

COMPUTATIONAL PHYSICS - EXERCISE SHEET 05

Nose-Hoover Thermostat

Matteo Garbellini*
Department of Physics, University of Freiburg

(Dated: June 7, 2021)

The following is the report for the Exercise Sheet 05. The goal of this exercise is to simulate a Lennard-Jones fluid using the Nose-Hoover NVT thermostat, and to study significant thermostastical quantities. Along this report, python scripts were also handed in. Additional code can be found at the github repository given at the end of this page. **As of today the code does not work properly, and shows unrealistic-unphysical results.**

I. CODE IMPLEMENTATION

The new code implements the Nose-Hoover thermostat, which include a Velocity-Verlet type of integrator for position, velocities and the additional thermostat quantities. Moreover a routine for computing the specific heat was implemented, together with a statistical routine for calculating the running average of a given quantity.

Additionally a new routine for generating random velocities directly distributed over a Maxwell-Boltzmann distribution was implemented. Although not necessary, some rescaling iterations are still performed.

II. PROBLEMS WITH THE CODE IMPLEMENTATION

Beginning from last week when running the code I noticed some inconsistent results, in particular related to the difference in magnitude between kinetic energy and potential energy. The resulting values of the specific heat using σ_E^2 and σ_U^2 were different by many order of magnitude. Indeed, by outputting the variance results, I discovered that the variance would increase over time, with peaks of $\sigma_E^2 \approx 10^6$.

A. Issues

In order to compare the results with some of the fellow students I decided to switch back to real units – considering that so far I had been using reduced units for convenience. This made the comparison much easier and gave an intuition of the possible errors. After some time of debugging I found the following major issues:

- The temperature after the first step would increase to incredibly high values $> 10^{44}$ in very little time
- The energy was off and the kinetic energy, after the first few iterations would remain constants, which was not a physical result.
- The velocity rescaling at the first step had some issues, resulting in a kinetic energy of $\approx 10^{-19}$ before and $\approx 10^{23}$ after the rescaling.
- Unrealistic specific heat values $C_v \approx 10^{200}$ (even after some fixes, see plot)

B. Fixes

So far I believe I have been able to fix some of the major issues, in particular I found the following bugs:

- The velocity update would add twice the new velocity at each time step, fault of a wrong +=
- Somehow the velocity rescaling would compute the temperature in the wrong way, leading to inconsistent results

* matteo.garbellini@studenti.unimi.it; <https://github.com/mgarbellini/Computational-Physics-Material-Science>

C. Unresolved yet

The following are some of the unresolved issues so far:

- Specific heat shows unphysical results
- Running average used with specific heat shows some error of the type *RuntimeWarning: overflow in true-divide*
- Outputting $\xi(t)$ the order of magnitude does not seem correct

D. Comments

By outputting the temperatures at each timestep it seems that the new implementation now produces physical meaningful results. This however took me almost a week of debugging, therefore I was not able to finish the simulations necessary for this exercise sheet. In the next section I will provide some preliminary results. Even though the temperature shows realistic results, the specific heat still does not show correct results. **Overall the code does not work properly yet.**

Additionally, it is to be noted that the issues rose from a clean-up and rewrite of some part of the code. Therefore I think that the results obtained in previous exercise sheets should be correct, although some minor inconsistency could also be explained – in particular some of the major fluctuations found in the density profiles.

III. SOME PRELIMINARY RESULTS

The following are some preliminary results showing that some of the code fixes were successfull.

A. Temperature

The temperature seems to be correct. The figure shows a simulation run with $N = 6^3$, with 2000 equilibration iterations at $T=300\text{k}$, 2000 and 2000 production iterations for $T=300\text{K}$ and $T=100\text{K}$, respectively.

