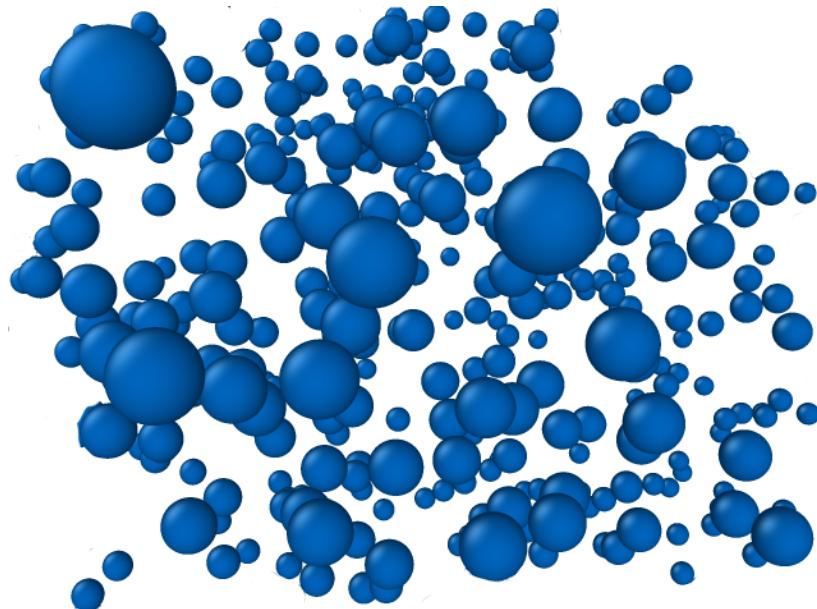




Technische Universität München

TECHNICAL UNIVERSITY OF MUNICH

ATOMISTIC MODELING OF MATERIALS



Final Project

Lukas Scheucher

March 7, 2017

Contents

1	Three-dimensional atomistic system	3
1.1	Determination of lattice constant	3
1.2	Constant temperature and molecular dynamics	3
1.2.1	Equilibration	4
1.2.2	Stability comparison between Nose-Hoover and Berendsen	4
1.2.3	Dependence on the system-size	4
1.2.4	MCMC	4
1.3	Constant pressure/temperature molecular dynamics	5
1.3.1	Finding the equilibrium box sizes	5
1.3.2	Evaluation of CV	5
1.4	Phase transition of Krypton	6
2	Two-dimensional atomistic tensile test	17
2.1	Equilibration	17
2.2	Tensile test	17

List of Figures

1.1	Lattice constant determination - convergence	4
1.2	Influence of the damping parameter on results - time convergence	7
1.3	Influence of the damping parameter on results - histograms	8
1.4	Comparisons between Nose-Hoover thermostat and weakly-coupled Berendsen thermostat - time convergence	9
1.5	Comparisons between Nose-Hoover thermostat and weakly-coupled Berendsen thermostat - histograms	10
1.6	Influence of the atom number on the results quality	11
1.7	Comparison between MCMC and NVT-MD	11
1.8	Determination of equilibrated box-size	12
1.9	Influence of the system size on the relative variance of pressure results	13
1.10	Density results over box-size	13
1.11	Equilibration under NPT ensemble	14
1.12	Equilibrium box-length estimation under NPT-ensemle	15
1.13	Phase transition of Krypton under NPT ensemble	15
1.14	Visualization of the trajectories during the phase transitions	16
2.1	Determination of appropriate damping parameters - Potential Energy	18
2.2	Determination of appropriate damping parameters - Temperature	19
2.3	System snapshots of the 2D system during equilibration	20
2.4	Stress-strain relations	20
2.5	Visualization of the trajectories during stress-strain simulation	21

Chapter 1

Three-dimensional atomistic system

All files for this task are located in [1_three-dimensional_atomic_system](#)

1.1 Determination of lattice constant

For the determination of the lattice constant and lattice type, 3 different configurations are considered:

1. Face-centered cubic (FCC)
2. Body-centered cubic (BCC)
3. Primitive cubic (PC)

The system will take those configuration, which shows the least total energy per atom. Since interactions happen not only with the closes neighbors, but also with other atoms, an simulation has to be performed to get the exact energies. This is done below.

However, a good first approximation can be made analytically by considering the following observations: All atoms within a lattice type are equal, meaning that an atom that appears as a center atom in BCC can also be viewed as a corner atom just by shifting the connections. Secondly, if one denotes the length of the cube as lattice constant a , one can easily determine the distances between all atom by simple geometric means.

By looking only at the closest neighbors, one can derive the following expression for the energy per atom in all 3 grids:

$$E_{FCC} = 6U(a_{FCC}) + 12U(a_{FCC}/\sqrt{2}) \quad (1.1)$$

$$E_{BCC} = 6U(a_{BCC}) + 8U(a_{BCC}\sqrt{3}/2) \quad (1.2)$$

$$E_{PC} = 6U(a_{PC}) \quad (1.3)$$

By solving for local minima, the following lattice constants are obtained

$$a_{FCC,min} = 5.7432\text{\AA} \quad (1.4)$$

$$a_{BCC,min} = 4.6143\text{\AA} \quad (1.5)$$

$$a_{PC,min} = 4.097\text{\AA} \quad (1.6)$$

$$(1.7)$$

Substituting this back in to the equation for the energies per atom, one can see, that the FCC configuration with an approximate lattice constant of 5.7432\AA gives the smallest total energy and is therefore the configuration obtained.

Scripts used for this question: [lattice_constant_approximation.m](#)

Of course, the lattice constant can also be obtained by simulations. This has been done with the input script [in.latticeconstant](#). To run the simulations just call [run_latticeconstant.sh](#).

The lattice-constant obtained after minimization is 5.64\AA . The minimization process is visualized in Figure 1.1.

1.2 Constant temperature and molecular dynamics

Through several simulations, a timestep of 10 fs has been determined as stable. The input-files for all simulations are [in.NVT_template](#) for the Noose-Hover thermostat and [in.NVEBer_template](#) for the "weak-coupling" Berendsen thermostat.

Lattice constant approximation

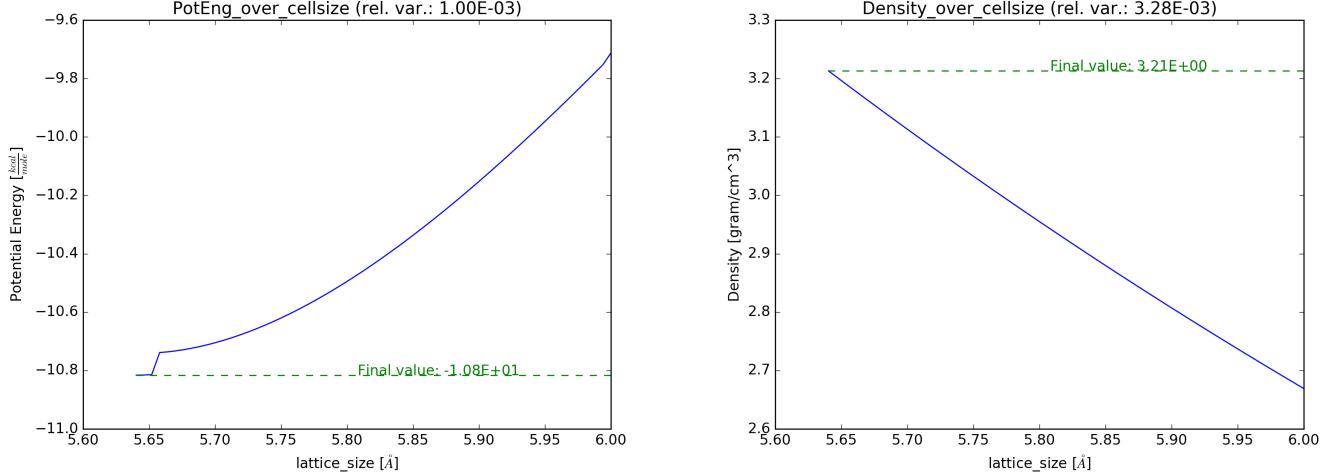


Figure 1.1: Convergence of Potential energy on the left, Final density value on the right. The final lattice-constant determined by the simulation is **5.640 Å**.

1.2.1 Equilibration

For the equilibration phase, the lattice constant of 5.90 Å, as states on the instruction, has been used. The results are plotted in Figure 1.2. All relative variances have been calculated after the system has reached equilibrium. The plots show, that the weakly coupled Berendsen thermostat capable of delivering the exact temperature value. However, while damping helps to reduce the fluctuation in Potential energy, it increases the variance of the temperature values. For the NVT thermostat on the other hand, damping can help to reduce fluctuations in all quantities of interest. From Figure 1.3 one can also see, that an increased time-damping parameter decreases the variance of the Total Energy results for a Berendsen algorithm, whereas the NVT ensemble can not benefit from that. Overall, the weakly-coupled Berendsen thermostat clearly performs superior compared to Nose-Hoover. This claim is also supported by the plots provided in Figure 1.5.

1.2.2 Stability comparison between Noose-Hoover and Berendsen

A comparison between the Nose-Hoover thermostat and the weakly-coupled Berendsen thermostat is provided in Figure 1.4. Overall, one can say that the weakly-coupled Berendsen delivers more accurate and stable results. Also, it does not benefit from damping, on the contrary, the relative variance increases with an increasing damping parameter.

1.2.3 Dependence on the system-size

To determine the dependence of the Nose-Hoover thermostat's convergence on the system size, simulations with a box-size of 5, 6, 7, 8, 9, 10, 11 and 12 have been carried out, which correlates to a total number of atoms of 500, 864, 1372, 2048, 2916, 4000, 5324 and 6912. The results are visualized in Figure 1.6. It can be observed, that the relative variance decays with the system size as $\frac{\langle T^2 \rangle - \langle T \rangle^2}{\langle T \rangle^2} \sim \frac{1}{N}$.

