



Computational Physics

Exercise Sheet 03

RENE-MARCEL LEHNER

Summer Term
2023

Vorwort	2	Formatierung des CLI outputs	3
Videos	2	E1: 2D Lennard-Jones-Fluid	4
E0: Comprehension questions.....	3		

Vorwort

Aus Zeit- und Ressourcen-Gründen ist diese Abgabe zum einen auf Deutsch, zum anderen hauptsächlich auf die Präsentation der Ergebnisse fokussiert, nicht auf übergreifende Herleitungen oder ausführliche Diskussionen.

Die Implementation und Erkundung der MD-Simulation war Spaßig und lehrreich, aber definit auch mit Frust behaftet, da das Debuggen des Codes teilweise viel Zeit gekostet hat und die Fehler alles andere als offensichtlich waren.

Zudem - weil es eine Abgabe ist, die von Studierende an Studierende geht - wird der Text nicht an allen Stellen hoch-wissenschaftlich verfasst sein; jedoch auf eine angemessen fachliche Art eine persönliche Note mittragen.

Videos

Der [.zip](#)-Datei sind drei Videos beigelegt (je etwa 30, 15 und 10 MB), die die Simulation für je b), c) und d) zeigen. Da das Datenlimit für die Abgaben bei 20 MB liegt, sind die Simulations-Videos unter diesen Links zu finden, je für b), c) und d). Diese Animationen waren vor allem für das Verständnis der Simulation und das Auffinden von Bugs äußerst hilfreich.

Es wird eine 16-fache Abspielgeschwindigkeit empfohlen.

E0: Comprehension questions

Task:

1. What should you be aware of, initializing an MD-simulation? Why is it necessary? Tipp: What would happen if two particles would occupy exactly the same position?
2. What is described by the pair correlation function? Which properties does this function have and what does it tell you about your system?

1. Für eine MD-Simulation sollte darauf geachtet werden, die Teilchen in einem geeigneten Abstand zu initialisieren. Besetzen Teilchen denselben Ort, entspräche der Distanzvektor dem Nullvektor und hätte die Länge Null, was zu einer Singularität in Potential und Kraft führen würde. Zudem sorgen kleine Abstände für sehr hohe Kräfte und damit indirekt für sehr hohe Startgeschwindigkeiten.

2. Die Paarkorrelationsfunktion ist eine bedingte Wahrscheinlichkeit als Funktion des Abstandes. Sie gibt jene Wahrscheinlichkeit an, ein Teilchen bei gegebenem Abstand unter der Voraussetzung, dass ein Teilchen im Zentrum platziert ist, anzufinden.

Diese Funktion wird gerne als Ordnungsparameter verwendet, da es ein Maß für die Ordnung des Fluids darstellt. Sie gibt Aufschluss über die örtliche Struktur, da bspw. geordnete Strukturen ein periodisches Muster in $g(r)$ aufweisen.

→ Auch Verschiebung entlang Gradientenlinien zum Minimum.

Formatierung des CLI outputs

Stolzerweise wurde während der Entwicklung des Codes herausgefunden, dass in der Linux-Konsole verschiedenste Formatierungen eingestellt werden können, die zunächst wie eine Spielerei wirken mögen, bei angemessener Anwendung jedoch für eine erhebliche Verbesserung der Übersicht sorgen.

Falls der Code ausgeführt wird und sofern eine Linux-Konsole verwendet wird, sei auf dieses nette Merkmal hingewiesen.

E1: 2D Lennard-Jones-Fluid

Task:

Molecular dynamics simulations are an essential tool in numerical physics for describing the behavior of larger quantities of atoms or molecules, if the potential of their interaction is known. In this task, such a molecular dynamics simulation shall now be exemplified for a simple bath of uncharged particles in two dimensions. The interaction of such particles can be described by the Lennard-Jones potential.

Implement a molecular dynamics simulation for N identical particles of mass $m = 1$ with a pairwise Lennard-Jones interaction.

$$V(r) = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right]$$

Set $\epsilon = \sigma = 1$, meaning lengths are given in units of σ and energies, as well as $k_B T$ in units of ϵ . Use periodic boundary conditions in a System of size $A = L \times L$. Use a cutoff radius of $r_c = L/2$ to determine the forces. Then use the Verlet algorithm with time steps of $h = 0.01$ (smaller time steps might be necessary for higher temperatures) as your integrator.

NOTE: A MD-simulation is a bigger project build up of several modules (like integrator, thermostat, etc.). That given, it might be helpful to do object orientated programming for this exercise. To help you with that, we provide you a template for python and C++. This, however, is just a recommendation and not mandatory for successfully completing this task.

a) Initialization:

To start off, take a square box of length $L = 2n\sigma$ and fill it with $N = n^2$ particles on equidistant positions. Choose your starting velocities such that $\sum_{i=1}^N \mathbf{v}_i(0) = 0$, so that your center of mass is stationary. Write your program in such a manner, that you are able to rescale velocities to set your starting temperature to $T(t=0)$.