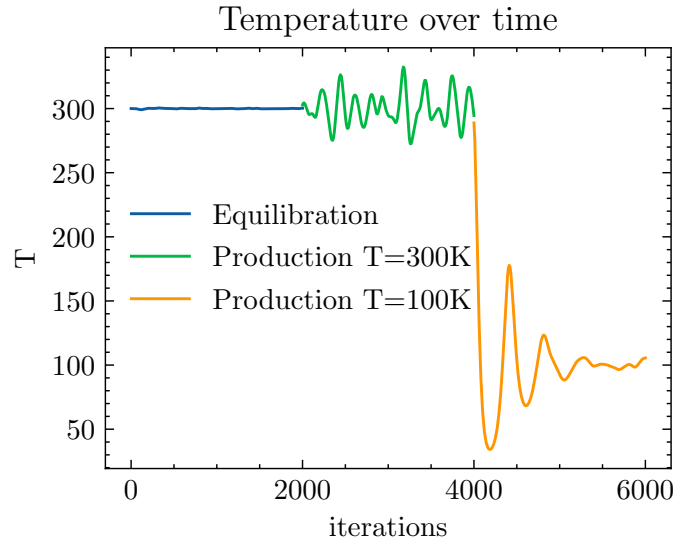


FIG. 1. The figure shows the temperature over time

B. Specific Heat

The specific heat still shows some unrealistic results. Additional debugging is needed. As can be seen from the figure, the specific heat computed with σ_E^2 does not even show in the range of the one with σ_U^2 .

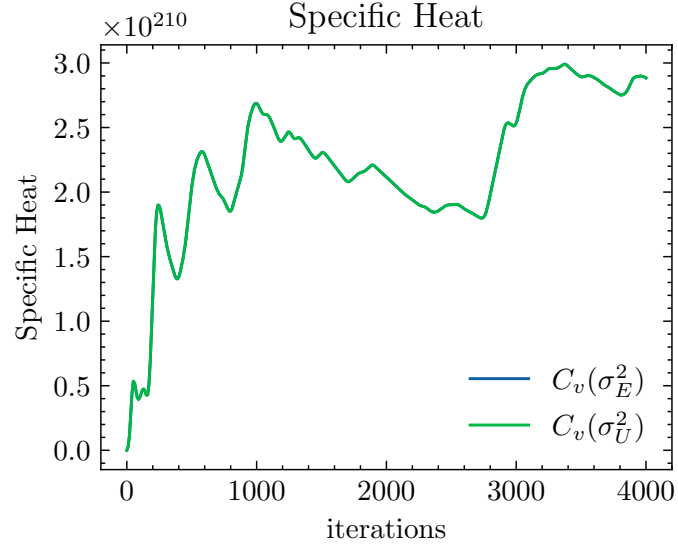


FIG. 2. The figure shows the specific heat over time

C. Plot of $\xi(t)$

The results for $\xi(t)$ seem quite unrealist as well, given the enormous fluctuations.

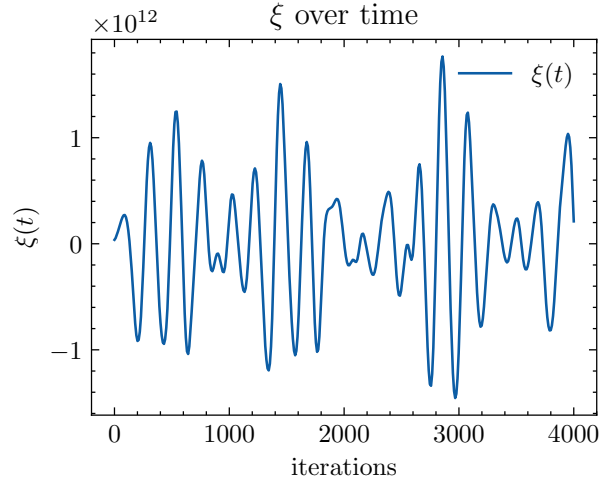


FIG. 3. The figure shows the value of $\xi(t)$ over time