1.2.4 MCMC

The input-file for the MCMC simulations was `in.NVT_MCMC`, the simulations were run by `run_NVT_MCMC.sh`, results were postprocessed with `postprocess_MCMC.sh` and plotted via `plot_MCMC_vs_NVT.py`. The results are stored in `plots/MCMC_vs_NVT/`.

A visualization of the Density and Potential Energy results is provided in Figure 1.7.

MCMC converges a little bit faster, but other than that neither the relative variance, nor the final distribution obtained are different. For this reason, I conclude that in this setup, the performance of MCMC does not justify the significantly increased cost.

Difference between MCMC and MD

Molecular dynamics is based on the solution of simple ODEs of motion for every particle. A time integrator is applied to move the system from time t^n to t^{n+1} . The most costly part of MCMC is the evaluation of the force on a particle, since all neighbor interactions must be taken into account (up to a defined cutoff radius). Also, parallelization is limited, since neighboring regions have to continuously exchange information about the particles on the border. For MCMC, no neighbor evaluations are required. Instead, every particles position (momenta) is subsequently updated by computing the Markov chain. MCMC can be advantageous for very large systems and high dimensions, in the example here, it could not utilize its advantages.

1.3 Constant pressure/temperature molecular dynamics

1.3.1 Finding the equilibrium box sizes

To determine the equilibrium box sizes, a regular simulation has been conducted and the final, equilibrium average has been used as a mean value. This is visualized in Figure 1.8. By dividing the final box-length by the original, one can obtain a scaling factor that is applied to the following simulations via the "change_box" command ([in.NPT_boxmod_template](#)). As a verification, the box-size is again plotted over time in the right hand side of Figure 1.8.

In Figure 1.9, the pressure distribution of the scaled simulation is plotted for different box sizes. We can clearly see a normal-like distribution centered around the target value of 1.0 atm and a shrinking variance with increasing system size.

The dependency of the relative variance on the number of atoms is also plotted in the bottom two figures. Clearly, the relative variance decays inverse proportional with the system size.

From the LAMMPS manual, we can see that the pressure is computed via the formula

$$P = \frac{Nk_B T}{V} + \frac{\sum_i^N r_i \cdot f_i}{dV} \quad (1.8)$$

where N is the number of atoms, k_B is the Boltzmann constant, T is the Temperature and V is the control volume. The second term contains all interactions where r_i and f_i are the position vector of atom i and dV is the change rate of the control volume. Since we are in equilibrium, the control volume remains constant, the second term can thus be neglected. Also, in the first term, $\frac{N}{V}$ is related to the density and also constant. We can therefore expect the convergence to be of the same rate as the convergence of T which has already been shown in the first part to be of first order.

1.3.2 Evaluation of CV

Equilibration

First, the system has been equilibrated under NPT ensemble as required in the instructions. The plots, confirming that equilibrium has been reached are shown in Figure 1.11.

The input-file for this task was [in.CV_equi](#).

Estimation of box length

The equilibrium box-length estimation under NPT ensemble is visualized in Figure 1.12. The equilibrium box length value calculates to 601 Å. The input-file for this task was [in.CV_equi](#).

Equilibration under NVT-ensemble

From the previous simulations we obtain a lattice size of 54.64 \AA , which corresponds to a scaling factor of 9.26 w.r.t the original box length of 5.9 \AA .

This scaling factor has been used in the input file for the following simulations. The input-file for this task was [in.CV_NVT](#).

Estimation of specific heat capacity

The specific heat capacity $C_v = (\frac{\partial E}{\partial T})_{V,N}$ can, according to the instruction be written as a function of the total energy variance: $C_v = \frac{\langle E^2 \rangle - \langle E \rangle^2}{k_B * T^2}$.

This formula was used in the first part of [calc_CV.py](#) and gave an approximation of $C_v \approx 0.1549 \frac{\text{kJ}}{\text{kgK}}$. However, one can also get a good approximation of C_V by using a finite difference approach. I did a central difference around 300.15 K with a stencil width of 1 K and obtained a value of $C_{v,FD} \approx 0.1516 \frac{\text{kJ}}{\text{kgK}}$.

Both values are nearly identical and coincide perfectly with values from the literature.

The input-files for this task were [in.CV_NVT_plus](#), post-processing was done via [postprocess_NVT.sh](#) and the finite difference approximation was carried out via [calc_CV.py](#).

1.4 Phase transition of Krypton

For the phase transition analysis, simulations between 25 K and 230 K have been carried out with a temperature-step of 1 K . Each time the final configuration is written to a file and read by the subsequent simulation as a initial configuration.

The input-files are [in.phasetrans](#) for the first simulation and [in.phasetrans_template](#) for all following.

The results are post-processed via [postprocess_phasetrans.py](#) and plotted via [plot_phasetrans.py](#).

For the non-dimensionalization, the equilibrium box length is read from the log-file for each temperature.

Figure 1.13 show the equilibrated density as a function of the equilibrated temperature. Two phase transitions at $\sim 130 \text{ K}$ and $\sim 177 \text{ K}$.

The trajectories during the transitions are visualized in Figure 1.14.

Influence of the damping parameter on results - time convergence

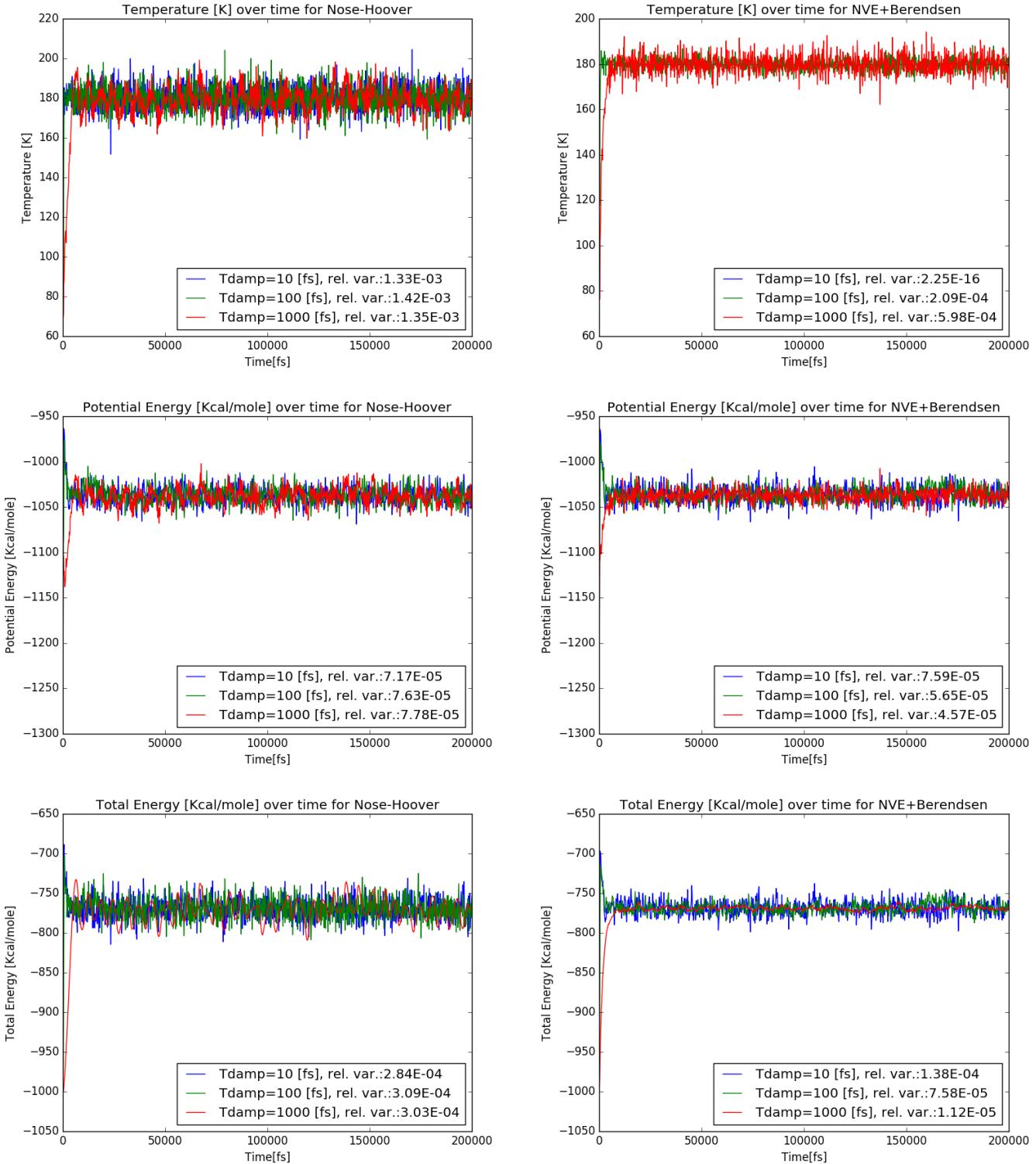


Figure 1.2: Equilibration of Temperature (top), Potential Energy (middle) and Kinetic Energy (bottom) for the Nose-Hoover thermostat (left) and the "weak-coupling" thermostat (right). The equilibration is plotted for different damping parameters and the relative variance of the equilibrated part is provided.

