2022S 260007-1 Advanced Computational Physics

Andreas Tröster

based on lecture notes by
Christoph Dellago

April 19, 2023

Contents

Appendices

Part I

Monte Carlo Simulations

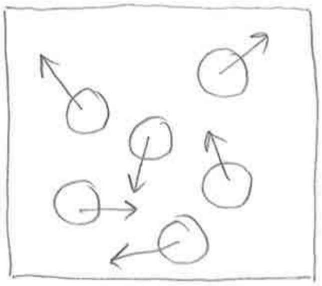
Chapter 1

Statistical Mechanics and Simulation

1.1 Introduction

Consider the problem of computing the pressure p of a system of N hard disks (in d=2) or spheres (in d=3) in a volume V at temperature T.

This problem of determining the equation of state can be solved analytically for the ideal gas, but not for hard disks or spheres. In 1953 a method for solving this problem with Monte Carlo (MC) simulations was suggested [Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller and Edward Teller, *Equation of State Calculations by Fast Computing Machines*, J. Chem. Phys. **21**, 1087 (1953)]. This was the birth of MC simulations in statistical mechanics.



The idea of using simulations in statistical mechanics was a real breakthrough. Since the times of Gibbs and Boltzmann, the general formalism to compute thermodynamic properties from knowledge of intermolecular interactions was known, but there was no way to compute the high-dimensional integrals appearing in this formalism analytically. Simulations provided an efficient way to evaluate these integrals. Today we can simulate very complex systems on the computer.
Simulations can provide “numerically exact” results. Using simulations, we can

- test assumptions and predictions of simplified theories.
- study realistic models under conditions not realizable in experiment.

Similar to experiments, simulations produce sets of data, but do not automatically yield physical insights. Further *analysis* is usually required. Also, simulations do not produce "new physics". However, they can give insight into the complex collective behavior of systems consisting of many simple components (building blocks). In this sense, simulations *can* produce "new phenomena".
In this course we discuss different particle-based simulation techniques:

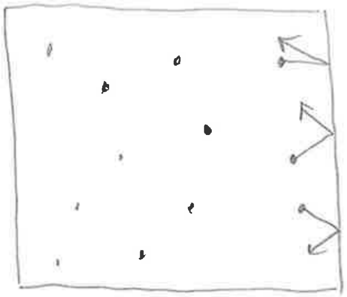
- *atomistic* simulations
- *mesoscopic* particles (colloids)

We shall mainly focus on Monte Carlo (MC) and molecular dynamics (MD). Before we talk about these methods in more detail, let us review some basic concepts of thermodynamics and statistical mechanics.

1.2 Statistical-Mechanical Averages

In general, we consider many particle systems (N-particle systems) consisting of identical building blocks (atoms, molecules, colloids, spins,. . .). The *goal of statistical mechanics* is to predict the properties of such a system based on knowledge of intermolecular interactions. Time and length scales of a macroscopic experiment are, however, much larger than molecular dimensions. For example,

- pressure exerted on a container wall by a gas or liquid in the container is the result of many collisions of atoms with the wall.
- pressure measured in a macroscopic experiment is the result of both a temporal and a spatial average.

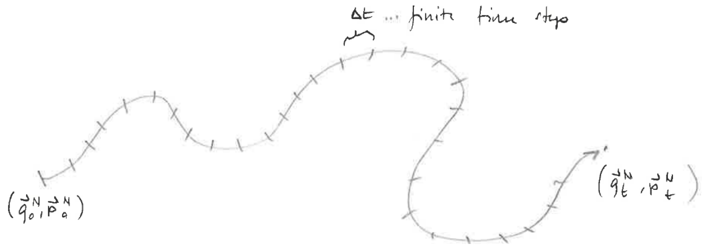


In principle, for a classical point particle system, such a microstate consists of a snapshot of all coordinates and momenta

$$\boldsymbol{r}^N \equiv \{\boldsymbol{r}_1, \ldots, \boldsymbol{r}_N\}, \qquad \boldsymbol{p}^N \equiv \{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_N\}$$

Its knowledge completely determines the state of the system and provides initial conditions for the solution of the equations of motion at times t extending arbitrarily far into the future (or past). However, in practice one can never control all degrees of freedom but only an *ensemble* of all microstates that are compatible with the macroscopic states of the system as described through a small number of parameters (V, N, T, p, magnetic field $H \ldots$).
In an MD simulation (discussed in detail in the next chapter) we integrate the equations of motion and determine the time evolution in the phase space

spanned by all coordinates and momenta. The simulation yields a *trajectory*, a series of snapshots of the system:


Δt ... finite time step
$(\vec{q}_0^N, \vec{p}_0^N)$
$(\vec{q}_t^N, \vec{p}_t^N)$

Each snapshot is a complete microstate of the system. The central assumption is that during a measurement the system visits all the microstates compatible with these macroscopic parameters. The measured quantities are therefore *averages* of the respective observable over all microstates.

To determine the statistical-mechanical expectation values of an observable, one averages over the snapshots of this “movie”. In equilibrium (and for quantities that depend only on one time i.e. not on two or more times separately), the sequence in which the snapshots occur in the movie is not important; it is sufficient to know the *probabilities* with which the microstates occur, but not their particular order in time. Once we know this probability distribution, which defines the *ensemble* compatible with the prescribed macroscopic parameters, we can carry out the averaging over the distribution rather than over a trajectory, which is therefore called an *ensemble average*.

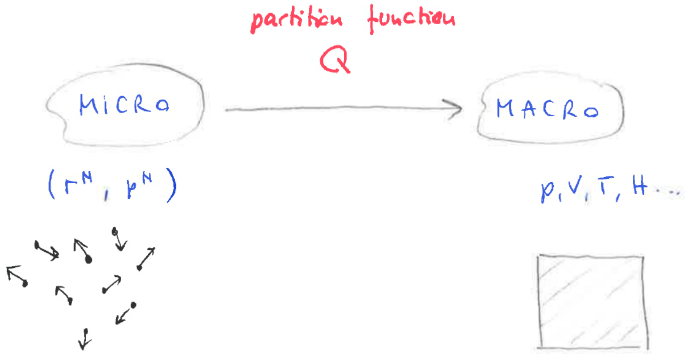
In a MC simulation, one also generates samples of microstates according to the probability distribution of the ensemble that one wants to study.

Before we discuss MC and MD simulations in detail, we need to review a few central notions of statistical mechanics and thermodynamics.

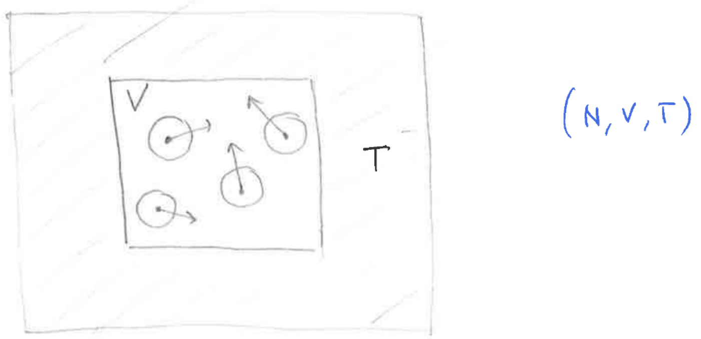
1.3 Classical Canonical Ensemble

1.3.1 Partition Function

The central quantity that connects the microscopic and the macroscopic world is the canonical partition function Q_{NVT}:



For simplicity we consider a system of N identical point particles, i.e. particles which have no internal degrees of freedom. The microstates of the system are therefore completely defined by the collection $(\boldsymbol{r}^N, \boldsymbol{p}^N)$ of all positions and momenta of the particles ($\boldsymbol{p}_i = m\boldsymbol{v}_i$). This system is enclosed in a fixed volume V and is in contact with a heat bath at fixed temperature T. This defines the *canonical ensemble (NVT-ensemble)*



The *classical canonical partition function* is

$$Q_{NVT} = \frac{1}{N!(2\pi\hbar)^{3N}} \int d\boldsymbol{r}^N d\boldsymbol{p}^N e^{-\beta H(\boldsymbol{r}^N, \boldsymbol{p}^N)} \tag{1.1}$$

where

- the integral is taken over the entire phase space.
- the factor $1/N!$ resolves the notorious “Gibbs paradox” by correcting the overcounting due to the fact that all permutations of identical particles over the microstates result in the same physical microstate.

- $h^3 = (2\pi\hbar)^3$ is a measure of the "cell size" in phase space and makes Q_{NVT} dimensionless.
- $\beta = 1/k_BT$ denotes the reciprocal/inverse temperature.
- k_B is the Boltzmann constant.
- $H(\boldsymbol{r}^N, \boldsymbol{p}^N)$ is the classical Hamilton function which gives the total energy of the classical microstate $(\boldsymbol{r}^N, \boldsymbol{p}^N)$.
- the so-called *Boltzmann factor* $e^{-\beta H(\boldsymbol{r}^N, \boldsymbol{p}^N)}$ is proportional to the probability of observing this microstate in the canonical ensemble.

We now assume that $H(\boldsymbol{r}^N, \boldsymbol{p}^N)$ is given by

$$H(\boldsymbol{r}^N, \boldsymbol{p}^N) = \underbrace{\sum_{i=1}^{N} \frac{\boldsymbol{p}_i^2}{2m}}_{\text{kinetic energy}} + \underbrace{U(\boldsymbol{r}^N)}_{\text{potential energy}} \tag{1.2}$$

Therefore, since the kinetic energy depends only on the momenta, while the potential energy depends only on the positions, the above partition function factorizes into a product of two separate integrals, one over configuration space and one over momentum space:

$$Q_{NVT} = \frac{1}{N!(2\pi\hbar)^{3N}} \left\{ \int d\boldsymbol{p}^N e^{-\beta \sum_{i=1}^N \frac{\boldsymbol{p}_i^2}{2m}} \right\} \left\{ \int_{V^N} d\boldsymbol{r}^N e^{-\beta U(\boldsymbol{r}^N)} \right\} \tag{1.3}$$

The momentum space integral can be factorized further into a product of Gaussian integrals

$$\int d\boldsymbol{p}^N e^{-\beta \sum_{i=1}^N \frac{\boldsymbol{p}_i^2}{2m}} = \left(\int_{-\infty}^{\infty} dp e^{-\frac{1}{2}\frac{p^2}{m/\beta}} \right)^{3N} = (2\pi m/\beta)^{3N/2} \tag{1.4}$$

Therefore

$$\begin{aligned} Q_{NVT} &= \frac{1}{N!} \frac{(2\pi m/\beta)^{3N/2}}{(2\pi\hbar)^{3N}} \int_{V^N} d\boldsymbol{r}^N e^{-\beta U(\boldsymbol{r}^N)} \\ &= \frac{1}{N!} \left(\frac{m/\beta}{2\pi\hbar^2} \right)^{3N/2} \int_{V^N} d\boldsymbol{r}^N e^{-\beta U(\boldsymbol{r}^N)} \end{aligned} \tag{1.5}$$

i.e.

$$Q_{NVT} = \frac{1}{N!} \frac{1}{\Lambda^{3N}} \int_{V^N} d\boldsymbol{r}^N e^{-\beta U(\boldsymbol{r}^N)} \tag{1.6}$$

where Λ is the so-called *thermal de Broglie wave length* defined by

$$\Lambda^2 = \frac{2\pi\hbar^2\beta}{m} \tag{1.7}$$

and $Z \equiv \int_{V^N} d\boldsymbol{r}^N e^{-\beta U(\boldsymbol{r}^N)}$ is called the *configuration integral.*

Excess formulation

For the example of an *ideal gas*, where by definition $U(\boldsymbol{r}^N) \equiv 0$, the above configuration integral is given by

$$\int_{V^N} d\boldsymbol{r}^N e^{-\beta U(\boldsymbol{r}^N)} = \int_{V^N} d\boldsymbol{r}^N 1 = V^N \tag{1.8}$$

such that the canonical partition function of the ideal gas is given by

$$Q_{NVT}^{(id)} = \frac{V^N}{N!\Lambda^{3N}} \tag{1.9}$$

Using this result, we can rewrite the partition function

$$Q_{NVT} = \frac{1}{N!}\frac{V^N}{\Lambda^{3N}}\frac{1}{V^N}\int_{V^N} d\boldsymbol{r}^N e^{-\beta U(\boldsymbol{r}^N)} \tag{1.10}$$

as a product

$$Q_{NVT} = Q_{NVT}^{(id)} \cdot Q_{NVT}^{(ex)} \tag{1.11}$$

of an *ideal part* $Q_{NVT}^{(id)}$ and an *excess part*

$$Q_{NVT}^{(ex)} = \frac{1}{V^N}\int_{V^N} d\boldsymbol{r}^N e^{-\beta U(\boldsymbol{r}^N)} \tag{1.12}$$

which contains all the non-trivial effects due to the interactions between the particles.

1.3.2 Relation with thermodynamics

Thermodynamics is a macroscopic, phenomenological theory which does not consider the microscopic structure of matter. Where thermodynamics and statistical mechanics deal with the same phenomena, the two theories need to be related to each other. This connection is given by the *partition function* and the related *thermodynamic potential* of the respective ensemble.
Thermodynamic potentials are *state functions*:

- they depend only on the parameters defining a macroscopic state.
- they do not depend on the particular way in which this state is reached; thermodynamic potentials are defined only for *equilibrium states.*

In the canonical ensemble, the relevant thermodynamic potential is the *Helmholtz free energy* $F(N, V, T)$ related to the canonical partition function Q_{NVT} by

$$F(N, V, T) = -k_B T \ln Q_{NVT} \tag{1.13}$$

$F(N, V, T)$ cannot be measured directly, but its derivatives can, and they have a clear physical meaning:

- *entropy*: $S = -\frac{\partial F}{\partial T}$
- *pressure*: $p = -\frac{\partial F}{\partial V}$
- *chemical potential*: $\mu = \frac{\partial F}{\partial N}$

In thermodynamics, the free energy is usually introduced as

$$F = E - TS \tag{1.14}$$

where E is the *internal energy.* To gain more insight into the nature of E, let us compute the partial derivative $\frac{\partial(\beta F)}{\partial \beta}$ both thermodynamically and statistical-mechanically

- thermodynamical calculation: Since $\frac{\partial T}{\partial \beta} = \frac{\partial}{\partial \beta} \frac{1}{k_B \beta} = -\frac{1}{k_B \beta^2}$, we have

$$\begin{aligned}\frac{\partial(\beta F)}{\partial \beta} &= F + \beta \frac{\partial F}{\partial \beta} = F + \beta \frac{\partial F}{\partial T} \frac{\partial T}{\partial \beta} = F - \beta \frac{\partial F}{\partial T} \frac{1}{k_B \beta^2} \\ &= F - T \underbrace{\frac{\partial F}{\partial T}}_{-S} \overset{(1.14)}{=} E - TS + TS \qquad (1.15)\end{aligned}$$

i.e.

$$\frac{\partial(\beta F)}{\partial\beta} = E \tag{1.16}$$

- statistical-mechanical calculation: starting from $F = -k_B T \ln Q_{NVT}$, we get

$$\begin{aligned}
\frac{\partial(\beta F)}{\partial\beta} &= -\frac{\partial \ln Q_{NVT}}{\partial\beta} = -\frac{1}{Q_{NVT}}\frac{\partial Q_{NVT}}{\partial\beta} \\
&= -\frac{1}{Q_{NVT}}\frac{\partial}{\partial\beta}\left[\frac{1}{N!}\frac{1}{(2\pi\hbar)^{3N}}\int d\boldsymbol{r}^N d\boldsymbol{p}^N e^{-\beta H(\boldsymbol{r}^N,\boldsymbol{p}^N)}\right] \\
&= -\frac{1}{Q_{NVT}}\frac{1}{N!}\frac{1}{(2\pi\hbar)^{3N}}\int d\boldsymbol{r}^N d\boldsymbol{p}^N e^{-\beta H(\boldsymbol{r}^N,\boldsymbol{p}^N)}[-H(\boldsymbol{r}^N,\boldsymbol{p}^N)] \\
&= \frac{\cancel{\frac{1}{N!}\frac{1}{(2\pi\hbar)^{3N}}}\int d\boldsymbol{r}^N d\boldsymbol{p}^N e^{-\beta H(\boldsymbol{r}^N,\boldsymbol{p}^N)} H(\boldsymbol{r}^N,\boldsymbol{p}^N)}{\cancel{\frac{1}{N!}\frac{1}{(2\pi\hbar)^{3N}}}\int d\boldsymbol{r}^N d\boldsymbol{p}^N e^{-\beta H(\boldsymbol{r}^N,\boldsymbol{p}^N)}} \qquad (1.17)
\end{aligned}$$

i.e.

$$\frac{\partial(\beta F)}{\partial\beta} = \langle H\rangle \tag{1.18}$$

In summary, we conclude that the thermodynamic inner energy is equal to the statistical-mechanical expectation value of the Hamilton function in the canonical ensemble:

$$E = \langle H\rangle \tag{1.19}$$

1.3.3 Averages and fluctuations

The *probability density function* of the canonical ensemble is

$$f(\boldsymbol{r}^N,\boldsymbol{p}^N) = \frac{1}{N!}\frac{1}{(2\pi\hbar)^{3N}}\frac{e^{-\beta H(\boldsymbol{r}^N,\boldsymbol{p}^N)}}{Q_{NVT}} \tag{1.20}$$

Here, the canonical partition function Q_{NVT} plays the role of a normalization factor that makes sure that

$$\int d\boldsymbol{r}^N d\boldsymbol{p}^N f(\boldsymbol{r}^N, \boldsymbol{p}^N) = 1 \tag{1.21}$$

The *canonical average* (or *expectation value*) of an observable (i.e. a phase space function) $A(\boldsymbol{r}^N, \boldsymbol{p}^N)$ is defined as

$$\langle A(\boldsymbol{r}^N, \boldsymbol{p}^N) \rangle = \int d\boldsymbol{r}^N d\boldsymbol{p}^N f(\boldsymbol{r}^N, \boldsymbol{p}^N) A(\boldsymbol{r}^N, \boldsymbol{p}^N) \tag{1.22}$$

Examples.

- If $A = A(\boldsymbol{p}^N)$ depends only on the momenta, we can often carry out the integration analytically. The reason is that the probability density of the momenta is simply a Gaussian. For instance, one can easily obtain the *equipartition theorem*

$$\begin{aligned} \frac{\langle \boldsymbol{p}_i^2 \rangle}{2m} &= \frac{\int d^3p \frac{\boldsymbol{p}^2}{2m} e^{-\beta \frac{p^2}{2m}}}{\int d^3p e^{-\beta \frac{p^2}{2m}}} = -\frac{\partial}{\partial \beta} \ln \int d^3p e^{-\beta \frac{p^2}{2m}} = -\frac{\partial}{\partial \beta} \ln(2\pi m/\beta)^{3/2} \\ &= \frac{3}{2} \frac{\partial}{\partial \beta} \ln \beta = \frac{3}{2\beta} = \frac{3}{2} k_B T \end{aligned} \tag{1.23}$$

or

$$\frac{\langle (p_i^x)^2 \rangle}{2m} = \frac{\langle (p_i^y)^2 \rangle}{2m} = \frac{\langle (p_i^z)^2 \rangle}{2m} = \frac{k_B T}{2} \tag{1.24}$$

A more interesting case occurs when $A = A(\boldsymbol{r}^N)$ depends only on the particle positions:

$$\begin{aligned} \langle A(\boldsymbol{r}^N) \rangle &= \int d\boldsymbol{r}^N d\boldsymbol{p}^N f(\boldsymbol{r}^N, \boldsymbol{p}^N) A(\boldsymbol{r}^N) \\ &= \frac{1}{Q_{NVT}} \underbrace{\frac{1}{N!} \frac{1}{(2\pi\hbar)^{3N}} \int d\boldsymbol{p}^N e^{-\beta \sum_{i=1}^N \frac{\boldsymbol{p}_i^2}{2m}}}_{Q_{NVT}^{(id)}/V^N} \int d\boldsymbol{r}^N A(\boldsymbol{r}^N) e^{-\beta U(\boldsymbol{r}^N)} \\ &\overset{(1.11)}{=} \frac{\cancel{Q_{NVT}^{(id)}}}{\cancel{Q_{NVT}^{(id)}} Q_{NVT}^{(ex)} V^N} \int d\boldsymbol{r}^N A(\boldsymbol{r}^N) e^{-\beta U(\boldsymbol{r}^N)} \end{aligned} \tag{1.25}$$

If we (re-)define the *configuration integral*

$$Z := Q_{NVT}^{(ex)} V^N \overset{(1.12)}{=} \int d\boldsymbol{r}^N e^{-\beta U(\boldsymbol{r}^N)} \tag{1.26}$$

we obtain

$$\langle A(\boldsymbol{r}^N) \rangle = \frac{1}{Z} \int d\boldsymbol{r}^N A(\boldsymbol{r}^N) e^{-\beta U(\boldsymbol{r}^N)} \tag{1.27}$$

Integrals (1.26), (1.27) are highly non-trivial and difficult (usually impossible) to compute by analytical means. This is why we need MC simulations!
The idea is to replace the expectation value (1.27) with an average over a large sample of configurations generated with the *Metropolis procedure*. Its big advantage is that the *probability density does not need to be normalized*, because in the acceptance step we only need the ratio of probabilities in which any constant factor cancels. In other words, the sample can be produced using the unnormalized probability density $e^{-\beta U(\boldsymbol{r}^N)}$ while the normalization factor Z can remain undetermined

- Usually, Z is unknown and very difficult to calculate. Knowledge of Z corresponds to knowing the true free energy of the system, but for the moment this is not needed.

In many cases it is interesting to know not just the average of an observable. Also its *fluctuations* (variances and covariances) carry important physical information. Let us calculate $\frac{\partial^2(\beta F)}{\partial \beta^2}$ both thermodynamically as well as statistical-mechanically:

- statistical-mechanical calculation: starting from $\beta F = -\ln Q_{NVT}$, we get

$$\begin{aligned} \frac{\partial^2(\beta F)}{\partial \beta^2} &= -\frac{\partial^2 \ln Q_{NVT}}{\partial \beta^2} = -\frac{\partial}{\partial \beta}\left(\frac{1}{Q_{NVT}} \frac{\partial Q_{NVT}}{\partial \beta}\right) \\ &= \frac{1}{Q_{NVT}^2}\left(\frac{\partial Q_{NVT}}{\partial \beta}\right)^2 - \frac{1}{Q_{NVT}} \frac{\partial^2 Q_{NVT}}{\partial \beta^2} \end{aligned} \tag{1.28}$$

We have

$$\frac{\partial Q_{NVT}}{\partial \beta} = -\frac{1}{N!} \frac{1}{(2\pi\hbar)^{3N}} \int d\boldsymbol{r}^N d\boldsymbol{p}^N e^{-\beta H} H = -Q_{NVT} \langle H \rangle \tag{1.29}$$

$$\frac{\partial^2 Q_{NVT}}{\partial \beta^2} = \frac{1}{N!} \frac{1}{(2\pi\hbar)^{3N}} \int d\boldsymbol{r}^N d\boldsymbol{p}^N e^{-\beta H} H^2 = Q_{NVT} \langle H^2 \rangle \tag{1.30}$$

Therefore

$$\frac{\partial^2(\beta F)}{\partial \beta^2} = \langle H \rangle^2 - \langle H^2 \rangle = -\langle \Delta H^2 \rangle = -\left\langle (H - \langle H \rangle)^2 \right\rangle \tag{1.31}$$

where

$$\Delta H \equiv H - \langle H \rangle \tag{1.32}$$

denotes the deviation of H from its thermal average (i.e. thermal *energy fluctuations*).

- thermodynamical calculation: Recalling $\frac{\partial T}{\partial \beta} = \frac{\partial}{\partial \beta}\frac{1}{k_B \beta} = -\frac{1}{k_B \beta^2}$, we have

$$\begin{aligned} \frac{\partial^2(\beta F)}{\partial \beta^2} &= \frac{\partial}{\partial \beta}\left(\frac{\partial(\beta F)}{\partial \beta}\right) \overset{(1.15)}{=} \frac{\partial}{\partial \beta}\left(F + TS\right) = \frac{\partial}{\partial T}\left(F + TS\right)\frac{\partial T}{\partial \beta} \\ &= \left(\cancel{\frac{\partial F}{\partial T}} + \cancel{S} + T\frac{\partial S}{\partial T}\right)\left(-\frac{1}{k_B \beta^2}\right) \end{aligned} \tag{1.33}$$

i.e.

$$\frac{\partial^2(\beta F)}{\partial \beta^2} = T\frac{\partial S}{\partial T}\left(-k_B T^2\right) \tag{1.34}$$

Recall that in thermodynamics, the heat capacity at constant volume is defined as the heat δQ needed to change the temperature by δT, i.e.

$$C_V = \frac{\delta Q}{\delta T} \tag{1.35}$$

and since the (reversible) entropy change is defined as $dS = \frac{\delta Q}{T}$, one arrives at

$$C_V = T\left(\frac{\partial S}{\partial T}\right)_V \tag{1.36}$$

Hence

$$\frac{\partial^2(\beta F)}{\partial \beta^2} = -C_V k_B T^2 \tag{1.37}$$

Combining the two results, we find

$$C_V = \frac{1}{k_B T^2}\left[\langle H^2 \rangle - \langle H \rangle^2\right] \tag{1.38}$$

i.e. the specific heat is proportional to the fluctuations of the energy. This relation can be used to compute C_V in MC or MD simulations.

