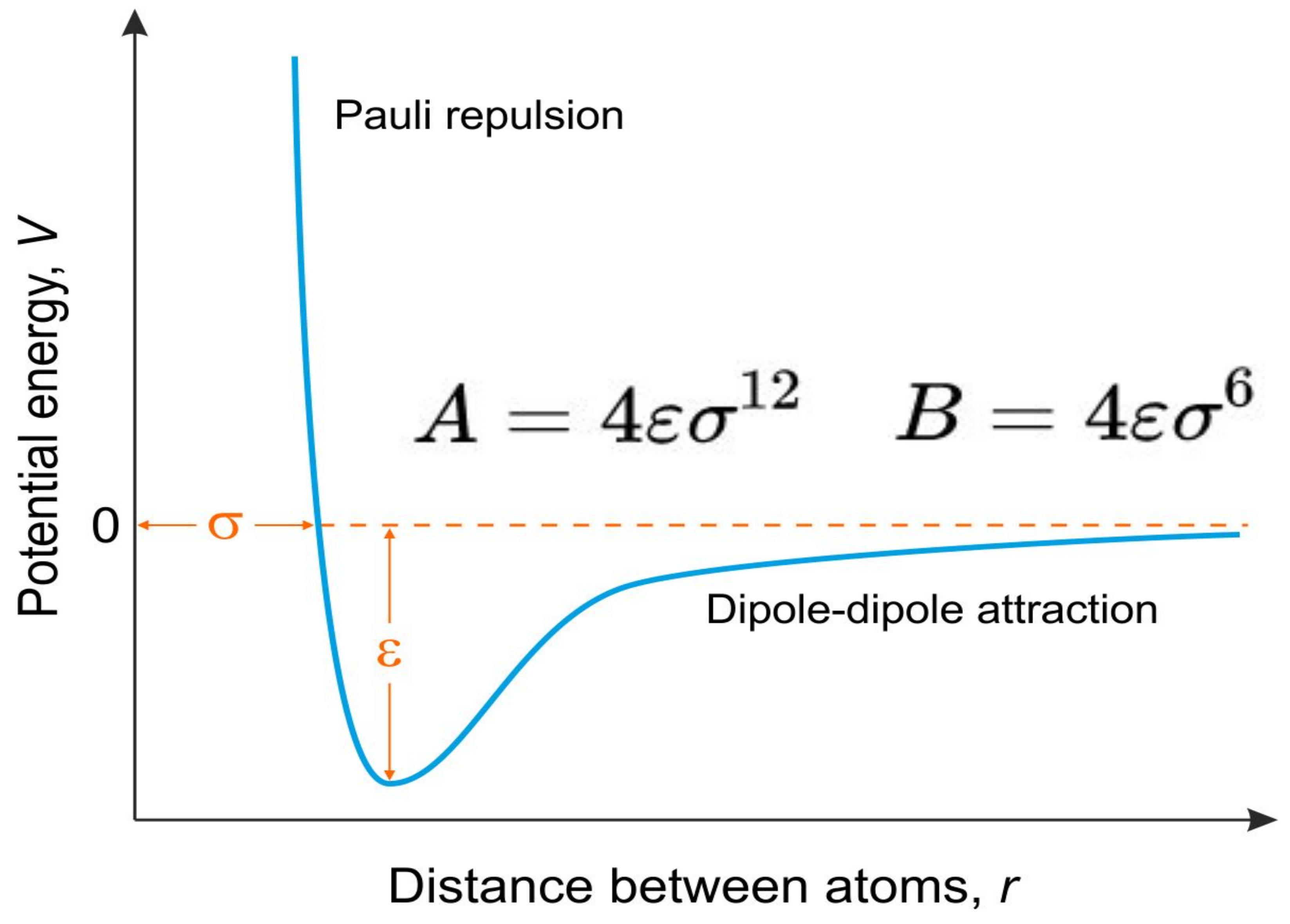
Force field setup

pair_style lj/cut/coul/cut 20 100

pair_coeff I J args

pair_coeff 1 1 $\boxed{0.0660}$ $\boxed{3.5000}$

ε σ



$$A = 4\varepsilon\sigma^{12} \quad B = 4\varepsilon\sigma^{6}$$

Skoltech

Bond description (e.g. harmonic)

bond_style harmonic

$$E = K(r - r_0)^2$$

K - (energy/distance^2)

r_0 - (distance)

bond_coeff *N* args

- *N* = bond type (see asterisk form below)
- args = coefficients for one or more bond types

Weights of bonding (pair, angle, dihedral) and non-bonding (vdW and Coulomb) contributions

special_bonds keyword values ...

```
special_bonds    lj    0    0    .5    coul    0    0    .5
```

turn off LJ and Coulomb for 1-2 and 1-3 atom pairs for which bonds are defined. Take 0.5 of LJ and 0.5 of Coulomb interactions for 1-4 atom pairs (dihedral).

Skoltech

System definition

There are 3 ways to define the simulation cell and reserve space for force field info and fill it with atoms in LAMMPS.

1. Read them in from (1) a data file or (2) a restart file via the **read_data** or **read_restart** commands, respectively. These files can also contain molecular topology information.
2. Or (3) create a simulation cell and fill it with atoms on a lattice (with no molecular topology), using these commands: **lattice**, **region**, **create_box**, **create_atoms** or **read_dump**.
3. The entire set of atoms can be duplicated to make a larger simulation using the **replicate** command.

Simulation setting

Force field coefficients are set by these commands (they can also be set in the read-in files): **pair_coeff**, **bond_coeff**, **angle_coeff**, **dihedral_coeff**, **improper_coeff**, **kspace_style**, **dielectric**, **special_bonds**.

Various simulation parameters are set by these commands: **neighbor**, **neigh_modify**, **group**, **timestep**, **reset_timestep**, **run_style**, **min_style**, **min_modify**.

Fixes impose a variety of boundary conditions, time integration, and diagnostic options. The **fix** command comes in many flavors.

Various computations can be specified for execution during a simulation using the **compute**, **compute_modify**, and **variable** commands.

Output options are set by the **thermo**, **dump**, and **restart** commands.

Commands by category

Fix setting

fix *ID group-ID style args*

fix 3 all nvt temp 300.0 300.0 0.01

Set a **fix** that will be applied to a group of atoms. In LAMMPS, a “fix” is any operation that is applied to the system during timestepping or minimization. Examples include updating of atom positions and velocities due to time integration, controlling temperature, applying constraint forces to atoms, enforcing boundary conditions, computing diagnostics, etc.

Fixes can be deleted with the **unfix** command.