Note: For random number generation, you are allowed to use already existing random number generators like f.e. `<random>` in C++:

```
#include <random>
mt19937 rnd;
uniform_real_distribution<> dist(0, 1);
random_number = dist(rnd);
```

b) Measurement/Equilibration:

Start at $T(0) = 1$ for $n = 4, 8, 16$. Calculate your center of mass velocity $\frac{1}{N} \sum_{i=1}^N \mathbf{v}_i(t)$ in dependence of time. Calculate the temperature $T(t)$ in dependence of time. Calculate the potential $E_{\text{pot}}(t) = \sum_{i < j-1}^N V(|\mathbf{r}_i - \mathbf{r}_j|)$ as well as the kinetic energy $E_{\text{kin}}(t) = \sum_{i=1}^N \frac{1}{2} \mathbf{v}_i^2$ as functions of time. How long does it take for your System to equilibrate?

c) Measurement:

After the equilibration is done, you can now measure the temperature T and the pair correlation function $g(r)$. To do so, start to average over your system for 10^4 to 10^6 time steps (depending on how long you are willing to wait).

Entdimensionalisiert: $\frac{m \sigma^2}{\epsilon v^2} \frac{1}{48} \ddot{\mathbf{r}} = \ddot{\mathbf{r}} = \frac{1}{r^7} \left(\left(\frac{1}{r} \right)^{12} - \frac{1}{2} \left(\frac{1}{r} \right)^6 \right)$

$$E_{\text{kin}} = \frac{N}{2} m \langle v^2 \rangle = \frac{d}{2} k_B T (N-1) \Rightarrow \langle v^2 \rangle = \frac{d}{m} k_B T \frac{N}{N-1} = \frac{\sigma^2}{\epsilon} \langle \tilde{v}^2 \rangle$$

$$\Rightarrow \langle \tilde{v}^2 \rangle = \frac{v^2 \epsilon}{\sigma^2 m} \hat{T} \frac{dN}{N-1} \Rightarrow \hat{T} = \frac{m}{\epsilon} \langle \tilde{v}^2 \rangle \frac{N-1}{N}$$

$$\hat{T}_{\text{ref}} \rightarrow \hat{T} \Rightarrow \tilde{v} = \alpha \hat{v} \Rightarrow \alpha = \frac{\hat{T}}{\hat{T}_{\text{ref}}}$$

Do these measurements for $n = 8$ at three different starting temperatures $T(0) = 1$, $T(0) = 0.01$ and $T(0) = 100$. Which phases do you notice?

d) Thermostat:

Now implement an isokinetic thermostat for your MD-simulation and couple it to your System to keep a constant temperature. What does now happen for $T = 0.01$? Look at the potential, kinetic and overall energy during equilibration and measure $g(r)$ in the newly equilibrated system. It might also be interesting to look at snapshots of your system.

e) Physical and Chemical Systems (Bonus: 5 Points)

- Which quantities are kept constant during your simulation (after turning on your thermostat)?
- Which thermodynamic ensemble is represented by that?
- You are using an isokinetic thermostat for your system. Is this kind of thermostat physically accurate? If not, name a physically accurate alternative.
- A chemist asks you to do a MD-simulation on a chemical solution. In which way might such a system differ from the ensemble we considered so far? If you have to keep different quantities constant, name a way to do so for your simulation.
- Which further potential should also be considered for a chemical solution (f.e. NaCl in water)?

Für den gesamten Code wurden alle Formeln **entdimensionalisiert**. Daraus folgen vereinfachte Formeln, auf die im weiteren Verlauf der Abgabe eingegangen wird.

Zudem wurde der Code in **C++** geschrieben und auf das **MD-template.cpp** zurückgegriffen.

- a) Die Initialisierungen der **MD**, **Thermostat**, **Data**, **Dataset** und **Potential** Klassen/structs sind dem Code zu entnehmen. Die verwendeten Algorithmen werden im weiteren Verlauf jedoch geschildert.

Es sei betont, dass alle Formeln entdimensionalisiert sind, weswegen bspw. $F = a$, $E_{kin} = v^2$, $L = 2n$ und $T = E_{kin}/N$ folgen.

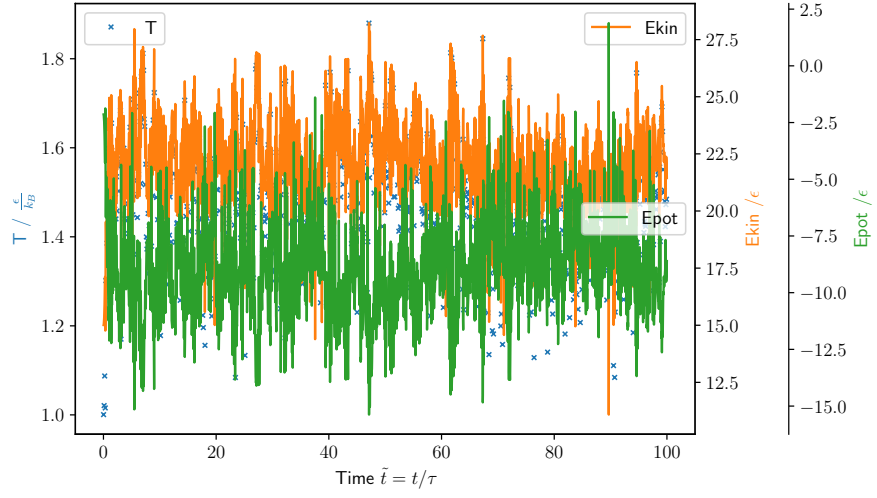
- b) Für die gesamte Simulation wurde der **Geschwindigkeits Verlet Algoritimus** verwendet, da es Schwierigkeiten mit dem gewöhnlichen Verlet gab (was sich später als ein "1/2 = 0"-Schmerz herausstellte).

