MONTE CARLO AND MOLECULAR DYNAMICS SIMULATION

WILLIAM JANKE[1] & ANDREAS HALLER[2]

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Twelve years before the start of the series, the Nine-Tailed Demon Fox attacked Konohagakure destroying much of the village and taking many lives. The leader of the village, the Fourth Hokage sacrificed his life to seal the Nine-Tails into a newborn, Naruto Uzumaki. Orphaned by the attack, Naruto was shunned by the villagers, who out of fear and anger, viewed him as the Nine-Tails itself. Though the Third Hokage outlawed speaking about anything related to the Nine-Tails, the children — taking their cues from their parents — inherited the same animosity towards Naruto. In his

thirst to be acknowledged, Naruto vowed he would one day become the greatest Hokage the village had ever seen.

[1] *wjanke@students.uni-mainz.de*
[2] *ahaller@students.uni-mainz.de*

1 INTRODUCTION

Many body problems often cannot be resolved analytically because of their huge amount of accessible states, for an instance. In order to solve such a problem with computational assistance, a variety of methods have been developed and all of them have their very own applications and limits. One of the first models is the Monte Carlo simulation with the Metropolis Criterion. Its nature is pure stochastic - the time progression evolves randomly and is not given a set of initial conditions. However, the Molecular Dynamics approach is used for the same initial System - with equal coordinate initialisation and boundary conditions - by solving Newton's equations of motions to all atoms simultaniously. This implies that this kind of simulations are deterministic and can be used to look into a system's time evolution.

The main goal of this work is an implementation of both methods in C++ for a set of noninteracting particles in a finite box with periodic boundary conditions and Lennard-Jones Potential. In order to deepen the similarities and differences in computational and physical aspects, a comparison at the very end is suffitient.

2 SYSTEM CONSIDERATIONS

If not explicitly mentioned, the system's settings are as follows:

Ensemble

N particles in a box of volume V and temperature T are considered. The closed box is placed in an external heat bath, hence the total energy is not fixed and the probability P_i for a given State $|i\rangle$ with Energy E_i and $\beta := \frac{1}{kT}$ is given by

$$P_i = \frac{1}{Z}\exp\left(-\beta E_i\right), \text{ and } Z = \sum_k \exp\left(-\beta E_k\right). \tag{1}$$

Such an ensemble is called canonical or NVT.

Potential and Energy

The particles are interacting via a normed Lennard-Jones Potential

$$V_{ij} = 4\left(\frac{1}{r_{ij}^{12}} - \frac{1}{r_{ij}^{6}} + \frac{2^7-1}{2^{14}}\right), \tag{2}$$

such that $V_{ij} = 0$, if $r_{ij} = 2\sqrt[6]{2}$. The total energy E_n of the system in a state $|n\rangle$ is given by the expression

$$E_n = \sum_i \sum_{j \neq i} V_{ij}. \tag{3}$$

Periodic Boundary Conditions

For both implementations, periodic boundary conditions (PBC) will be used. With PBC there are two important consequences for our System.

First, when a particle reaches the border of the System, it is not reflected back, but transfered to the other side of the box.

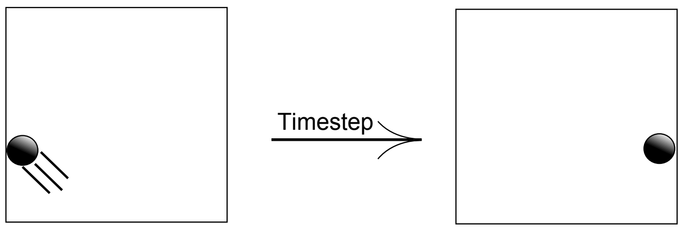

Figure 1: A particle in a box with PBC jumps from one side to the other.

Second, particles can interact with images of other particles behind the border of the box. As a result, the distance vector between two particles is not definite.