Influence of the damping parameter on results - histograms

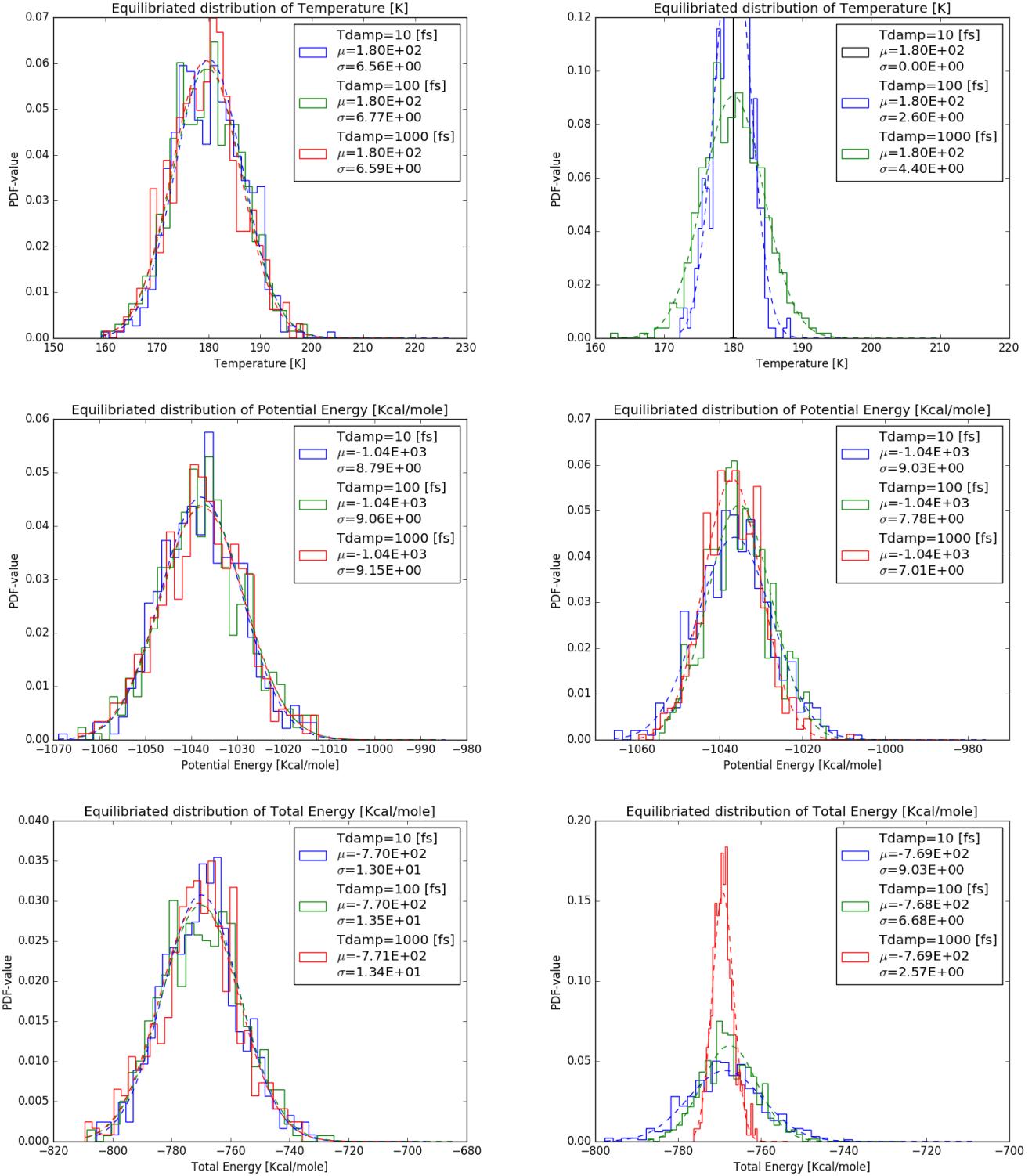


Figure 1.3: Histograms of the equilibrated Temperatures (top), Potential Energies (middle) and Total Energies (bottom) for the NVT ensemble (left) and the Berendsen thermostat (right).

Comparisons between Nose-Hoover thermostat and weakly-coupled Berendsen thermostat

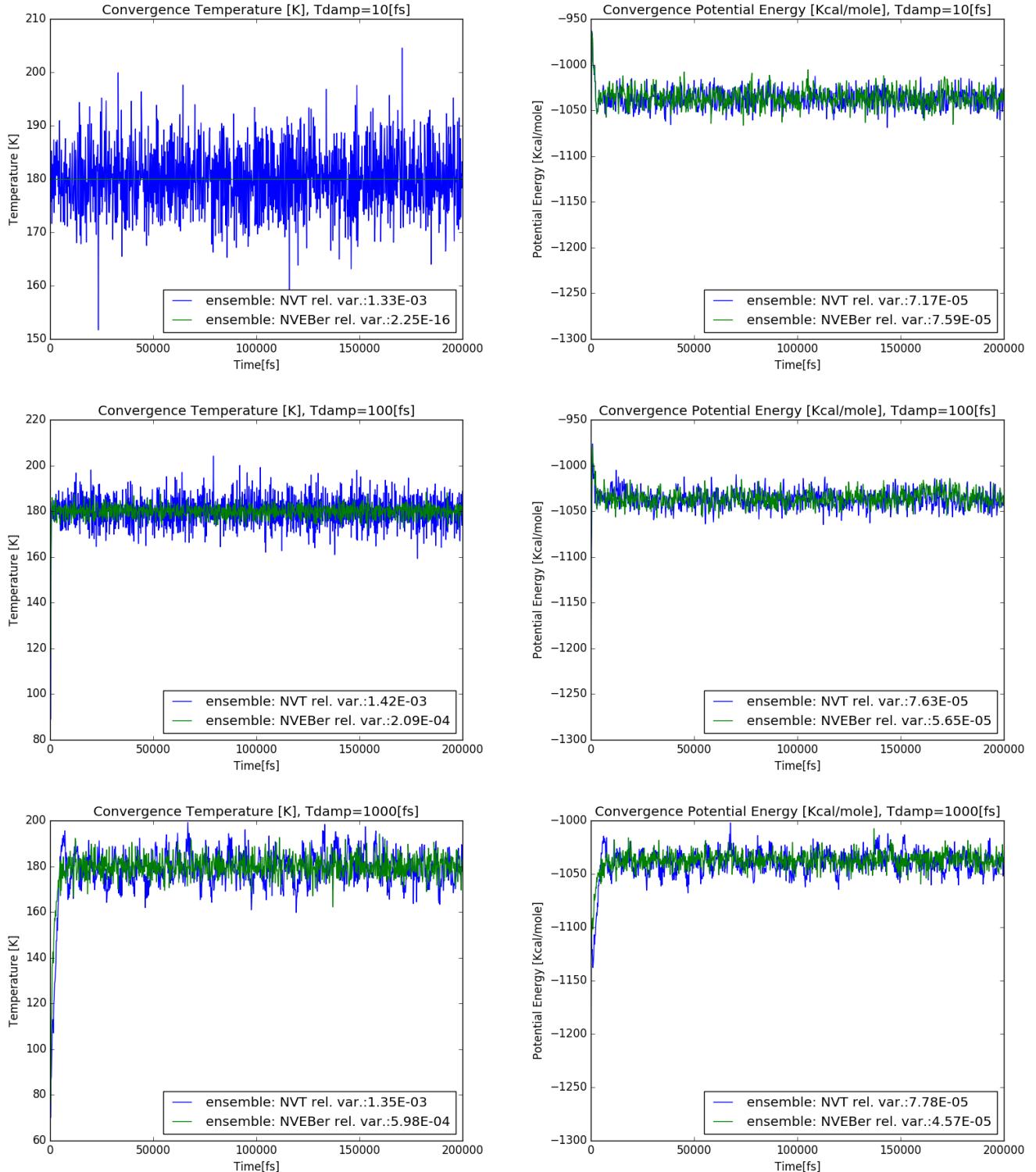


Figure 1.4: Comparisons between Nose-Hoover thermostat(blue) and weakly-coupled Berendsen(green) for temperature(left) and Potential Energy(right). The 3 rows show the convergence for 3 different damping parameters.

Comparisons between Nose-Hoover thermostat and weakly-coupled Berendsen thermostat