Der Rest ist relativ einfach dem Code zu entnehmen, da die relevanten Formeln bereits in der Aufgabenstellung angegeben sind.

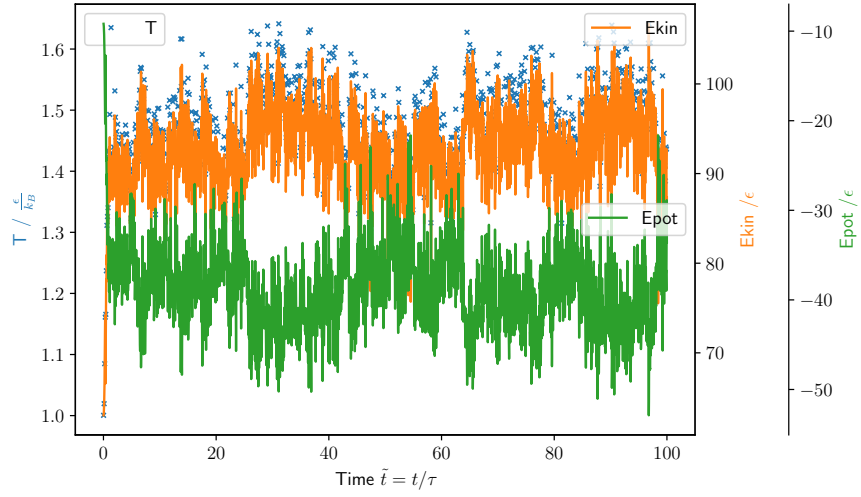
Für die Berechnung der potentiellen Energie wurde von dem errechneten Wert das Cut-Off-Potential $V(L/2)$ abgezogen, um eine Unstetigkeit zu vermeiden.

Um die Frage in b) zu beantworten: Die Zeit zur Äquilibration des Systems hängt von Temperatur und Anzahl der Teilchen ab. Da die Temperatur in b) für alle Szenarien gleich ist, gibt es also hier nur eine Abhängigkeit von der Teilchenzahl. Da die Frage nicht direkt nach einem quantitativen Wert fragt und dieser nicht so einfach zu bestimmen wäre, kann er stattdessen aus Abbildung 1.1 grafisch abgelesen werden. Hier wird nämlich ein sofortiger Transfer (nach Start der Simulation) der potentiellen Energie in kinetische Energie sichtbar. Ein Gleichgewicht stellt sich nach relativ kurzer Zeit ein.

Potential energy, kinetic energy and temperature $T(0) = 1$ for $N = 16$ particles
no prior equilibrium



Potential energy, kinetic energy and temperature $T(0) = 1$ for $N = 64$ particles
no prior equilibrium



Potential energy, kinetic energy and temperature $T(0) = 1$ for $N = 256$ particles
no prior equilibrium