Skoltech

How to run a simulation

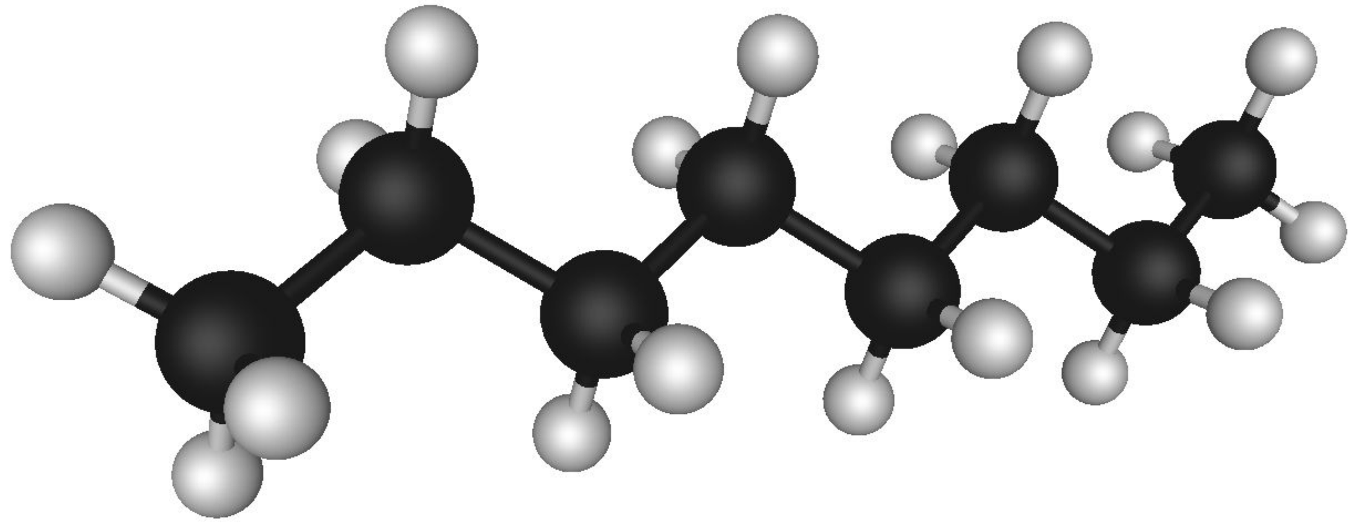
A molecular dynamics simulation is run using the **run** command. Energy minimization (molecular statics) is performed using the **minimize** command. A parallel tempering (replica-exchange) simulation can be run using the **temper** command.

```
run 1000
```

Output file

- **Thermodynamic output**, which is a list of quantities printed every few timesteps to the screen and logfile.
- **Dump** files, which contain snapshots of atoms and various per-atom values and are written at a specified frequency.
- Certain fixes can output user-specified quantities to files: **fix ave/time** for time averaging, **fix ave/chunk** for spatial or other averaging, and fix **print** for single-line output of **variables**. Fix print can also output to the screen.
- **Restart files**.

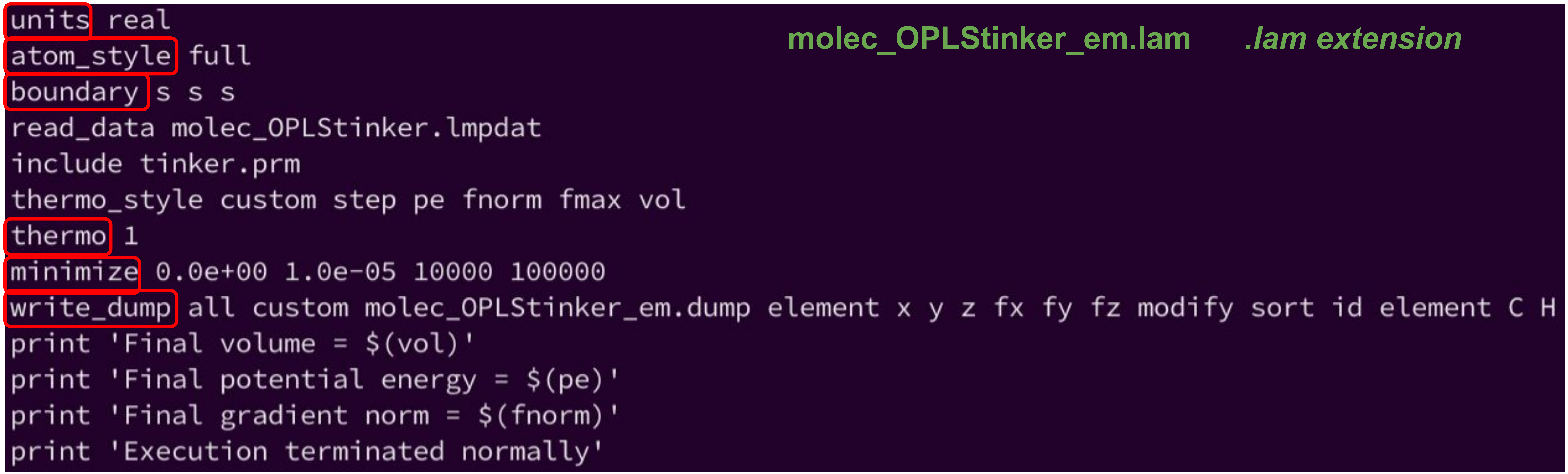
Lab 2. LAMMPS. Benzene



Skoltech

1. Geometry optimization

units – sets the style of units used for a simulation.
atom_style – define what style of atoms to use in a simulation. This determines what attributes are associated with the atoms.
boundary – set the style of boundaries for the global simulation box in each dimension. *S* is non-periodic condition, where the position of the face is set so as to encompass the atoms in that dimension (shrink-wrapping), no matter how far they move.