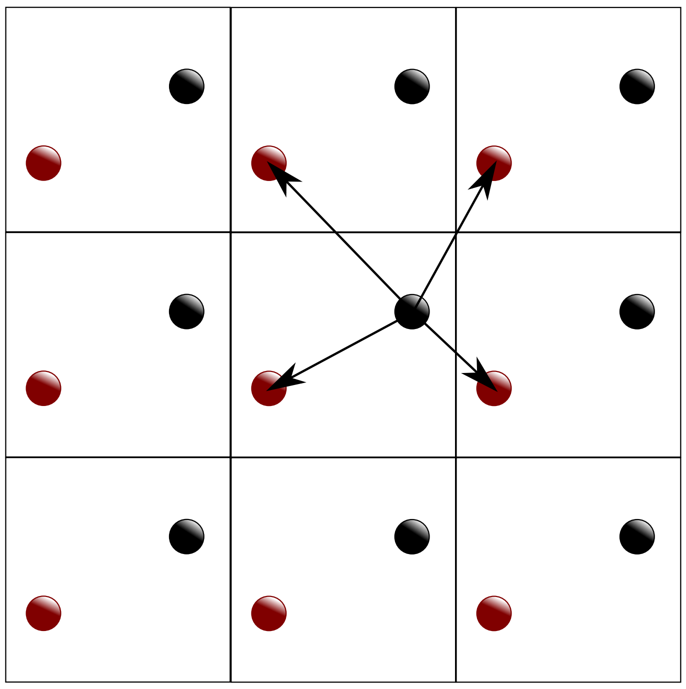

Figure 2: In the middle is the main simulation box with two particles being simulated, the surounding eight boxes are periodic images of the main box. The distance between the two particles is not definite with PBC because there are multiple ways to get from one particle to the other.

Of course a particle doesn't interact with all the images of one and the same particle, thats why the minimum image convention is used, which means a particle only interacts with the closest image of another particle. In figure (2) that would be the image to the right middle.

3 MONTE CARLO SIMULATION

It has been already mentioned, that MC approaches can be used to research many body systems which are not solvable analytically. Numerical methods simplify this task, but they introduce other challenges - in most cases (especially for large scaling sizes), it is not possible to acces every single state. However, for the explanation of the used algorithm, this can be neglected for now.

3.1 Detailed Balance and the Metropolis Algorithm

Consider the particle flow out of state $|i\rangle$

$$j_i^{\text{out}} := \sum_j P_i(t) W_{ij}, \tag{4}$$

with jump probability W_{ij} from state $|i\rangle$ to $|j\rangle$. Analogue, the particle flow into state $|i\rangle$ can be written as

$$j_i^{\text{in}} := \sum_j P_j(t) W_{ji}. \tag{5}$$

This yields to the master equation

$$\frac{dP_i(t)}{dt} = \sum_j \left(P_j(t) W_{ji} - P_i(t) W_{ij} \right), \tag{6}$$

which is fulfilled in the equilibrium (left hand side vanishs) by the stricter condition of detailed balance[1]

$$P_i W_{ij} = P_j W_{ji}. \tag{7}$$

In other words - the system moves towards equilibrium if detailed balance is enforced[2]. The key of success is to choose W_{ij} such that detailed balance is fulfilled. This is accomplished with the Metropolis Criterion

$$W_{ij} = \begin{cases} \exp(-\beta\Delta E) & \text{, if} \Delta E = E_j - E_i > 0 \\ 1 & \text{, if } \Delta E \leqslant 0 \end{cases} . \tag{8}$$

The according proof is short accepting $\Delta E > 0$ without qualification

$$\frac{W_{ij}}{W_{ji}} = \frac{\exp(-\beta\Delta E)}{1} = \frac{\exp(-\beta E_j)}{Z} \cdot \frac{Z}{\exp(-\beta E_i)} = \frac{P_j}{P_i}. \tag{9}$$

Explanatory Note

This criterion does always accept the trial state, if the energy is lowered and exponentially suppresses those, which raise the energy. For the sake of illustration, all simulation steps will be described in detail.

1. Initialise the coordinates of all particles - random or sorted
2. Perform M trial moves ($M \approx$??), i.e.

1 This is a strong, but not necessary condition to prevent the resulting Markov chain to be trapped in a limit cycle [?].

2 But not in the classical, newtonian way of motion, every snapshot of the system towards equilibrium is the result of random initialised trial steps.