- The left hand side of this relation corresponds to the reaction of the system to a change in an external control variable (here β), while the right hand side measures the fluctuations of the freely evolving system. Eq. (1.38) is an example of a *fluctuation-response relation*.

Note that using $S = -\frac{\partial F}{\partial T}$, we can also write C_V as the second derivative

$$C_V = -T\frac{\partial^2 F}{\partial T^2} \tag{1.39}$$

Second derivatives of thermodynamics are generally related to *fluctuations*. Analogous expressions can be derived for the isochoral compressibility, the thermal expansion coefficient and others.

- Of course, these relations depend on the underlying ensemble; for instance, in the microcanonical NVE ensemble $\langle \Delta H \rangle = 0$ and the above relation is not applicable.

1.3.4 Equation of state

The *equation of state* is the relation

$$p = p(\rho, T) \tag{1.40}$$

between pressure p, density ρ and temperature T of a system. Here ρ is the *number density*

$$\rho = \frac{N}{V} \tag{1.41}$$

- We could just as well consider $p(N, T)$ at fixed V or $p(V, T)$ at fixed N.

We would like to express $p(\rho, T)$ as a canonical average. To accomplish that, we use the microscopic/macroscopic strategy applied before. We know that $p = -\frac{\partial F}{\partial V}$ and thus

$$\beta p = -\beta\frac{\partial F}{\partial V} = -\frac{\partial(\beta F)}{\partial V} = \frac{\partial \ln Q_{NVT}}{\partial V} = \frac{\partial \ln Q^{(id)}_{NVT}}{\partial V} + \frac{\partial \ln Q^{(ex)}_{NVT}}{\partial V} \tag{1.42}$$

The first term is the ideal gas term corresponding to the ideal gas equation of state

$$p^{id} = \rho k_B T \quad \Rightarrow \quad \beta p^{(id)} = \rho \tag{1.43}$$

Therefore we are left with

$$\beta p = \rho + \frac{1}{Q^{(ex)}_{NVT}}\frac{\partial Q^{(ex)}_{NVT}}{\partial V} \tag{1.44}$$

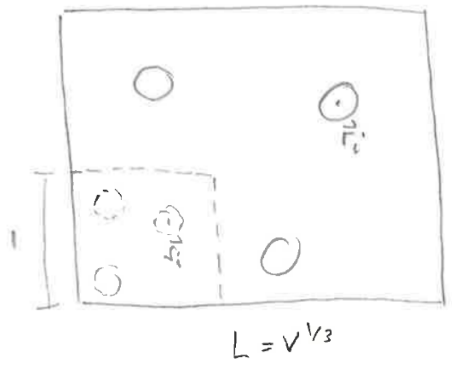
where the last term contains all the non-trivial contributions to the pressure. We realize that

$$Q^{(ex)}_{NVT} = \frac{1}{V^N}\int_{V^N} d\boldsymbol{r}^N e^{-\beta U(\boldsymbol{r}^N)} \tag{1.45}$$

depends on V both explicitly (through $1/V^N$) and implicitly (through the integration limits). To be able to carry out the differentiation w.r.t. the volume, we introduce *scaled coordinates* that allow to shrink and expand the volume conveniently. Let us write the coordinates as

$$\boldsymbol{r}_i = V^{1/3}\boldsymbol{s}_i, \qquad i = 1, \dots N \tag{1.46}$$

In this way, we rescale our coordinates to lie inside a box $\widehat{V}$ of unit volume. For a cubic box



Transforming from $\boldsymbol{r}_i$ to $\boldsymbol{s}_i$, we have to take into account the Jacobian

$$\left|\frac{\partial(\boldsymbol{r}_1, \boldsymbol{r}_2, \dots, \boldsymbol{r}_N)}{\partial(\boldsymbol{s}_1, \boldsymbol{s}_2, \dots, \boldsymbol{s}_N)}\right| = (V^{1/3})^{3N} = V^N \tag{1.47}$$

we rewrite the integral as

$$Q^{(ex)}_{NVT} = \frac{1}{\cancel{V^N}} \int_{\widehat{V}} d\boldsymbol{s}^N \cancel{V^N} e^{-\beta U((V^{1/3}\boldsymbol{s})^N)} \tag{1.48}$$

where $(V^{1/3}\boldsymbol{s})^N$ is a shorthand for $(V^{1/3}\boldsymbol{s}_1, \dots, V^{1/3}\boldsymbol{s}_N)$. This integral runs over the unit volume $\widehat{V}$. So, with scaled coordinates, the integration volume does not depend on V, and we can carry out the derivative with respect to V:

$$\frac{\partial Q^{(ex)}_{NVT}}{\partial V} = \int_{\widehat{V}} d\boldsymbol{s}^N e^{-\beta U((V^{1/3}\boldsymbol{s})^N)} \left(-\beta \frac{\partial U(V^{1/3}\boldsymbol{s}_1, \dots, V^{1/3}\boldsymbol{s}_N)}{\partial V}\right) \tag{1.49}$$

We compute

$$\begin{aligned}
-\frac{\partial U(V^{1/3}\boldsymbol{s}_1, \dots, V^{1/3}\boldsymbol{s}_N)}{\partial V} &= -\sum_{i=1}^{N} \frac{\partial U(V^{1/3}\boldsymbol{s}_1, \dots, V^{1/3}\boldsymbol{s}_N)}{\partial(V^{1/3}\boldsymbol{s}_i)} \frac{\partial V^{1/3}\boldsymbol{s}_i}{\partial V} \\
&= \sum_{i=1}^{N} \boldsymbol{F}_i((V^{1/3}\boldsymbol{s})^N) \frac{1}{3} V^{-2/3} \boldsymbol{s}_i \qquad (1.50)
\end{aligned}$$

where

$$\boldsymbol{F}_i((V^{1/3}\boldsymbol{s})^N) = -\frac{\partial U(V^{1/3}\boldsymbol{s}_1, \dots, V^{1/3}\boldsymbol{s}_N)}{\partial(V^{1/3}\boldsymbol{s}_i)} \tag{1.51}$$

is the force on particle i. So

$$\begin{aligned}\frac{\partial Q_{NVT}^{(ex)}}{\partial V} &= \beta \int_{\widehat{V}} d\boldsymbol{s}^N e^{-\beta U((V^{1/3}\boldsymbol{s})^N)} \sum_{i=1}^{N} \boldsymbol{F}_i((V^{1/3}\boldsymbol{s})^N)\frac{1}{3}V^{-2/3}\boldsymbol{s}_i \\ &= \beta \int_V \frac{d\boldsymbol{r}^N}{V^N} e^{-\beta U(\boldsymbol{r}^N)} \sum_{i=1}^{N} \boldsymbol{F}_i(\boldsymbol{r}^N)\frac{1}{3}V^{-2/3}\frac{\boldsymbol{r}_i}{V^{1/3}} \\ &= -\frac{\beta}{3V}\frac{1}{V^N}\int_V d\boldsymbol{r}^N e^{-\beta U(\boldsymbol{r}^N)} \left(-\sum_{i=1}^{N} \boldsymbol{F}_i(\boldsymbol{r}^N)\boldsymbol{r}_i\right) \qquad (1.52)\end{aligned}$$

In terms of the *virial*

$$W(\boldsymbol{r}^N) = -\sum_{i=1}^{N} \boldsymbol{F}_i(\boldsymbol{r}^N)\boldsymbol{r}_i \qquad (1.53)$$

we have shown that

$$\frac{1}{Q_{NVT}^{(ex)}}\frac{\partial Q_{NVT}^{(ex)}}{\partial V} = -\frac{\beta}{3V}\langle W\rangle \qquad (1.54)$$

and the equation of state turns into

$$p = \rho k_B T - \frac{1}{3V}\langle W\rangle \qquad (1.55)$$

- At first sight, the definition (1.53) of the virial seems to be problematic, since it appears to depend on the choice of origin $\mathbf{0}$ for the particle coordinates $\boldsymbol{r}_i$. However, suppose that we shift our coordinate system to a new origin $\mathbf{0}'$ by a constant vector $\boldsymbol{R}$. With respect to this shifted coordinate system $\boldsymbol{r}'_i = \boldsymbol{r}_i - \boldsymbol{R}$, and the virial is

  $$W(\boldsymbol{r}'^N) \overset{(1.53)}{=} -\sum_{i=1}^{N} \boldsymbol{F}'_i(\boldsymbol{r}'^N)\boldsymbol{r}'_i = W(\boldsymbol{r}^N) + \boldsymbol{R}\sum_{i=1}^{N} \boldsymbol{F}'_i(\boldsymbol{r}'^N) \qquad (1.56)$$

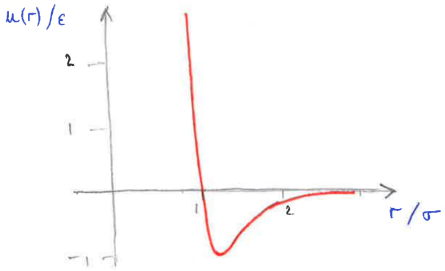
  Since there are only internal forces, the sum $\sum_{i=1}^{N} \boldsymbol{F}'_i(\boldsymbol{r}'^N) \equiv \mathbf{0}$ over all forces should vanish by Newton's third law, such that $W(\boldsymbol{r}'^N) \equiv W(\boldsymbol{r}^N)$.

After all, in the absence of an external field the pressure should depend only on relative coordinates, i.e. on the differences $\boldsymbol{r}_i - \boldsymbol{r}_j$. So we would like to rewrite the virial in a way that manifestly exhibits this dependence on relative coordinates. This task is particularly easy if the particles interact only through a *pairwise additive potential* that depends only on the relative distance

$$U(\boldsymbol{r}^N) = \sum_{i<j} u(r_{ij}) = \frac{1}{2}\sum_{i\neq j} u(r_{ij}), \qquad r_{ij} = |\boldsymbol{r}_{ij}| \tag{1.57}$$

In general, such a decomposition is not possible for a real material, but for some systems it may be a valid approximation.
A simple and frequently used pair potential to describe the noble gases like He, Ne, Ar, . . . is the *Lennard-Jones (LJ) potential*

$$u_{LJ}(r) := 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] \tag{1.58}$$



The parameters ϵ and σ depend on the substance one considers:

- Parameter σ can be viewed as a measure of the diameter of the (spherically assumed) particles.

- The LJ potential has a minimum at $r = 2^{1/6}\sigma$.

- Parameter ϵ, which equals the depth of the well at this minimum, describes the *strength* of the potential.

- For $r > 2^{1/6}\sigma$, the LJ potential is *attractive.* The r^{-6} form of this attractive part is motivated by the quantum-mechanical analysis of the fluctuation-induced dipole interaction (the so-called *dispersion* or *Van der Waals* forces.)

- For $r < 2^{1/6}\sigma$, it is *repulsive.* The r^{-12} form, which should account for the Pauli exclusion principle of electrons is chosen purely for computational convenience.

In the case of a pairwise additive potential, the total force $\boldsymbol{F}_i = -\boldsymbol{\nabla}_{\boldsymbol{r}_i} U(\boldsymbol{r}^N)$ becomes

$$\boldsymbol{F}_i = -\frac{1}{2} \sum_{\substack{j,l \\ j \neq l}} \boldsymbol{\nabla}_{\boldsymbol{r}_i} u(r_{jl}) \tag{1.59}$$

Since we differentiate w.r.t. $\boldsymbol{r}_i$, only terms for which either j or l equals i are different from zero. Therefore, the above double sum reduces to two single sums

$$\boldsymbol{F}_i = -\frac{1}{2} \left\{ \sum_{\substack{l \\ l \neq i}} \boldsymbol{\nabla}_{\boldsymbol{r}_i} u(r_{il}) + \sum_{\substack{j \\ j \neq i}} \boldsymbol{\nabla}_{\boldsymbol{r}_i} u(r_{ji}) \right\} \tag{1.60}$$

and since

$$\begin{aligned} \boldsymbol{\nabla}_{\boldsymbol{r}_i} u(r_{il}) \quad &= \quad u'(r_{il}) \boldsymbol{\nabla}_{\boldsymbol{r}_i} r_{il} = u'(r_{il}) \boldsymbol{\nabla}_{\boldsymbol{r}_i} \sqrt{(\boldsymbol{r}_l - \boldsymbol{r}_i)^2} = u'(r_{il}) \frac{\cancel{2}(\boldsymbol{r}_l - \boldsymbol{r}_i)(-1)}{\cancel{2}\sqrt{(\boldsymbol{r}_l - \boldsymbol{r}_i)^2}} \\ &= \quad -u'(r_{il}) \frac{\boldsymbol{r}_{il}}{r_{il}} \end{aligned} \tag{1.61}$$

and similarly

$$\boldsymbol{\nabla}_{\boldsymbol{r}_i} u(r_{ji}) = u'(r_{ji}) \frac{\boldsymbol{r}_{ji}}{r_{ji}} = -u'(r_{ij}) \frac{\boldsymbol{r}_{ij}}{r_{ij}} \tag{1.62}$$

Thus, the total force on particle i

$$\boldsymbol{F}_i = -\frac{1}{2} \left\{ -\sum_{\substack{l \\ l \neq i}} u'(r_{il}) \frac{\boldsymbol{r}_{il}}{r_{il}} - \sum_{\substack{j \\ j \neq i}} u'(r_{ij}) \frac{\boldsymbol{r}_{ij}}{r_{ij}} \right\} = \sum_{\substack{j \\ j \neq i}} u'(r_{ij}) \frac{\boldsymbol{r}_{ij}}{r_{ij}} \tag{1.63}$$

can be written as a sum of *pair forces*

$$\boldsymbol{F}_i = \sum_{\substack{j \\ j \neq i}} \boldsymbol{f}_{ij} \tag{1.64}$$

where

$$\boldsymbol{f}_{ij} = u'(r_{ij}) \frac{\boldsymbol{r}_{ij}}{r_{ij}} \tag{1.65}$$

is the *force exerted by particle* j *on particle* i. This formula explicitly shows that $\boldsymbol{f}_{ij} = -\boldsymbol{f}_{ji}$, illustrating Newton's principle of "actio=reactio", i.e. pair forces are of equal magnitude but opposite directions.
Using these results, we obtain the alternative representation

$$\begin{aligned} W(\boldsymbol{r}^N) &= -\sum_i \boldsymbol{r}_i \cdot \boldsymbol{F}_i = -\sum_i \boldsymbol{r}_i \cdot \sum_{\substack{j \\ j\neq i}} \boldsymbol{f}_{ij} = -\sum_{i\neq j} \boldsymbol{r}_i \cdot \boldsymbol{f}_{ij} \\ &\overset{(i\leftrightarrow j)}{=} -\sum_{i\neq j} \boldsymbol{r}_j \cdot \underbrace{\boldsymbol{f}_{ji}}_{-\boldsymbol{f}_{ij}} = \sum_{i\neq j} \boldsymbol{r}_j \cdot \boldsymbol{f}_{ij} \qquad (1.66) \end{aligned}$$

of the virial. Symmetrizing, we arrive at

$$\begin{aligned} W(\boldsymbol{r}^N) &= \frac{1}{2}\left[W(\boldsymbol{r}^N) + W(\boldsymbol{r}^N)\right] = \frac{1}{2}\left[-\sum_{i\neq j} \boldsymbol{r}_i \cdot \boldsymbol{f}_{ij} + \sum_{i\neq j} \boldsymbol{r}_j \cdot \boldsymbol{f}_{ij}\right] \\ &= \frac{1}{2}\sum_{i\neq j} \underbrace{(\boldsymbol{r}_j - \boldsymbol{r}_i)}_{\boldsymbol{r}_{ij}} \cdot \boldsymbol{f}_{ij} \qquad (1.67) \end{aligned}$$

and since the scalar product $\boldsymbol{r}_{ij}\boldsymbol{f}_{ij} = \boldsymbol{r}_{ji}\boldsymbol{f}_{ji}$ is symmetric in the indices i, j, we can write the virial as a sum over ordered pairs of particles (like we did for the energy) that, as promised, manifestly only depends on relative coordinates:

$$W(\boldsymbol{r}^N) = \frac{1}{2}\sum_{i\neq j} \boldsymbol{r}_{ij} \cdot \boldsymbol{f}_{ij} = \sum_{i<j} \boldsymbol{r}_{ij} \cdot \boldsymbol{f}_{ij} \qquad (1.68)$$

In fact, for spherically symmetric particles the pair forces $\boldsymbol{f}_{ij}$ are parallel to $\boldsymbol{r}_{ij}$ (see the derivation Eq. (1.63)), such that this simplifies further to

$$W(\boldsymbol{r}^N) = \sum_{i<j} \boldsymbol{r}_{ij} \cdot \boldsymbol{f}_{ij} = \sum_{i<j} \boldsymbol{r}_{ij} \cdot u'(r_{ij})\frac{\boldsymbol{r}_{ij}}{r_{ij}} = \sum_{i<j} u'(r_{ij})\frac{r_{ij}^2}{r_{ij}} \qquad (1.69)$$

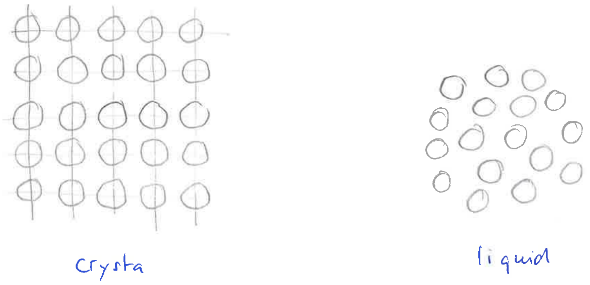
i.e.

$$W(\boldsymbol{r}^N) = \sum_{i<j} r_{ij}u'(r_{ij}) \qquad (1.70)$$

which depends only on scalar quantities.

1.3.5 Pair distribution function

Consider a *gas* or a *liquid* (the term *fluid* includes both of them). In contrast to a crystalline solid, where atoms are arranged on a lattice with long-ranged order, atoms in a fluid lack long-range order:



Nevertheless, a liquid is not completely disordered, simply because a place occupied by a particle cannot be taken by another particle. This leads to a certain short-ranged structural order in a liquid, which can be quantitatively described with the *pair distribution function* $g(r)$ (also called *radial distribution function* or *pair correlation function*).

$g(r)$ is important for a number of reasons:

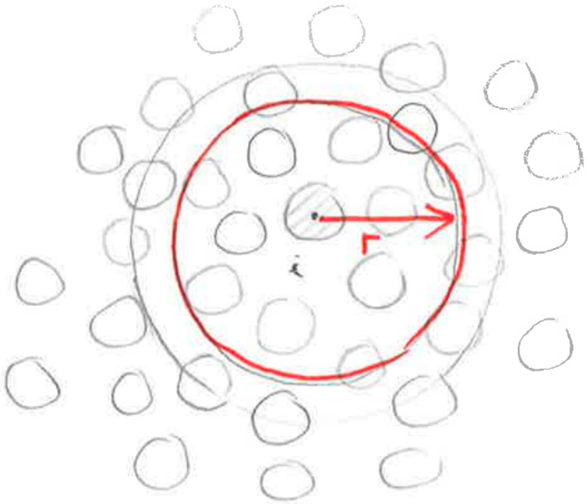
- It characterizes local structures around an atom. It can be measured via the structure factor $S(\boldsymbol{k})$ which can be determined via x-ray and neutron scattering.

- Thermodynamic properties of systems with additive pair potentials are completely determined by $g(r)$.

- There are theoretical approaches that allow an approximate analytical calculation of $g(r)$.

Definition of $g(r)$ for a system of atoms.

- Consider a system of N atoms and pick particle i as *reference particle.*



- Count how many particles there are in a sphere of radius r around particle i. In terms of the Heaviside step function $\theta_H(x)$, the average number of such particles can be written as

$$n(r) = \left\langle \sum_{\substack{j \\ j \neq i}} \theta_H(r - r_{ij}) \right\rangle \tag{1.71}$$

- Taking the derivative of $n(r)$ with respect to r gives

$$\frac{dn(r)}{dr} = \left\langle \sum_{\substack{j \\ j \neq i}} \delta(r - r_{ij}) \right\rangle \tag{1.72}$$

  where $\delta(x)$ is the Dirac delta distribution. The average number of particles in a thin shell of thickness dr is then given by

$$\frac{dn(r)}{dr} dr \approx \left\langle \sum_{\substack{j \\ j \neq i}} \delta(r - r_{ij}) \right\rangle dr \tag{1.73}$$

- In a completely disordered system with a homogeneous density ρ of particles (an ideal gas), the corresponding number of particles in such a shell would be given by

$$\frac{dn(r)}{dr} dr \approx \rho \cdot 4\pi r^2 dr \tag{1.74}$$

  The pair correlation function describes the *relative deviation of the number of particles at distance r compared to this completely homogeneous* *distribution:*

$$g(r) = \frac{1}{4\pi r^2 \rho} \left\langle \sum_{\substack{j \\ j\neq i}} \delta(r - r_{ij}) \right\rangle \tag{1.75}$$

- Thus, for the ideal gas $g(r) = 1$.

- Since all particles are equivalent, we can average over the N different choices of reference particle i, such that we arrive at

$$g(r) = \frac{1}{4\pi r^2 \rho N} \left\langle \sum_{i\neq j} \delta(r - r_{ij}) \right\rangle = \frac{1}{2\pi r^2 \rho N} \left\langle \sum_{i<j} \delta(r - r_{ij}) \right\rangle \tag{1.76}$$

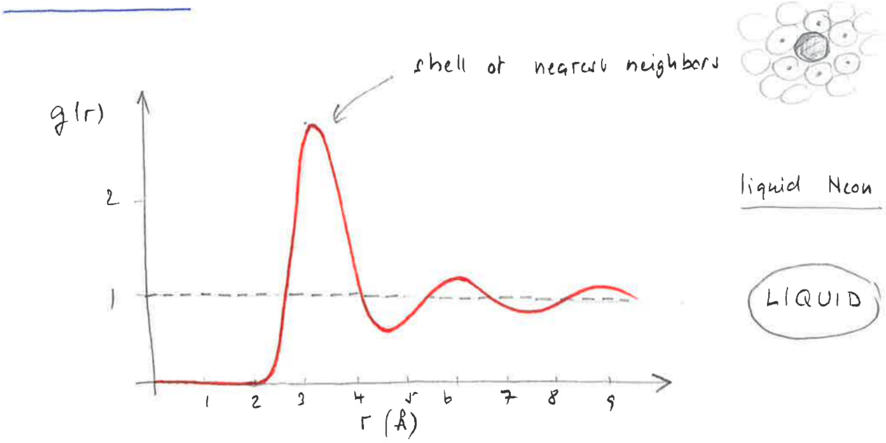
Integration over r from zero to infinity yields

$$4\pi\rho \int_0^\infty dr\, r^2 g(r) = N - 1 \tag{1.77}$$

which is just the number of all particles minus the reference particle. If we instead integrate only up to a certain finite radius r, we obtain the number of particles in the sphere with radius r minus the reference particle:

$$n(r) = 4\pi\rho \int_0^r dr'\, r'^2 g(r') \tag{1.78}$$

Typical form



We can regard $g(r)$ as a measure of the *local density* at distance r from the reference particle:

- If $g(r) < 1$ it is lower than ρ. In particular, $g(r) \equiv 0$ for small r, because the presence of the reference particle at the origin excludes all other particles from being there.

- If $g(r) > 1$ the local density is higher than the macroscopic density $\rho = \frac{N}{V}$.

- Particles in the *first shell* (or *first coordination shell*) around the reference particle all have approximately the same distance from it, so there is a maximum. These particles are the *nearest neighbors* of the reference particle.

- For growing r, $g(r)$ decreases again below 1 because of similar exclusion effects (particles cannot be in the space already occupied by those in the first shell). The maximum corresponding to the second shell is lower because as r increases the order gets successively smeared out.

- Depending on the thermodynamic state there may be more maxima and minima. As a rule, these are more distinct at high ρ and low T.

- For $r \to \infty$, correlations should die out completely such that we expect $g(r) \to 1$ in this limit. This, however, is strictly true only in the *grand-canonical ensemble*, where N can fluctuate. In the canonical ensemble, we instead obtain

  $$\lim_{r\to\infty} g(r) = 1 + O(1/N) \tag{1.79}$$

  which only approaches 1 in the thermodynamic limit.