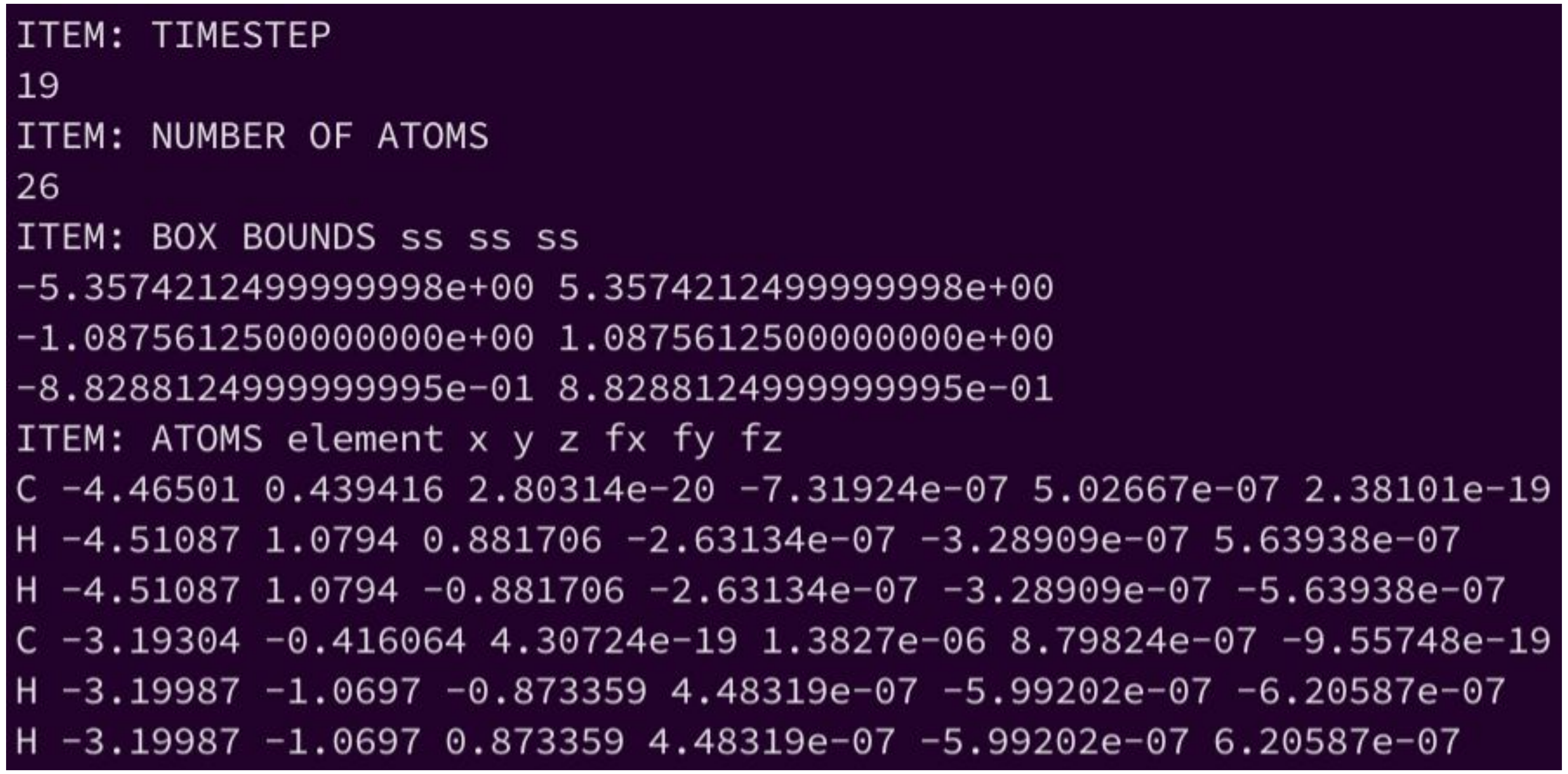
molec_OPLStinker_em.lam *.lam extension*

```
units real
atom_style full
boundary s s s
read_data molec_OPLStinker.lmpdat
include tinker.prm
thermo_style custom step pe fnorm fmax vol
thermo 1
minimize 0.0e+00 1.0e-05 10000 100000
write_dump all custom molec_OPLStinker_em.dump element x y z fx fy fz modify sort id element C H
print 'Final volume = $(vol)'
print 'Final potential energy = $(pe)'
print 'Final gradient norm = $(fnorm)'
print 'Execution terminated normally'
```

thermo_style – set the style and content for printing thermodynamic data to the screen and log files.
thermo – compute and print thermodynamic info on timesteps that are a multiple of *N*.
minimize (etol ftol maxiter maxeval) – perform an energy minimization of the system, by iteratively adjusting atom coordinates. Iterations are terminated when one of the stopping criteria is satisfied.
write_dump – dump a single snapshot of atom quantities to one or more files for the current state of the system.