- Select a particle and move it with vector δr
- Compute the energy difference $\Delta E = E_j - E_i$

3. Generate a uniformly distributet random number x in $[0, 1]$ and accept the trial move, if eighter $\Delta E > 0$ or $x < \exp(-\beta \Delta E)$, discard otherwise.

4 MOLECULAR DYNAMICS SIMULATION

Molecular Dynamics (MD) gained very much popularity in the 1950 and 1970's to simulate real time behaviour of many body systems or complex structures like proteins. To do this an MD algorithm numerically solves the equations of motion of a given system. In Case of our System these equations are Newton's equations of motion with forces f_i given by the Lennard-Jones-Potential. The basic structure of the MD algorithm, that we implement, is the following:

1. Initialize a starting configuration.
2. Loop
   a) Calculate resulting forces for each Particle.
   b) Integrate Newton's equations of motion. To do this we choose the Velocity-Verlet algorithm.
   c) After predefined time intervals, save the current configuration or measure an observable.
3. Calculate time averages of the measured observables.

These steps will now be discussed in more detail.

1. Initialization

To initialize the System, locations and velocities of the particles must be generated. Parameters for the initialization are the number of particles N, particlemass m, dimension of the system d, Temperature T and Size of the System L. For each particle d random numbers $\in [0, L]$ are generated to initialize the locations. The intial velocities however, are a little more difficult to generate because there are two conditions which have to be satisfied:

1. $\sum_i \vec{v_i} = 0$ (The system itself is resting).
2. $\langle \vec{v}_i^2 \rangle = \frac{d k_B T}{m}$ (satisfies the Maxwell-Boltzmann distribution).

To meet the first condition, velocities are intialized in pairs, f.i. after generating the first velocity v_1, the second particle will get the velocity $v_2 = -v_1$. The program then resumes by generating the third velocity. This procedure requires $N/2$ random Numbers to be generated per Dimension and limits N to be an even number. The second condition can be satisfied by ??? ...

2.a) Calculation of Forces between the Particles

The force, that a particle j applies on a particle i can be calculated by using

$$\vec{f}_{ij} = -\frac{\partial V(r_{ij})}{\partial r_{ij}} \cdot \hat{e}_{r,ij}, \tag{10}$$

where r_{ij} is the distance between the two particles, $V(r)$ is the Lennard-Jones-Potential and $\hat{e}_{r,ij}$ is the normalized distance vector pointing from particle j to particle i. By inserting (2) the force can explicitly be written as

$$\vec{f}_{ij} = -\frac{\partial V(r_{ij})}{\partial r_{ij}} \cdot \hat{e}_{r,ij} = -4 \cdot \left(-\frac{12}{r_{ij}^{13}} + \frac{6}{r_{ij}^{7}} \right) \cdot \frac{\vec{r}_{ij}}{r_{ij}} \tag{11}$$

$$= 24 \cdot \left(\frac{2}{r_{ij}^{14}} - \frac{1}{r_{ij}^{8}} \right) \cdot \vec{r}_{ij}. \tag{12}$$

The effective force on one particle i is now given by

$$\vec{f}_i = \sum_{j \neq i} \vec{f}_{ij}$$

which has to be calculated for every particle in the system before doing a Molecular Dynamic step.

2.b) Integration of Newton's equations of motion with the Velocity-Verlet algorithm

In 1. and 2.a) the locations $\vec{r}_i(t)$, velocities $\vec{v}_i(t)$ and forces $\vec{f}_i(t)$ have been saved. These can now be used in the Velocity-Verlet algorithm

$$\vec{r}(t+\Delta t) = \vec{r}(t) + \vec{v}(t)\Delta t + \frac{\vec{f}}{2m}\Delta t^2 \tag{13}$$

$$\vec{v}(t+\Delta t) = \vec{v}(t) + \frac{\vec{f}(t+\Delta t) + f(t)}{2m}\Delta t \tag{14}$$

to calculate the state of the system after a timestep Δt.

2.c) & 3) Snapshots and Observables

Throughout the MD Simulation Snapshots of the Particlelocations will be saved in files after a predefined amount of timesteps (For Visualization). Additionally observables like potential and kinetic energy can be measured to calculate a time-average.

5 CONCLUSION

REFERENCES