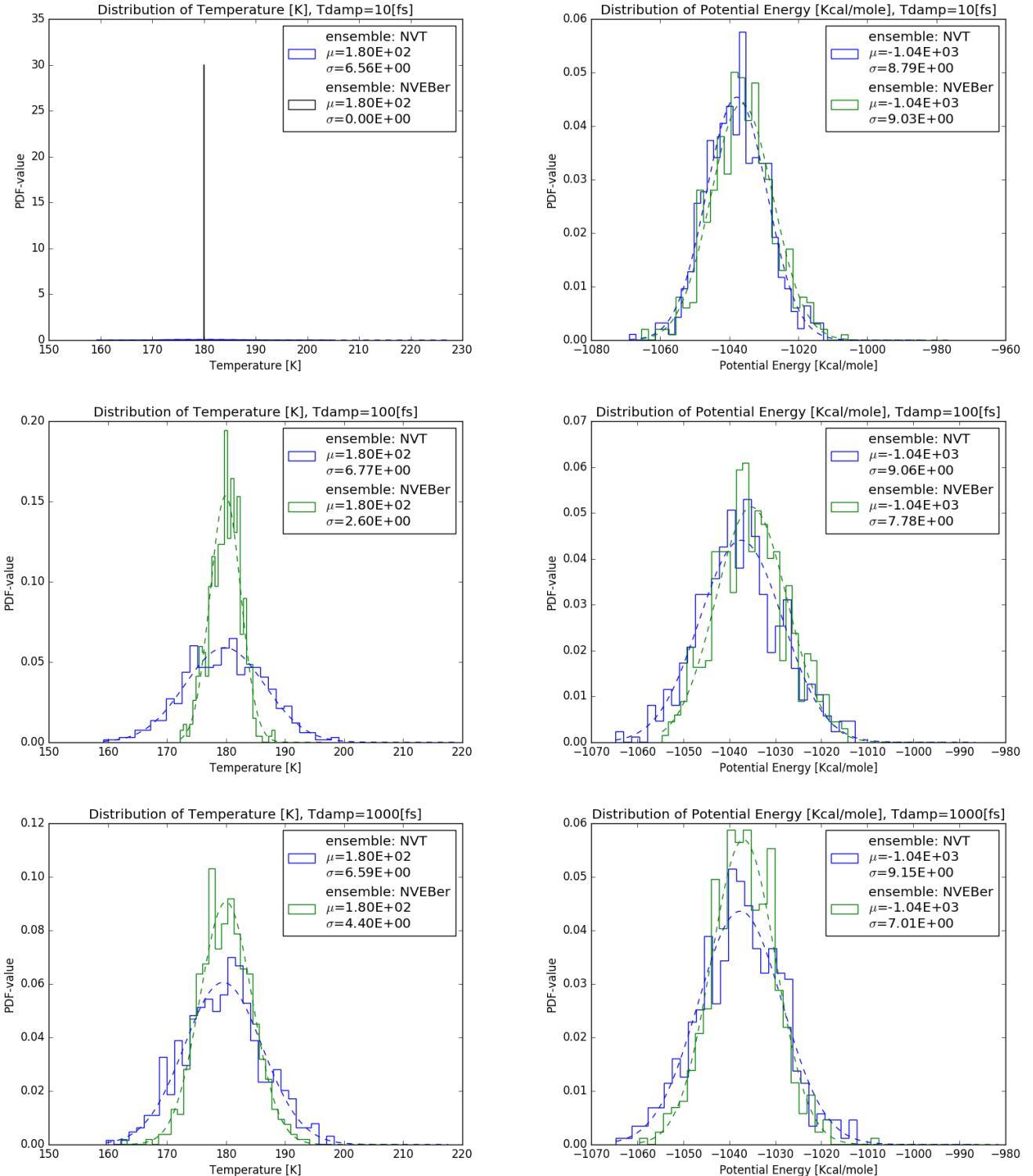


Figure 1.5: Distributions of Temperature(left) and Potential Energy(right) for different damping values(increasing from top to bottom) for the Nose-Hoover thermostat(blue) and the weakly-coupled Berendsen thermostat (blue). One can conclude, that the weakly coupled Berendsen thermostat leads to better results, especially for the Temperature it can achieve a significantly reduced standard deviation, provided that the timd damping parameter is not too big.

Influence of the atom number on the results quality

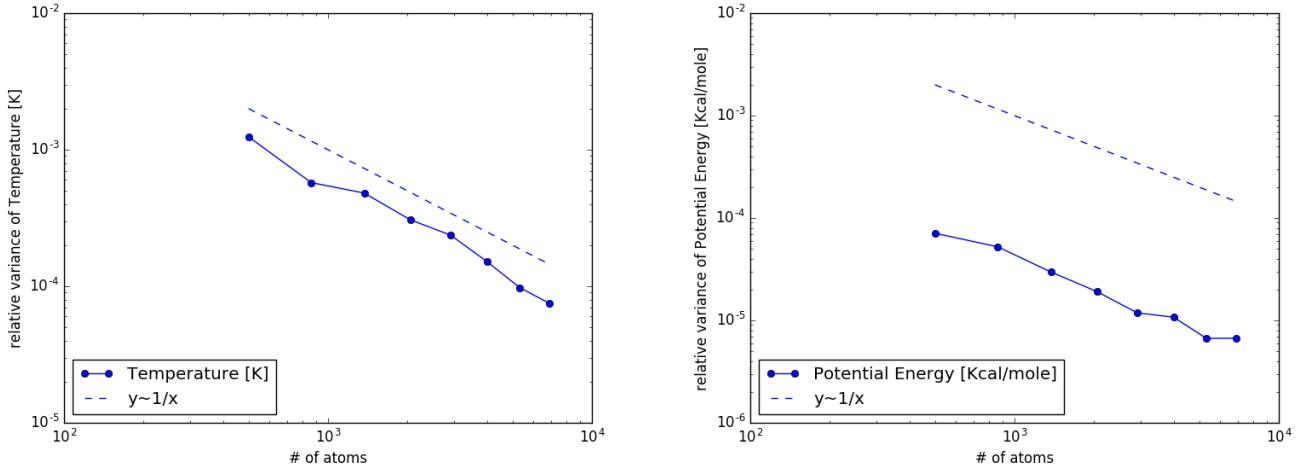


Figure 1.6: Relative variance of Temperature(left) and Potential Energy(right) with respect to the number of atoms in the simulation. As the dashed line shows, the relative variance decays with the system size as $\frac{\langle T^2 \rangle - \langle T \rangle^2}{\langle T \rangle^2} \sim \frac{1}{N}$.

Comparison between MCMC and NVT-MD

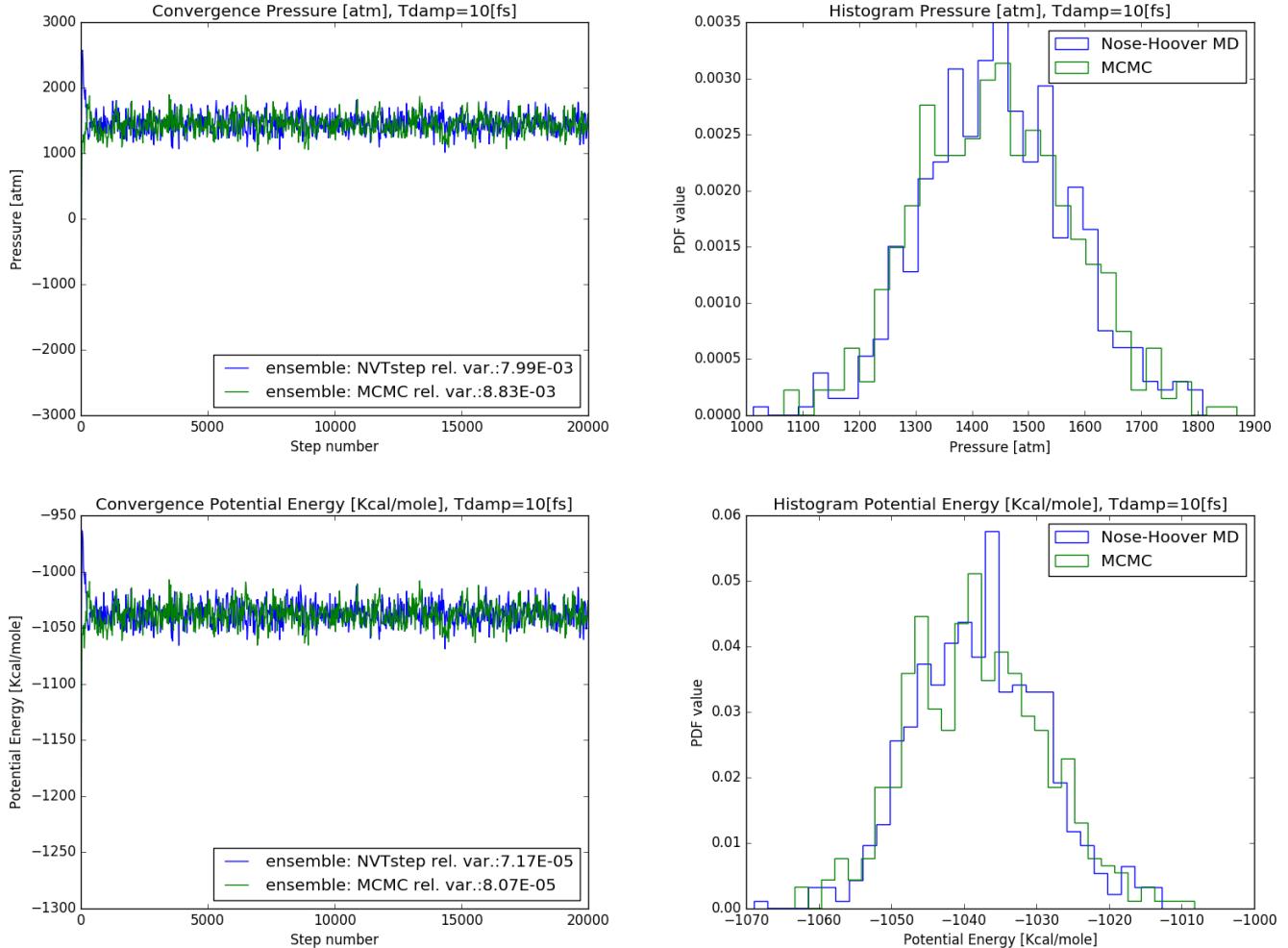


Figure 1.7: Comparison between MCMC and NVT-MD. Both algorithms deliver very similar results. MCMC converges a little bit faster, but other than that neither the relative variance, nor the final distribution obtained are different.