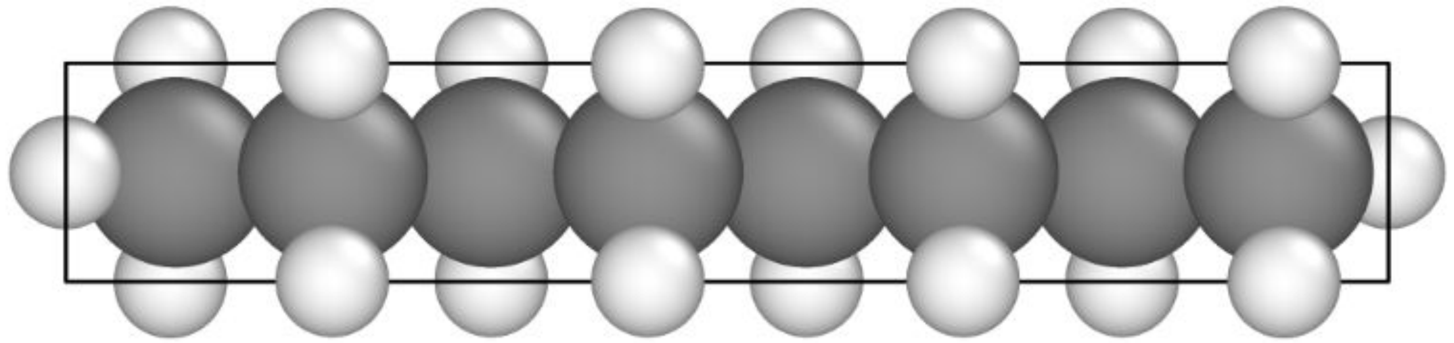
1. Geometry optimization

Optimized structure

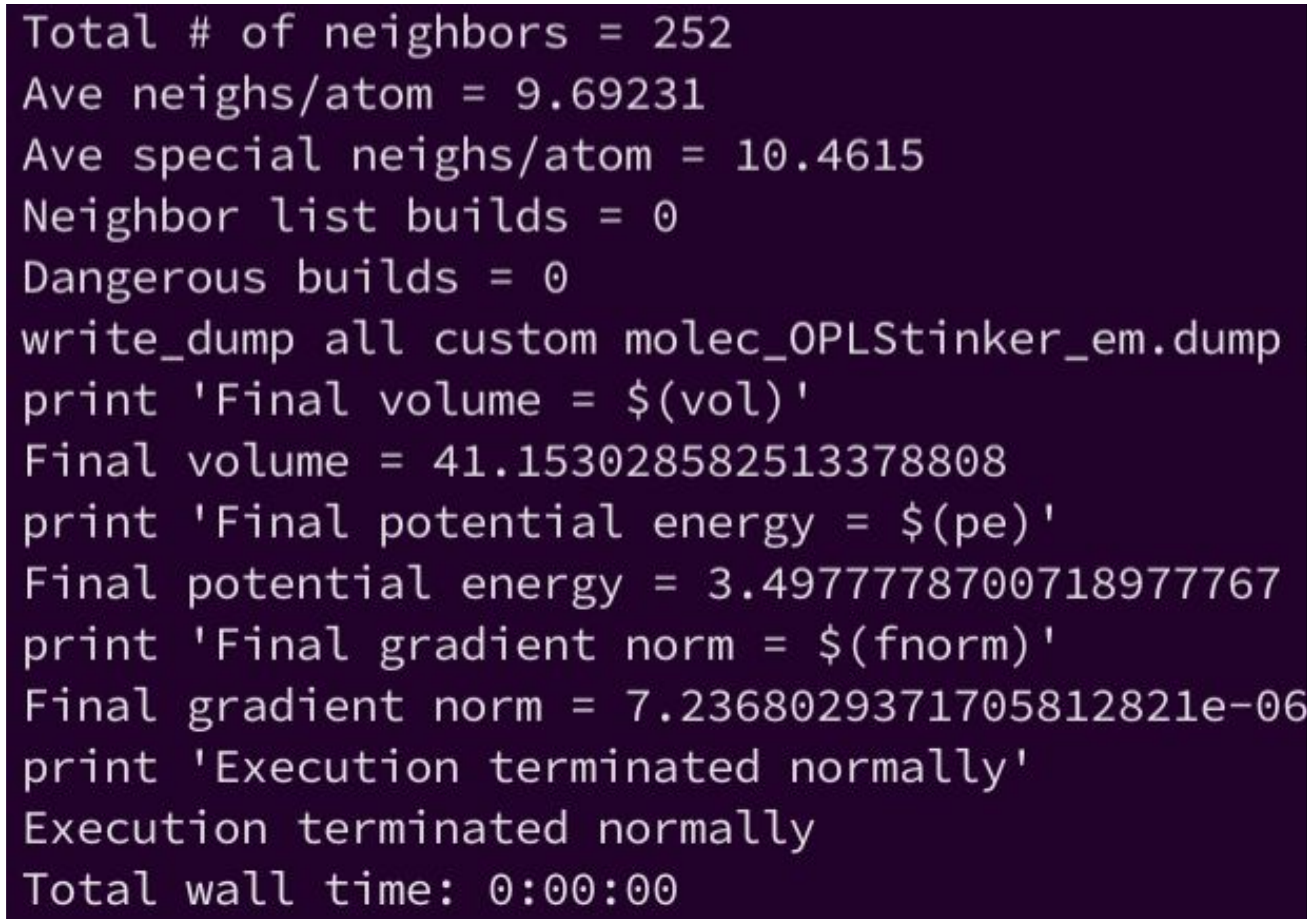


```
ITEM: TIMESTEP
19
ITEM: NUMBER OF ATOMS
26
ITEM: BOX BOUNDS ss ss ss
-5.3574212499999998e+00 5.3574212499999998e+00
-1.0875612500000000e+00 1.0875612500000000e+00
-8.8288124999999995e-01 8.8288124999999995e-01
ITEM: ATOMS element x y z fx fy fz
C -4.46501 0.439416 2.80314e-20 -7.31924e-07 5.02667e-07 2.38101e-19
H -4.51087 1.0794 0.881706 -2.63134e-07 -3.28909e-07 5.63938e-07
H -4.51087 1.0794 -0.881706 -2.63134e-07 -3.28909e-07 -5.63938e-07
C -3.19304 -0.416064 4.30724e-19 1.3827e-06 8.79824e-07 -9.55748e-19
H -3.19987 -1.0697 -0.873359 4.48319e-07 -5.99202e-07 -6.20587e-07
H -3.19987 -1.0697 0.873359 4.48319e-07 -5.99202e-07 6.20587e-07
```

molec_OPLStinker_em.dump *.dump extension*



Data on the optimization and calculated output values like energies, volume and forces.



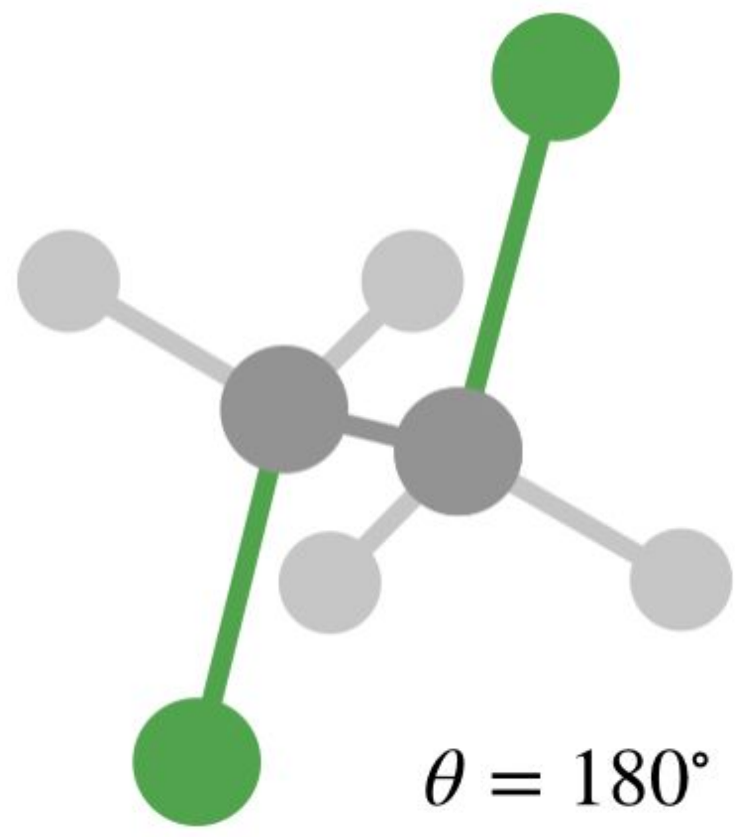
```
Total # of neighbors = 252
Ave neighs/atom = 9.69231
Ave special neighs/atom = 10.4615
Neighbor list builds = 0
Dangerous builds = 0
write_dump all custom molec_OPLStinker_em.dump
print 'Final volume = $(vol)'
Final volume = 41.153028582513378808
print 'Final potential energy = $(pe)'
Final potential energy = 3.4977787007189777 67
print 'Final gradient norm = $(fnorm)'
Final gradient norm = 7.2368029371705812821e-06
print 'Execution terminated normally'
Execution terminated normally
Total wall time: 0:00:00
```