So far, we have assumed that particles are arranged isotropically, i.e. that there are no preferred directions. Accordingly, $g(r)$ depends only on the distance r but not on the direction $\boldsymbol{r}$. One can, however, also define a direction-dependent pair correlation function

$$g(\boldsymbol{r}) := \frac{1}{\rho N}\left\langle \sum_{\substack{j\\ j\neq i}} \delta^3(\boldsymbol{r} - \boldsymbol{r}_{ij}) \right\rangle \tag{1.80}$$

In a slightly more formal way, the pair correlation function can be introduced via the *two-particle density*

$$g^{(2)}(\boldsymbol{r}_1, \boldsymbol{r}_2) = N(N-1)\frac{1}{Z}\int d\boldsymbol{r}_3 \ldots d\boldsymbol{r}_N e^{-\beta U(\boldsymbol{r}^N)} \tag{1.81}$$

- Note that $N(N-1) = \frac{N!}{(N-2)!}$

This is (essentially) the *marginal distribution* where only the positions of particles 1 and 2 are of interest and all other degrees of freedom have been integrated out. If the system is not subject to an external potential, the two-particle density depends only on the difference $\boldsymbol{r}_{12} = \boldsymbol{r}_2 - \boldsymbol{r}_1$, and so

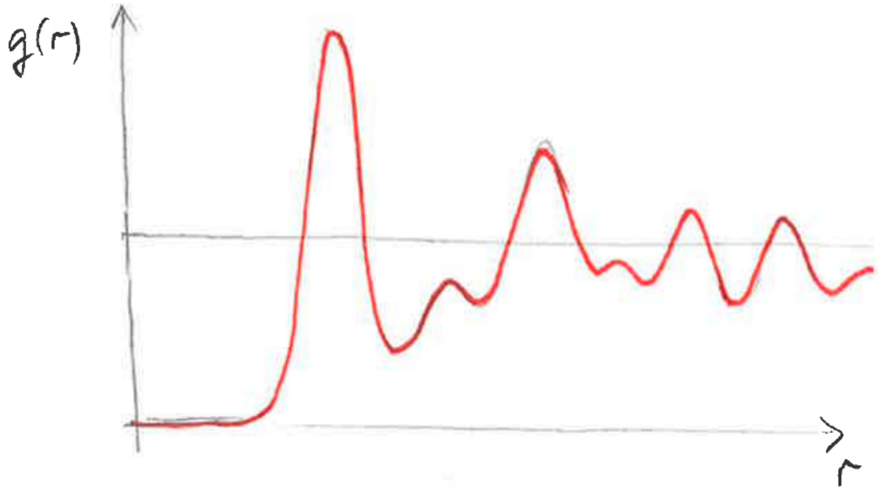
$$g^{(2)}(\boldsymbol{r}_1, \boldsymbol{r}_2) = g^{(2)}(\boldsymbol{r}_{12}) \tag{1.82}$$

The pair correlation function is then defined as

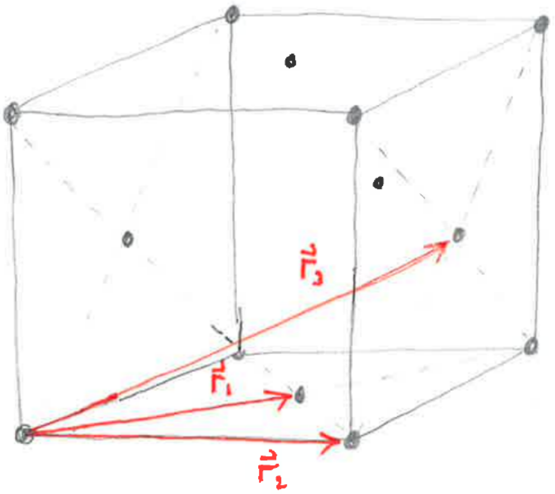
$$g(\boldsymbol{r}_{12}) = \frac{1}{\rho^2} g^{(2)}(\boldsymbol{r}_1, \boldsymbol{r}_2) \tag{1.83}$$

In fact, in analogy to the two-particle density, there exists a whole hierarchy of n-particle densities $\rho^{(n)}(\boldsymbol{r}_1, \ldots, \boldsymbol{r}_n)$, which are also obtained by marginalizing with respect to particle positions.

For a *solid*, $g = g(\boldsymbol{r})$ depends on magnitude *and* direction of the distance vector $\boldsymbol{r}$. Nevertheless, one often considers the pair correlation function obtained by averaging over all directions. Such a $g(r)$ for a crystal might look like this:



For a crystal, $g(r)$ exhibits more peaks than for a liquid, and they are more pronounced. The positions of these peaks depend, of course, on the particular structure of the crystal.



fcc crystal

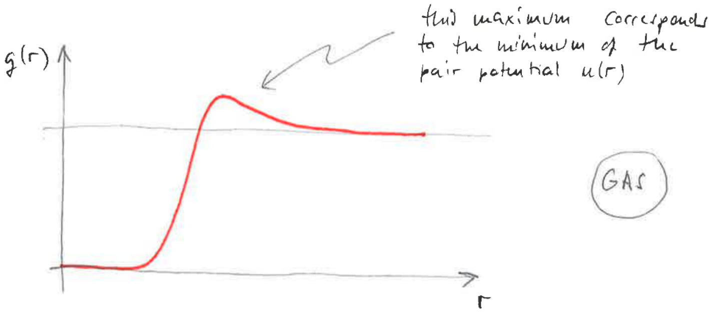
In an fcc crystal, for example, the typical distances of particles corresponding to the first three peaks in $g(r)$ have ratios

$$r_1 : r_2 : r_3 = 1 : \sqrt{2} : \sqrt{3} \tag{1.84}$$

The finite width of the peaks is due to the thermal fluctuations of the particles around the sites of the perfect lattice.
At low density, i.e. for a dilute gas, the pair correlation function allows for a simple approximation. In fact, to first order in the density, $g(r)$ is given simply by the Boltzmann factor of the pair potential:

$$g(r) \approx e^{-\beta u(r)} \tag{1.85}$$

- This result follows from Eq. (1.81) under the assumption that when two particles interact, the others are far away. A derivation can be found in J.P. Hansen and I.R. McDonald, "Theory of simple liquids", Academic Press.



For high temperature and low density, $g(r)$ thus becomes a step function.

1.3.6 Virial equation of state

$g(r)$ contains a lot of thermodynamic information about a given system. For instance, based on the knowledge of $g(r)$, one can compute the expectation of any observable

$$A(\boldsymbol{r}^N) = \sum_{i<j} a(r_{ij}) = \frac{1}{2}\sum_{i\neq j} a(r_{ij}) \tag{1.86}$$

that can be written as a sum of pair contributions $a(r_{ij})$, *irrespective of whether the interactions are pairwise additive or not*! To see this, we rewrite the expectation of $A(\boldsymbol{r}^N)$ by inserting 1 in the form of an integral over a delta function

$$\begin{aligned}\langle A(\boldsymbol{r}^N)\rangle &= \left\langle \frac{1}{2}\sum_{i\neq j} a(r_{ij})\right\rangle = \left\langle \frac{1}{2}\sum_{i\neq j}\int_0^\infty dr\, a(r_{ij})\delta(r-r_{ij})\right\rangle \\ &= \frac{1}{2}\int_0^\infty dr\, a(r)\left\langle \sum_{i\neq j}\delta(r-r_{ij})\right\rangle \end{aligned} \tag{1.87}$$

Recalling the definition

$$g(r) \overset{(1.76)}{=} \frac{1}{4\pi r^2 \rho N}\left\langle \sum_{i\neq j}\delta(r-r_{ij})\right\rangle \tag{1.88}$$

we obtain

$$\langle A(\boldsymbol{r}^N)\rangle = \int_0^\infty dr\, a(r) 2\pi r^2 \rho N g(r) \tag{1.89}$$

If A is an *extensive* variable, the average of A per particle is given by

$$\frac{\langle A(\boldsymbol{r}^N)\rangle}{N} = 2\pi\rho\int_0^\infty dr\, a(r) r^2 g(r) \tag{1.90}$$

Example: pairwise additive interactions. If the interactions are pairwise additive, i.e. if

$$U(\boldsymbol{r}^N) = \sum_{i<j} u(r_{ij}) \tag{1.91}$$

and thus the virial is

$$W = \sum_{i<j} w(r_{ij}), \qquad w(r) = r\, u'(r) \tag{1.92}$$

then the average potential energy and virial per particle are given by

$$
\begin{aligned}
\frac{\langle U\rangle}{N} &= 2\pi\rho\int_0^\infty dr\, r^2 g(r)u(r) \qquad (1.93)\\
\frac{\langle W\rangle}{N} &= 2\pi\rho\int_0^\infty dr\, r^2 g(r)[ru'(r)] \qquad (1.94)
\end{aligned}
$$

For pairwise additive potentials, the general form of the equation of state can therefore be rewritten explicitly as

$$
p \overset{(1.55)}{=} \rho k_B T - \frac{1}{3V}\langle W\rangle = \rho k_B T - \frac{2\pi\rho N}{3V}\int_0^\infty dr\, r^2 g(r)[ru'(r)] \qquad (1.95)
$$

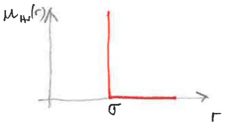
i.e. as the *virial equation of state*

$$
p = \rho k_B T - \frac{2\pi\rho^2}{3}\int_0^\infty dr\, r^2 g(r)[ru'(r)] \qquad (1.96)
$$

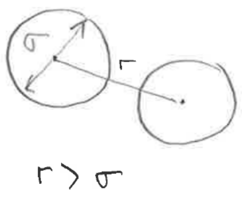
1.3.7 Hard spheres equation of state

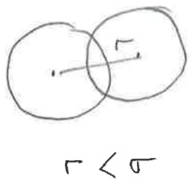
For hard spheres (HSs), the pair potential is given by

$$
u_\sigma(r) = \begin{cases} \infty, & r < \sigma \\ 0, & r > \sigma \end{cases} \qquad (1.97)
$$



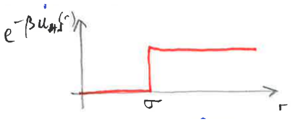
σ is the diameter of the spheres. This particular form of the pair interaction potential simply means that in the N-particle system any particle configuration is forbidden that contains two or more *overlapping* particles, while all other configurations are assigned the same energy.





Since the potential $u_\sigma(r)$ is discontinuous at $r = \sigma$, we cannot directly apply the virial equation of state. Note, however, that the Boltzmann factor of the HS potential is just the Heaviside unit step function shifted to $r = \sigma$:

$$e^{-\beta u_\sigma(r)} = \begin{cases} 0, & r < \sigma \\ 1, & r > \sigma \end{cases} = \theta_H(r-\sigma) \quad (1.98)$$



Thus we have (in the distribution sense)

$$\delta(r-\sigma) = \frac{d}{dr}\theta_H(r-\sigma) = \frac{d}{dr}\left[e^{-\beta u_\sigma(r)}\right] = -\beta u_\sigma'(r) e^{-\beta u_\sigma(r)} \tag{1.99}$$

Therefore, we can express

$$u_\sigma'(r) = -k_B T e^{\beta u_\sigma(r)} \delta(r-\sigma) \tag{1.100}$$

Plugging this expression in the virial equation of state (1.96), we obtain

$$\begin{aligned} \frac{p}{k_B T \rho} &\overset{(1.96)}{=} 1 - \frac{2\pi\rho}{3k_B T}\int_0^\infty dr\, r^2 g_\sigma(r)[r u_\sigma'(r)] \\ &\overset{(1.100)}{=} 1 + \frac{2\pi\rho}{3}\int_0^\infty dr\, r^3 g_\sigma(r) e^{\beta u_\sigma(r)} \delta(r-\sigma) \end{aligned} \tag{1.101}$$

We introduce the so-called *cavity function* $y_\sigma(r)$ for the hard sphere system. For a general pair potential $u(r)$, the corresponding cavity function $y(r)$ is defined by

$$y(r) := g(r) e^{\beta u(r)} \tag{1.102}$$

In terms of $y_\sigma(r)$, Eqn. (1.101) becomes

$$\frac{p}{k_B T \rho} = 1 + \frac{2\pi\rho}{3}\int_0^\infty dr\, r^3 y_\sigma(r) \delta(r-\sigma) \tag{1.103}$$

The important point to recognize is now that even if the correlation function or the potential $u(r)$ happen to be discontinuous, the cavity function $y(r)$ still *is continuous.*

- *Proof.* In fact, recall the formal definition of $g(r_{12})$ via Eqs. (1.81) and (1.83)

$$g(r_{12}) = \rho^{-2}\frac{N!}{(N-2)!}\frac{1}{Z}\int d\boldsymbol{r}_3 \dots d\boldsymbol{r}_N\, e^{-\beta \sum\limits_{i<j} u(r_{ij})} \tag{1.104}$$

Since the integration does not run over $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$, the factor $e^{-\beta u(r_{12})}$ can be pulled out of the integral:

$$g(r_{12}) = \frac{N!}{(N-2)!}\frac{e^{-\beta u(r_{12})}}{Z}\rho^{-2}\int d\boldsymbol{r}_3 \dots d\boldsymbol{r}_N\, e^{-\beta \sum\limits_{i<j}' u(r_{ij})} \tag{1.105}$$

Here $\sum'_{i<j}$ indicates a sum without the term $(i = 1, j = 2)$. This formula shows that any discontinuity in $g(r_{12})$ must originate from a discontinuity of $e^{-\beta u(r_{12})}$. In the cavity function

$$y(r_{12}) = g(r_{12})e^{\beta u(r_{12})} = \frac{N!}{(N-2)!}\frac{\rho^{-2}}{Z}\int d\boldsymbol{r}_3 \ldots d\boldsymbol{r}_N \, e^{-\beta \sum'_{i<j} u(r_{ij})} \tag{1.106}$$

precisely this factor has, however, been absorbed. ✓

We may therefore evaluate the delta function in (1.103), which yields

$$\frac{p_\sigma}{\rho k_B T} = 1 + \frac{2\pi\rho\sigma^3}{3} y_\sigma(\sigma) \tag{1.107}$$

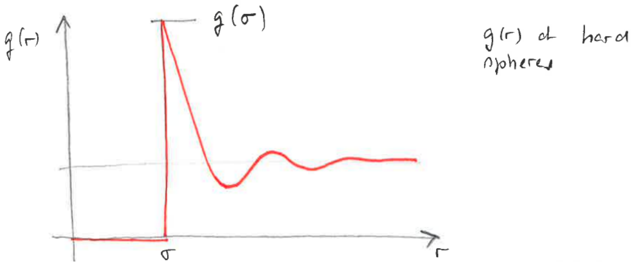
For the hard sphere potential, however, we obviously have

$$y_\sigma(r) = g_\sigma(r) \qquad \forall\, r > \sigma \tag{1.108}$$

Thus, we conclude that due to continuity

$$y_\sigma(\sigma) = \lim_{r\to\sigma} y_\sigma(r) = \lim_{r\searrow\sigma} g_\sigma(r) \equiv g_\sigma(\sigma^+) \tag{1.109}$$

The $g_\sigma(\sigma^+)$ value on the right hand side is known as the *pair correlation function at contact.*



We therefore finally are in the position to evaluate the delta function in (1.103), which yields the hard sphere equation of state

$$\frac{p_\sigma}{\rho k_B T} = 1 + \frac{2\pi\sigma^3\rho}{3} g_\sigma(\sigma^+) \tag{1.110}$$

Since the volume of one hard sphere of diameter σ is $\frac{4\pi}{3}\left(\frac{\sigma}{2}\right)^3 = \frac{\pi}{6}\sigma^3$, the *volume fraction* ϕ_σ of these spheres at number density ρ is

$$\phi_\sigma = \frac{\pi}{6}\sigma^3\rho \tag{1.111}$$

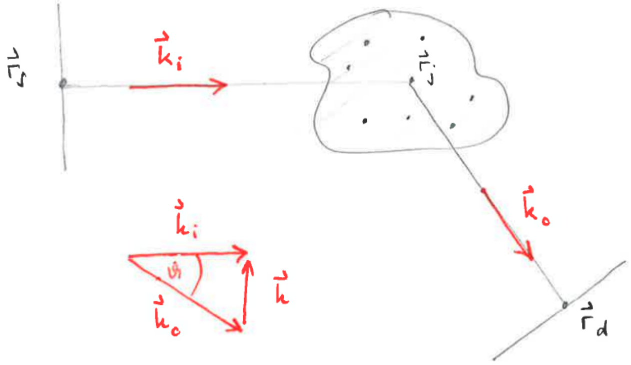
The *hard sphere equation of state* then takes the form

$$\frac{p_\sigma}{\rho k_B T} = 1 + 4\phi_\sigma g_\sigma(\sigma^+) \qquad (1.112)$$

This expression is also known as the *contact formula*.

1.3.8 The structure factor

The pair correlation function $g(r)$ can be obtained from x-ray or neutron scattering experiments. In such experiments, a *monochromatic* wave hits a sample with N scattering centers located at positions $\boldsymbol{r}_j$. Each of these scattering centers leads to a *spherical wave* propagating outward and interfering constructively or destructively with the spherical waves coming from the other scattering centers.



We assume that both

- the position $\boldsymbol{r}_s$ of the source
- the position $\boldsymbol{r}_d$ of the detector

are located sufficiently far away from the sample, such that the incoming waves hitting the sample as well as the outgoing wave reaching the detector can be approximated as plane waves. Denoting the incoming and outcoming wave vector by $\boldsymbol{k}_i$ and $\boldsymbol{k}_o$, respectively, we have

$$k_i = |\boldsymbol{k}_i| = \frac{2\pi}{\lambda} = \begin{cases} p/\hbar, & \text{for neutrons} \\ \omega/c, & \text{for x-rays} \end{cases} \qquad (1.113)$$

The amplitude A of the radiation reaching the detector

$$\begin{aligned} A &\propto \sum_{j=1}^{N} e^{i\boldsymbol{k}_i \overbrace{(\boldsymbol{r}_j - \boldsymbol{r}_s)}^{\text{incoming optical path length}}} \times e^{i\boldsymbol{k}_o \overbrace{(\boldsymbol{r}_d - \boldsymbol{r}_j)}^{\text{outgoing optical path length}}} \\ &= e^{-i\boldsymbol{k}_i\boldsymbol{r}_s} e^{i\boldsymbol{k}_o\boldsymbol{r}_d} \sum_{j=1}^{N} e^{i(\boldsymbol{k}_i - \boldsymbol{k}_o)\boldsymbol{r}_j} \end{aligned} \tag{1.114}$$

only depends on the relative phases of the contributions coming from the scattering centers. For *elastic scattering* defined by

$$k_i = k_o = \frac{2\pi}{\lambda} \tag{1.115}$$

having the same wave length λ, the wave vectors $\boldsymbol{k}_i$, $\boldsymbol{k}_o$ and the *scattering vector*

$$\boldsymbol{k} = \boldsymbol{k}_i - \boldsymbol{k}_o \tag{1.116}$$

form an isosceles triangle, and the *scattering angle* θ is related to $k = |\boldsymbol{k}|$ by

$$k^2 = (\boldsymbol{k}_i - \boldsymbol{k}_o)^2 = k_i^2 - 2\boldsymbol{k}_i\boldsymbol{k}_o + k_o^2 = \left(\frac{2\pi}{\lambda}\right)^2 2\underbrace{(1 - \cos\theta)}_{2\sin^2\frac{\theta}{2}} \tag{1.117}$$

i.e. the *Bragg law*

$$k = \frac{4\pi}{\lambda}\sin\frac{\theta}{2} \tag{1.118}$$

The *intensity* of the radiation arriving at the detector is proportional to the square of the amplitude

$$I(\boldsymbol{k}) \propto \left|\sum_{j=1}^{N} e^{i\boldsymbol{k}\boldsymbol{r}_j}\right|^2 = \sum_{j,l=1}^{N} e^{i\boldsymbol{k}\boldsymbol{r}_j} e^{-i\boldsymbol{k}\boldsymbol{r}_l} = \sum_{j,l=1}^{N} e^{-i\boldsymbol{k}\boldsymbol{r}_{jl}} \tag{1.119}$$

where $\boldsymbol{r}_{jl} = \boldsymbol{r}_l - \boldsymbol{r}_j$ is the vector going from particle j to particle l.
For a statistical-mechanical many-particle system we now have to average this appropriately over all the positions of the scattering centers. The *instantaneous particle number density* of the fluid under consideration is a sum

$$\rho(\boldsymbol{r}) = \sum_{j=1}^{N} \delta^3(\boldsymbol{r} - \boldsymbol{r}_j) \tag{1.120}$$

of δ-functions at the particle positions. The Fourier transform of this particle number density is

$$\rho_{\boldsymbol{k}} \;=\; \int d\boldsymbol{r} e^{-i\boldsymbol{k}\boldsymbol{r}} \rho(\boldsymbol{r}) = \sum_{j=1}^{N} \int d\boldsymbol{r} e^{-i\boldsymbol{k}\boldsymbol{r}} \delta^3(\boldsymbol{r} - \boldsymbol{r}_j) = \sum_{j=1}^{N} e^{-i\boldsymbol{k}\boldsymbol{r}_j} \tag{1.121}$$

The *structure factor* is defined as the statistical average of the squared modulus of $\rho_{\boldsymbol{k}}$ per particle, i.e.

$$S(\boldsymbol{k}) \equiv \frac{1}{N}\left\langle \rho_{\boldsymbol{k}}\rho_{-\boldsymbol{k}}\right\rangle = \frac{1}{N}\left\langle \sum_{j,l=1}^{N} e^{-i\boldsymbol{k}\boldsymbol{r}_j} e^{i\boldsymbol{k}\boldsymbol{r}_l}\right\rangle = \frac{1}{N}\left\langle \sum_{j,l=1}^{N} e^{-i\boldsymbol{k}\boldsymbol{r}_{lj}}\right\rangle \tag{1.122}$$

or

$$S(\boldsymbol{k}) = 1 + \frac{1}{N}\left\langle \sum_{\substack{j,l=1 \\ j\neq l}}^{N} e^{-i\boldsymbol{k}\boldsymbol{r}_{lj}}\right\rangle \tag{1.123}$$

Up to the prefactor (and the average), this expression is identical to Eq. (1.119) for the intensity $I(\boldsymbol{k})$, i.e.

$$S(\boldsymbol{k}) \propto I(\boldsymbol{k}) \tag{1.124}$$

Since $S(\boldsymbol{k})$ is a statistical average of a sum of pair contributions, we can express it in terms of $g(r)$. In fact, recalling that

$$g(\boldsymbol{r}) \overset{(1.80)}{=} \frac{1}{\rho N}\left\langle \sum_{\substack{j=1 \\ j\neq i}}^{N} \delta^3(\boldsymbol{r}-\boldsymbol{r}_{ij})\right\rangle \tag{1.125}$$

we derive

$$\begin{aligned}\int d\boldsymbol{r}\, e^{-i\boldsymbol{k}\boldsymbol{r}} g(\boldsymbol{r}) &= \frac{1}{\rho N}\int d\boldsymbol{r}\, e^{-i\boldsymbol{k}\boldsymbol{r}}\left\langle \sum_{\substack{j=1 \\ j\neq i}}^{N} \delta^3(\boldsymbol{r}-\boldsymbol{r}_{ij})\right\rangle \\ &= \frac{1}{\rho N}\left\langle \sum_{\substack{j=1 \\ j\neq i}}^{N} e^{-i\boldsymbol{k}\boldsymbol{r}_{lj}}\right\rangle \end{aligned} \tag{1.126}$$

Comparing Eqs. (1.123) and (1.126), we conclude that

$$S(\boldsymbol{k}) \equiv 1 + \rho \int d\boldsymbol{r} e^{-i\boldsymbol{k}\boldsymbol{r}} g(\boldsymbol{r}) \tag{1.127}$$

The structure factor is therefore essentially the Fourier transform of the pair correlation function. However, the above expression contains a singularity, because for $r \to \infty$ we know that $g(r) \to 1$, for which there is a contribution

$$\int d\boldsymbol{r} e^{-i\boldsymbol{k}\boldsymbol{r}} 1 = (2\pi)^3 \delta^3(\boldsymbol{k}) \tag{1.128}$$

to the Fourier transform that corresponds to forward ($\boldsymbol{k} = \boldsymbol{0}$) scattering and needs to be isolated. We therefore define a *reduced structure factor*

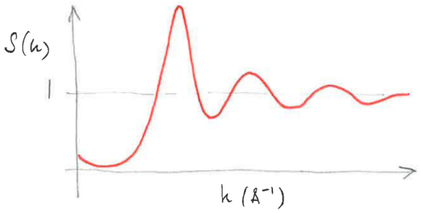
$$\widehat{S}(\boldsymbol{k}) := S(\boldsymbol{k}) - (2\pi)^3\delta^3(\boldsymbol{k}) = 1 + \rho \int d\boldsymbol{r} e^{-i\boldsymbol{k}\boldsymbol{r}} \left[g(\boldsymbol{r}) - 1\right] \tag{1.129}$$

Inverting this formula, we obtain

$$g(\boldsymbol{r}) = 1 + \frac{1}{(2\pi)^3\rho} \int d\boldsymbol{k} e^{i\boldsymbol{k}\boldsymbol{r}} \left[\widehat{S}(\boldsymbol{k}) - 1\right] \tag{1.130}$$

For an *isotropic* fluid, $g(\boldsymbol{r})$ and $\widehat{S}(\boldsymbol{k})$ depend only on the magnitude of their arguments. Transformation to spherical coordinates and integration over the angular variable yields

$$\begin{aligned} \widehat{S}(k) &= 1 + 4\pi\rho \int_0^\infty dr r^2 \frac{\sin kr}{kr} \left[g(r) - 1\right] && \text{(1.131a)} \\ g(r) &= 1 + \frac{1}{2\pi^2\rho} \int_0^\infty dk k^2 \frac{\sin kr}{kr} \left[\widehat{S}(k) - 1\right] && \text{(1.131b)} \end{aligned}$$



typical "look" of a structure factor of a liquid

(liquid neon)

- In the limit $\boldsymbol{k} \to \boldsymbol{0}$ one can show that the *compressibility relation*

$$\lim_{\boldsymbol{k}\to\boldsymbol{0}} \widehat{S}(\boldsymbol{k}) = \widehat{S}(\boldsymbol{0}) = \rho k_B T \kappa_T \tag{1.132}$$

  holds, which relates the long-wavelength limit $\widehat{S}(\boldsymbol{0})$ to the isothermal compressibility

$$\kappa_T = -\frac{1}{V}\left(\frac{\partial V}{\partial p}\right)_T \tag{1.133}$$

  For the ideal gas, we have

$$pV = Nk_BT \quad \Rightarrow \quad \kappa_T^{id} = -\frac{p}{\cancel{Nk_BT}} \frac{\partial}{\partial p}\bigg|_T \frac{\cancel{Nk_BT}}{p} = \frac{1}{\rho k_B T} \tag{1.134}$$

  such that the above compressibility relation can actually be written as

$$\widehat{S}(\boldsymbol{0}) = \kappa_T / \kappa_T^{id} \tag{1.135}$$

1.4 Cutoffs

In computer simulations, a large part of the computer time is usually spent calculating forces and energies. For short-ranged interactions, one can reduce the computational effort by neglecting contributions to the potential energy due to pairs of particles that are farther apart than a certain *cutoff radius* r_c. Of course, this can be done only if the total potential energy is dominated by interactions with neighboring particles closer than r_c (this is just what the term "short-ranged" means).