Determination of equilibrated box-size

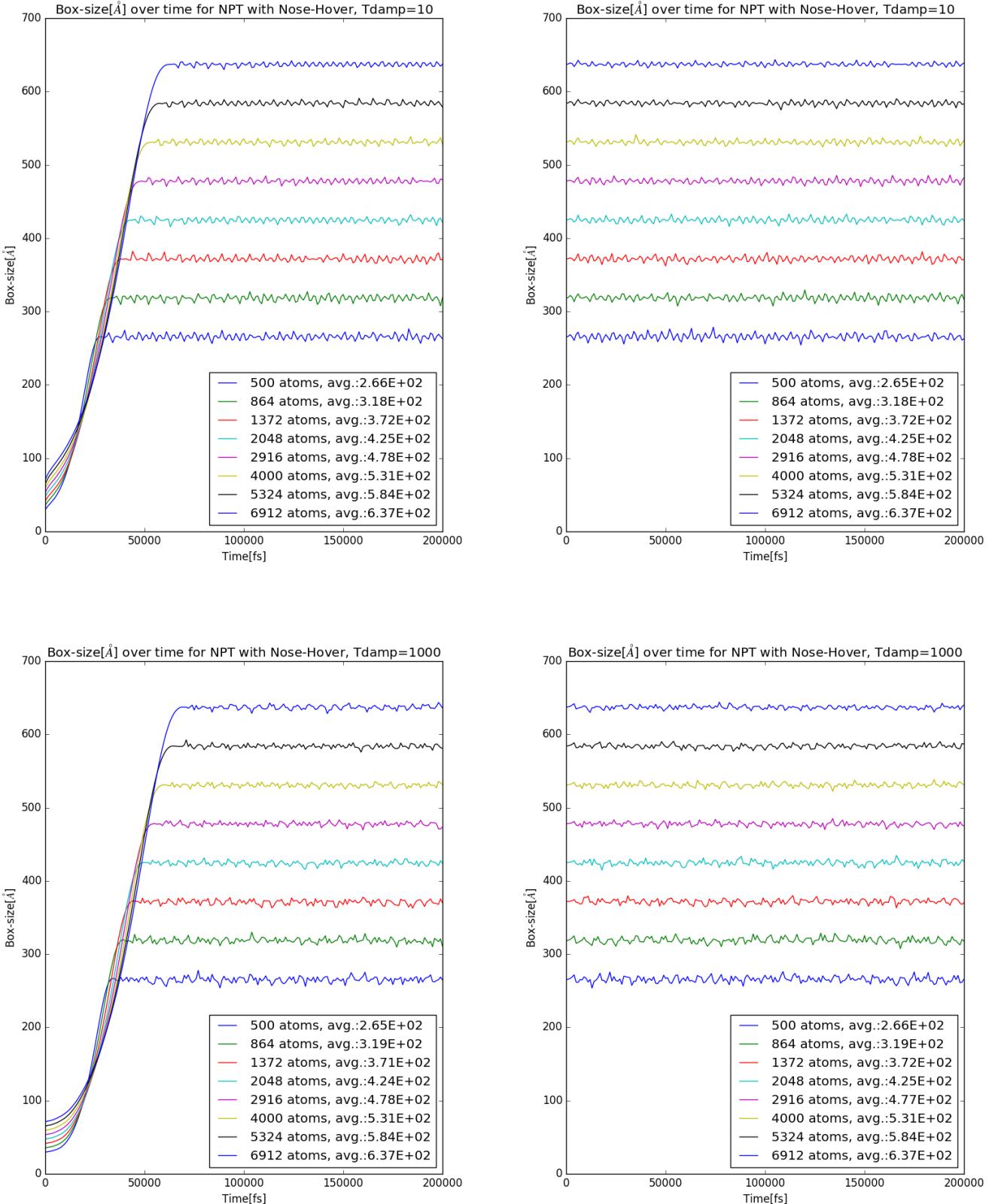


Figure 1.8: Determination of the box-size by equilibration(left) and Time convergence after box-size modification (right).

Influence of the system size on the relative variance of pressure results

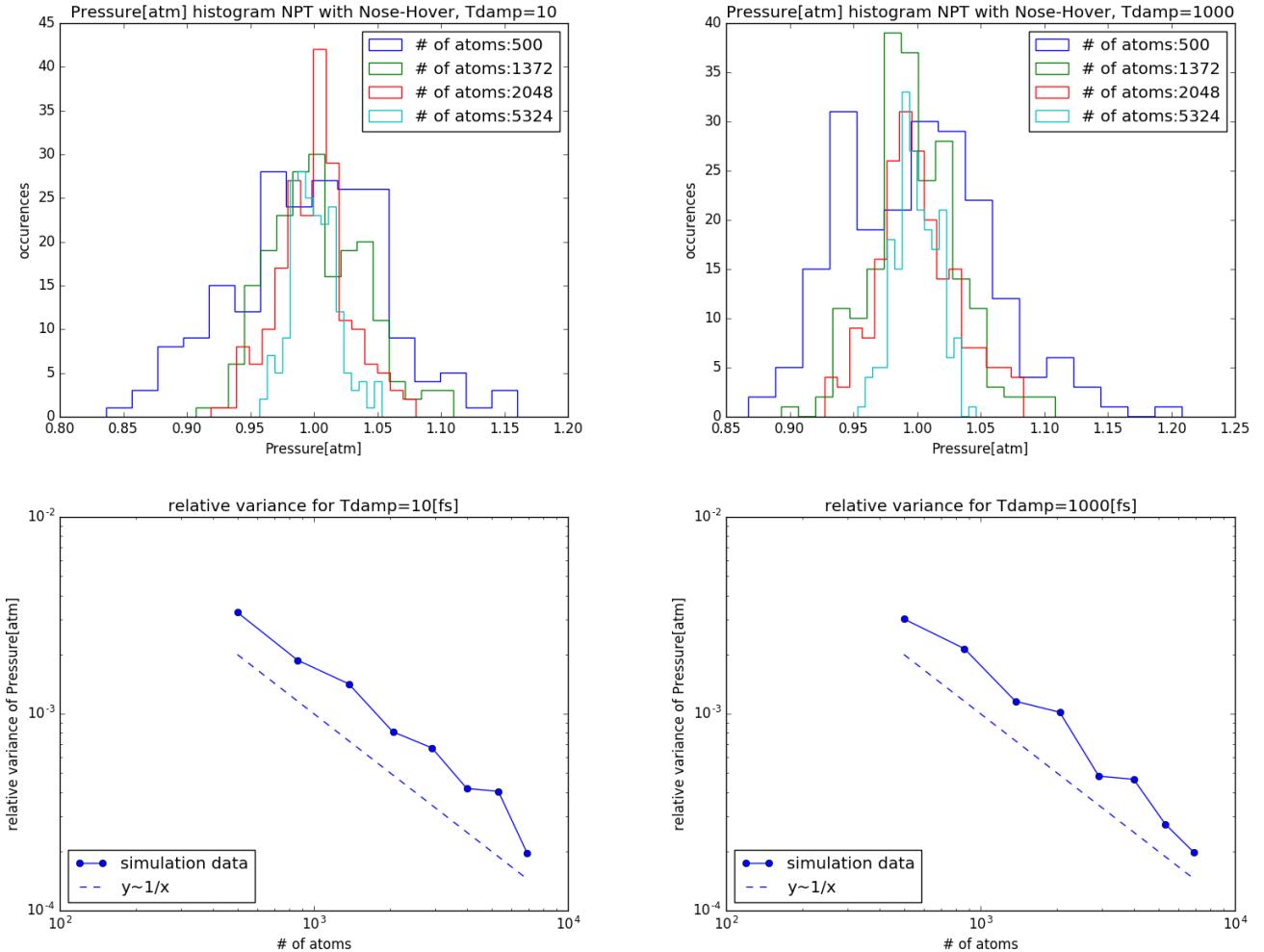


Figure 1.9: Influence of the system size on the simulation. The top two figures show histograms of the obtained pressure values, the bottom two figures show the dependency of the relative variance on the system size. The left two have been calculated with a damping value of 10 fs the right two with 1000 fs .

Density results over box-size

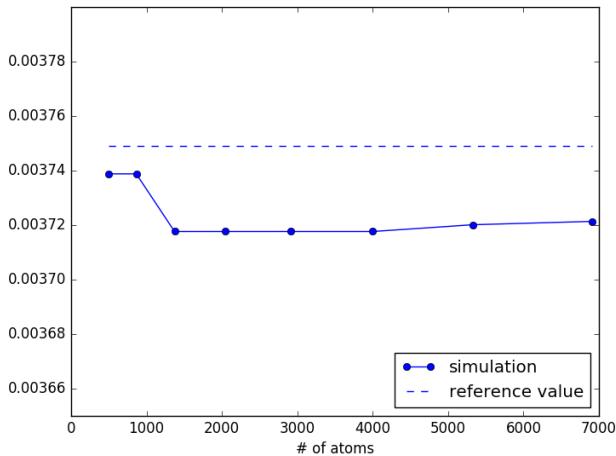


Figure 1.10: Density results as a function of the simulation box size. Clearly, the reference density is very well approximated. One should also take into account, that the 275.15 K that the instruction requires is slightly higher than the $\circ C$ that the provided reference value is based upon.

equilibration under NPT ensemble

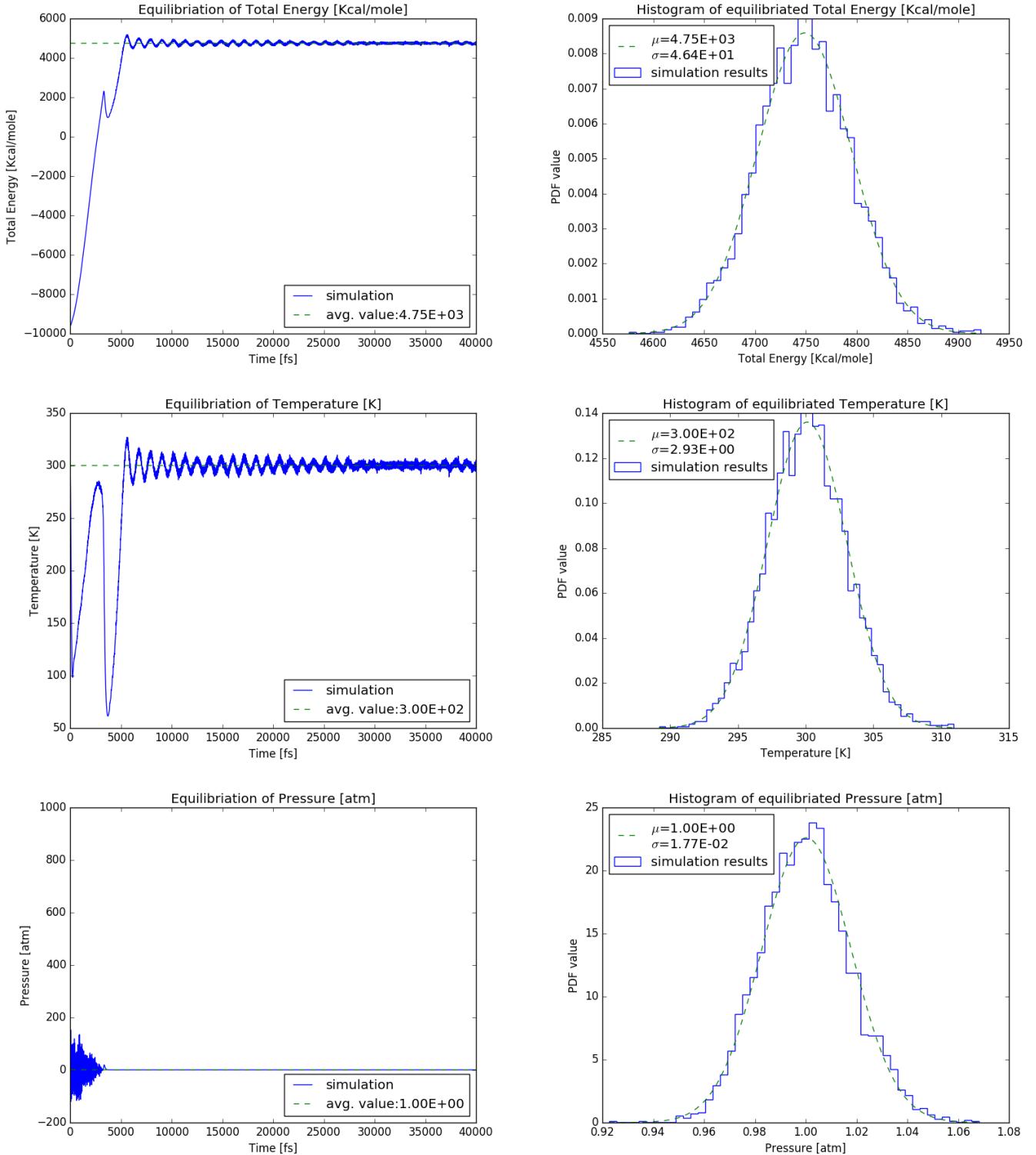


Figure 1.11: Equilibration of the system under NPT-Ensemble. The left part shows the development of thermodynamic quantities over time, the right figures show the distribution of those quantities after equilibration has been reached. Both the target Temperature and target Pressure have been reached with an acceptable standard deviation.