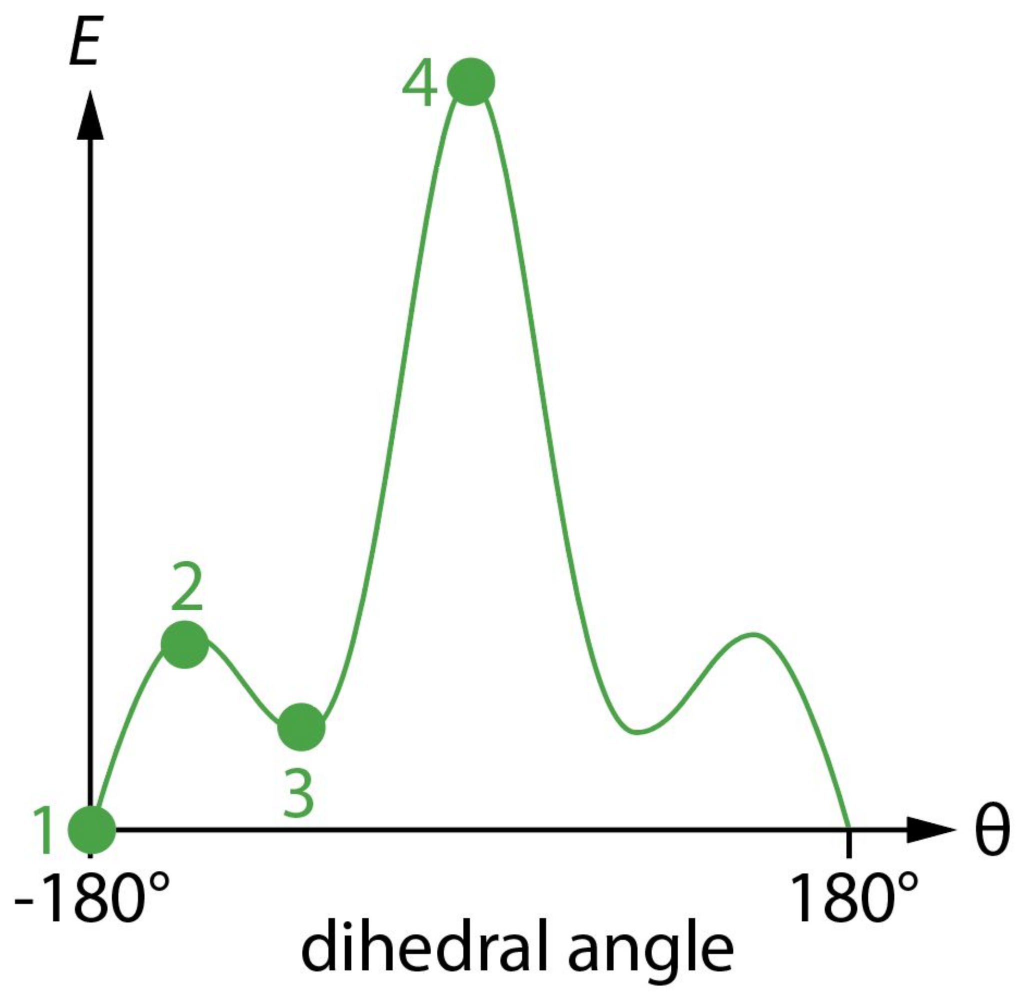
log.lammps *.lammps extension*

Skoltech

2. Conformers

Conformational isomerism is a form of spatial isomerism in which the isomers can be interconverted just by rotations about formally single bonds. While any two arrangements of atoms in a molecule that differ by rotation about single bonds can be referred to as different conformations, conformations that correspond to local minima on the potential energy surface are specifically called *conformational isomers* or *conformers*.





Skoltech

2. Conformers

lammps_pes_script.py – generates conformers for a specified molecule, using axis of rotation and rotating atoms.

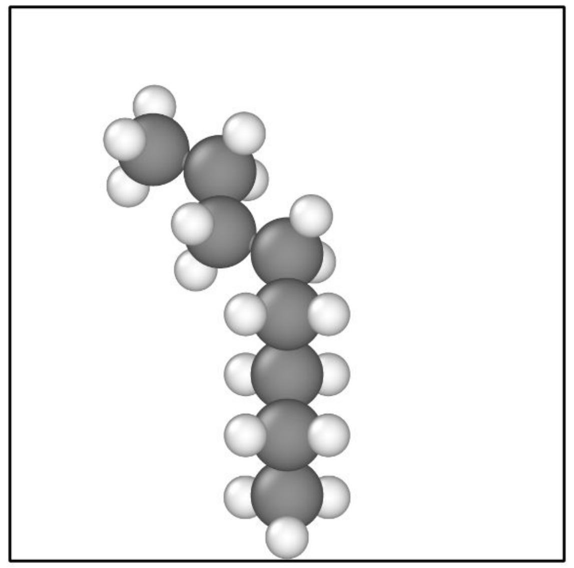
template.lam– auxiliary files, which are used to generate an input file with extension .lam for each conformer.

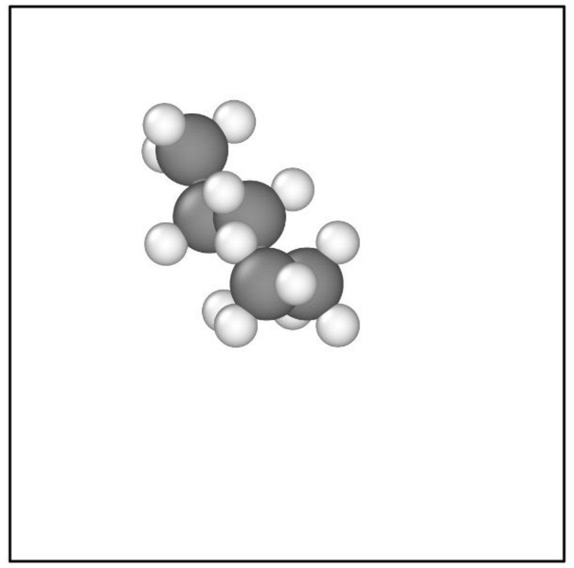
run_pes.sh – slurm script to launch calculation for all conformers. The variable with names: <INP_STRUCTURE="octane_${i}.lmpdat">should be adjusted to your molecule.

octane.lmpdat – file with the initial structure of a molecule for which you want to study conformers.

tinker.prm – file with parametrization.

make_plot.py – generates a plot from log files.