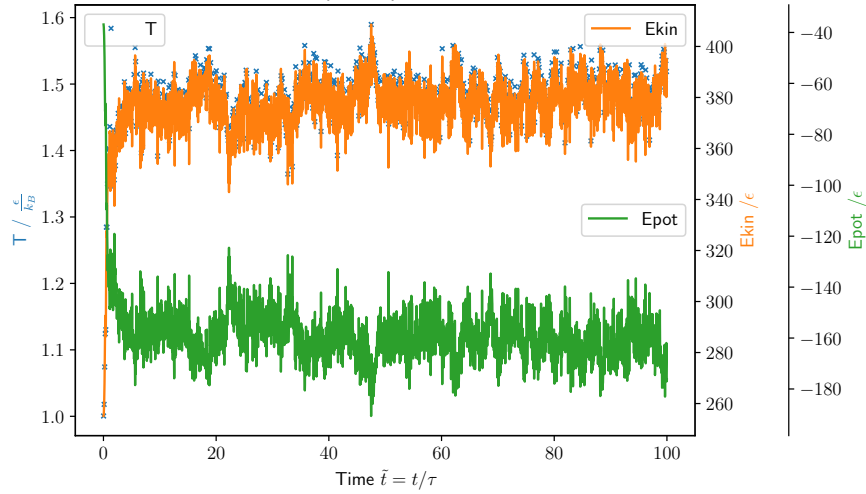


Figure 1.1

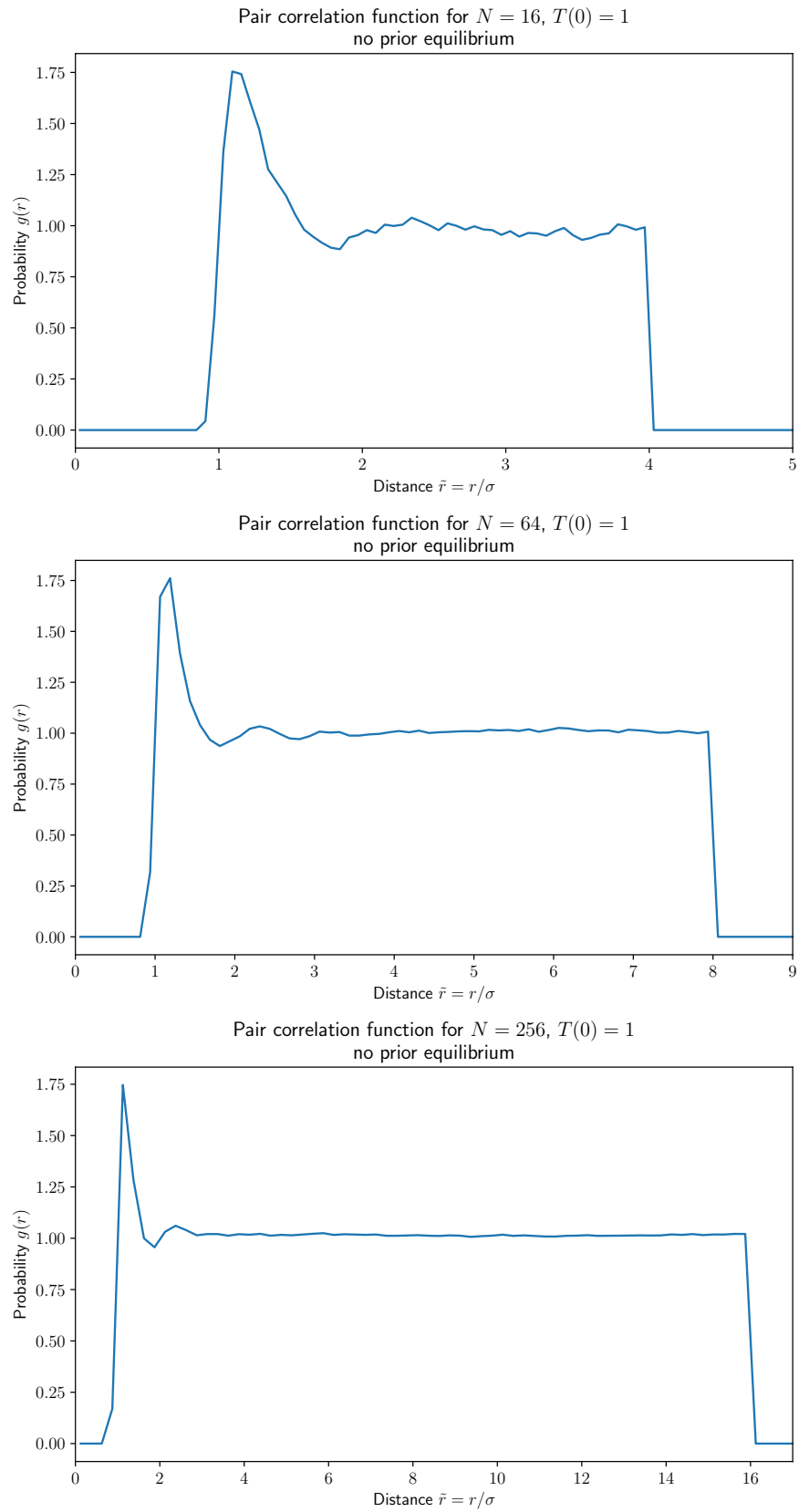


Figure 1.2

- c) ¹ Die Videos bieten hier eine visuelle Interpretation. Im Allgemeinen sehen alle Simulationen äußerst ungeordnet aus.

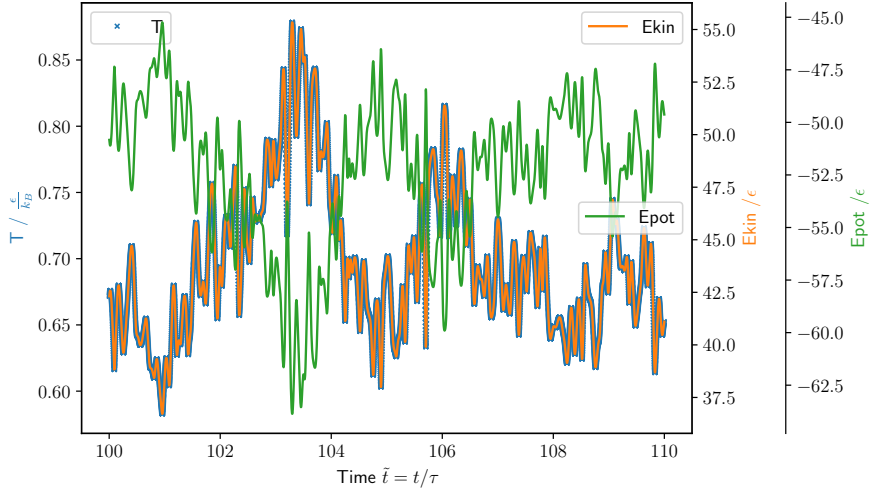
Jedoch liegen bei genauerer Untersuchung für $T(0) = 0.01$ und $T(0) = 1$ flüssige Phasen vor, was unter anderem der Abbildung 1.3, also dem Wechselspiel der potentiellen und kinetischen Energie zu entnehmen ist.