Equilibrium box-length estimation under NPT-ensemble

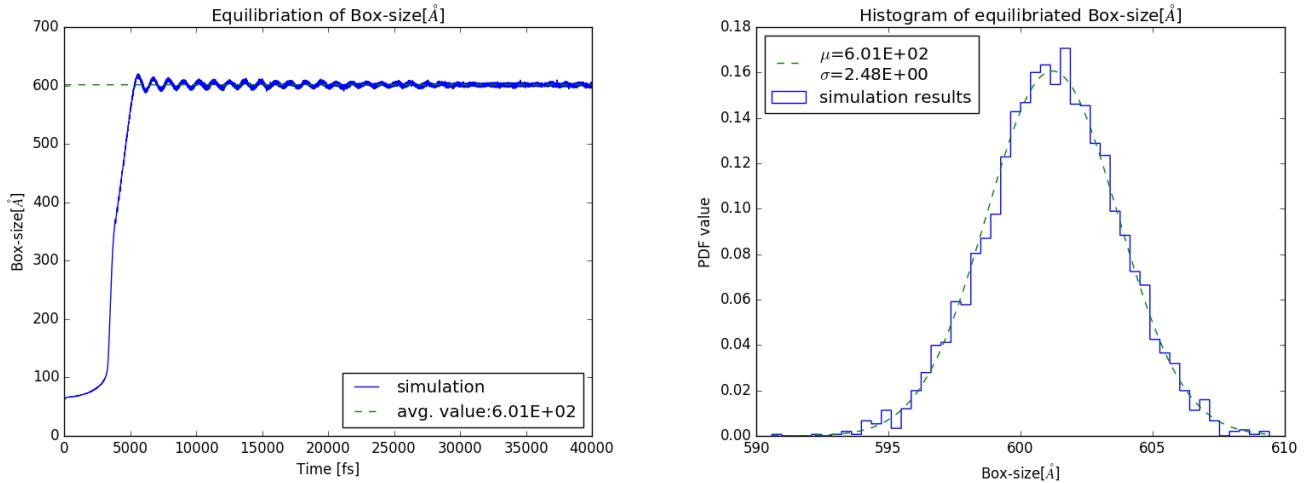


Figure 1.12: Estimation of the equilibrium Box length under the NPT ensemble. An equilibrated value of 601 \AA is obtained with a standard deviation of 2.48 \AA .

Phase transition of Krypton under NPT ensemble

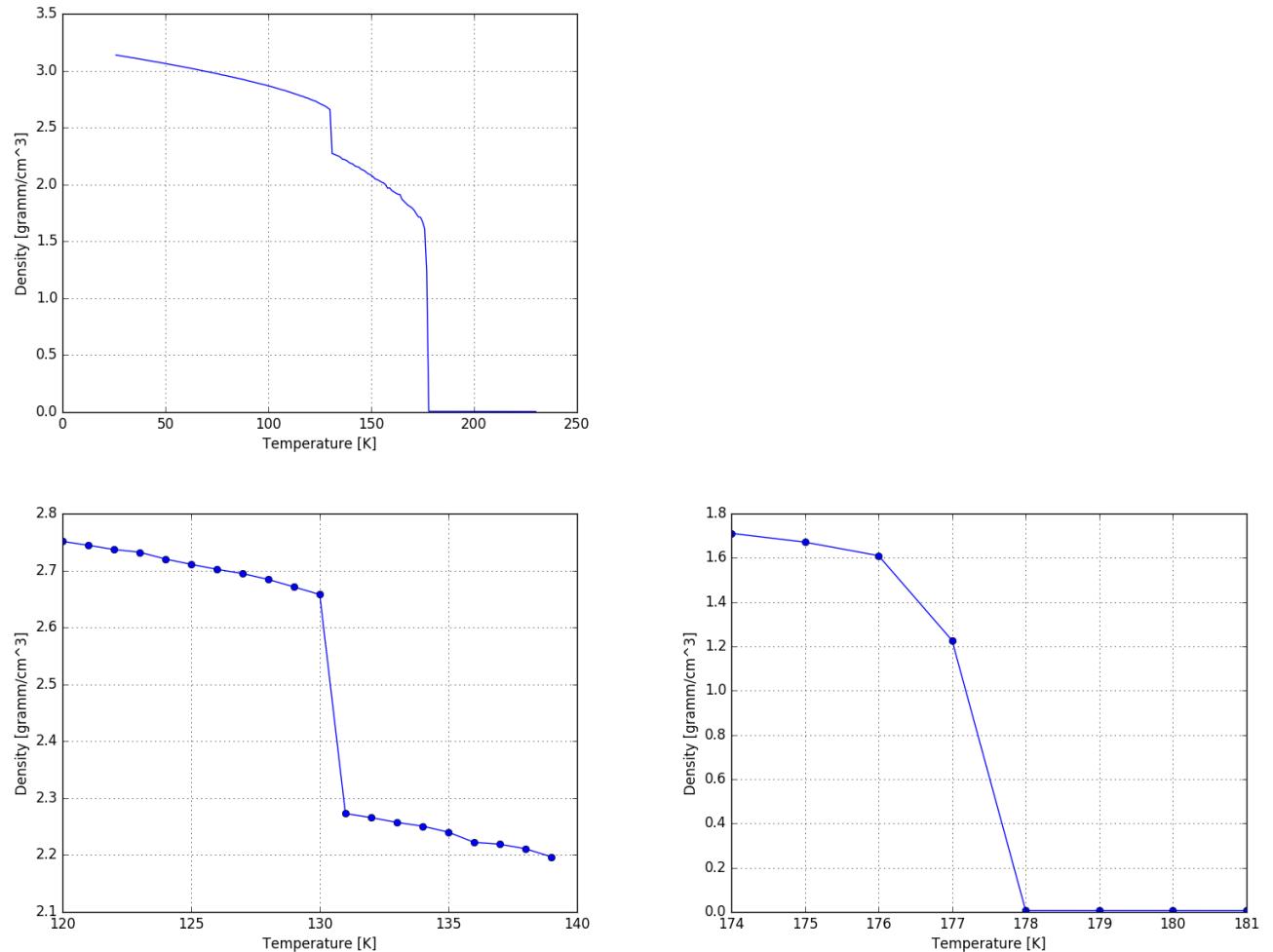


Figure 1.13: Phase transition analysis of Krypton in NPT ensemble. Simulations have been carried out every 1 K . As the first graph shows, we get 2 distinct phase transitions. The two graphs below show the transitions in greater detail. One can conclude, that the melting occurs at $\sim 130 \text{ K}$, and Boiling at $\sim 177 \text{ K}$.