Truncated potentials

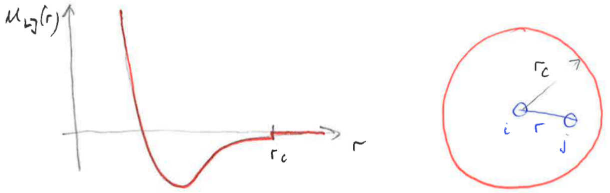
For the LJ potential

$$u_{LJ}(r) \overset{(1.58)}{=} 4\epsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} \right] \tag{1.136}$$

the simplest way to implement a cutoff is to simply *truncate* the interaction, setting

$$u_{trunc}(r) = \begin{cases} u_{LJ}(r), & r < r_c \\ 0, & r \geq r_c \end{cases} \tag{1.137}$$

This corresponds to simply neglecting energies and forces beyond a distance r_c.



The potential energy then becomes

$$U(\boldsymbol{r}^N) = \frac{1}{2} \sum_{\substack{i \neq j \\ r_{ij} < r_c}} u(r_{ij}) \tag{1.138}$$

How much computer time can we save by this strategy?

- Without cutoff, we have to consider all $\frac{N(N-1)}{2}$ pairs of particles to compute the energy, which means that $O(N^2)$ pair energy terms need to be evaluated.
- With cutoff r_c, each particle sees (on average) $\frac{4\pi r_c^3}{3}\rho$ other particles, so if we increase the system size, which is proportional to N, while keeping r_c =const, the total number of pair energies to evaluate scales like

$$\frac{1}{2} \cdot N \cdot \frac{4\pi r_c^3}{3} \rho \sim O(N) \tag{1.139}$$

> In molecular simulations, the goal often is to achieve such a linear scaling with system size. Otherwise, the simulation of large systems would be impossible.

Of course, truncating a potential that is not strictly zero beyond r_c, one certainly introduces an error. One can make this error arbitrarily small by making r_c large, but that, of course, increases the computational load of the simulation. A smarter alternative is to try to estimate the so-called *tail corrections*, by which the results obtained for small r_c differ from those that would result from using the untruncated potential.
Let us consider the expectation value of the total potential energy per particle. Assuming that r_c is sufficiently large, such that $g(r) \approx 1$ for $r > r_c$, we split

$$\begin{aligned} \frac{\langle U \rangle}{N} &= 2\pi\rho \int_0^\infty dr\, r^2\, g(r) u(r) \\ &\approx \underbrace{2\pi\rho \int_0^{r_c} dr\, r^2\, g(r) u(r)}_{(1)} + \underbrace{2\pi\rho \int_{r_c}^\infty dr\, r^2\, u(r)}_{(2)} \end{aligned} \tag{1.140}$$

- the first contribution

  $$(1) = \frac{1}{N} \left\langle \sum_{\substack{i<j \\ r_{ij}<r_c}} u(r_{ij}) \right\rangle \tag{1.141}$$

  may be determined by simulating the system with the truncated potential.

- the second contribution

  $$(2) = 2\pi\rho \int_{r_c}^\infty dr\, r^2\, u(r) \tag{1.142}$$

  resembles the *tail correction* as the total interaction energy per particle beyond r_c. For the example of the LJ potential, it can be computed analytically with the result

  $$(2) = \frac{8}{3}\pi\rho\epsilon\sigma^3 \left[\frac{1}{3}\left(\frac{\sigma}{r_c}\right)^9 - \left(\frac{\sigma}{r_c}\right)^3 \right] \tag{1.143}$$

  – *Proof.*

  $$\begin{aligned} 2\pi\rho \int_{r_c}^\infty dr\, r^2\, u_{LJ}(r) &= 2\pi\rho \int_{r_c}^\infty dr\, r^2\, 4\epsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^6 \right] \\ &= 8\pi\epsilon\sigma^3\rho \int_{r_c}^\infty d\left(\frac{r}{\sigma}\right) \left(\frac{r}{\sigma}\right)^2 \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^6 \right] \\ &= 8\pi\epsilon\sigma^3\rho \int_{r_c/\sigma}^\infty dx \left(x^{-10} - x^{-4} \right) \\ &= 8\pi\rho\epsilon\sigma^3 \left[\frac{1}{9}\left(\frac{r_c}{\sigma}\right)^{-9} - \frac{1}{3}\left(\frac{r_c}{\sigma}\right)^{-3} \right] \\ &= \frac{8}{3}\pi\rho\epsilon\sigma^3 \left[\frac{1}{3}\left(\frac{\sigma}{r_c}\right)^9 - \left(\frac{\sigma}{r_c}\right)^3 \right] \end{aligned} \tag{1.144}$$

A similar tail correction can be applied to the virial by splitting

$$\frac{\langle W\rangle}{N} \approx \frac{1}{N}\left\langle \sum_{\substack{i<j \\ r_{ij}<r_c}} r_{ij}u'(r_{ij}) \right\rangle + 2\pi\rho \int_{r_c}^{\infty} dr\, r^2\,[ru'(r)] \tag{1.145}$$

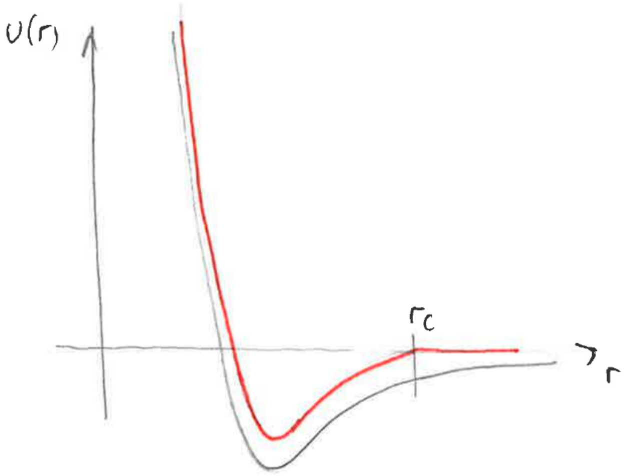
and again assuming that $g(r) \sim 1$ for $r > r_c$.

Truncated and shifted potentials

Another possibility to implement a cutoff is to *truncate and shift* the potential:

$$u_{tr-sh}(r) \equiv \begin{cases} u(r) - u(r_c), & r < r_c \\ 0, & r \geq r_c \end{cases} \tag{1.146}$$

This approach is often used in MD, as it avoids a discontinuity in the energy (and a singularity in the forces). Some authors modify the potential further to also avoid a discontinuity in the forces. Again, the error introduced by the cutoff and shifting can be take care of by appropriate tail corrections.



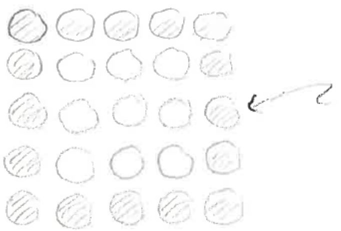
- **Warning.** If the above tail correction integrals are to converge, the potential $u(r)$ must decay stronger than r^{-3}, which excludes e.g. Coulomb and dipolar potentials.

1.5 Boundary Conditions

Often the goal of MC or MD simulations is to determine the properties of a macroscopic sample. However, macroscopic samples have sizes that exceed those that are accessible to simulations on a computer by many orders of magnitude. While macroscopic systems have particle numbers that are of the order of Avogadro's number $N_A = 6.02 \cdot 10^{23}$, systems simulated on a computer usually consist of a few hundred to a few thousand particles simply because larger systems are computationally more expensive to simulate. Now, in small systems, the choice of boundary conditions may have a strong effect, because in a small system a large fraction of particles are located near the surface.

- Imagine, for instance, a simple cubic crystal of $10 \times 10 \times 10 = 1000$ particles, almost half of them are located on its surface!
    - Indeed, $8 \times 8 \times 8 = 512$ are located beneath its surface, leaving $1000 - 512 = 488$ particles on its surface, which corresponds to 48.8%.
- Even for a crystal of 10^6 particles, 5.8808% of the particles are surface particles.

Surface particles, whose fraction scales like $N^{-1/3}$ with the total particle number N, experience a different local environment, and thus have a different influence on the physical properties of the system than so-called *bulk particles* that are located well in the interior of the system



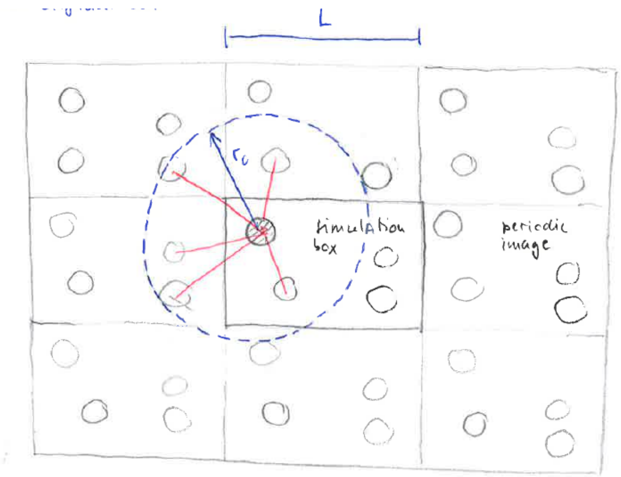
the local environment of surface particles is different from that of particles in the interior

To better reproduce the properties of a bulk system in the thermodynamic limit, molecular simulations are typically carried out with *periodic boundary conditions* (PBC). With such boundary conditions, there are no surfaces such that finite size effects should be less pronounced than for systems enclosed in a container. With PBC, one imagines that the volume containing the particles is the primitive cell of an infinite lattice of identical cells. Efficiently, one *replicates the system periodically* in all directions. Each of the replicated cells is an exact copy of the original one. Each particle then interacts with all other particles in this infinite arrangement, i.e. it interacts with all other particles in its own cell, but also with all their *periodic images* (including its own periodic image!).
For pairwise additive interactions, the total energy with PBC is given by

$$U(\boldsymbol{r}^N) = \frac{1}{2} \sum_{i,j,\boldsymbol{n}}{}^{\prime} u(|\boldsymbol{r}_{ij} + \boldsymbol{n}L|) \tag{1.147}$$

- This *infinite sum* runs over all particle pairs i, j *and* over all cells.
- L is the side length of the simulation box (for simplicity assumed cubic).
- The vector $\boldsymbol{n} = (n_x, n_y, n_z) \in \mathbb{Z}^3$ consists of three integers that specify the particular periodic image.

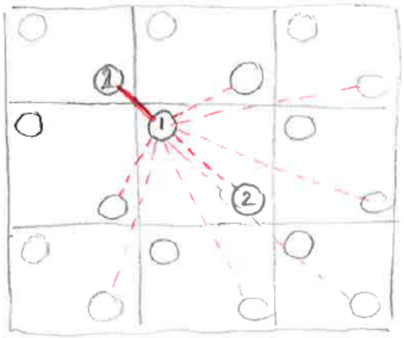
- In particular, $\boldsymbol{n} = \mathbf{0}$ corresponds to the original simulation box. The prime $\sum'$ implies that for $\boldsymbol{n} = \mathbf{0}$ the terms $i = j$ are excluded from the sum to avoid self-interactions.



As it stands, the infinite sum (1.147) is not particularly practical for simulations, as calculating the total energy would require an infinite number of operations. For systems with short-ranged interactions, however, only a finite number of particles in the vicinity of a given particle need to be considered as indicated in the above figure by the cutoff-sphere.

While PBC are a practical way to mimic a macroscopic bulk system, it should be noted that *finite size effects* may still affect the result. For instance, if long wavelength fluctuations with a wavelength λ larger than the linear box size L are important, large finite size effects are to be expected. This may, for instance, happen in the vicinity of a continuous phase transition, where critical fluctuations become dominant.

With PBC, sometimes pair interactions are truncated using the so-called *minimum image convention*. This convention stipulates that one can always use the *nearest periodic image* in the calculation of pair interactions.



In the calculation of the interaction between particles 1 and 2 one uses the periodic image of 2 that is closest to 1 (solid line)

Chapter 2

Canonical Monte Carlo Simulations

In this section we discuss the main igredients of a MC simulation for a classical many-particle system in the canonical ensemble. Before we do that, we will briefly review the theoretical basis for the Metropolis algorithm.

2.1 The Metropolis Algorithm

As discussed in the previous chapter, our goal often is to compute averages of a phase space observable $A(\boldsymbol{r}^N)$. For this task, we have seen that it suffices to average over configuration space, i.e.

$$\langle A \rangle = \frac{\int d\boldsymbol{r}^N e^{-\beta U(\boldsymbol{r}^N)} A(\boldsymbol{r}^N)}{\int d\boldsymbol{r}^N e^{-\beta U(\boldsymbol{r}^N)}} \tag{2.1}$$

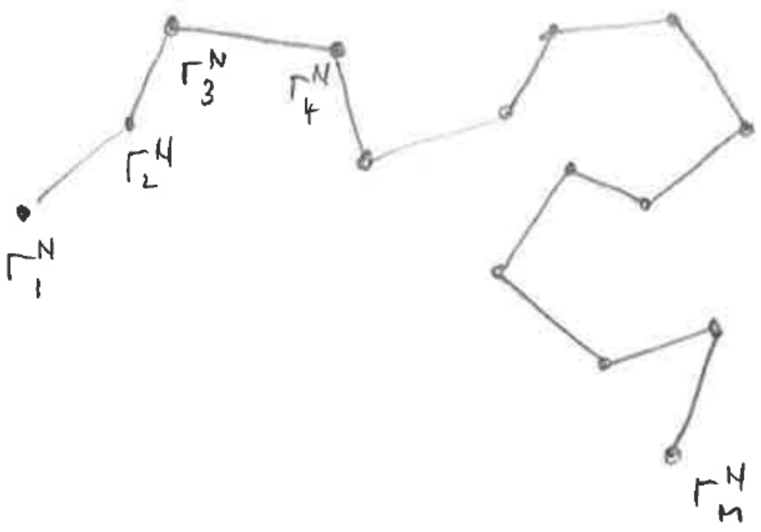
If we can generate a *sample* of M configuration space points $\boldsymbol{r}_i^N$, $i = 1, \ldots M$ according to the probability density

$$f(\boldsymbol{r}^N) = \frac{e^{-\beta U(\boldsymbol{r}^N)}}{\int d\boldsymbol{r}^N e^{-\beta U(\boldsymbol{r}^N)}} \tag{2.2}$$

then we can approximate the above canonical average $\langle A \rangle$ by the average over the sample

$$\langle A \rangle \approx \frac{1}{M} \sum_{i=1}^{M} A(\boldsymbol{r}_i^N), \qquad \boldsymbol{r}_i^N \text{ distributed according to (2.2)} \tag{2.3}$$

The difficulty, of course, lies in generating such a sample with the correct distribution. This is a particular challenge, since in general the normalization factor $\int d\boldsymbol{r}^N e^{-\beta U(\boldsymbol{r}^N)}$ in (2.2) is unknown. A solution of this problem was provided by the Metropolis algorithm suggested in 1953 by Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller and Edward Teller [J. Chem. Phys. **21**, 1087 (1953)]. The basic idea of this very powerful method is to generate a *Markov chain* of configurations $\{\boldsymbol{r}_i\}$ by taking a given configuration, generating a new configuration and accepting or rejecting it. Repeating these basic steps generates a sequence of configurations which, provided the generation and acceptance/rejection of configurations is done in the correct way, samples the desired distribution $f(\boldsymbol{r}^N)$. Since the generation and acceptance/rejection steps involve a stochastic step based on random numbers, this algorithm is a "Monte Carlo" method.



Such a sequence is a *Markov chain*, because new configurations are produced randomly with a probability that depends only on the current state and not on the prior history of the system (this is the *Markov property*).
To make the basic idea of Metropolis MC more precise, consider a sequence of configurations

$$\{\boldsymbol{r}_1^N, \boldsymbol{r}_2^N, \boldsymbol{r}_3^N, \ldots, \boldsymbol{r}_M^N\} \tag{2.4}$$

generated according to a stochastic procedure (i.e., involving random numbers) to be specified later. Such a sequence is called a *Markov chain* if the conditional probability that the configuration at time n is $\boldsymbol{r}_n^N$ depends only on $\boldsymbol{r}_{n-1}^N$ but not on prior configurations:

$$p(\boldsymbol{r}_n^N|\boldsymbol{r}_{n-1}^N, \boldsymbol{r}_{n-2}^N, \ldots, \boldsymbol{r}_2^N, \boldsymbol{r}_1^N) \overset{!}{=} p(\boldsymbol{r}_n^N|\boldsymbol{r}_{n-1}^N) \tag{2.5}$$

- Note that the "time" we are talking about here is not the physical time but rather a kind of "MC-time" that just specifies the position in the sequence.

The conditional probability $p(\boldsymbol{r}^{N'}|\boldsymbol{r}^N)$ is also called the *transition probability*, and is denoted by

$$p(\boldsymbol{r}^N \to \boldsymbol{r}^{N'}) = p(\boldsymbol{r}^{N'}|\boldsymbol{r}^N) \tag{2.6}$$

The particular form of transition probability $p(\boldsymbol{r}^N \to \boldsymbol{r}^{N'})$ depends on how exactly $\boldsymbol{r}^{N'}$ is generated from $\boldsymbol{r}^N$.
Before discussing the form of $p(\boldsymbol{r}^N \to \boldsymbol{r}^{N'})$ in a Metropolis MC simulation, let's consider the general properties the transition probability needs to have such that the sample produced by the procedure follows the desired distribution.

- To simplify our discussion and save some writing effort, let $x \equiv \boldsymbol{r}^N$ denote the entire configuration of the system, and $p(x \to y)$ for the transition probabilities $x \to y$.

Let us consider a large collection of configurations x which are distributed according to some distribution $f^{(n)}(x)$, and imagine that for each of these states x you attempt to carry out transitions $x \to y$ to all states y (including the possibility of a trivial transition $x \to x$ itself), success governed by the transition probability $p(x \to y)$. The resulting new distribution of steps will then be given by

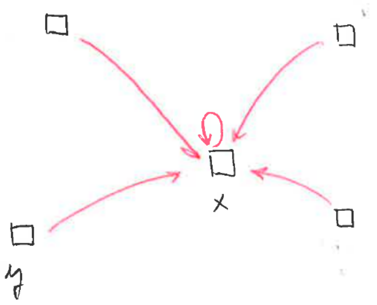
$$f^{(n+1)}(x) = \sum_y f^{(n)}(y)p(y \to x) \tag{2.7}$$

Since we would like the procedure to sample the equilibrium distribution $f(x)$, we require that $f^{(n)}(x)$ converges towards $f(x)$ for $n \to \infty$, i.e.

$$\lim_{n\to\infty} f^{(n)}(x) \equiv f(x) \tag{2.8}$$

A *necessary* condition for that to happen is that $f(x)$ is *stationary* with respect to the "dynamics" mediated by $p(x \to y)$. In other words, applying $p(y \to x)$ to a large set of configurations distributed according to $f(y)$ should not change this distribution but leave it *invariant*, such that the following *stationarity condition* for the transition probabilities $p(y \to x)$ should hold:

$$f(x) = \sum_y f(y)p(y \to x) \qquad \text{for all states}\, x \tag{2.9}$$



the probability to find the system in x at time n+1 is the sum (integral) to find the system in y at time n times the transition probability from y to x .

2.1.1 Global balance

To understand the meaning of stationarity condition (2.9), let us slightly rewrite it. Recognizing that since the transition probabilities are normalized as

$$\sum_y p(x \to y) = 1 \tag{2.10}$$

we multiply the left-hand side of Eq. (2.9) by 1, such that

$$f(x) \sum_y p(x \to y) = \sum_y f(y) p(y \to x) \tag{2.11}$$

and pull $f(x)$ into the sum on the left hand side. Then we get

$$\sum_y p(x \to y) f(x) = \sum_y f(y) p(y \to x) \qquad \text{for all states } x \tag{2.12}$$

- The left hand side describes the decrease in probability of the state x due to transitions away from x to any one of the other states y.
- The right hand side describes the increase in probability of the state x due to transitions to x from any one of the other states y.

The above stationarity condition means that *in equilibrium for all states x the loss in probability described by the left hand side must be exactly compensated by the increase in probability described by the right hand side.* Therefore, Eq. (2.12) is also called the *condition of global balance* or *condition of full balance.*

2.1.2 Detailed balance

The stationarity condition (2.9) for $p(x \to y)$ can be guaranteed to hold by imposing a much stronger condition than the global balance condition (2.12): one may require that in equilibrium transitions from x to y are exactly compensated by transitions from y to x, *separately* for each state x and y. In other words, we impose the *condition of detailed balance* or *condition of microscopic balance*

$$f(x) p(x \to y) = f(y) p(y \to x) \qquad \text{for all states } x, y \tag{2.13}$$

Obviously, if detailed balance holds, the global balance is automatically satisfied. Note, however, that the converse is not necessarily true.

In practical implementations of MC simulations, one usually requires detailed rather than global balance. The simple reason is that for a given algorithm detailed balance is usually much easier to demonstrate than global balance.

- *For the curious:* This is not to say that there are no correct MC algorithms that satisfy global balance while violating detailed balance. Examples include

* the class of *sequential update algorithms* in which particles are moved or spins are flipped according to a prescribed sequential order instead of picking them at random (see e.g. [R. Ren & G. Orkoulas, J. Chem. Phys. **124**, 064109 (2006)])

* Event-chain Monte Carlo algorithms originally devised for for hard-spheres [E.P. Bernard, W. Krauth, and D.B. Wilson, Phys. Rev. E **80**, 056704 (2009)] and subsequently generalized to arbitrary pair potentials

2.1.3 Metropolis MC

Let us now consider what detailed balance means for the particular form of the transition probability $p(x \to y)$ corresponding to a basic *Metropolis MC* step. As mentioned before, a step of a Metropolis MC procedure consists of two parts:

1. First, a new configuration y is *generated* from a given (old) configuration. This *trial move* usually contains an element of randomness (for instance, a particle is displaced by a random amount in a random direction), and so a *generation probability*

$$p_{gen}(x \to y) \tag{2.14}$$

as the probability density for generating the trial move from x to y.

2. In the second stage of the MC step, the new configuration is either *accepted* or *rejected.* The trial move (i.e. the new configuration y) is accepted with probability

$$p_{acc}(x \to y) \tag{2.15}$$

$p_{acc}(x \to y)$ is the probability for accepting y, which was generated from x:

- If y is accepted, it becomes the new configuration of the system.
- If y is rejected, the system stays in x, i.e. the old state is *counted again* for the calculation of averages of the quantities of interest.

Since the generation and acceptance probabilities are statistically independent, the total transition probability $p(x \to y)$ for a transition from $x \to y$ is simply the *product* of the generation and acceptance probability:

$$p(x \to y) = p_{gen}(x \to y)p_{acc}(x \to y) \tag{2.16}$$

In other words, $p(x \to y)$ is the product of the probability to *attempt* a move $x \to y$ and the probability to accept it.