Hier bewegen sich die Teilchen vorzugsweise in übergeordneten Gruppen, die sich in sich scheren. Sie bilden keine feste Gitterstruktur aus, haben in der Geschwindigkeit lokal aber eine Vorzugsrichtung.

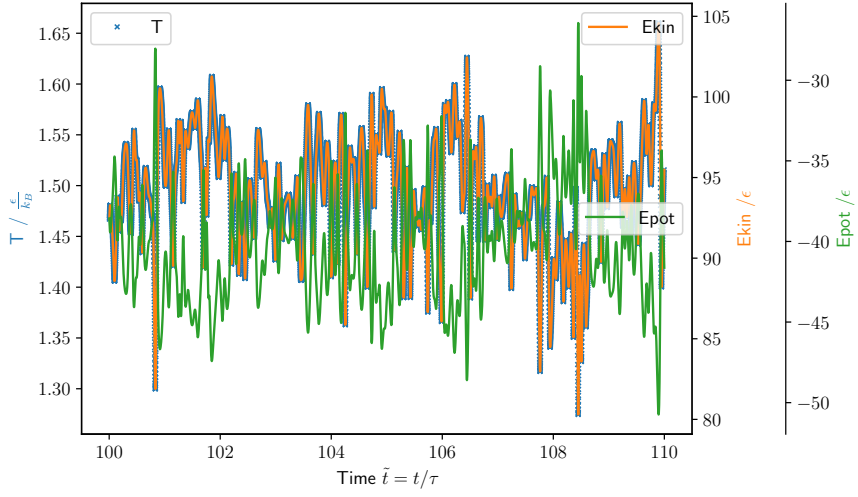
Im Vergleich dazu kann bei $T = 100$ beobachtet werden, dass es keine lokalen Ausprägungen gibt, weder zeitlich noch örtlich, was charakteristische Merkmale einer Gas-Phase sind.

¹Nicht sicher, warum Absätze hier derart formatiert werden. Aus Zeitgründen wird dies aber mal hingenommen.

Potential energy, kinetic energy and temperature $T(0) = 0.01$ for $N = 64$ particles with prior equilibrium



Potential energy, kinetic energy and temperature $T(0) = 1$ for $N = 64$ particles with prior equilibrium



Potential energy, kinetic energy and temperature $T(0) = 100$ for $N = 64$ particles with prior equilibrium