Visualization of the trajectories during the phase transitions

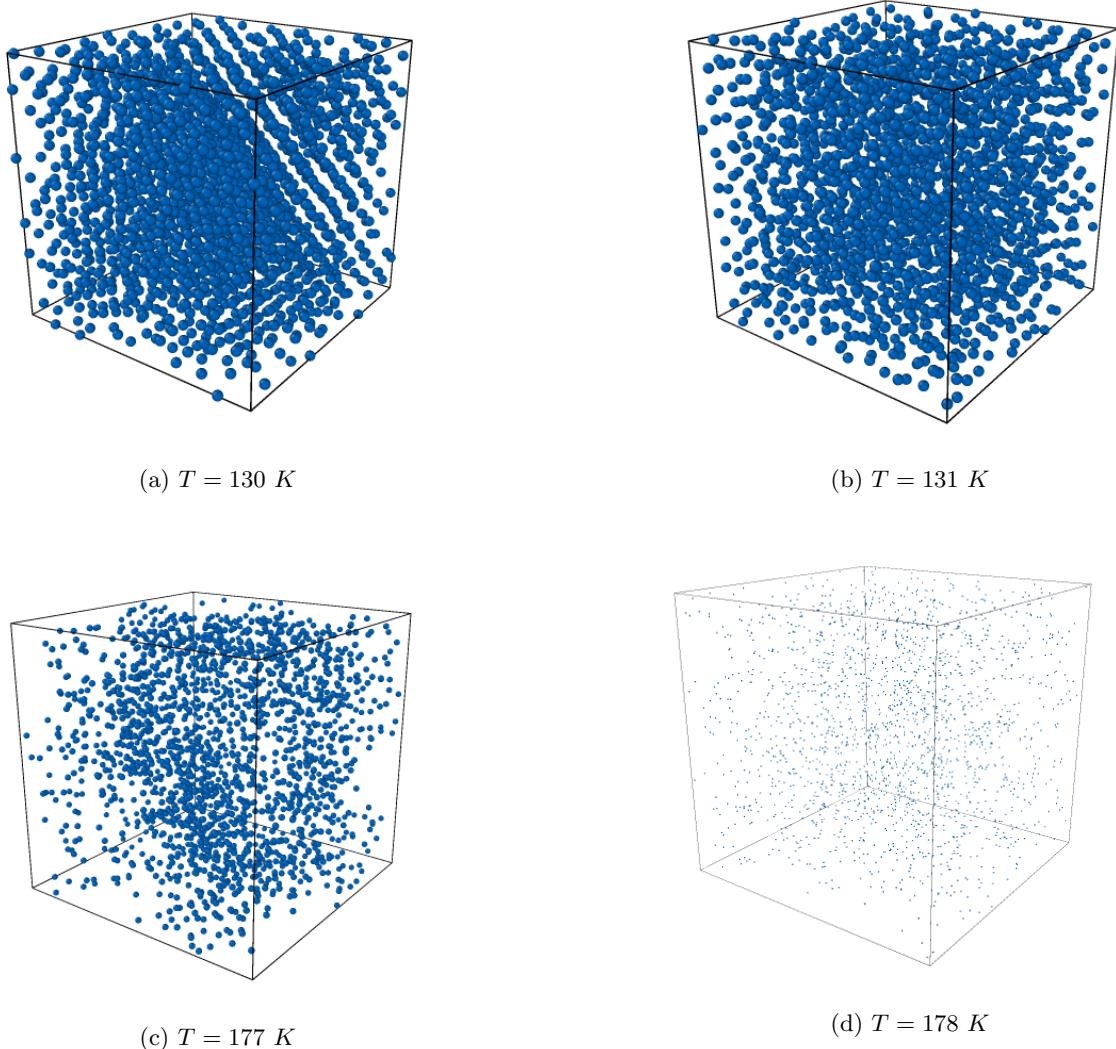


Figure 1.14: Visualization of the trajectories during the phase transitions at four different Temperatures. The top two figures show the melting process around 130 K , the bottom two figures show the boiling around 177 K . Please note that the box-size are not up to scale from one figure to another. Instead, one can use the atom size as a reference for the total system size.

Chapter 2

Two-dimensional atomistic tensile test

All files for this task are located in [2_Two-dimensional_atomic_tensile_test](#)

2.1 Equilibration

First, a system of $30 \times 30 \times 1$ cells has been set up and equilibrated under Nose-barostat. Several different computations of T_{damp} and p_{damp} have been used to determine the influence of both. The input file was [in.equi_template](#). The influence of the damping parameters on the equilibration of the Potential Energy is visualized in Figure 2.1. First of all, if the damping parameter, particularly the pressure damping parameter, is chosen too low, the simulation becomes unstable and crashes. This was e.g. the case for $p_{damp} = 10 \text{ fs}$.

What is more, the graph shows, that the choice of T_{damp} effects the convergence properties. The value should not be chosen too small. The choice of p_{damp} on the other hand, seems not to influence the convergence of the potential energy greatly, as long as it is chosen such that the simulation is stable.

Additionally, the convergence of the system Temperature w.r.t the damping parameters is provided in Figure 2.1. One can observe, that a too big choice of T_{damp} leads to spurious oscillations at the beginning of the simulations and periodic oscillations even in the equilibrated regime.

Based on Figures 2.1 and 2.2 the following choice is made: $T_{damp} = 100 \text{ fs}$, $p_{damp} = 100 \text{ fs}$. The equilibrium configuration is plotted in Figure 2.3.

2.2 Tensile test

Based on the considerations before, the tensile test was conducted with damping parameters of $T_{damp} = 100 \text{ fd}$ and $p_{damp} = 100 \text{ fs}$.

The input file for the tensile-test was [in.tensiletest](#), the results were plotted via [plot_tensiletest.py](#).

The results of the tensile test are visualized in Figure 2.4.

Determination of appropriate damping parameters - Potential Energy

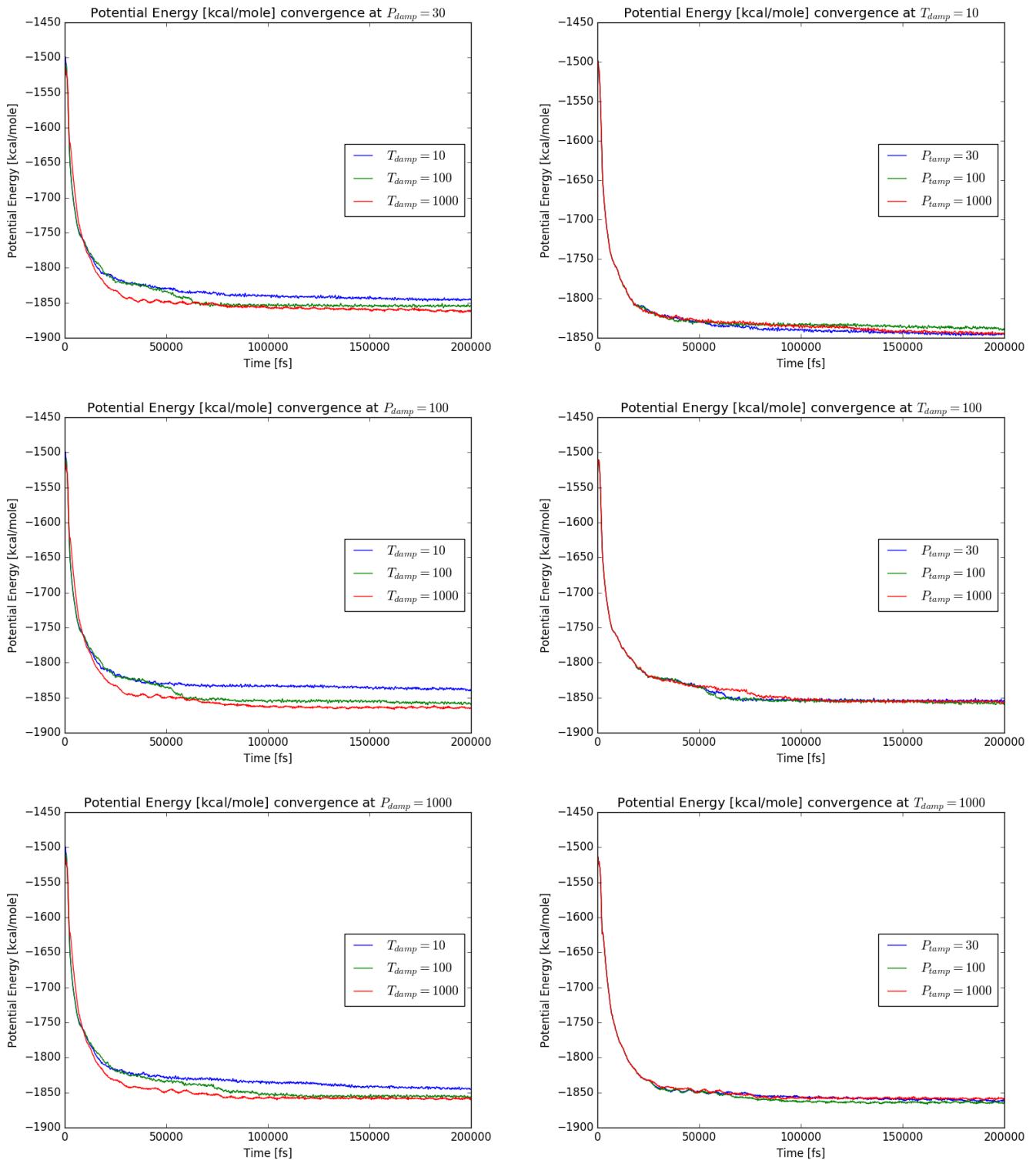


Figure 2.1: Influence of the damping parameters on the convergence of the potential energy. The plots on the left show a variation of T_{damp} under constant values of p_{damp} , the plots on the right show a variation of p_{damp} under constant values of T_{damp} .