Since we would like the algorithm to sample the equilibrium distribution $f(x)$, we *require this MC transition probability* $p(x \to y)$ *to satisfy detailed balance:*

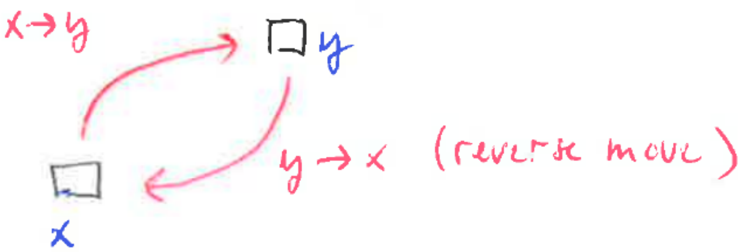
$$f(x)p_{gen}(x \to y)p_{acc}(x \to y) = f(y)p_{gen}(y \to x)p_{acc}(y \to x) \tag{2.17}$$

For a given fixed form of the generation probability $p_{gen}(x \to y)$ this equation can be trivially rewritten as a condition for the acceptance probability of the move $x \to y$ and the *reverse* move $y \to x$:

$$\frac{p_{acc}(x \to y)}{p_{acc}(y \to x)} = \frac{f(y)p_{gen}(y \to x)}{f(x)p_{gen}(x \to y)} \tag{2.18}$$

Often, *symmetric* generation probabilities are used, i.e. the probability of generating y from x and the probability of generating x from y are the same, i.e.

$$p_{gen}(x \to y) = p_{gen}(y \to x) \qquad \text{(symmetric generation)} \tag{2.19}$$

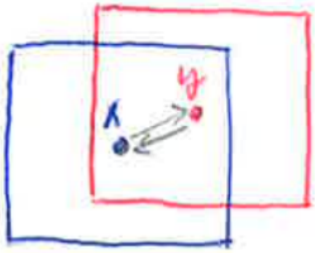


In this case, condition (2.18) simplifies to

$$\frac{p_{acc}(x \to y)}{p_{acc}(y \to x)} = \frac{f(y)}{f(x)} \qquad \text{(symmetric generation)} \tag{2.20}$$

Note that there are many useful MC algorithms in which the generation probabilities are not symmetric, and we will get to know some of them later in the course.

- As an example of a symmetric move, consider the random displacement of a particle to a new position uniformly distributed in a small region around the old position.



If $f(x)$ is the canonical distribution, i.e. $f(x) = \frac{e^{-\beta H(x)}}{Z}$, condition (2.20) becomes

$$\frac{p_{acc}(x \to y)}{p_{acc}(y \to x)} = \frac{e^{-\beta H(y)}}{e^{-\beta H(x)}} = e^{-\beta \Delta H} \qquad \text{(symmetric generation)} \qquad (2.21)$$

where

$$\Delta H = H(y) - H(x) \qquad (2.22)$$

is the energy difference between the old and new configuration. Note that, as promised, in this expression *the normalization factor Z does not appear!* This is important because it means that MC can be applied without knowing the normalization factor Z of the canonical distribution function!

- If Z (and thus the canonical free energy) were known, there would be no need to carry out MC simulations in the first place.

Also note that for putting (2.21) to work, only energy **changes** ΔH need to be calculated, which is e.g. for short range interactions computationally *much* cheaper than calculating the total energy itself.

2.1.4 Metropolis rule

Any acceptance probability satisfying the detailed balance condition in the symmetric form (2.20) can be used in a MC simulation, and many forms of $p_{acc}(x \to y)$ have been proposed.

- Note that $p_{acc}(x \to y)$ also needs to be a number between 0 and 1.

A particularly simple and efficient acceptance probability was suggested by Metropolis et al. in 1953. For the choice of Metropolis

$$p_{acc}(x \to y) = \begin{cases} \frac{f(y)}{f(x)}, & \text{if } f(y) < f(x) \\ 1, & \text{if } f(y) \geq f(x) \end{cases} \qquad \text{(symmetric generation)} \; (2.23)$$

the trial move is accepted with certainty if the new configuration has a larger statistical weight than the old one, while it is only accepted with probability $f(y)/f(x)$ if its weight is smaller.

- To show that this choice indeed satisfies detailed balance, we separately consider to following two cases:
  - $f(y) < f(x)$: Then
$$\frac{p_{acc}(x \to y)}{p_{acc}(y \to x)} = \frac{\frac{f(y)}{f(x)}}{1} = \frac{f(y)}{f(x)} \qquad \checkmark \tag{2.24}$$
  - $f(y) \geq f(x)$: Then
$$\frac{p_{acc}(x \to y)}{p_{acc}(y \to x)} = \frac{1}{\frac{f(x)}{f(y)}} = \frac{f(y)}{f(x)} \qquad \checkmark \tag{2.25}$$
- An alternative to the Metropolis rule is the so-called *symmetric rule*
$$p_{acc}(x \to y) = \frac{f(y)}{f(x) + f(y)} \tag{2.26}$$
which is used frequently for simulating spin systems.

The Metropolis acceptance rule can be expressed in a compact way as

$$p_{acc}(x \to y) = \min\left[1, \frac{f(y)}{f(x)}\right] \qquad \text{(symmetric generation)} \tag{2.27}$$

For the canonical ensemble

$$p_{acc}(x \to y) = \min\left[1, e^{-\beta\Delta H}\right], \quad \Delta H = H(y) - H(x) \qquad \text{(symm. gen.)} \tag{2.28}$$

In other words,

- moves downhill in energy are always accepted.
- moves in which the energy increases are accepted with probability $e^{-\beta\Delta H}$.

According to the Metropolis acceptance rule, a trial move $x \to y$ with $f(y) < f(x)$ needs to be accepted with probability $f(y)/f(x)$, which is a number between 0 and 1. One can implement this *rejection/acceptance decision* by drawing a random number ξ from a uniform distribution in the interval $[0, 1]$. If

$$\xi \leq \frac{f(y)}{f(x)} \tag{2.29}$$

then the trial move is accepted and is rejected otherwise.

2.1.5 Rejected moves

In applying the Metropolis MC scheme it is essential that if a trial move is rejected, the old configuration is counted again. Let us look at this issue in some more detail.

So far we had argued that the transition probability of Metropolis MC is given by

$$p(x \to y) = p_{gen}(x \to y) p_{acc}(x \to y) \tag{2.30}$$

However, this transition probability is *not normalized*:

$$\sum_y p(x \to y) = \sum_y p_{gen}(x \to y) p_{acc}(x \to y) \leq 1 \tag{2.31}$$

The reason is that the generation probability $p_{gen}(x \to y)$ itself *is* normalized, i.e. $\sum_y p_{gen}(x \to y) = 1$, but in the above sum, it is multiplied by $p_{acc}(x \to y)$, a number smaller than or equal to 1. Therefore, the above sum is also smaller than or equal to 1. In fact, the transition probability $p(x \to y)$ fails to be normalized precisely because *it misses all rejected moves*! In order to be useful, a transition probability *must* be normalized, because something rather than nothing should always happen and we need a prescription of what to do. *So what's wrong here?*

We can fix this problem by explicitly including the rejected moves into the transition probability by counting the old configurations again if the trial move is rejected. To do that we first define the *total acceptance probability*

$$P_{acc}(x) := \sum_y p_{gen}(x \to y) p_{acc}(x \to y) \tag{2.32}$$

for trial moves starting from configuration x, from which we read off the *total rejection probability* $P_{rej}(x)$ for trial moves starting from x:

$$P_{rej}(x) := 1 - P_{acc}(x) \tag{2.33}$$

In the full transition probability, we need to also include this probability of rejected moves! Therefore, we really should define

$$p(x \to y) \equiv \underbrace{p_{gen}(x \to y) p_{acc}(x \to y)}_{\text{accept}} + \underbrace{P_{rej}(x)\delta_{x,y}}_{\text{reject}} \tag{2.34}$$

where $\delta_{x,y}$ is the Kronecker delta. The last term on the right hand side says that with probability $P_{rej}(x)$ the system stays at x. This transition probability, which reflects the prescription that after a rejection the old configuration is counted again, *is* normalized:

$$\begin{aligned}\sum_y p(x \to y) &\equiv \underbrace{\sum_y p_{gen}(x \to y) p_{acc}(x \to y)}_{\overset{(2.32)}{=} P_{acc}(x)} + P_{rej}(x) \underbrace{\sum_y \delta_{x,y}}_{=1} \\ &= P_{acc}(x) + P_{rej}(x) = 1 \quad \checkmark \end{aligned} \tag{2.35}$$

We can also explicitly check that the correct transition probability as defined in Eq. (2.34) obeys the condition of detailed balance:

- For $y \neq x$, Eq. (2.34) reduces to

  $$p(x \to y) = p_{gen}(x \to y) p_{acc}(x \to y) \tag{2.36}$$

  for which we have already verified that, using the Metropolis rule, this transition probability satisfies detailed balance (in fact, it was just constructed such that it does).✓

- For $y = x$, Eq. (2.34) reduces to

  $$p(x \to x) = p_{gen}(x \to x) p_{acc}(x \to x) + P_{rej}(x) \cdot 1 \tag{2.37}$$

  which trivially satisfies detailed balance:

  $$f(x)[p_{gen}(x \to x) + P_{rej}(x)] = f(x)[p_{gen}(x \to x) + P_{rej}(x)] \quad \checkmark \tag{2.38}$$

In detail, according to the Metropolis rule

$$p_{acc}(x \to x) = \min[1, f(x)/f(x)] = 1 \tag{2.39}$$

and therefore

$$p(x \to x) = p_{gen}(x \to x) + P_{rej}(x) \tag{2.40}$$

As encoded in Eq. (2.40), there are thus two possibilities for the system to stay at x:

- either the identical state x is generated in the generation state (and then, of course, accepted), which is the trivial case.
- or the system stays in the state x because a trial step to some other state y had been generated but was rejected.

In summary, we have shown that - upon including rejected moves by recounting the old configuration after a rejection - the transition probability (2.34) corresponding to the Metropolis procedure (2.28)

- satisfies detailed balance.

- is correctly normalized to 1.

Also, we concluded that if a trial move is rejected, the old configuration must be counted again. If this is not done, the simulation samples the *wrong* ensemble.

- It is instructive to try out how the results of a MC simulation change, if instead of recounting the old configuration again, a new trial move is attempted over and over until another configuration is found that is accepted. That such a strategy gives erroneous results is very obvious taking the example of a two-state system with just two states x_0, x_1 at energy levels $E_0 < E_1$, for which the simulation would just jump between these energies back and forth, leading to an energy average $\langle E\rangle = (E_0 + E_1)/2$, *regardless of which temperature T is imposed*! This is clearly wrong.

2.2 Ergodicity

Another condition that an MC simulation should obey is that it should be *ergodic*: any physically admissible configuration of the system should be reachable from any other configuration in a *finite* number of steps.
Ergodicity can be proven for some simple MC schemes, but these are not necessarily the most efficient ones. On the other hand, there are more efficient algorithms which have either not been proven to be ergodic or, even worse, have been shown to be explicitly *non-ergodic*. The solution to this dilemma is to mix a more efficient but possibly non-ergodic MC scheme with a less efficient but ergodic one. The resulting MC scheme will then, of course, as a whole be ergodic.

2.3 A basic MC simulation code

Now that we know the theoretical foundation of Metropolis MC (or, more generally, of Markov chain MC), let's have a look on how an MC simulation of a simple N-particle system is implemented in practice.
The general structure of a generic MC program is as follows.

- Note that we will discuss the individual steps in greater detail in subsequent sections.

To simplify the discussion, we imagine that we are simulating a system of N identical classical particles with energy $U(\boldsymbol{r}^N)$.

```
program monte carlo
{
  read in parameters;
  initialize system (or read in initial positions from file);
  perform $N_{steps}$ Monte Carlo steps:
  {
    compute energy $E_{old} = U(\boldsymbol{r}^N)$ of current configuration $\boldsymbol{r}^N$;
    pick random vector $\boldsymbol{\Delta}$;
    pick random particle $\boldsymbol{r}_{\text{picked}}$;
    displace $\boldsymbol{r}'_{\text{picked}} := \boldsymbol{r}_{\text{picked}} + \boldsymbol{\Delta}$;
    compute energy $E_{new} = U(\boldsymbol{r}^{N'})$ of trial configuration $\boldsymbol{r}^{N'}$;
```

```
        accept trial configuration $\boldsymbol{r}^{N'}$ with probability
            $p_{acc} = \min[1, \exp\{-\beta[E_{new} - E_{old}]\}]$;
        update averages;
    }
    compute and output averages;
}
```

2.3.1 For the curious: So why does it work?

So why does Markov MC work? Let us give a brief sketch of proof which, while being mathematically far from thourough, should provide a hint on why the MC receipe outlined above actually samples the target equilibrium distribution after discarding a initial equilibration period.

We shall consider the (for most practical purposes over-simplified) case of a finite state space with N states labeled $n = 1, \ldots, N$ with transition probabilities $p_{nm} \equiv p(m \to n)$, such that a given probability distribution $p_n(t)$ evolves with time according to the master equation

$$\frac{df_n(t)}{dt} = \sum_m \Big[\underbrace{p_{nm} f_m(t)}_{\text{gain}} - \underbrace{p_{mn} f_n(t)}_{\text{loss}}\Big] \tag{2.41}$$

We are, of course, looking for a *stationary solution* $\dfrac{d\boldsymbol{f}^s(t)}{dt} = 0$ of this equation, which should then correspond to the sought-after equilibrium distribution. Again, such a stationary solution will only exists if global balance $0 \equiv \sum_m \Big[p^s_{nm} f_m(t) - p^s_{mn} f_n(t)\Big]$ holds.

Let us now introduce a more compact notation for the above master equation. We introduce the matrix $\boldsymbol{P}$ of so-called *rate constants* defined by

$$P_{nm} \equiv \begin{cases} p_{nm}, & n \neq m \\ -\sum\limits_{k \neq n} p_{kn}, & n = m \end{cases} \tag{2.42}$$

Then our master equation can be rewritten in the compact form

$$\frac{d\boldsymbol{f}(t)}{dt} = \boldsymbol{P} \cdot \boldsymbol{f}(t) \tag{2.43}$$

Moreoveror, a stationary distribution $\boldsymbol{f}^s$ then must satisfy the relation $\boldsymbol{P} \cdot \boldsymbol{f}^s = \boldsymbol{0}$, i.e. it must be an *eigenvector of* $\boldsymbol{P}$ *for eigenvalue* $\lambda = 0$.

Now note that $\boldsymbol{P}$ is, of course, a real matrix, but generally *not* a symmetric one. Therefore, it has right and left eigenvectors $\boldsymbol{\psi}_\alpha$ and $\boldsymbol{\phi}_\beta$, respectively, such that

$$\boldsymbol{P}\boldsymbol{\psi}_\alpha = \lambda_\alpha \boldsymbol{\psi}_\alpha, \qquad \boldsymbol{\phi}_\beta \boldsymbol{P} = \mu_\beta \boldsymbol{\phi}_\beta \tag{2.44}$$

are different but their spectrum $\{\lambda_\alpha\} = \{\mu_\beta\}$ agrees.

- Of course, $\boldsymbol{\phi}_\alpha$ is formally a row vector, while $\boldsymbol{\psi}_\alpha$ is a column vector. What is meant is that the relations

$$\boldsymbol{P}\boldsymbol{\psi}_\alpha = \lambda_\alpha \boldsymbol{\psi}_\alpha, \qquad \boldsymbol{P}^t \boldsymbol{\phi}^t_\beta = \mu_\beta \boldsymbol{\phi}^t_\beta \tag{2.45}$$

hold.

In addition, orthogonality $\boldsymbol{\phi}_\beta \cdot \boldsymbol{\psi}_\alpha = \sum_i \phi_{\beta i}\psi_{\alpha i} \propto \delta_{\alpha\beta}$ of left and right eigenvectors to different eigenvalues $\lambda_\alpha \neq \lambda_\beta$ holds.
Let us expand $\boldsymbol{f}(t) = \sum\limits_\alpha c_\alpha(t)\boldsymbol{\psi}_\alpha$ in the left eigenbasis $\{\boldsymbol{\psi}_\alpha\}$ of $\boldsymbol{P}$. If we insert this expansion into the master equation, then $\sum_\alpha \dot{c}_\alpha(t)\boldsymbol{\psi}_\alpha \equiv \sum_\alpha c_\alpha(t)\underbrace{\boldsymbol{P}\cdot\boldsymbol{\psi}_\alpha}_{\lambda_\alpha}$.
Using orthogonality (and ignoring the possibility of degenerate eigenvalues), it follows that $\dot{c}_\alpha(t) = \lambda_\alpha c_\alpha(t)$ for all α. These equations are solved as $c_\alpha(t) = c_\alpha(0)e^{t\lambda_\alpha}$, such that the time evolution of $bmf(t)$ is given by

$$bmf(t) = \sum_\alpha e^{\lambda_\alpha t} c_\alpha(0)\boldsymbol{\psi}_\alpha \tag{2.46}$$

Observe now that from the definition of P_{ij} we see that the relation $\sum_i P_{ij} = 0$ holds for all j. We may interpret this relation by saying the the row vector $\boldsymbol{\phi}_0 \equiv (1, 1, \ldots 1)$ is a *left* eigenvector $\boldsymbol{\phi}_0 \cdot \boldsymbol{P} = \boldsymbol{0}$ for eigenvalue 0. Thus, there also exists *at least* one *right* eigenvector $\boldsymbol{\psi}_0$ with $\boldsymbol{P}\cdot\boldsymbol{\psi}_0 = \boldsymbol{0}$, and one shows that $\boldsymbol{\psi}_0 \geq 0 \,\forall\, i$.
The central ingredient for understanding the convergence properties of the above time evolution of $\boldsymbol{f}(t)$ is now that one can also show that all eigenvalues λ_α are either real or come in complex conjugate pairs. In addition, all eigenvalues have *non-positive real parts* $\Re\lambda_\alpha \leq 0$. Therefore, we can write

$$\boldsymbol{f}(t) = c_0\boldsymbol{\psi}_0 + \sum_{\alpha\neq 0} \underbrace{e^{\Re\lambda_\alpha t}}_{\overset{(t\to\infty)}{\longrightarrow} 0} e^{i\Im\lambda_\alpha t} c_\alpha(0)\boldsymbol{\psi}_\alpha \overset{(t\to\infty)}{\longrightarrow} = c_0\boldsymbol{\psi}_0 \tag{2.47}$$

It follows that all components other than $\boldsymbol{\psi}_0$ decay exponentially. In addition, one can also show that all components $\boldsymbol{\psi}_{0i} \geq 0$ are non-negative, such that it qualifies for a probability distribution upon normalizing by chppsing the factor $c_0 \equiv 1/\sum_{i=1}^N \boldsymbol{\psi}_{0i}$.
In summary, while not be 100% mathematically bullet-proof, the above reasoning should make it clear why indeed *any initial distribution $\boldsymbol{f}$ will converge towards the stationary distribution $\boldsymbol{f}^s \equiv c_0\boldsymbol{\psi}_0$ with normalization* $c_0 \equiv 1/\sum_{i=1}^N \boldsymbol{\psi}_{0i}$ provided the transition probabilities of out Markov chain have been properly set up as explained in the previous sections.

2.3.2 Trial moves

One of the reasons that MC simulation is such a powerful tool is that one has great freedom in designing *trial moves* that lead to efficient sampling of configuration space. By using detailed balance, one can then easily derive the acceptance probability suitable for a particular trial move. For instance, trial moves do not need to be related to the actual way the system naturally evolves in physical time. In fact, often trial moves that are completely "artificial" and have nothing to do with the "natural" dynamics lead to the most efficient MC algorithm.

- By the way, what does "efficient" mean for a MC simulation? The most effective simulation is the one that, for a given amount of simulation time, yields the smallest statistical error in the computed averages of interest. We will elaborate on the concept of efficiency later.

In order to discuss issues related to generation of trial moves, let us consider a system of N identical classical particles in a simulation box of volume V. The particles are supposed to interact via a pairwise additive potential $U(\boldsymbol{r}^N) = \sum_{i<j} u(r_{ij})$ and PBC are used.
Perhaps the simplest trial move we can think of consists of the translation of a single particle by a random amount. To do this, we

1. pick one of the N particles at random. For this purpose, we generate a random number in the interval $[0, 1]$. The index $i \in \{1, 2, \ldots, N\}$ of the chosen particle is the given by

$$i = \lfloor N\xi \rfloor + 1 \tag{2.48}$$

where

$$\lfloor x \rfloor \equiv \text{floor}(x) \equiv \max\{i \in \mathbb{N} : i < x\} \tag{2.49}$$

2. The chosen particle i is displaced randomly. This is done by picking three more random numbers $\xi_x, \xi_y, \xi_z \in [0, 1]$ from a uniform distribution. The new position $\boldsymbol{r}'_i = (x'_i, y'_i, z'_i)$ of particle i is then obtained from the original position $\boldsymbol{r}_i = (x_i, y_i, z_i)$ by adding the random displacement

$$x'_i = x_i + \Delta\left(\xi_x - \frac{1}{2}\right) \tag{2.50}$$

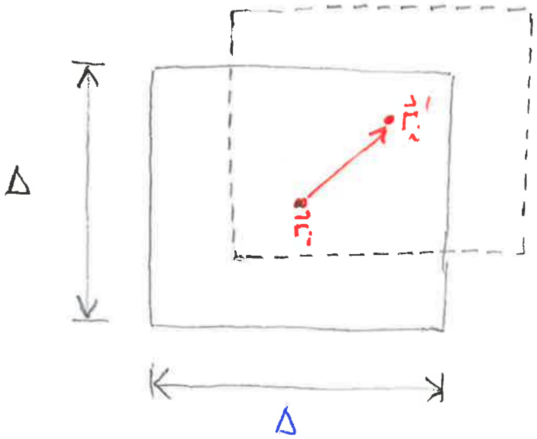
$$y'_i = y_i + \Delta\left(\xi_y - \frac{1}{2}\right) \tag{2.51}$$

$$z'_i = z_i + \Delta\left(\xi_z - \frac{1}{2}\right) \tag{2.52}$$

in each coordinate direction, where

- subtracting $1/2$ from each of the random numbers ξ_x, ξ_y, ξ_z makes sure that the respective displacements occur in positive and negative directions with equal probability.
- the parameter Δ governs the *maximum size* of the displacements.

This procedure generates a new position $\boldsymbol{r}'_i$ uniformly distributed over a cube of side length Δ with center at the old position $\boldsymbol{r}_i$.



Now consider a trial move starting from the new position $\boldsymbol{r}_i'$ and generated following the same prescription as described above. What is the probability that this trial step will lead back to the old position $\boldsymbol{r}_i$? Since both the cube centered around $\boldsymbol{r}_i$ and the one centered around $\boldsymbol{r}_i'$ have the same volume Δ^3, the probability of generating the *reverse move* $\boldsymbol{r}_i' \to \boldsymbol{r}_i$ is exactly equal to the probability of generating the *forward move* $\boldsymbol{r}_i \to \boldsymbol{r}_i'$. In addition, the probability $1/N$ of picking exactly i is the same for both the forward and the reverse move and all the other particles remain untouched in both cases, the *generation probability* corresponding for this *single particle displacement* move is *symmetric*:

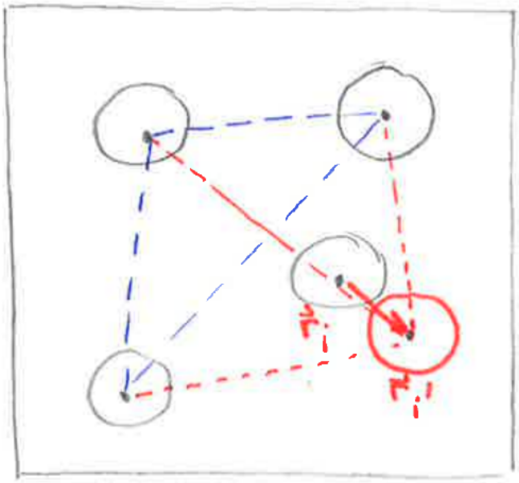
$$p_{gen}(\boldsymbol{r}^N \to \boldsymbol{r}^{N'}) = p_{gen}(\boldsymbol{r}^{N'} \to \boldsymbol{r}^N) \tag{2.53}$$

Hence, in this case we can use the simple acceptance criterion

$$p_{acc}(\boldsymbol{r}^N \to \boldsymbol{r}^{N'}) = \min\left[1, \exp\left(-\beta \Delta U\right)\right], \qquad \Delta U = U(\boldsymbol{r}^{N'}) - U(\boldsymbol{r}^N) \tag{2.54}$$

2.3.3 How to calculate ΔU efficiently

Displacing particle i changes only the $N-1$ distances between particle i and all other particles, while the distances between all other particles remain unchanged. For pairwise additive potentials the calculation of the energy difference ΔU caused by the displacement of one single particle therefore requires only $O(N)$ calculations of the pair potential:



We decompose the potential energy of a pairwise additive potential $U(\boldsymbol{r}^N) = \frac{1}{2}\sum_{k\neq l} u(r_{kl})$ in the "old" configuration as

$$U(\boldsymbol{r}^N) = \frac{1}{2}\sum_{\substack{k\neq l\\ k,l\neq i}} u(r_{kl}) + \frac{1}{2}\sum_{\substack{k\\ k\neq i}} u(r_{ki}) + \frac{1}{2}\sum_{\substack{l\\ l\neq i}} u(r_{il}) \tag{2.55}$$

to isolate the contributions of particle i. However, since $r_{ki} = r_{ik}$, we actually have

$$U(\boldsymbol{r}^N) = \frac{1}{2}\sum_{\substack{k\neq l\\ k,l\neq i}} u(r_{kl}) + \sum_{\substack{k\\ k\neq i}} u(r_{ki}) \tag{2.56}$$

Likewise, for the "new" configuration we obtain

$$U(\boldsymbol{r}^{N\prime}) = \frac{1}{2}\sum_{\substack{k\neq l\\ k,l\neq i}} u(r_{kl}) + \sum_{\substack{k\\ k\neq i}} u(r'_{ki}) \tag{2.57}$$

Hence

$$\Delta U = \sum_{\substack{k\\ k\neq i}} [u(r'_{ki}) - u(r_{ki})] \tag{2.58}$$

This sum contains $N-1$ terms, one for each distance that has changed due to the displacement of particle i.