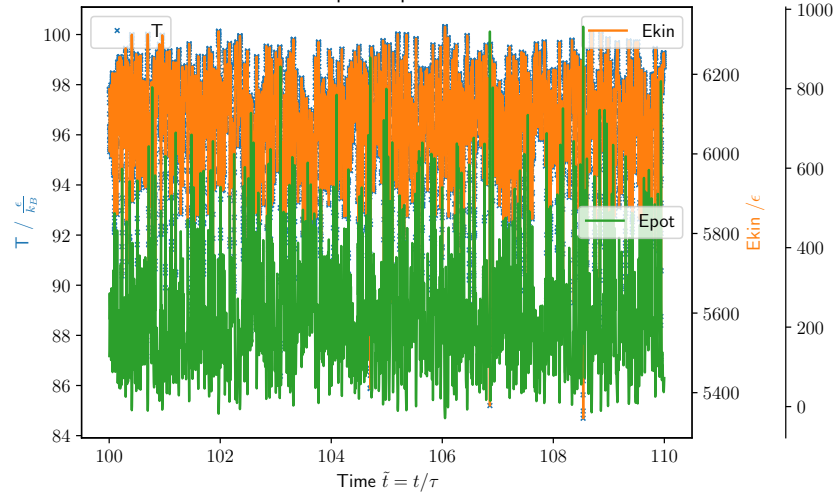


Figure 1.3

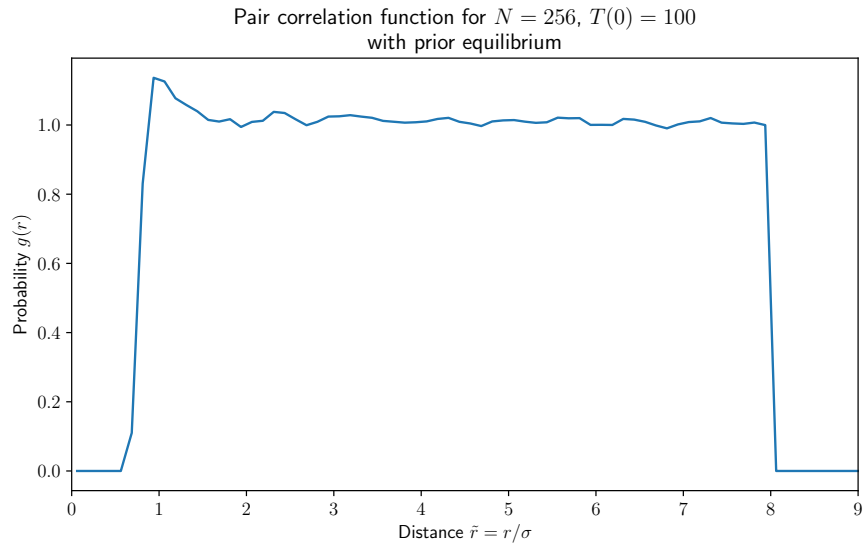
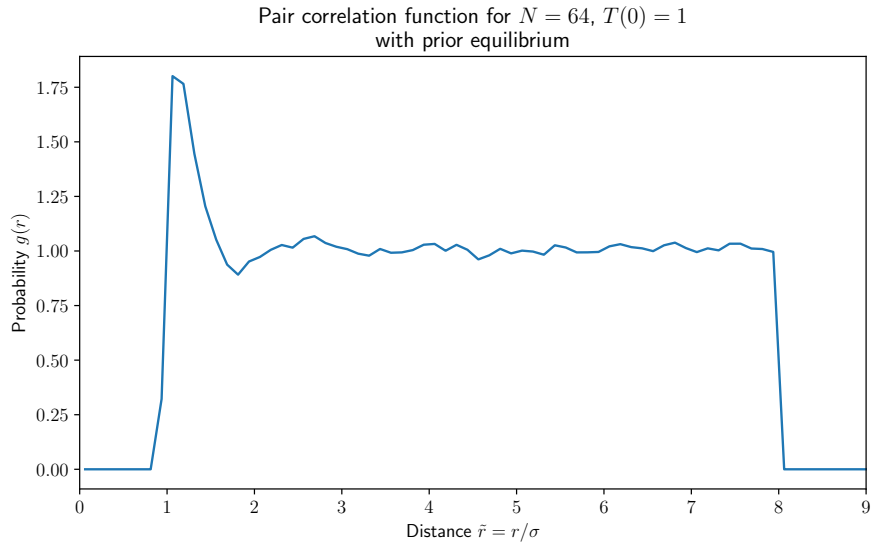
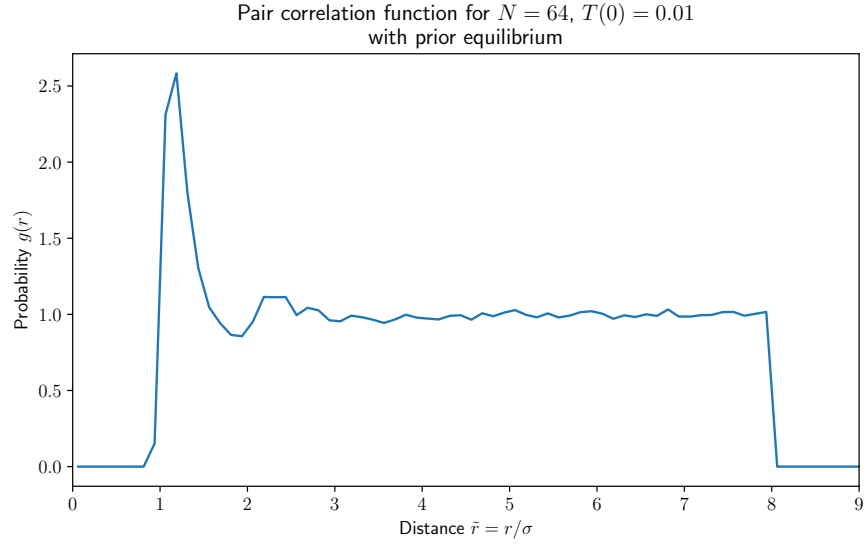


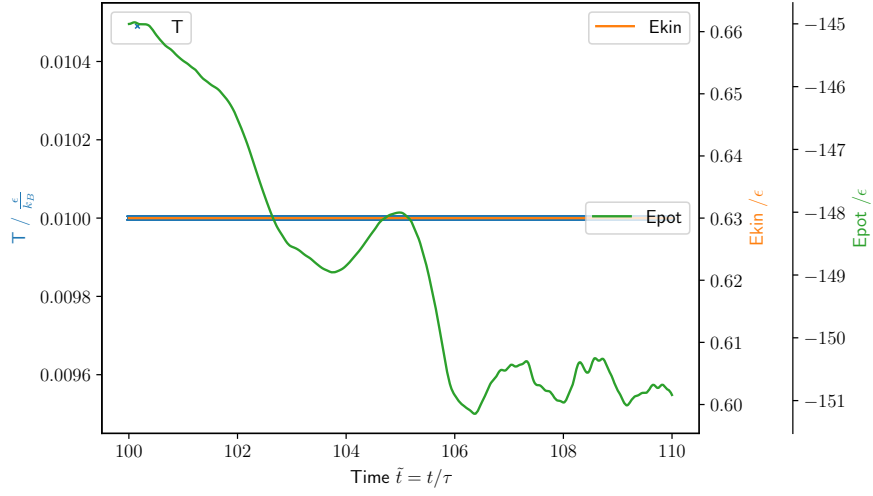
Figure 1.4

- d) Bei sehr niedrigen Temperaturen bilden die Teilchen Cluster, die ein unterliegendes Raster haben. Dies entspricht einer Festkörper-Eigenschaft. Aus Mangel an Teilchen ist das Cluster lokal begrenzt, weswegen es bei spontaner Betrachtung auch als Fluid angenommen werden kann. Da die Teilchen jedoch nicht einander scheren, sich als Gruppe bewegen und vor allem relativ zueinander in festen Gitterplätzen verharren, ist diesem LJ-Fluid die Phase **fest** zuzuordnen.