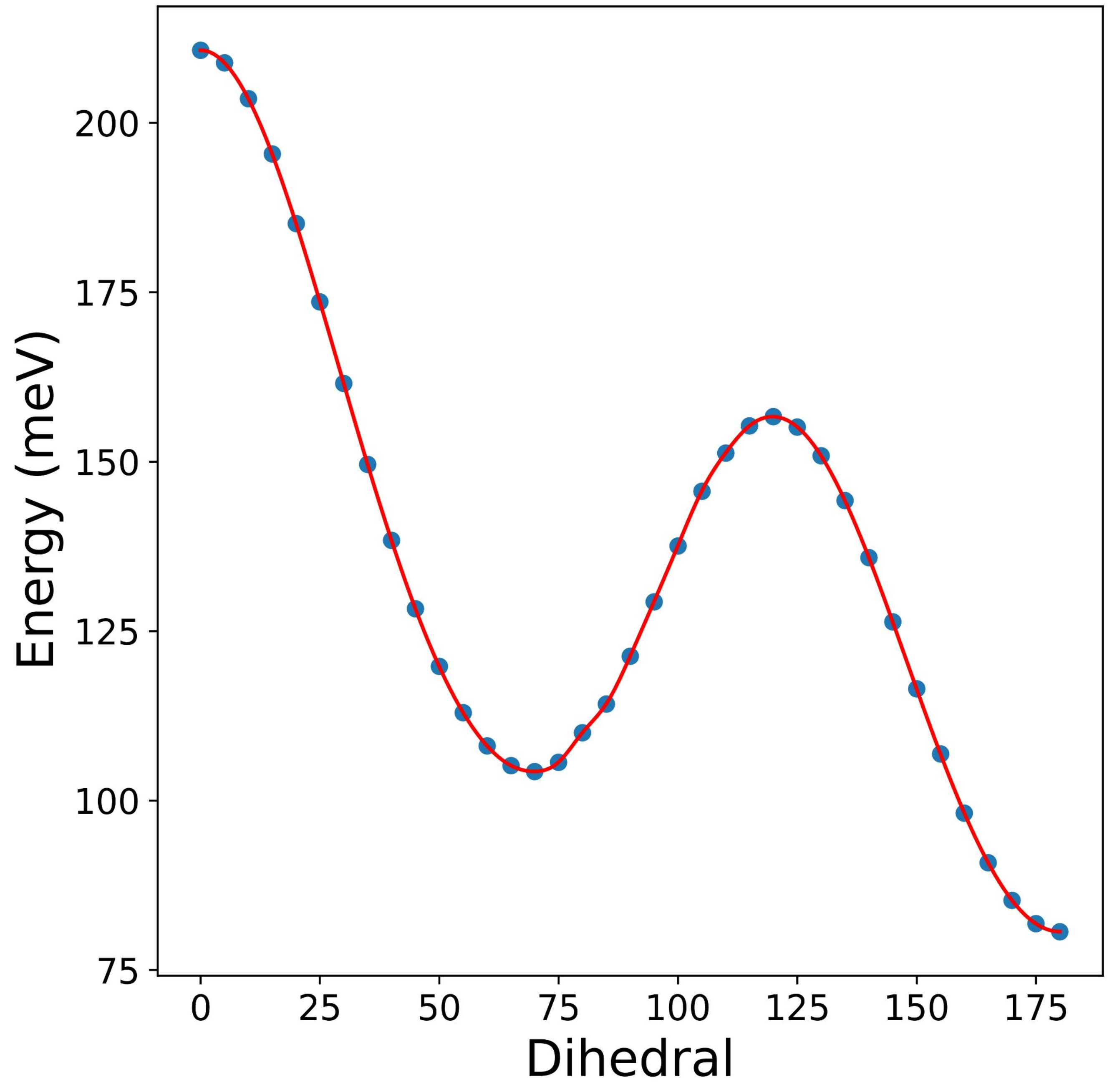


Skoltech

2. Conformers

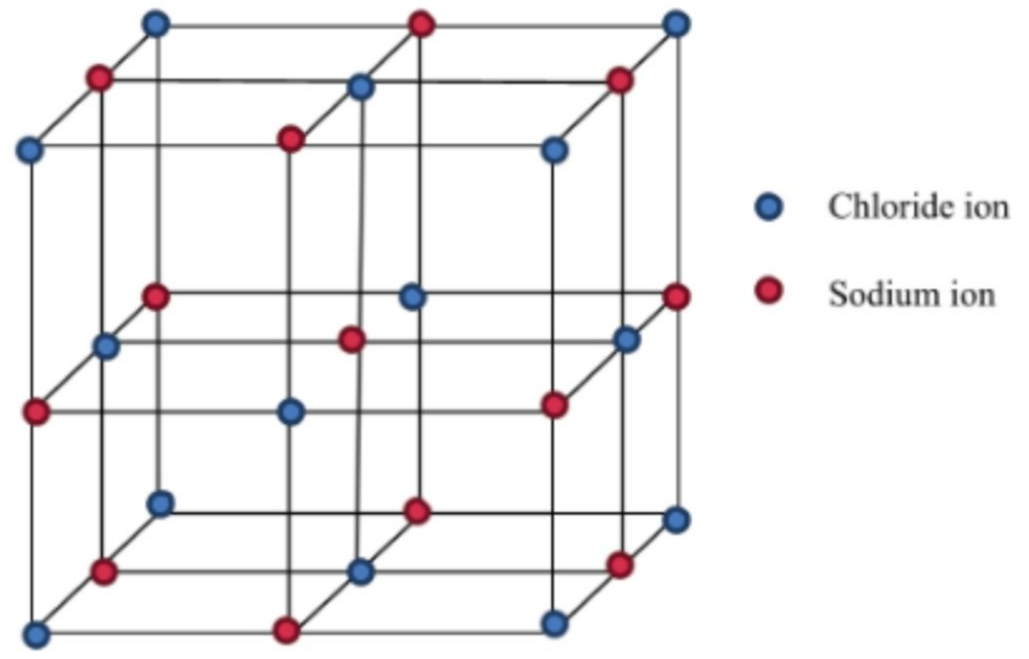
How to repeat the calculation, located in sampleinputs/pes

1. Replace **octane.lmpdat** with your molecule.
2. Determine an axis of rotation using two atoms and specify the atoms which you want to rotate. Also, provide the maximum angle of rotation and step or rotation. Remember, that based on your molecule, the dihedral angle can be defined as 360-α,180-α or α, where α is the rotation angle, used in **lammps_pes_script.py** script.
3. Launch **lammps_pes_script.py** to get the list of structures.
4. Launch **run_pes.sh** to calculate all structures.
5. Make a plot using **make_plot.py** and analyze the results.