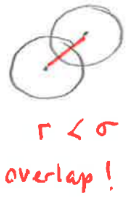
- *For the curious.* In a simple program, it might be advantageous to store the pair potential contributions $u(r_{ki})$ as a two-dimensional array in memory. In this way, every time we shift particle i we precisely have to update the vector $u(r_{\cdot i}) = (u(r_{ki}))_{k=1}^{N}$. Obviously, this still requires an order of N potential calculations. If the pair potential is of short range and the system is large it is more efficient to set up Verlet or cell lists to be discussed later.

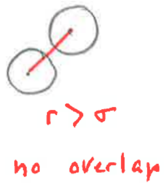
Acceptance criterion for hard spheres

For hard spheres, the acceptance step following a random particle displacement is even simpler. According to the Metropolis criterion, the acceptance probability

$$p_{acc}(\boldsymbol{r}^N \rightarrow \boldsymbol{r}^{N'}) = \min\left[1, \exp\left(-\beta\Delta U\right)\right] \tag{2.59}$$

for hard spheres is either 1 (because $\Delta U = 0$ and thus $\exp(-\beta\Delta U) = 1$ if there is no overlap in both old and new configuration) or 0 (because $\Delta U = \infty$ and thus $\exp(-\beta\Delta U) = 0$ if the displacement yields an overlap with another particle). So, to decide whether to accept or to reject a trial configuration y, one has to check if the displaced particle i overlaps with any other particle, i.e. if the center of particle i is closer than σ to the center of any other particle:





As soon as one overlap is detected, one immediately rejects the trial configuration, as there is no need to further check for other overlaps with the remaining particles.

2.3.4 Would it be more efficient to displace all (or groups of) particles?

One might be tempted to move all (or groups of) particles in the basic step of a MC simulation rather than just one particle. One would hope that such a *multi-particle* or *collective* move leads to a faster sampling of configuration space, and hence to a more efficient simulation. Is this really true?

In order to answer this question, we first have to define what we mean by *efficiency*. In a MC simulation we compute averages over the sample produced by the algorithm. As such, the averages are affected by a statistical error, which would like to be as small as possible. So we can say that one algorithm is *more efficient* than another one if it requires less computing time (CPU-time is the currency of simulations) to achieve a prescribed statistical error in the quantity of interest.

For instance, if we would like to compute the pressure of a system for a given density and temperature, we would like to use the particular MC algorithm (i.e. the generation procedure for the trial move) that produces the smallest statistical error in the virial within the budget of CPU hours at our disposal. Clearly, the definition of efficiency also *depends on the particular quantity* one considers!

Nevertheless, to obtain a more practical definition of efficiency, we expect that the statistical error, or more precisely, the *mean square error* in the observables should be inversely proportional to the *statistically independent* configurations visited during the course of the simulation.
On the other hand, the number of visited configurations is a measure of the *distance* that the random walk of the MC simulation has covered in configuration space.
To get a handle on this idea, we consider the *sum of all squares of all accepted trial displacements* $\mathbf{\Delta}_j^2$ divided by the total computing time as a measure of the efficiency of the simulation:

$$\text{efficiency} \equiv \frac{\sum_j \mathbf{\Delta}_j^2}{t_{CPU}} \tag{2.60}$$

- Note that the sum over these squares is different from the mean square displacement per computer time. For instance, for a solid the mean square displacement per computer time goes to zero for increasing computer time, while (2.60) converges to a finite value.

Using this - admittedly a bit *ad hoc* - definition, we now address the question if it is better to attempt of one or many particles.
In a many-particle system, we expect a trial move to be rejected if the potential energy increases by much more than $k_B T$, because in this case $e^{-\beta\Delta U} = e^{-\Delta U/k_B T} \ll 1$. To move through configuration space quickly, we would like to make the particle displacements large. But at the same time we would still like to have a reasonable acceptance probability (if the acceptance probability approaches zero, the system does not move any more), i.e. ΔU should not be much larger than $k_B T$. So, how large can we make the displacement without getting an energy increase exceeding $k_B T$?
Imagine that we displace particle i by $\mathbf{\Delta}_i$. Expanding ΔU in a Taylor series and averaging yields

$$\langle \Delta U \rangle = \sum_{\alpha=1}^{3} \left\langle \frac{\partial U}{\partial r_i^\alpha} \right\rangle \overline{\Delta_i^\alpha} + \sum_{\alpha,\beta=1}^{3} \left\langle \frac{\partial^2 U}{\partial r_i^\alpha \partial r_i^\beta} \right\rangle \overline{\Delta_i^\alpha \Delta_i^\beta} + \ldots \tag{2.61}$$

Here the angular brackets denote an ensemble average and the overbars an average over the displacement distribution. The superscripts $\alpha, \beta, \ldots$ denote the spatial coordinate directions. If the displacement distribution is symmetric and the displacements in different Cartesian directions are independent like in our choice above, then

$$\overline{\Delta_i^\alpha} = 0, \qquad \overline{\Delta_i^\alpha \Delta_i^\beta} \equiv \overline{\Delta^2}\delta_{\alpha\beta} \tag{2.62}$$

and we obtain

$$\langle \Delta U \rangle = f(U)\overline{\Delta^2} + O(\Delta^4) \qquad \text{(single particle displaced)} \tag{2.63}$$

where $f(U)$ is a function depending on the second derivatives of the potential energy, whose specific form will not be important. If we now *require that* $\langle \Delta U \rangle \approx k_B T$, which is necessary to achieve a decent acceptance rate, we find

$$\overline{\Delta^2} \approx \frac{k_B T}{f(U)} \qquad \text{(single particle displaced)} \tag{2.64}$$

- Imagine that we carry out N single-particle moves. In each move the largest part of the computing time will go into the calculation of the energy. Assuming local interactions and that we are using tricks like neighbor lists (to be discussed later) to calculate energies, each energy calculation will cost us an amount of CPU times proportional to the number of neighbors n. Thus, to carry out N single particle moves will require a CPU-time proportional to nN. The *total sum of displacements* will be

  $$\overline{\sum_{j=1}^{N} \mathbf{\Delta}^2_{i(j)}} \approx \frac{N k_B T}{f(U)} \qquad (N \text{ single-particle displacements}) \tag{2.65}$$

  The total mean square displacement per unit of CPU-time of these N single-particle displacements will be proportional to

  $$\text{efficiency} \overset{(2.60)}{\approx} \frac{\frac{N k_B T}{f(U)}}{nN} = \frac{k_B T}{n f(U)} \qquad (N \text{ single-particle moves}) \tag{2.66}$$

- Instead of doing N single particle moves, we may also *instantaneously displace* N *the particles by displacement vectors* $\mathbf{\Delta}_j$, $j = 1, \dots N$. The average change in energy is then given by

  $$\langle \Delta U \rangle = f(U) N \overline{\Delta^2} + O(\Delta^4) \tag{2.67}$$

  and hence the mean squared displacement amplitude for each single-particle displacement in the total of N displacements is estimated as

  $$\overline{\Delta^2} \approx \frac{k_B T}{N f(U)} \tag{2.68}$$

  Not unexpectedly, compared to the single-particle displacement case this is smaller by a factor of N. For the total mean square displacement achieved by this move, we obtain

  $$\overline{\sum_{j=1}^{N} \mathbf{\Delta}^2_j} \approx \cancel{N} \frac{k_B T}{\cancel{N} f(U)} \qquad (N\text{-particle displacement}) \tag{2.69}$$

  The total mean square displacement per unit of CPU-time of these N single-particle displacements will be proportional to

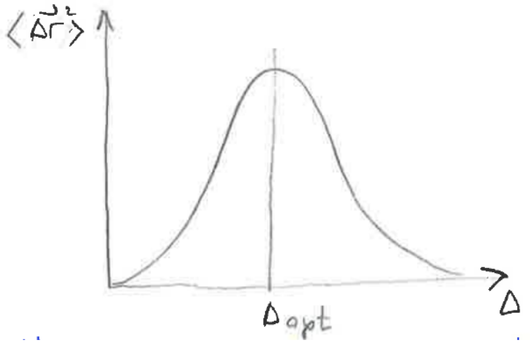
$$\text{efficiency} \overset{(2.60)}{\approx} \frac{\frac{k_BT}{f(U)}}{nN} = \frac{1}{N}\frac{k_BT}{nf(U)} \qquad (N\text{-particle displacement}) \quad (2.70)$$

In comparison to the efficiency (2.66) of N consecutive single-particle displacements, the efficiency (2.70) of one instantaneous N-particle displacement is down by a factor of $1/N$!
Note that here we have assumed that all particles are displaced independently from one another. Indeed, it is possible to construct *quite efficient collective many-particle moves*, in which many particles are moved in a coordinated, non-independent way.

2.3.5 How large should the displacement Δ be chosen?

The efficiency of a MC simulation crucially depends on the choice of the displacement size Δ.

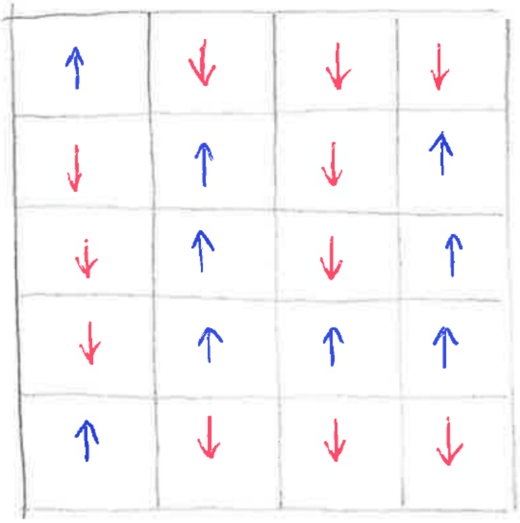
- For very *small* displacements, the energy changes are small and the trial moves have a very high likelihood to be accepted. However, since the displacements are small, the system moves through configuration space slowly, leading to a low efficiency.
- If, on the other hand, displacements are very *large* on average, trial moves most likely lead to large energy changes. Such moves have a very small likelihood to be accepted. Thus, for large displacements the efficiency will also be low, albeit for different reasons than for small displacements.
- In between these two extreme cases, there is an intermediate displacement size Δ_{opt} - with intermediate acceptance probability that yields the largest mean square displacement $\langle(\Delta \boldsymbol{r}^N)^2\rangle$, and hence the *optimum efficiency.*



Often one finds statements like “An optimal efficiency will be obtained for acceptance rates around 50%.” However, this can be regarded as merely a rough rule of thumb. Indeed, a detailed error analysis often reveals that the optimal acceptance rate may be considerably lower. For instance, for hard core systems, for which rejected moves are computationally cheaper than accepted moves, the optimal acceptance probability is actually closer to 20% rather than 50%.

2.4 Lattice models

Many fundamental phenomena in statistical physics can be studied with the help of lattice models, which are physical models defined on a discrete regular lattice rather than in continuous space like the many-particle systems we have considered so far. Particularly in the field of condensed matter physics, lattice models have provided many important insights. For instance, phase transitions were studied using lattice models, which offer a particularly transparent way to discuss and understand collective effects arising from interparticle interactions. Lattice models have the advantage that they sometimes are amenable to analytical solutions. In many cases, however, computer simulations are the only way to study the statistical mechanics of these systems.
To understand the significance and usefulness of lattice models, let us consider the *Ising model*. In two dimensions, this model, which is named after Ernst Ising (1900-1948) who originally investigated it in one dimension with the goal to study ferromagnetism, consists of N spins arranged on a square lattice:



- Note that the Ising model can be mapped to the so-called *lattice gas*, in which each lattice site is either occupied by a particle or is empty.

Each spin is represented by a discrete variable s_i which can take only the two values

$$s_i = \pm 1 \tag{2.71}$$

A particular state ν of the system is specified by the values of all N spins

$$\nu = \{s_1, \ldots, s_N\} \tag{2.72}$$

In the presence of an external magnetic field H, the energy of state ν is given by

$$E_\nu = -J \sum_{i,j}{}' s_i s_j - H\mu \sum_i s_i \tag{2.73}$$

Here μ is the *magnetic moment* of a spin and J is the *coupling constant* that controls the strength of interactions of spins. The sum with the prime $\sum'$ extends over all nearest neighbor pairs of spins.

- For a simple cubic d-dimensional lattice with lattice constant a, the nearest neighbors of a lattice site $\boldsymbol{x}$ are the sites $\boldsymbol{x} \pm a\boldsymbol{e}_\mu$, $\mu = 1, \ldots, d$, where $\boldsymbol{e}_\mu$ denotes the unit vector in direction μ.

In the canonical ensemble, i.e. if the spin lattice is in contact with a heat bath of temperature T, the probability $\rho(\nu)$ to find the system in configuration ν is given by

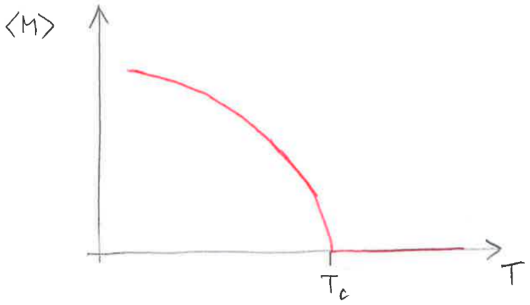
$$\rho(\nu) = \frac{e^{-\beta E_\nu}}{\sum_\nu e^{-\beta E_\nu}} \tag{2.74}$$

where the sum in the denominator extends over all spin configurations ν. Hence configurations with low energies have a higher probability to occur.

Let us now take a closer look at the energy appearing in Eq. (2.73). If the coupling constant $J > 0$, neighboring spins that are aligned (i.e. they point in the same direction and are both either $+1$ or -1) produce a lower contribution to the energy than if they are of opposite sign. Therefore, configurations with many aligned pairs of neighboring sites are energetically favored. For sufficiently low temperatures, this stabilization leads to a cooperative phenomenon called *spontaneous magnetization*: due to the interactions between nearest neighbors, a large fraction of spins points in the same direction, leading to a *net magnetization* even in the absence of a magnetic field. More specifically, in this case the magnetization

$$M = \sum_{i=1}^{N} \mu s_i \tag{2.75}$$

has an average value different from zero.



For temperatures larger than the *critical temperature* T_C (also called the *Curie temperature*), the magnetization vanishes because thermal fluctuations destroy the long range order. Representing a milestone in theoretical physics, the partition function

$$Q = \sum_{\nu} e^{-\beta E_{\nu}} \tag{2.76}$$

of the 2d Ising model has been solved analytically for vanishing external field $H = 0$ by Lars Onsager in 1944, so all its thermodynamic properties are known exactly.
Consider now the average of a given quantity $A(\nu)$ in the canonical ensemble

$$\langle A \rangle = \frac{\sum_{\nu} A(\nu) e^{-\beta E_{\nu}}}{\sum_{\nu} e^{-\beta E_{\nu}}} \tag{2.77}$$

In general, such averages cannot be computed analytically and one has to resort to computer simulations to determine them. So in the following we will discuss how - using the MC approach - we can generate a sequence of configurations ν_m that samples the canonical ensemble such that the average $\langle A \rangle$ can be approximated as an average

$$\langle A \rangle \approx \frac{1}{M} \sum_{m=1}^{M} A(\nu_m) \tag{2.78}$$

over this sequence. In order to do that we need to

- generate a new configuration $\nu^{(n)}$ from an old one, $\nu^{(o)}$, and then
- accept or reject the new configuration according to an appropriate criterion.

Repeating this basic step will yield the desired sequence of configurations.
The simplest way to generate a new configuration from an old one is the *single spin flip algorithm* consisting of the following steps:

1. Pick one of the N spins at random. One can do that by generating a random number ξ from a uniform distribution in the interval $(0, 1)$ and determining the index $i \in \{1, \ldots, N\}$ of the spin as (cf. (2.49))

$$i = \lfloor N\xi \rfloor + 1 \tag{2.79}$$

2. Next, the selected spin is flipped:

$$s_i^n = -s_i^o \tag{2.80}$$

3. The energy difference between the new and the old configuration is computed:

$$\Delta E = E_{\nu^n} - E_{\nu^o} \tag{2.81}$$

4. The new configuration ν^n is accepted with probability

$$p_{acc}(\nu^o \to \nu^n) = \min\left[1, e^{-\beta \Delta E}\right] \tag{2.82}$$

   For $\Delta E \leq 0$ the move is therefore always accepted, while for $\Delta E > 0$ the move is accepted with probability $e^{-\beta \Delta E}$.

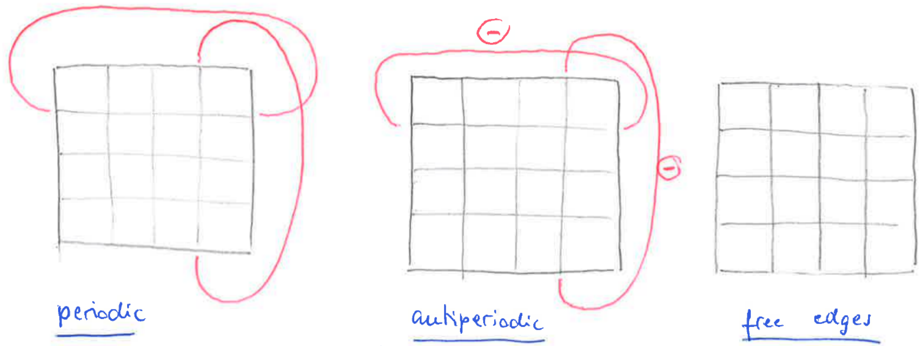
5. Update averages. If the move was rejected, the old configuration is counted again.

6. Go to 1.

Obviously, to start this procedure, one needs an initial configuration, which can e.g. be generated by setting all spins to 1 or -1. One can also generate an initial configuration by assigning ±1 randomly to each spin.

2.4.1 Boundary conditions

In the calculation of the energy difference in step 3., it is important to take into account the appropriate *boundary conditions*. Which boundary conditions are used depends on the physical situation one wants to study. The following boundary conditions are frequently used (others exist as well):



(the sign of the coupling J is reversed for interactions across the boundaries

2.4.2 Calculating the energy difference

For the 2d Ising model with nearest neighbor interaction, step 3. requires, of course, only to consider the *nearest neighbors* of the selected spin:

$$\Delta E = -J \sum_{\substack{j \\ j \text{ n.n. of } i}} (s_i^n - s_i^o)s_j^o - H\mu(s_i^n - s_i^o) \tag{2.83}$$

The difference $s_i^n - s_i^o$ can, however, only be -2 (if we flip s_i^o from $+1$ to -1) or $+2$ (if we flip s_i^o from -1 to $+1$), so we can write this difference in both cases as

$$s_i^n - s_i^o = 2s_i^n \tag{2.84}$$

and the above energy difference turns into

$$\Delta E = -2J \sum_{\substack{j \\ j \text{ n.n. of } i}} s_i^n s_j^o - 2H\mu s_i^n \tag{2.85}$$

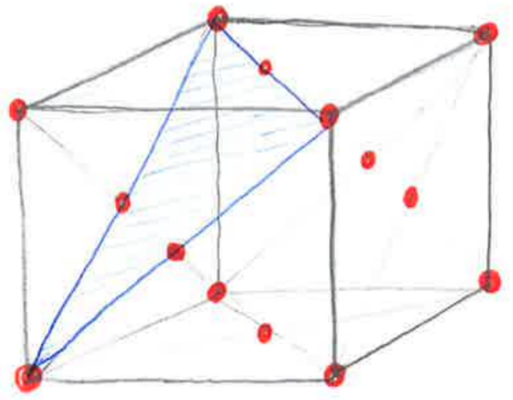
2.5 Initialization of MC simulations

Every MC simulation needs a configuration to start with. Often, this initial configuration will be the final (and therefore hopefully well equilibrated) configuration of a prior simulation. In many cases, however, no initial configuration is available, and we need to construct it from scratch. In principle, it is not required that the initial configuration is a typical equilibrium configuration, since the MC simulation will always relax towards equilibrium. However, if one chooses a very untypical (i.e. unphysical) initial configuration, this relaxation may take a very long time, thus wasting a lot of CPU time, since during such an *equilibration period* averages should not be collected.

- If one plans to simulate a *crystalline* phase, it makes sense to prepare the system in the crystalline structure one wants to study. One then needs to pick an appropriate particle number and geometry for the simulation box such that the system fits into the simulation box without causing any defects such as vacancies.

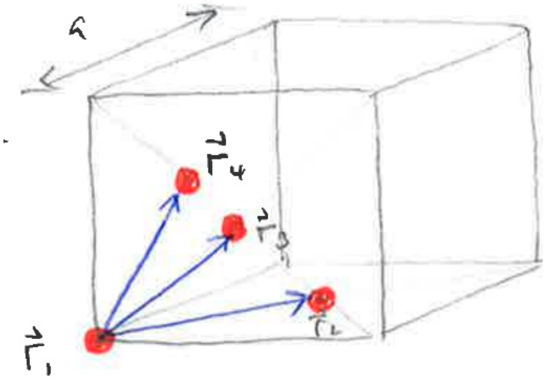
- To simulate a *liquid*, one may be inclined to place the individual particle into the simulation box at random using a uniform distribution. However, at high densities, this procedure will produce large particle "overlaps" (i.e. high repulsive energies for soft particles) and it may be very time-consuming to relax to equilibrium. The alternative is again to prepare a crystalline system by placing the particles at the sites of an appropriate regular lattice. This allows to generate initial configurations without pronounced particle overlaps even at high densities. This crystalline configuration is then equilibrated at a high temperature until it melts (this step may still take a while, because the system may reside in the overheated crystalline state for considerable time until it finally overcomes the nucleation barrier). Slowly cooling the resulting liquid then brings the system to the desired state point.

Frequently, a *face-centered cubic* (fcc) structure is used as the initial structure to produce a high density liquid.



In the fcc structure, atoms are located at the corners and the centers of the faces of a cubic unit cell (hence the name “face centered”). Since the atoms at the corners are shared among eight neighboring unit cells and the atoms on the faces among two neighboring unit cells, a (conventional) fcc unit cell with lattice constant a contains 8/8+6/2 =1+3=4 atoms at sites

$$\boldsymbol{r}_1 = \begin{pmatrix}0\\0\\0\end{pmatrix}, \quad \boldsymbol{r}_2 = \frac{a}{\sqrt{2}}\begin{pmatrix}1\\1\\0\end{pmatrix}, \quad \boldsymbol{r}_3 = \frac{a}{\sqrt{2}}\begin{pmatrix}1\\0\\1\end{pmatrix}, \quad \boldsymbol{r}_4 = \frac{a}{\sqrt{2}}\begin{pmatrix}0\\1\\1\end{pmatrix} \tag{2.86}$$



An fcc crystal is generated by replicating this unit cell. In a simulation cubic box with PBC, one can arrange n^3 fcc unit cells in a cubic way, where n is the number of cubic unit cells per side length of the box. The total number of atoms in such a box is then

$$N = 4n^3 \tag{2.87}$$

which implies that in order to completely fill a cubic simulation box with a perfect fcc crystal one has to choose N such that it equals $4n^3$ for an integer n. Possible particle numbers are therefore the so-called *magic numbers* of the fcc lattice

$$N = 4,\ 32,\ 108,\ 256,\ 500,\ 864, \ldots \tag{2.88}$$

The fcc lattice is a *close packed structure*: identical hard spheres arranged on an fcc lattice such that they touch each other have the highest possible density. The *packing fraction*, i.e. the fraction of space actually occupied by the spheres, is

$$\phi = \frac{\pi}{3\sqrt{2}} \approx 0.74048 \tag{2.89}$$

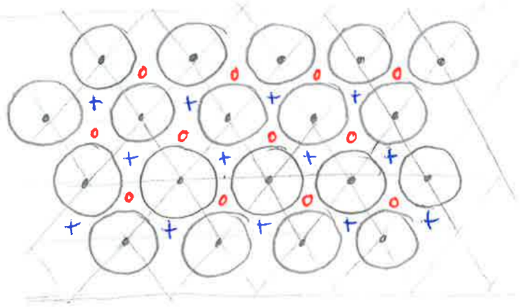
- *Proof.* In the fcc structure, two nearest neighbor atom centers are at distance $d = a\frac{\sqrt{2}}{2}$. If they touch each other, they therefore have radius $r = \frac{d}{2} = a\frac{\sqrt{2}}{4}$ and volume

$$\frac{4\pi}{3}r^3 = \frac{4\pi}{3}\left(a\frac{\sqrt{2}}{4}\right)^3 = \frac{4\pi}{3}\frac{2\sqrt{2}}{4^3}a^3 = \frac{\sqrt{2}\pi}{24}a^3 \tag{2.90}$$

And since there are 4 atoms per unit cell, we obtain a packing fraction of

$$\phi = \frac{4\cdot\frac{\sqrt{2}\pi}{24}a^3}{a^3} = \frac{\pi}{3\sqrt{2}} \qquad \checkmark \tag{2.91}$$

In the fcc structure each atom has 12 nearest neighbors. One can think of this structure as arising in the following way.