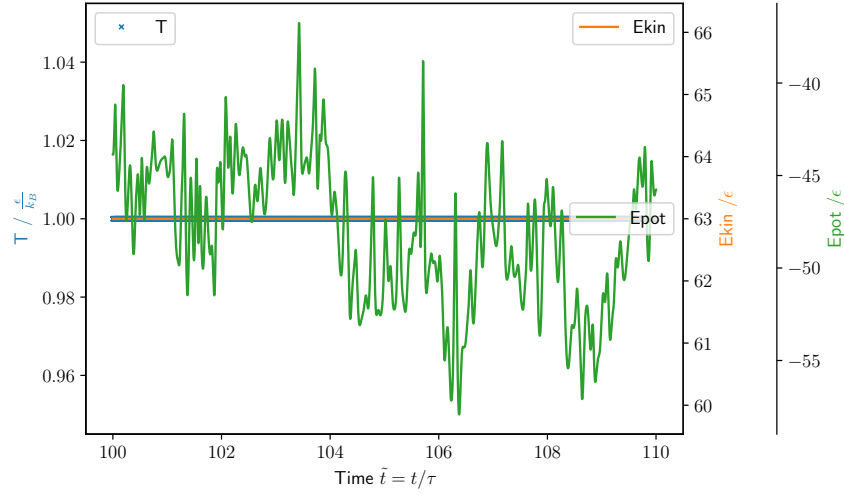
Letztlich zeigt sich auch in Abbildung 1.6 ein für eine feste Phase typisches Verhalten, bei dem die Wahrscheinlichkeiten für das Antreffen von anderen Teilchen sehr ausgeprägt und lokalisiert sind und abschnittsweise auch gegen Null gehen können.

Dies deutet auf die Existenz eines "Schalen-Modells" hin.

Potential energy, kinetic energy and temperature $T(0) = 0.01$ for $N = 64$ particles with prior equilibrium and isokinetic thermostat



Potential energy, kinetic energy and temperature $T(0) = 1$ for $N = 64$ particles with prior equilibrium and isokinetic thermostat



Potential energy, kinetic energy and temperature $T(0) = 100$ for $N = 64$ particles with prior equilibrium and isokinetic thermostat