Skoltech

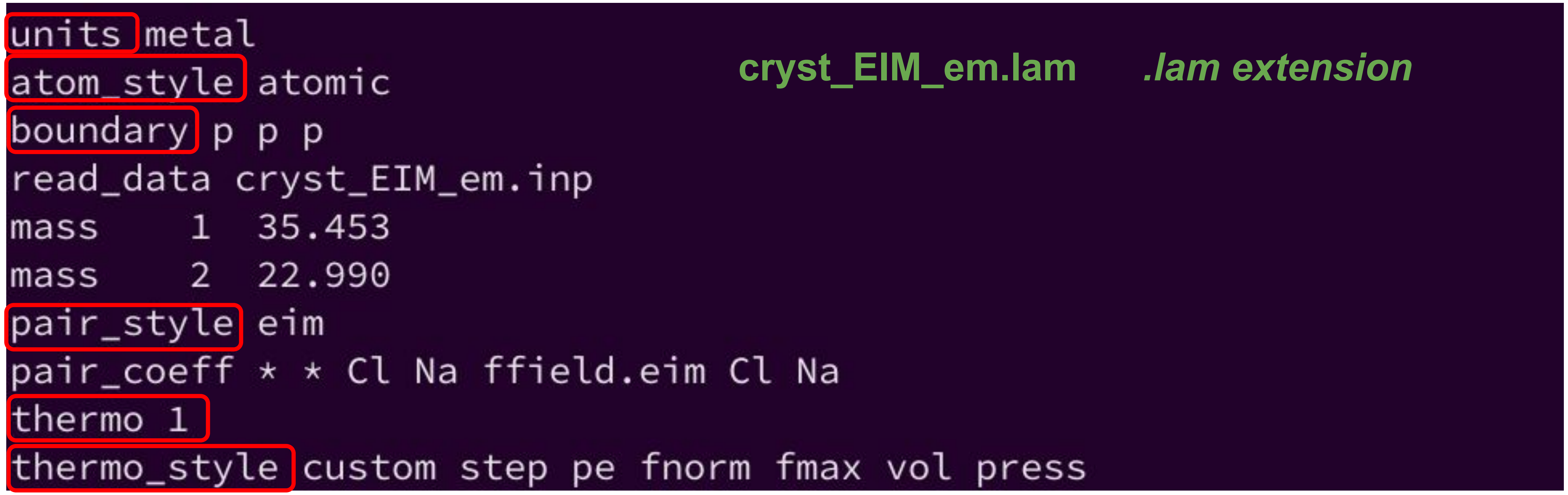
Lab 2. LAMMPS. NaCl



Skoltech

1. Geometry optimization

units – sets the style of units used for a simulation.
atom_style – define what style of atoms to use in a simulation. This determines what attributes are associated with the atoms.
boundary – set the style of boundaries for the global simulation box in each dimension. *S* is non-periodic condition, where the position of the face is set so as to encompass the atoms in that dimension (shrink-wrapping), no matter how far they move.



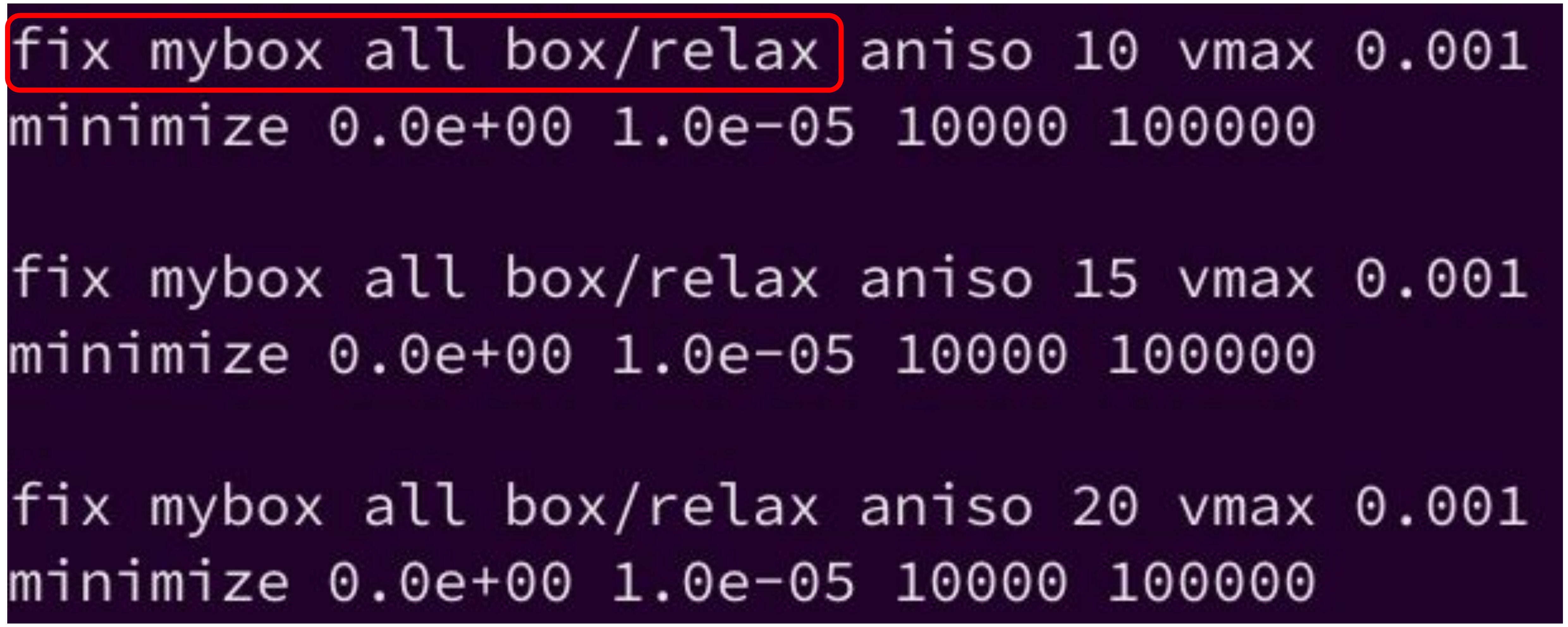
```
units metal
atom_style atomic
boundary p p p
read_data cryst_EIM_em.inp
mass     1  35.453
mass     2  22.990
pair_style eim
pair_coeff * * Cl Na ffield.eim Cl Na
thermo 1
thermo_style custom step pe fnorm fmax vol press
```

thermo_style – set the style and content for printing thermodynamic data to the screen and log files.
thermo – compute and print thermodynamic info on timesteps that are a multiple of *N*.
pair_style (etol ftol maxiter maxeval) – perform an energy minimization of the system, by iteratively adjusting atom coordinates. Iterations are terminated when one of the stopping criteria is satisfied.