1. One first places a regular hexagonal *first layer* of particles in the plane.
2. A *second layer* is then placed on top of the first layer at the positions indicated by the red circles ○.
3. For the *third layer* one has two choices:

- One can either place the particles exactly above those of the first layer (positions indicated by the gray dots ●). This kind of stacking the layers is called $ABABA\ldots$ stacking and it results in the *hexagonal close-packed structure* (hcp). In this stacking every other layer is the same in the xy-plane.
- The other possibility is to place the third layer atoms at the positions indicated by the blue crosses +. This type of stacking is called $ABCABC\ldots$-stacking and it is the one that produces an *fcc*-crystal. In this stacking, every third layer is the same. The horizontal layers considered in this stacking correspond to 111-lattice planes of the fcc lattice.

Note, however, that other stacking sequences of A, B and C layers also produce close-packed structures. This includes periodic structures of more complicated nature, or even *random stackings*.

2.6 Equilibration

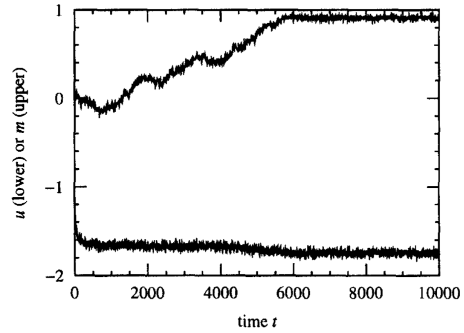
To obtain meaningful estimates of the measured observables in a MC simulation, it is very important that the simulation actually samples the equilibrium state for the prescribed thermodynamic conditions. Should we include all measurements recorded since the very start of the simulation?
In general, the answer is "No!". The only exception may be the fortunate situation that e.g. a configuration from an already equilibriated previous simulation is available to start from. Most of the time, however, we need to start from scratch. As discussed above, it is then wise to try assemble an initial configuration in a way that is expected to resemble as closely as possible a typical equilibrium state of the system. Nevertheless, despite our best efforts, once we then start the simulation we will frequently observe a noticeable relaxation period, in which the values of the monitored observables systematically evolve from their initial "manufactured" towards their true equilibrium values. Once equilibrium is actually reached, the values A_i measured for an observable A will eventually be found to fluctuate around a specific plateau value $\langle A\rangle$ corresponding to its equilibrium value. At this stage, any "memory" the system had of its initial non-equilibrium configuration has decayed. Apart from the specific physics of the system, this *equilibration time* $\tau_{A,eq}$ for observable A needed to complete equilibration is found to depend on

- the specific algorithm used
- the size of the system
- and (important!) the specific observable under consideration
  - We illustrate this point taking the example of the 2d Ising model of size 100 × 100 at vanishing external field $H = 0$ and temperature $T = 2$ with periodic boundary conditions (units chosen such that $k_B = J = 1$). Suppose we start the simulation with a pure random spin configuration, which actually resembles a typical configuration at $T = \infty$.

In MC, it is desirable to monitor the progress of a simulation as a function of some internal "MC time" measured in a unit that allows a fair comparison of the properties of simulation algorithms at different system sizes. It has therefore become standard to measures MC time in units of so-called *sweeps*, where

$$1 \text{ sweep} := N \times \text{time for 1 MC step} \tag{2.92}$$

The following plot, taken from the book of Newman & Barkema shows the evolution of the internal energy (lower curve) and magnetization per spin (upper curve) measured in sweeps:



The plot suggests that after about 6000 initial sweeps the system may have reached equilibrium, but also indicates that the internal energy seems to equilibrate somewhat faster than the magnetization.

Clearly, if we want to calculate $\langle A\rangle$ from an average $\bar{A} = M^{-1}\sum_{i=1}^{M} A_i$, it is mandatory to discard all data A_i observed during this equilibration period. The obvious problem is, however, how to make sure that equilibrium has truely been reached. In unfavorable scenarios the system could remain trapped in metastable states for a very long time before finally escaping to the true equilibrium state. To minimize the chance of something like this happening, we may run separate simulations

- from distinct initial states (e.g. a fully disordered vs. a completely ordered spin state in the Ising model)
- and/or using different seeds for the random number generator

and let them evolve until the values of the observables under scrutiny are seen to merge at common plateaus.

We will continue the discussion of this topic in Sec. 3.4. Before we can do that, however, let us concentrate on the general topic of data evaluation and assessment of statistical errors in MC and other simulations.

Chapter 3

Data evaluation and statistical errors

3.1 Correlations

Since averages in simulations are calculated over a finite sample, these averages are affected by statistical errors. In this section we will estimate these errors, paying particular attention to the effect of correlations.
Suppose that our goal is to estimate the thermodynamic expectation value $\langle A \rangle$ of the quantity $A(\boldsymbol{r}^N)$. Assume that we have successfully generated M configurations $\{\boldsymbol{r}_i^N : i = 1, \dots M\}$ distributed according to the canonical distribution. Let us abbreviate

$$A_i \equiv A(\boldsymbol{r}_i^N), \qquad i = 1, \dots, M \tag{3.1}$$

As we shall see below, in a MC simulation it hardly makes sense to perform a measurement after every single MC move. Indeed, "time" in MC simulations is usually measured in units of *sweeps*, where, as we have noted above, a sweep is - somewhat loosely - defined by N successive MC moves, where N is a measure of the number of relevant degrees of freedom of the system (total number of particles, number of lattice sites, ...). We consider the *sample average*

$$\overline{A} = \frac{1}{M} \sum_{i=1}^{M} A_i \tag{3.2}$$

While $\overline{A}$ will deviate from the desired ensemble average $\langle A \rangle$ due to the finite sample size $M < \infty$, it is an *unbiased* (see Sec. 3.2 below) estimator of $\langle A \rangle$, because

$$\langle \overline{A} \rangle = \left\langle \frac{1}{M} \sum_{i=1}^{M} A(\boldsymbol{r}_i^N) \right\rangle = \frac{1}{M} \sum_{i=1}^{M} \langle A \rangle = \frac{1}{M} M \langle A \rangle = \langle A \rangle \ \checkmark \tag{3.3}$$

where we used the linearity of expectation. The statistical error of $\overline{A}$ is estimated by the standard deviation $\sigma_{\overline{A}}$ of the sample mean, which is the square root of the variance of the sample average

$$\sigma_{\overline{A}}^2 = \left\langle (\overline{A} - \langle \overline{A} \rangle)^2 \right\rangle = \left\langle \overline{A}^2 - 2\overline{A}\langle \overline{A} \rangle + \langle \overline{A} \rangle^2 \right\rangle = \langle \overline{A}^2 \rangle - \langle \overline{A} \rangle^2 \qquad (3.4)$$

Inserting (3.2), we obtain

$$\begin{aligned} \sigma_{\overline{A}}^2 &\overset{(3.3)}{=} \left\langle \frac{1}{M^2} \sum_{i,j=1}^{M} A_i A_j \right\rangle - \left\langle \frac{1}{M} \sum_{i=1}^{M} A_i \right\rangle^2 \\ &= \frac{1}{M^2} \sum_{i,j=1}^{M} \left[\langle A_i A_j \rangle - \langle A_i \rangle \langle A_j \rangle \right] \qquad (3.5) \end{aligned}$$

In equilibrium, time translation invariance holds.

- For finite M, this is at least true for any choice of i, j satisfying $i, j \ll M$, since we will see that $\langle A_i A_j \rangle - \langle A_i \rangle^2 \to 0$ for $|j - i| \to \infty$.

This is now utilized twice:

1. The "diagonal" contribution $i = j$ in the above formula corresponds to the variance

   $$\sigma_A^2 = \langle A_i^2 \rangle - \langle A_i \rangle^2 \qquad (3.6)$$

   of the distribution of the individual data A_i, which does not depend on the "time" i.

   Factorizing σ_A^2 out of the summands of Eq. (3.5), we write

   $$\sigma_{\overline{A}}^2 = \frac{\sigma_A^2}{M^2} \sum_{i,j=1}^{M} \mathcal{A}_{ij} \qquad (3.7)$$

   where

   $$\mathcal{A}_{ij} \equiv \frac{\langle A_i A_j \rangle - \langle A_i \rangle \langle A_j \rangle}{\sigma_A^2}, \qquad (\mathcal{A}_{ii} = 1) \qquad (3.8)$$

   denotes the *normalized autocorrelation function*.

2. Again using translational invariance in time, we further rewrite

   $$\mathcal{A}_k \equiv \frac{\langle A_i A_{i+k} \rangle - \langle A_i \rangle^2}{\sigma_A^2}, \qquad (\mathcal{A}_0 = 1) \qquad (3.9)$$

   independent of the chosen time i appearing in the time series.

Then

$$\sigma_{\overline{A}}^2 = \frac{\sigma_A^2}{M} \sum_{k=0}^{M} \mathcal{A}_k \tag{3.10}$$

For completely uncorrelated data we would have $\mathcal{A}_{ij} = \delta_{ij}$ i.e. $\mathcal{A}_k = \delta_{0,k}$. In this case, (3.10) would reduce to

$$\sigma_{\overline{A}}^2 = \frac{\sigma_A^2}{M} \qquad \text{(uncorrelated data)} \tag{3.11}$$

In reality, however, efficient MC algorithms always generate data that are correlated to some degree. Use of Eq. (3.11) is therefore only justified if we are able to select an uncorrelated subset of M_{eff} measurements from our M measured data. In principle, this is done by simply waiting between subsequent measurements for correlations to decay. The corresponding waiting time is of the order of the so-called (proper) *integrated autocorrelation time* $\tau_{A,\text{int}}$. If this time is larger than the measurement interval +1 (in units of MC sweeps) at which the M data A_i are collected, then we expect the (3.11) to be replaced by

$$\sigma_{\overline{A}}^2 = \frac{\sigma_A^2}{M_{\text{eff}}}, \qquad \text{where} \quad M_{\text{eff}} = \frac{M}{2\tau_{A,\text{int}}} \tag{3.12}$$

which is larger then the naive variance by a factor of $2\tau_{A,\text{int}}$, because the effective number of uncorrelated measurements M_{eff} is smaller than the number of measured data by the same factor!

To compute $\tau_{A,\text{int}}$, we return to Eq. (3.7), rewriting

$$\begin{aligned}
\sigma_{\overline{A}}^2 &\overset{(3.7)}{=} \frac{\sigma_A^2}{M^2} \sum_{i,j=1}^{M} \mathcal{A}_{ij} = \frac{\sigma_A^2}{M^2} \left(M + 2 \sum_{\substack{i,j=1 \\ j>i}}^{M} \mathcal{A}_{ij} \right) \\
&= \frac{\sigma_A^2}{M} \left(1 + \frac{2}{M} \sum_{i=1}^{M} \sum_{j=i+1}^{M} \mathcal{A}_{ij} \right)
\end{aligned} \tag{3.13}$$

Upon substituting $j = i + k$ we get the scheme

$$\begin{aligned}
\sum_{i=1}^{M} \sum_{j=i+1}^{M} \mathcal{A}_{ij} &= \sum_{i=1}^{M} \sum_{k=1}^{M-i} \mathcal{A}_{i,i+k} \overset{(3.9)}{=} \sum_{i=1}^{M} \sum_{k=1}^{M-i} \mathcal{A}_k \\
&= (i=1): \quad \mathcal{A}_1 + \mathcal{A}_2 + \cdots + \mathcal{A}_{M-3} + \mathcal{A}_{M-2} + \mathcal{A}_{M-1} \\
& (i=2): \quad +\mathcal{A}_1 + \mathcal{A}_2 + \cdots + \mathcal{A}_{M-3} + \mathcal{A}_{M-2} \\
& (i=3): \quad +\mathcal{A}_1 + \mathcal{A}_2 + \cdots + \mathcal{A}_{M-3} \\
& \vdots \\
& (i=M-1): +\mathcal{A}_1
\end{aligned} \tag{3.14}$$

from which we read off that

$$\sum_{i=1}^{M} \sum_{j=i+1}^{M} \mathcal{A}_{ij} = \sum_{k=1}^{M} (M-k)\mathcal{A}_k \tag{3.15}$$

Hence, confirming our expectation (3.12), the result for the variance is

$$\sigma^2_{\overline{A}} = \frac{\sigma^2_A}{M} \cdot 2\tau_{A,\mathrm{int}} \tag{3.16}$$

where the *(integrated) autocorrelation time* in units of MC sweeps is

$$\tau_{A,\mathrm{int}} = \frac{1}{2} + \sum_{k=1}^{M} \mathcal{A}_k \left(1 - \frac{k}{M}\right) \tag{3.17}$$

- For uncorrelated data $\mathcal{A}_k = \delta_{0k}$ and so $\tau_{A,\mathrm{int}} = 1/2$ ✓.
- As a rule of thumb, a lag of $2 \times \tau_{A,\mathrm{int}}$ gives a new independent configuration.

For large k, the autocorrelation function (3.9) decays exponentially,

$$\mathcal{A}_k \overset{(k\to\infty)}{\longrightarrow} a_0 e^{-k/\tau_{A,\exp}} \tag{3.18}$$

Thus, since for a meaningful simulation $M \gg \tau_{A,\exp}$ must hold, the factor $\left(1 - \frac{k}{M}\right)$ in (3.17), which decreases linearly with increasing k, can be safely neglected, which yields the usually employed definition of the integrated autocorrelation time

$$\tau'_{A,\mathrm{int}} := \frac{1}{2} + \sum_{k=1}^{M} \mathcal{A}_k \tag{3.19}$$

In the simple case that the decay of $\mathcal{A}_k \equiv e^{-k/\tau_{A,\exp}}$ is ideally exponential, sending $M \to \infty$ and using the trapezoidal rule

$$\int_a^b dx\, f(x) \approx \sum_{k=1}^{M} \frac{f(x_{k+1}) + f(x_k)}{2} \Delta x_k \overset{(\text{all } \Delta x_k \text{ equal})}{\longrightarrow} \frac{\Delta x}{2} \left[f(a) + 2 \sum_{k=1}^{M-1} f(x_k) + f(b) \right]$$

with $a = 0,\ b \gg 1,\ f(0) = 1,\ f(b) \approx 0,\ \Delta x = 1$ yields the approximation

$$\tau'_{A,\mathrm{int}} \approx \int_0^\infty dk\, e^{-k/\tau_{A,\exp}} = \tau_{A,\exp} \tag{3.20}$$

which should clarify the prefix "integrated" and also explain the extra factor of 2 in the introduction (3.16) of $\tau_{A,\text{int}}$. In general, however, the decay of $\mathcal{A}_k$ will be governed by a whole distribution of exponential decay times, and then $\tau_{A,\text{int}}, \tau'_{A,\text{int}} \neq \tau_{A,\text{exp}}$. In fact, for realistic models one can show [Sokal & Thomas, J. Stat. Phys. **63** 867 (1991)] that

$$\tau_{A,\text{int}} \leq \tau_{A,\text{exp}} \tag{3.21}$$

3.2 Bias

Finite sampling time is not just the source of statistical errors in the calculated averages for the observables of interest that we have discussed in the previous section, but can also lead to additional systematic errors in the form of *bias*. In statistics, an estimator $\widetilde{o}(A)$ of an observable $o(A)$ for finite sample size M is called *biased*, if its expectation value

$$\langle \widetilde{o}(A) \rangle \neq \langle o(A) \rangle, \qquad M < \infty \tag{3.22}$$

systematically differs from the exact expectation value $\langle o(A) \rangle$ for finite sample size. Otherwise the estimator is called *unbiased*.

- As we have shown in (3.3), the sample average $\overline{A} = \frac{1}{M}\sum\limits_{i=1}^{M} A(\boldsymbol{r}_i^N)$ is an example of an unbiased estimator for $\langle A \rangle$.
- However, the naive estimator

  $$\tilde{\sigma}_A^2 := \overline{(A - \overline{A})^2} = \overline{A^2} - \overline{A}^2 \tag{3.23}$$

  for σ_A^2 *is biased*. To see this, we evaluate

  $$\langle \tilde{\sigma}_A^2 \rangle = \langle \overline{A^2} \rangle - \langle \overline{A}^2 \rangle \tag{3.24}$$

  Adding and subtracting $\langle A \rangle^2 = \langle \overline{A} \rangle^2$, we obtain

  $$\langle \tilde{\sigma}_A^2 \rangle = \underbrace{\langle \overline{A^2} \rangle - \langle \overline{A} \rangle^2}_{\sigma_A^2} - \underbrace{\left(\langle \overline{A}^2 \rangle - \langle \overline{A} \rangle^2 \right)}_{\overset{(3.4)}{=} \sigma_{\overline{A}}^2 \overset{(3.12)}{=} \frac{\sigma_A^2}{M_{\text{eff}}}} = \sigma_A^2 \left(1 - 1/M_{\text{eff}}\right) \neq \sigma_A^2 \checkmark$$

  - In fact, we have shown that $\tilde{\sigma}_A^2$ is *weakly biased*, i.e. its bias $\propto 1/M_{\text{eff}}$ is asymptotically smaller than the statistical error $\propto 1/\sqrt{M_{\text{eff}}}$; the possibility of a large numerical prefactor should, however, not be overlooked. . . .
  - Note that $\tilde{\sigma}_A^2$ is biased even for completely uncorrelated data, since in this case M_{eff} is merely replaced by M, leading to the celebrated replacement of $1/M$ by $1/(M-1)$ due to "one missing degree of freedom".

We can, however, easily come up with a *bias-corrected estimator* $\widehat{\sigma}_A^2$ *for* σ_A^2:

$$\widehat{\sigma}_A^2 := \frac{\tilde{\sigma}_A^2}{1 - 1/M_{\text{eff}}} = \frac{M_{\text{eff}}}{M_{\text{eff}} - 1}\tilde{\sigma}_A^2 = \frac{M_{\text{eff}}}{M_{\text{eff}} - 1}\left(\overline{A^2} - \overline{A}^2\right) \tag{3.25}$$

Again using $\overline{A^2} - \overline{A}^2 = \overline{\left(A - \overline{A}\right)^2}$, we rewrite this in more practical terms as

$$\widehat{\sigma}_A^2 = \frac{M_{\text{eff}}}{M_{\text{eff}} - 1}\overline{\left(A - \overline{A}\right)^2} = \frac{M_{\text{eff}}}{M_{\text{eff}} - 1}\frac{1}{M}\sum_{i=1}^{M}\left(A_i - \overline{A}\right)^2 \tag{3.26}$$

Recalling the relation (3.12) between σ_A^2 and $\sigma_{\overline{A}}^2$, we obtain a bias-corrected estimator for the variance of the average $\overline{A}$

$$\widehat{\sigma}_{\overline{A}}^2 = \frac{1}{M(M_{\text{eff}} - 1)}\sum_{i=1}^{M}\left(A_i - \overline{A}\right)^2 \tag{3.27}$$

Our final bias-corrected estimator for the error of the average $\overline{A}$ in the presence of correlations is therefore

$$\widehat{\sigma}_{\overline{A}} = \sqrt{\frac{1}{M(M_{\text{eff}} - 1)}\sum_{i=1}^{M}\left(A_i - \overline{A}\right)^2} \tag{3.28}$$

3.3 Block averages

Error estimates that are directly based on Eq. (3.28) can be very difficult to carry out in practice, since the calculation of $\tau_{A,\text{int}}$ is tedious and time-consuming.

- To calculate $\tau_{A,\text{int}}$, Eq. (3.17) can be evaluated numerically. However, for large $k \gg \tau_{A,\text{int}}$ the numerical estimator

$$\widetilde{\mathcal{A}}_k = \frac{\frac{1}{M-k}\sum_{i=1}^{M-k} A_i A_{i+k} - \left(\frac{1}{M-k}\sum_{i,j=1}^{M-k} A_i\right)\left(\frac{1}{M-k}\sum_{i,j=1}^{M-k} A_{j+k}\right)}{\sigma_A^2} \tag{3.29}$$

for the normalized autocorrelation function (3.9) is observed to be very noisy. A way out is to cut off the summation self-consistently at, say, $6\tau_{A,\text{int}}$, such that we obtain the equation

$$\tau_{A,\text{int}} = \frac{1}{2} + \sum_{k=1}^{6\tau_{A,\text{int}}} \widetilde{\mathcal{A}}_k \left(1 - \frac{k}{M}\right) \tag{3.30}$$

which can be numerically solved for $\tau_{A,\text{int}}$.

There is, however, a more convenient method to estimate $\tau_{A,\mathrm{int}}$ and thus $\sigma_{\overline{A}}$. Observe that if we choose the measurement interval of the order of $\tau_{A,\mathrm{int}}$, then the measurements become effectively uncorrelated. But then $M_{\mathrm{eff}} \approx M$, and (3.28) approaches the naive estimate $\sigma_{\overline{A}} = \sqrt{[1/M(M-1)] \sum_{i=1}^{M} \left(A_i - \overline{A}\right)^2}$. This observation is the general idea behind the blocking method.

- Of course, deliberately taking more than M_{eff} measurements by decreasing the measurement interval still produces correct averaging values, but merely increases the redundancy in the data without further improving the statistics. In fact, since computing the required averages may be computationally expensive, the efficiency of a simulation is actually likely to decrease upon choosing measurement intervals smaller than $\tau_{A,\mathrm{int}}$.

Let us partition our M measured points

$$\{A_i, i = 1, \ldots, M\} = \bigcup_{k=1}^{K} \left\{ A_{(k-1)M^{(B)}+m^{(B)}} : m^{(B)} = 1, \ldots, M^{(B)} \right\} \tag{3.31}$$

into K "blocks", each containing $M^{(B)}$ consecutive datapoints, such that $M = K \times M^{(B)}$. If the block size $M^{(B)} \gg \tau_{A,\mathrm{int}}$, then the data in different blocks will be *effectively uncorrelated*. We can then think of these blocks as the results of K short independent simulations (e.g. with different initializations). For each block with index k, we can compute its arithmetic average

$$\overline{A}_k^{(B)} = \frac{1}{M^{(B)}} \sum_{m^{(B)}=1}^{M^{(B)}} A_{(k-1)M^{(B)}+m^{(B)}}, \qquad k = 1, \ldots, K \tag{3.32}$$

Trivially, the arithmetic average over all these K block averages

$$\overline{\overline{A}^{(B)}} = \frac{1}{K} \sum_{k=1}^{K} \overline{A}_k^{(B)} = \overline{A} \tag{3.33}$$

is equal to the total arithmetic sample average $\overline{A}$.
Since the blocks are assumed to be mutually uncorrelated (i.e. $K_{\mathrm{eff}} = K$), the variance of these averages $\overline{A}_k^{(B)}$, $k = 1, \ldots, K$ can be determined from the bias-corrected estimator

$$\widehat{\sigma}^2_{\overline{A}^{(B)}} \overset{(3.26)}{=} \frac{1}{K-1} \sum_{k=1}^{K} \left(\overline{A}_k^{(B)} - \overline{\overline{A}^{(B)}} \right)^2 \overset{(3.33)}{=} \frac{1}{K-1} \sum_{k=1}^{K} \left(\overline{A}_k^{(B)} - \overline{A} \right)^2 \tag{3.34}$$

and the bias-corrected estimator for the variance of the arithmetic mean of the blocks is

$$\widehat{\sigma}^2_{\overline{\overline{A}^{(B)}}} \overset{(3.28)}{=} \frac{\widehat{\sigma}^2_{\overline{A}^{(B)}}}{K} \overset{(3.34)}{=} \frac{1}{K(K-1)} \sum_{k=1}^{K} \left(\overline{A}_k^{(B)} - \overline{A} \right)^2 \tag{3.35}$$

Since, however, $\overline{\overline{A}^{(B)}} = \overline{A}$ according to (3.33), we must also have $\sigma^2_{\overline{\overline{A}^{(B)}}} \overset{(3.33)}{=} \sigma^2_{\overline{A}}$. Therefore, we can summarize the situation as follows:

$$\widehat{\sigma}^2_{\overline{A}} = \frac{\widehat{\sigma}^2_{\overline{A}^{(B)}}}{K} = \frac{1}{K(K-1)} \sum_{k=1}^{K} \left(\overline{A}_k^{(B)} - \overline{A} \right)^2 , \qquad (M^{(B)} \text{ large enough}) \quad (3.36)$$

is a bias-corrected estimator for the variance $\sigma^2_{\overline{A}}$ of the average $\overline{A}$. In particular, a comparison of the above relation $\widehat{\sigma}^2_{\overline{A}} = \frac{\widehat{\sigma}^2_{\overline{A}^{(B)}}}{K}$ of estimators to the general result $\sigma^2_{\overline{A}} \overset{(3.12)}{=} \frac{\sigma^2_A}{M_{\text{eff}}}$, allows to obtain an estimator

$$\widehat{M}_{\text{eff}}^{-1} \equiv \frac{1}{K} \frac{\widehat{\sigma}^2_{\overline{A}^{(B)}}}{\sigma^2_A} \qquad (3.37)$$

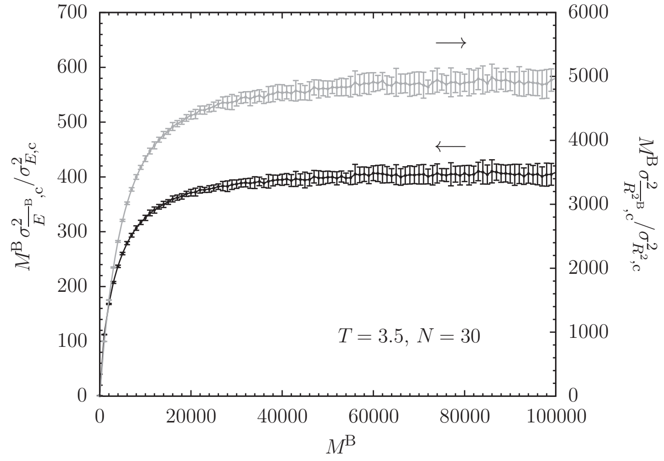
for the inverse effective number of statistically independent measurements. Recalling $2\tau_{A,\text{int}} = M/M_{\text{eff}}$ and $M = KM^{(B)}$, this finally yields an *estimator for the integrated autocorrelation time*:

$$\widehat{\tau}_{A,\text{int}} \equiv \frac{M^{(B)}}{2} \cdot \frac{\widehat{\sigma}^2_{\overline{A}^{(B)}}}{\sigma^2_A} \qquad (3.38)$$

The variance σ^2_A and the autocorrelation time $\tau_{A,\text{int}}$ are, however, intrinsic properties of the given time series and therefore independent of any blocking prescription. Hence, both sides of this equation must be *independent of* $M^{(B)}$ *provided* $M^{(B)} \gg \tau_{A,int}$ *is chosen large enough*! This observation therefore suggests a simple strategy to numerically determine $\tau_{A,\text{int}}$ without having to determine the autocorrelation function explicitly and perform the tedious sum (3.17):

- For a block size $M^{(B)} = 1$, the variance $\widehat{\sigma}^2_{\overline{A}^{(B)}}$ is equal to the naive variance of $\overline{A}$, which will vastly underestimate the correct variance in the presence of correlations. Thus, the right hand side of Eq. (3.38) is expected to severely underestimate $\tau_{A,\text{int}}$.
- Gradually increasing the block size $M^{(B)}$, we will observe an increase in the value of $M^{(B)} \cdot \frac{\widehat{\sigma}^2_{\overline{A}^{(B)}}}{\sigma^2_A}$.
- Once $M^{(B)}$ becomes large enough for the block averages $\overline{A}_k^{(B)}$ to be uncorrelated, the values of the right hand side of Eq. (3.38) should level off, forming a plateau, and further increase in $M^{(B)}$ only yields an increase in its statistical fluctuations. The height of this plateau then agrees, of course, to a very good approximation with the desired integrated autocorrelation function $\tau_{A,\text{int}}$.