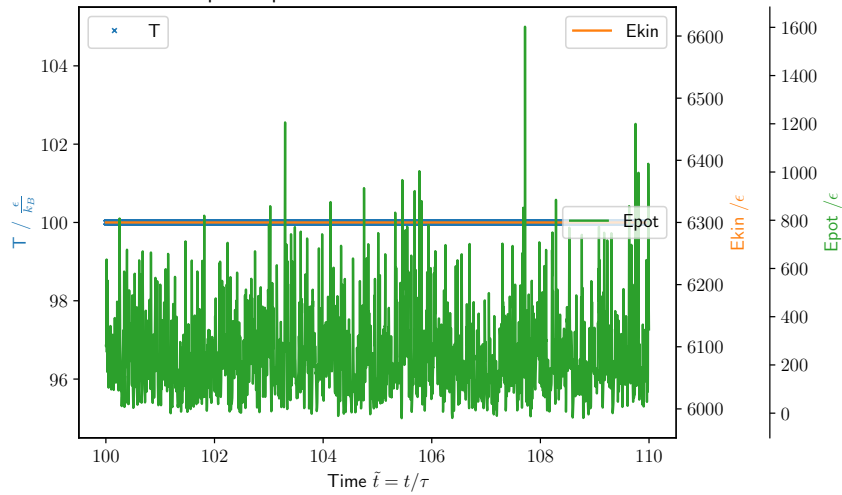


Figure 1.5

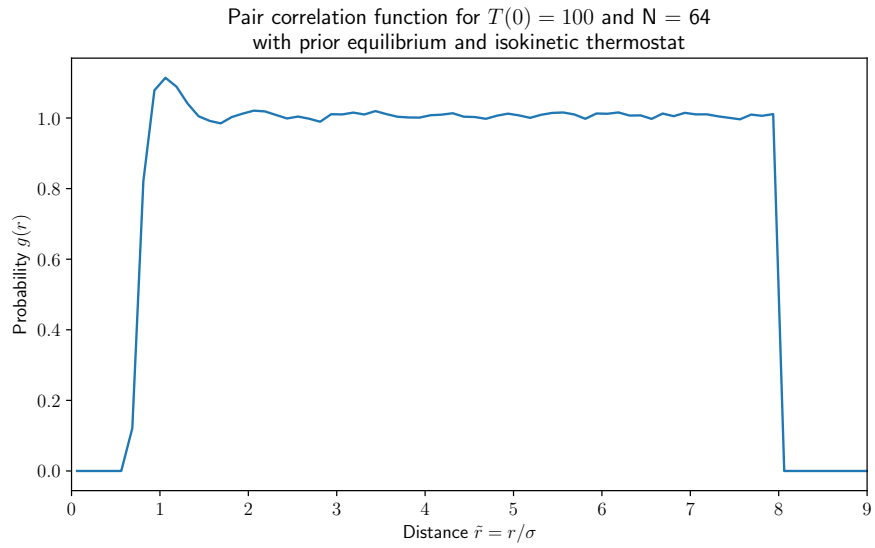
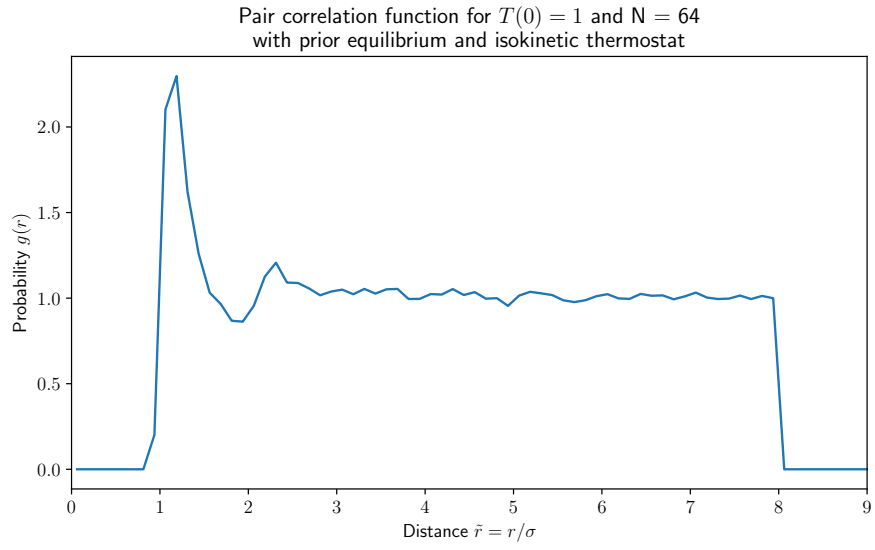
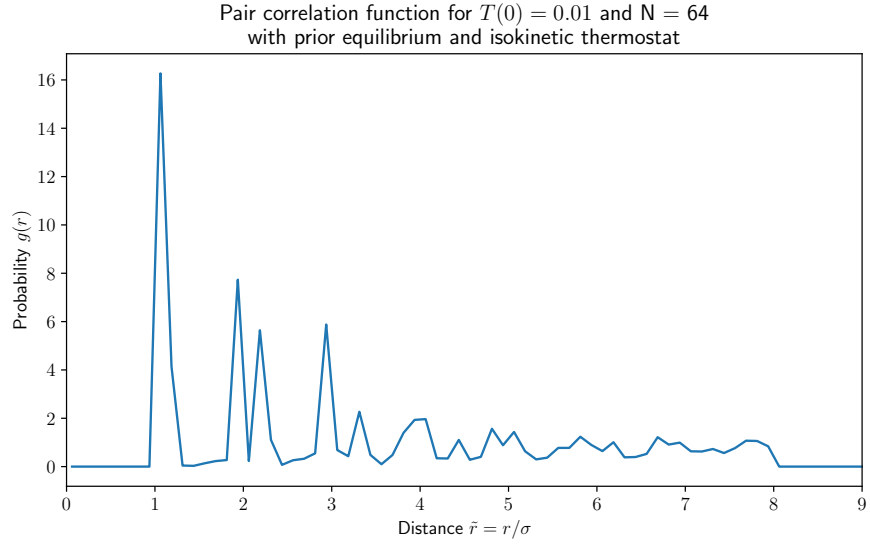


Figure 1.6

- e) (a) Der Verlet-Algorithmus erhält die Gesamtenergie, sofern keine weiteren Randbedingungen eingeführt werden (wie das isokinetsiche Thermostat). Konstant bleiben aber auch Schwerpunkts-
geschwindigkeit und bei dem isokinetischen Thermostat vor allem auch die kinetische Energie
(streng) und die Temperatur (ebenfalls streng), dafür aber nicht mehr die Gesamtenergie (da die
potentielle Energie weiterhin fluktuiert).
- (b) Da sowohl Teilchenzahl N , als auch Temperatur T und Volumen V erhalten sind, handelt es sich
um ein Kanonisches Ensemble.
- (c) Nein, ein isokinetisches Thermostat ist nicht physikalisch präzise, da es die Geschwindigkeiten
derart modifiziert, dass diese nicht mehr der gewünschten Boltzmann-Verteilung entsprechen.
Diese ist aber gerade die Grundlage für die Statistische Physik und den daraus abgeleiteten
quantitativen Eigenschaften thermodynamischer Systeme.
Eine Alternative stellt das **Nosé-Hoover Thermostat** dar, das den Austausch von Energie
erlaubt und die Boltzmann-Verteilung der Geschwindigkeiten gewährleistet.
- (d) Bei chemischen Reaktionen ändern sich die Teilchenzahlen, da die Teilchen miteinander reagieren
und sich kombinieren oder aufteilen können. Dies wird erst bei dem Großkanonischen Ensemble
berücksichtigt. Hier muss ein chemisches Potential eingeführt werden, um die Teilchenzahl
konstant zu halten.
- (e) Das Lennard-Jones Potential behandelt neutrale Teilchen. Für Ionen muss zusätzlich ein
elektrostatisches Potential betrachtet werden, welches sich mit dem LJ Potential überlagern
würde. Hier muss vor allem beachtet werden, dass diese Kräfte sehr hohe Reichweiten haben.