1. Geometry optimization

fix box/relax – Apply an external pressure or stress tensor to the simulation box during an **energy minimization**. This allows the box size and shape to vary during the iterations of the minimizer so that the final configuration will be both an energy minimum for the potential energy of the atoms, and the system pressure tensor will be close to the specified external tensor.

aniso – anisotropic, when x, y, and z dimensions are controlled independently using the Pxx, Pyy, and Pzz.
vmax – limit the fractional change in the volume of the simulation box that can occur in one iteration of the minimizer.



```
fix mybox all box/relax aniso 10 vmax 0.001
minimize 0.0e+00 1.0e-05 10000 100000

fix mybox all box/relax aniso 15 vmax 0.001
minimize 0.0e+00 1.0e-05 10000 100000

fix mybox all box/relax aniso 20 vmax 0.001
minimize 0.0e+00 1.0e-05 10000 100000
```

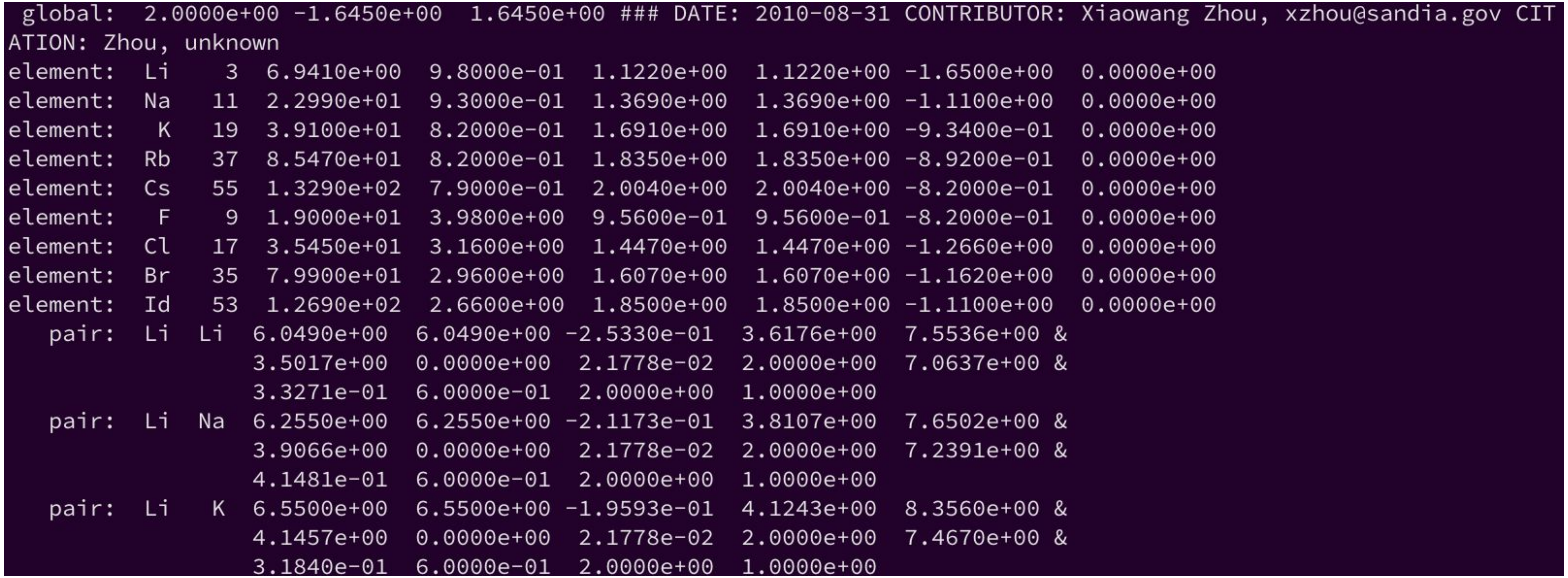
molec_OPLStinker_em.lam *.lam extension*

Skoltech

1. Geometry optimization

Embedded ion model:

$$E = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=i_1}^{i_N}\phi_{ij}\left(r_{ij}\right) + \sum_{i=1}^{N} E_i\left(q_i, \sigma_i\right)$$



```
 global:  2.0000e+00 -1.6450e+00  1.6450e+00 ### DATE: 2010-08-31 CONTRIBUTOR: Xiaowang Zhou, xzhou@sandia.gov CIT
ATION: Zhou, unknown
element:  Li    3  6.9410e+00  9.8000e-01  1.1220e+00  1.1220e+00 -1.6500e+00  0.0000e+00
element:  Na   11  2.2990e+01  9.3000e-01  1.3690e+00  1.3690e+00 -1.1100e+00  0.0000e+00
element:   K   19  3.9100e+01  8.2000e-01  1.6910e+00  1.6910e+00 -9.3400e-01  0.0000e+00
element:  Rb   37  8.5470e+01  8.2000e-01  1.8350e+00  1.8350e+00 -8.9200e-01  0.0000e+00
element:  Cs   55  1.3290e+02  7.9000e-01  2.0040e+00  2.0040e+00 -8.2000e-01  0.0000e+00
element:   F    9  1.9000e+01  3.9800e+00  9.5600e-01  9.5600e-01 -8.2000e-01  0.0000e+00
element:  Cl   17  3.5450e+01  3.1600e+00  1.4470e+00  1.4470e+00 -1.2660e+00  0.0000e+00
element:  Br   35  7.9900e+01  2.9600e+00  1.6070e+00  1.6070e+00 -1.1620e+00  0.0000e+00
element:  Id   53  1.2690e+02  2.6600e+00  1.8500e+00  1.8500e+00 -1.1100e+00  0.0000e+00
   pair:  Li  Li  6.0490e+00  6.0490e+00 -2.5330e-01  3.6176e+00  7.5536e+00 &
                  3.5017e+00  0.0000e+00  2.1778e-02  2.0000e+00  7.0637e+00 &
                  3.3271e-01  6.0000e-01  2.0000e+00  1.0000e+00
   pair:  Li  Na  6.2550e+00  6.2550e+00 -2.1173e-01  3.8107e+00  7.6502e+00 &
                  3.9066e+00  0.0000e+00  2.1778e-02  2.0000e+00  7.2391e+00 &
                  4.1481e-01  6.0000e-01  2.0000e+00  1.0000e+00
   pair:  Li   K  6.5500e+00  6.5500e+00 -1.9593e-01  4.1243e+00  8.3560e+00 &
                  4.1457e+00  0.0000e+00  2.1778e-02  2.0000e+00  7.4670e+00 &
                  3.1840e-01  6.0000e-01  2.0000e+00  1.0000e+00
```

ffield.eim   *.eim extension*

Skoltech

Extra materials

More to read:

- Lammps docs: unpad
- Lammps tutorial: Gravelle
- Interatomic potentials: NIST, openKIM
- Python scripts for LAMMPS: Pizza
- Conformer search: Avogadro (ver1, ver2, ver2_bin), mcsorkun

Lammps tutorials:

- LAMMPS workshops
- Gravelle, Atomsk
- Tutorials set1, Tutorials set2, Tutorials set3

Thx

Skoltech