Determination of appropriate damping parameters - Temperature

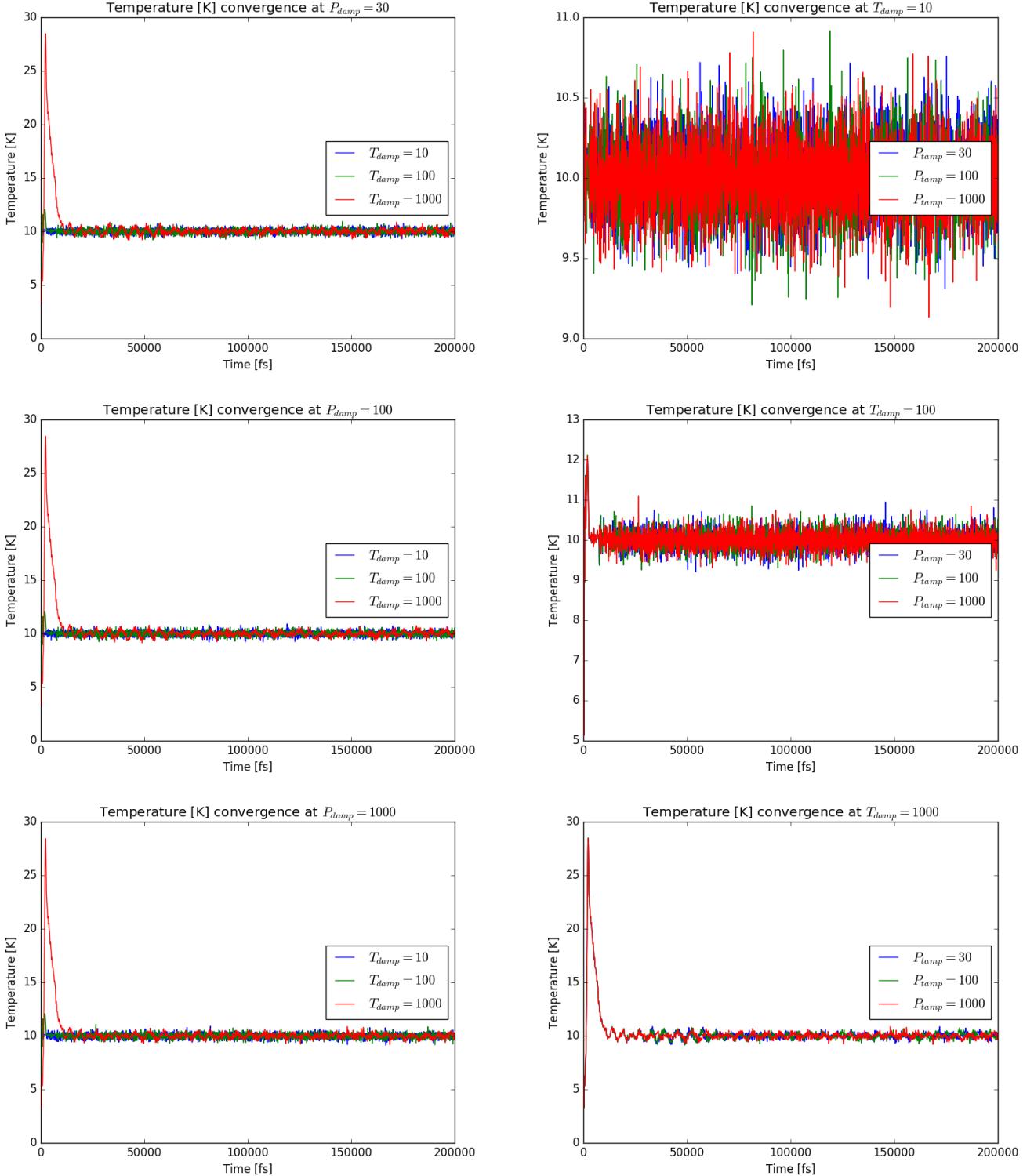


Figure 2.2: Influence of the damping parameters on the convergence of the system temperature. The plots on the left show a variation of T_{damp} under constant values of p_{damp} , the plots on the right show a variation of p_{damp} under constant values of T_{damp} .

System snapshots of the 2D system during equilibration

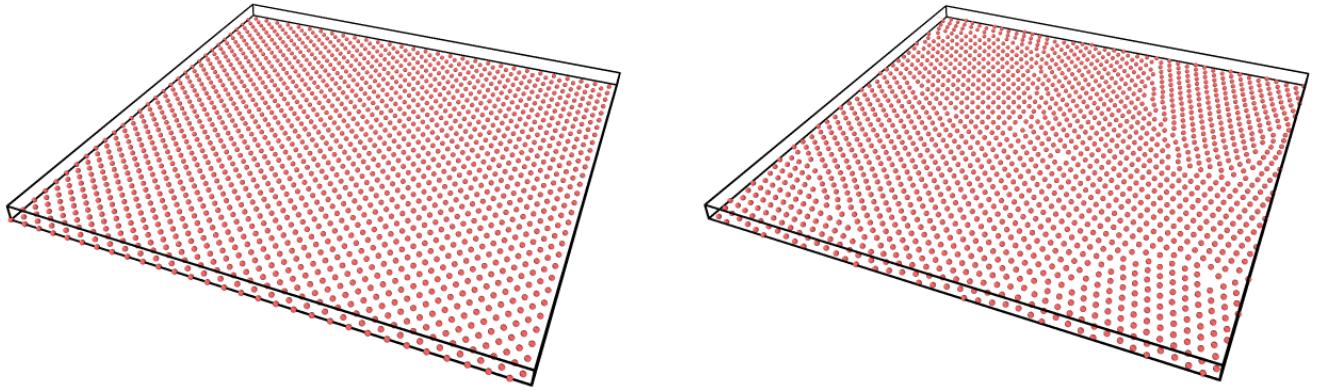


Figure 2.3: Initial(left) and equilibrated configuration(right) of the 2D-Krypton system. The atom radius is up to scale.

Stress-strain relations

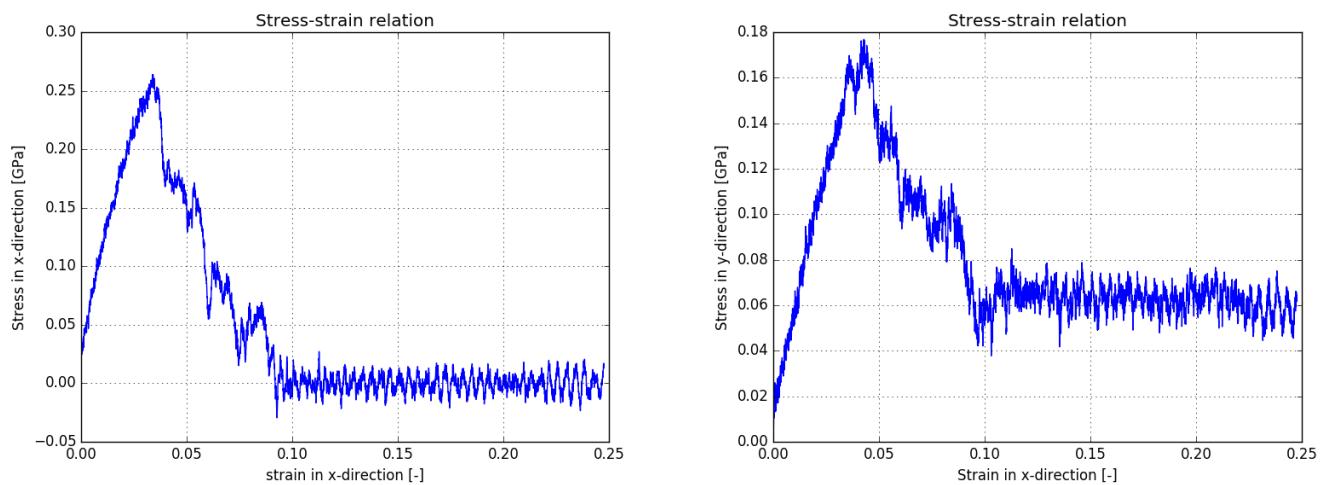


Figure 2.4: Stress in x-direction(left) and y-direction(right) as a function of the strain in x-direction. A clear point of material failure can be observed

Visualization of the trajectories during stress-strain simulation

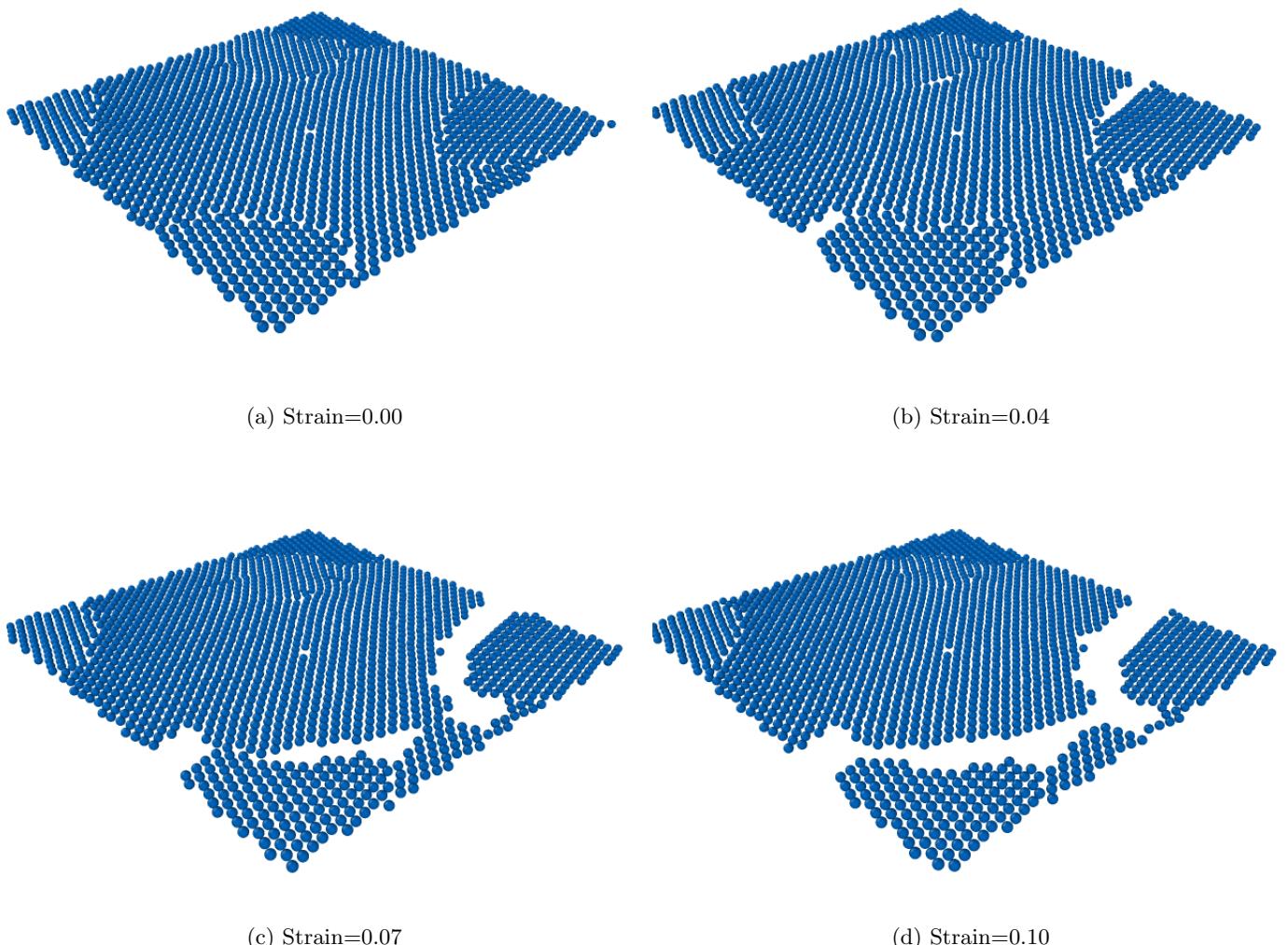


Figure 2.5: The figure shows 4 snapshots of the system for 4 different strain values. All snapshots show equilibrated systems. The visualization coincides perfectly with Figure 2.4, that suggests a critical strain of 0.035.