To illustrate the above blocking procedure, we reproduce estimates of the integrated autocorrelation times for the energy (black data, label E) and the squared radius of gyration (label R^2, gray data) of a flexible polymer in the random-coil phase (data from [Qi & Bachmann, J. Chem. Phys. **141**, 074101 (2014)]):



- The small arrows above the two data sets refer to their corresponding scale on the left and right axis, respectively.

This plot also illustrates that different physical observables may exhibit quite distinct autocorrelation times.

3.4 Equilibration revisited

Recall the previous discussion of the equilibration time $\tau_{A,eq}$ from Sec. 2.6. In principle, the specific choice of $\tau_{A,eq}$ will clearly have an effect on the statistical quality of the average $\bar{A}$ we are aiming to compute. Obviously, there is a trade-off between

- a certain systematic (non-statistical) bias inflicted on $\bar{A}$ if we choose $\tau_{A,eq}$ too small, thereby unintentionally including (a small fraction of) nonequilibrium configurations in the average.
- an increase in statistical error (and possibly a small additional statistical bias) by discarding valid data if we choose $\tau_{A,eq}$ too large.

In most practical cases, it will certainly suffice to estimate $\tau_{A,eq}$ from visual inspection as we have explained above. However, in certain situations (for instance, if we have to deal with a large number of simulations for different values of external parameters) it would be desirable to have a method at our disposal that allows to determine the optimal choice of $\tau_{A,eq}$ automatically from a given dataset. Fortunately, such an approach has been worked out [J.D. Chodera, *A Simple Method for Automated Equilibration Detection in Molecular Simulations*, J. Chem. Theory Comput. **12**, 1799-1805 (2016)]. Here we only sketch the basic idea.

Recall from (3.12) that correlations between M successive measurements, signaled by an integrated autocorrelation time $\tau_{A,int}$ effectively lead to a reduction $M_{\text{eff}} = \frac{M}{2\tau_{A,\text{int}}}$ of the effectively statistically uncorrelated measurements. Dropping the first t_0 of the M measurements and performing this analysis with the remaining $M(t_0) = M - t_0$ data results in a slight but well-defined t_0-dependence of $\tau_{A,\text{int}} = \tau_{A,\text{int}}(t_0)$ and thus of $M_{eff} = M_{\text{eff}}(t_0) = \frac{M-t_0}{2\tau_{A,\text{int}}(t_0)}$. On lowering t_0 starting from some value well above $\tau_{A,eq}$, we should observe an increase in the number of effectively uncorrelated samples $M_{\text{eff}}(t_0)$. Once we reach values $t_0 < \tau_{A,eq}$, such that we are starting to include highly atypical initial data, $\tau_{A,\text{int}}$ will increase and thus we should observe a decrease in $M_{\text{eff}}(t_0)$. Thus, the value of t_0 at which we observe a maximum in the quantity $M_{\text{eff}}(t_0)$ should give a reasonable estimate of $\tau_{A,eq}$. This approach has been implemented as part of the package `pymbar` (see `https://github.com/choderalab/automatic-equilibration-detection`) by the authors of the cited paper, in which further details may be found.

3.5 Resampling Methods

Up to now we have discussed how to deal with the effects of time correlations in the data A_i measured for a single observable A produced from a Markov chain MC algorithm. In particular, we have

- shown that the sample average $\overline{A}$ is bias-free
- explained how to eliminate correlations between the data by the blocking strategy
- derived bias-free estimator $\widehat{\sigma}_{\overline{A}}$ of its error bar

We found, however, that the naive estimator $\tilde{\sigma}^2_A \overset{(3.23)}{=} \overline{(A-\overline{A})^2} = \overline{A^2} - \overline{A}^2$, which is built after the nonlinear function $\langle A^2\rangle - \langle A\rangle^2$ of the two different(!) observables A and A^2, was biased. Moreover, the question arises how to calculate an "error bar for the error bar", i.e. how to determine the statistical error bar of $\widehat{\sigma}_{\overline{A}}$.

Actually, in a simulators everyday life one may encounter quite a number of very important physical quantities that similarly depend not just on a single one but on several expectation values of different observables simultaneously, and it remains unclear how to devise a general method for removing the bias of their estimator and calculating a meaningful error bar.

To illustrate the situation with a concrete example, consider a MC simulation of a d-dimensional Ising type of model, in which we recorded the magnetizations $m_k = M_k/N$, $k = 1, \ldots, K$ per site and energies $e_k = E_k/N$, $k = 1, \ldots, K$

per site. Among the top priority targets of such a simulation are the following quantities:

- the *isochoric specific heat* $c_V = N\frac{\langle e^2\rangle - \langle e\rangle^2}{k_B T^2}$ requires to calculate
- the *isothermal magnetic susceptibilty per site* $\chi = N\frac{\langle m^2\rangle - \langle m\rangle^2}{k_B T}$ we need to determine
- In a finite size scaling analysis, so-called *Binder cumulant* for the energy $E_4 = \frac{1}{2}\left(3 - \frac{\langle e^4\rangle}{\langle e^2\rangle^2}\right)$ allows to distinguish first from second order transitions, while the magnetization cumulant $B_4 = \frac{1}{2}\left(3 - \frac{\langle m^4\rangle}{\langle m^2\rangle^2}\right)$ gives a convenient way to estimate the critical temperature of the infinite system. The fraction of expectation values of type $\frac{\langle x^4\rangle}{\langle x^2\rangle^2}$ appearing in these expressions essentially measures the "kurtosis" of the underlying respective distribution.

Suppose therefore that we have successfully performed such an MC simulation, in which we have gathered M data $A_i^{(\nu)}$, $i = 1, \dots, M$ for n observables $A^{(\nu)}, \nu = 1, \dots, n$. As a first pre-processing step, we divide the data into K blocks of size $M^{(B)}$ large enough to eliminate correlations between these blocks as explained in Section 3.3.

- For simplicity, let us assume that the observables A^ν share integrated autocorrelation times of similar magnitude, such that the block size $M^{(B)}$ that leads to a temporal decorrelation of our date and the resulting number of blocks K are also the same.

To save some writing, let

$$x_k^\nu \equiv \overline{A}_k^{(\nu,B)} \overset{(3.32)}{=} \frac{1}{M^{(B)}} \sum_{m^{(B)}=1}^{M^{(B)}} A^{(\nu)}_{(k-1)M^{(B)}+m^{(B)}}, \quad k = 1, \dots, K \tag{3.39}$$

denote these block-averaged decorrelated data, and introduce the formal n-dimensional vectors $\boldsymbol{x}_k \equiv (x_k^1, \dots, x_k^n)^T$.
From our previous work, we know that the sample averages $\overline{x}^\nu$ are bias-free, and we also have derived bias-free estimators $\widehat{\sigma}^2_{\overline{x}^\nu}$ of their variances. Generalizing the examples given above, we now study the problem how to obtain an estimator $\widehat{f}$ and an error bar for the quantity

$$f(\langle\boldsymbol{x}\rangle) \equiv f(\langle x^1\rangle, \dots, \langle x^n\rangle) \tag{3.40}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a (possibly nonlinear) scalar differentiable function. Investigating the above examples, an obvious candidate for the estimator $\widehat{f}$ is

$$\widehat{f} = f(\overline{\boldsymbol{x}}), \qquad \text{where} \quad \overline{\boldsymbol{x}} \equiv (\overline{x^1}, \dots, \overline{x^n}) \tag{3.41}$$

We would then like to answer the following questions:

- What is the bias of $\widehat{f}$ and how can we eliminate or at least reduce it?

- How can we calculate a meaningful error bar σ_f for $\widehat{f}$?

First we will try to tackle these problems with the "traditional" approach of Gaussian error propagation. While this effort will provide some insight into the problem, will turn out to lead to a quite cumbersome formalism due to the presence of the *mutually correlations* between the expectation values mentioned above. In the following sections, we will then move on to discuss more modern ways to deal with these problems: the *jackknife* and the *bootstrap*.

3.5.1 Preparation: Gaussian bias analysis and error propagation

In principle, Gaussian analysis constitutes the traditional way to calculate bias and error bars based on the knowledge of the individual standard deviations $\sigma_{\bar{x}^\nu}$ of the mean of the variables x^ν. The trouble is, however, that even after we have eliminated the *temporal* correlation in the data by blocking, the block-averaged data of different observables will still be *mutually cross-correlated* due to the trivial fact that they are derived from common raw data that were recorded along one and the same Markov chain. Hence, beyond the standard deviations one also needs to take cross-correlations of these multiple variables into account, which yields a complicated and impractical formalism. Nevertheless, the Gaussian approach serves as a valuable device for comparing and verifying the formulas of the more modern and much more convenient jackknife and bootstrap resampling strategies to be discussed later on.
Assuming f to be sufficiently smooth, we start by Taylor-expanding $f(\overline{\boldsymbol{x}})$ to second order in powers of $\delta_{\overline{x^\nu}} \equiv \overline{x^\nu} - \langle x^\nu \rangle$ and abbreviate $f \equiv f(\langle \boldsymbol{x} \rangle)$, $f_\mu \equiv \frac{\partial f(\boldsymbol{x})}{\partial x^\mu}|_{\boldsymbol{x}=\langle \boldsymbol{x} \rangle}$ etc. We have

$$f(\overline{\boldsymbol{x}}) = f + \sum_\mu f_\mu \delta_{\overline{x^\mu}} + \frac{1}{2} \sum_{\mu\nu} f_{\mu\nu} \delta_{\overline{x^\mu}} \delta_{\overline{x^\nu}} + \dots \tag{3.42}$$

Gaussian bias reduction

Taking the statistical average $\langle \dots \rangle$ of (3.42), we obtain

$$\langle f(\overline{\boldsymbol{x}}) \rangle - f \quad = \quad \sum_\mu f_\mu \underbrace{\langle \delta_{\overline{x^\mu}} \rangle}_{=0} + \frac{1}{2} \sum_{\mu\nu} f_{\mu\nu} \underbrace{\langle \delta_{\overline{x^\mu}} \delta_{\overline{x^\nu}} \rangle}_{\mathrm{Cov}(\overline{x^\mu}\, \overline{x^\nu})} + \dots$$

- Generalizing the variance $\mathrm{Var}(X) \equiv \sigma_X^2$ of a single random variable X, the *covariance* $\mathrm{Cov}(X, Y)$ of two random variables X, Y is defined as

$$\mathrm{Cov}(X, Y) = \langle XY \rangle - \langle X \rangle \langle Y \rangle \tag{3.43}$$

In particular, the variance of X is $\sigma_X^2 = \mathrm{Cov}(X, X)$, which may explain why one sometimes encounters notations like σ_{XY}^2 for $\mathrm{Cov}(X, Y)$. Such a notation, is, however, quite dangerous, since in contrast to the variance of X, the covariance of two arbitrary random variables X, Y has no reason to be non-negative.

Similar to (3.12) (for $K_{\text{eff}} \equiv K$), the covariances $\text{Cov}(\overline{x^\mu}, \overline{x^\nu})$ of the sample means $\overline{x^\mu}$ and $\overline{x^\nu}$ are related to the sample covariances $\text{Cov}(x^\mu, x^\nu)$ of x^μ and x^ν by

$$\text{Cov}(\overline{x^\mu}, \overline{x^\nu}) = \text{Cov}(x^\mu, x^\nu)/K \tag{3.44}$$

Thus, the bias of $f(\overline{\boldsymbol{x}})$ is

$$\langle f(\overline{\boldsymbol{x}})\rangle - f = \frac{1}{2K}\sum_{\mu\nu} f_{\mu\nu}\,\text{Cov}(x^\mu, x^\nu) + \dots \tag{3.45}$$

from which we see that – unless f is a purely linear function – the estimator $\widehat{f}$ is biased, and its the leading bias is proportional to $1/K$. This is not too bad, since it decays stronger than the typical statistical error $\sim 1/\sqrt{K}$, i.e. we have a *weak bias*, which is usually not a big deal. Taking into account higher order terms in the expansion, we anticipate that the total bias of $f(\overline{\boldsymbol{x}})$ will be of the form

$$\langle f(\overline{\boldsymbol{x}})\rangle - f = \frac{A}{K} + \frac{B}{K^2} + O(K^{-3}) \tag{3.46}$$

for certain constants $A, B, \dots$.

- It is instructive to analyze the bias of another estimator for $f = f(\langle x\rangle, \langle y\rangle)$. In fact, let us study the estimator

$$\widehat{f}^{(BAD)} := \overline{f(x,y)} = \frac{1}{K}\sum_{k=1}^{K} f(x_k, y_k) \tag{3.47}$$

which may appear to look equally reasonable at first glance. Using this estimator is, however, a very bad idea, since it is *strongly biased* (actually is rather an estimator for $\langle f(x,y)\rangle$ than for $f(\langle x\rangle, \langle y\rangle)$)! To see this, we carry out an analysis simular to what we have done above. We Taylor-expand

$$f(x_k, y_k) = f + f_x\delta_{x_k} + f_y\delta_{y_k} + \frac{f_{xx}}{2}\delta_{x_k}^2 + f_{xy}\delta_{x_k}\delta_{y_k} + \frac{f_{yy}}{2}\delta_{y_k}^2 + \dots \tag{3.48}$$

where $\delta_{x_k} = x_k - \langle x\rangle$ and so forth. Averaging results in

$$\begin{aligned}\langle f(x_k, y_k)\rangle &= f + 0 + 0 + \frac{f_{xx}}{2}\langle\delta_{x_k}^2\rangle + f_{xy}\langle\delta_{x_k}\delta_{y_k}\rangle + \frac{f_{yy}}{2}\langle\delta_{y_k}^2\rangle + \dots\\ &= f + \frac{f_{xx}}{2}\sigma_x^2 + f_{xy}\,\text{Cov}(x,y) + \frac{f_{yy}}{2}\sigma_y^2 + \dots \end{aligned} \tag{3.49}$$

Note that here it is *not* the variances and covariances of the sample means that appear but instead the sample variances and covariances themselves, which are larger than the former by a factor K as in the previous derivation! Inserting (3.49) into the definition of $\widehat{f}^{BAD}$, we obtain

$$\begin{aligned}\langle\widehat{f}^{(BAD)}\rangle &= \frac{1}{K}\sum_{k=1}^{K}\left[f + \frac{f_{xx}}{2}\sigma_x^2 + f_{xy}\,\text{Cov}(x,y) + \frac{f_{yy}}{2}\sigma_y^2 + \dots\right]\\ &= f + \frac{f_{xx}}{2}\sigma_x^2 + f_{xy}\,\text{Cov}(x,y) + \frac{f_{yy}}{2}\sigma_y^2 + \dots \end{aligned} \tag{3.50}$$

i.e. the bias

$$\langle\widehat{f}^{(BAD)}\rangle - f = \frac{f_{xx}}{2}\sigma_x^2 + f_{xy}\,\text{Cov}(x,y) + \frac{f_{yy}}{2}\sigma_y^2 + \dots \tag{3.51}$$

which is of order $O(1)$ i.e. does not die out for $K \to \infty$ unless f is a purely linear function, such that all its higher order derivatives vanish identically!

We may try to eliminate the leading bias contribution to this estimator. If we use bias-free estimators $\widehat{\mathrm{Cov}}(x^\mu, x^\nu)$ constructed according to (3.25), such that

$$\widehat{\mathrm{Cov}}(x^\mu, x^\nu) = \frac{K}{K-1}\widetilde{\mathrm{Cov}}(x^\mu, x^\nu), \qquad \widetilde{\mathrm{Cov}}(x^\mu, x^\nu) = \overline{x^\mu x^\nu} - \overline{x^\mu}\,\overline{x^\nu} \tag{3.52}$$

to substitute for expectation values in (3.45), we obtain the relation

$$f(\overline{\boldsymbol{x}}) - f = \frac{1}{2\cancel{K}}\frac{\cancel{K}}{K-1}\sum_{\mu\nu} f_{\mu\nu}\widetilde{\mathrm{Cov}}(x^\mu, x^\nu) + \dots \tag{3.53}$$

from which we read off a new estimator

$$\widehat{f}^G \equiv f(\overline{\boldsymbol{x}}) - \frac{1}{K-1}\sum_{\mu\nu}\frac{1}{2}\frac{\partial^2 f(\boldsymbol{x})}{\partial x_\mu \partial x_\nu}\Big|_{\boldsymbol{x}=\langle\boldsymbol{x}\rangle}\widetilde{\mathrm{Cov}}(x^\mu, x^\nu) \tag{3.54}$$

that is unbiased to leading order in $1/K$.

- Let us consider the case $f(x, y) = x - y^2$, where $x_i = m_i^2$, $y_i = m_i$ which corresponds to finding (up to proportionality) a biased-reduced estimator for the magnetic susceptibility correlation function $\langle m_i^2\rangle - \langle m_i\rangle^2$. Since here $f_{xx} = f_{xy} = 0$ and $f_{yy} = -2$, according to (3.54) a bias-reduced estimator is given by

  $$\widehat{f}^G = \widehat{f} + \frac{\tilde{\sigma}_y^2}{K-1} = \overline{x} - \overline{y}^2 + \frac{\overline{y^2} - \overline{y}^2}{K-1} \tag{3.55}$$

  Since here $\overline{y^2} = \overline{m_i^2} = \overline{x}$, this reduces to

  $$\widehat{f}^G = \frac{(K-1)(\overline{x} - \overline{y}^2) + \overline{x} - \overline{y}^2}{K-1} = \frac{K}{K-1}(\overline{x} - \overline{y}^2) \tag{3.56}$$

  which agrees with what we could have anticipated from our previous work.

Gaussian error estimation

From (3.42) we derive

$$\begin{aligned} f^2(\overline{\boldsymbol{x}}) &\overset{(3.42)}{=} \left(f + \sum_\mu f_\mu \delta_{\overline{x^\mu}} + \frac{1}{2}\sum_{\mu\nu} f_{\mu\nu}\delta_{\overline{x^\mu}}\delta_{\overline{x^\nu}} + \dots\right)^2 \\ &= f^2 + 2f\sum_\mu f_\mu \delta_{\overline{x^\mu}} + \sum_{\mu\nu}\left(f\, f_{\mu\nu} + f_\mu f_\nu\right)\delta_{\overline{x^\mu}}\delta_{\overline{x^\nu}} + \dots \end{aligned} \tag{3.57}$$

and so, similarly to what we did above,

$$\langle \widehat{f}^2\rangle = f^2 + \frac{1}{K}\sum_{\mu\nu}\left(f\, f_{\mu\nu} + f_\mu f_\nu\right)\mathrm{Cov}(x^\mu, x^\nu) + \dots \tag{3.58}$$

Altogether, replacing again the argument $\langle\boldsymbol{x}\rangle$ of the derivatives by $\overline{\boldsymbol{x}}$, we see that the error $\sigma_f^2 = \langle\hat{f}^2\rangle - \langle\hat{f}\rangle^2$ of f is

$$\sigma_f^2 = \frac{1}{K}\sum_{\mu\nu}\left(f\, f_{\mu\nu} + f_\mu f_\nu\right)\mathrm{Cov}(x^\mu, x^\nu) + \dots \tag{3.59}$$

and, replacing Cov by $\widehat{\mathrm{Cov}} \overset{(3.52)}{=} \frac{K}{K-1}\widetilde{\mathrm{Cov}}$, that the best estimator is

$$(\widehat{\sigma}_f^G)^2 \equiv \frac{1}{K-1}\sum_{\mu\nu}(f\, f_{\mu\nu} + f_\mu f_\nu)\, \widetilde{\mathrm{Cov}}(x^\mu, x^\nu) \tag{3.60}$$

In summary, using the above formalism it is possible to obtain

- an estimator $\widehat{f}^G$ for $f(\langle \boldsymbol{x} \rangle)$ that is unbiased to leading order in $1/K$, and
- an estimator $\widehat{\sigma}_f^G$ for the error σ_f^2 of order $1/\sqrt{K}$.

To order $1/\sqrt{K}$, our results therefore imply

$$f(\langle \boldsymbol{x} \rangle) = f(\overline{\boldsymbol{x}}) \pm \widehat{\sigma}_f^G \tag{3.61}$$

where the error $\widehat{\sigma}_f^G$ is computed from (3.60). Unfortunately, however, using these formulas it is cumbersome to keep track of all the partial derivatives, correlations and cross-correlations in practical computations. As we shall see, resampling methods like the jackknife or the bootstrap provide a much more convenient framework to reduce bias and determine error bars.

3.5.2 The jackknife

We define the i-th *jackknife* averages of the data vectors $\boldsymbol{x}_k$ as

$$\boldsymbol{x}_i^J := \frac{1}{K-1}\sum_{\substack{k=1\\k\neq i}}^{K} \boldsymbol{x}_k, \qquad i = 1, \dots K \tag{3.62}$$

- Note that trivially

$$\boldsymbol{x}_i^J = \frac{1}{K-1}\left[\sum_{k=1}^{K}\boldsymbol{x}_k - \boldsymbol{x}_i\right] = \frac{K\overline{\boldsymbol{x}} - \boldsymbol{x}_i}{K-1} = \overline{\boldsymbol{x}} + \frac{\overline{\boldsymbol{x}} - \boldsymbol{x}_i}{K-1} \tag{3.63}$$

such that these jackknife averages are much less scattered around $\overline{\boldsymbol{x}}$ than the $\boldsymbol{x}_k$.

Rather than estimating $f(\langle \boldsymbol{x} \rangle)$ by $\widehat{f} = f(\overline{\boldsymbol{x}})$ i.e. by evaluating f on the sample means, we now obtain K jackknife estimates of f averages by evaluating it on the K jackknife estimates:

$$\widehat{f}_i^J := f(\boldsymbol{x}_i^J), \qquad i = 1, \dots, K \tag{3.64}$$

- For example, determination of the Binder cumulant involves estimating the ratios $f(\langle x\rangle, \langle y\rangle) = \frac{\langle x\rangle}{\langle y\rangle^2}$, $x_k = m_k^4$, $y_k = m_k^2$ of powers of the K measured magnetizations per site m_k. Based on the sample means, we would then obtain the estimate

$$\widehat{f} = \frac{\overline{x}}{\overline{y}^4} = \frac{\frac{1}{K}\sum\limits_{k=1}^{K} x_k}{\left(\frac{1}{K}\sum\limits_{k=1}^{K} y_k\right)^2} = \frac{\frac{1}{K}\sum\limits_{k=1}^{K} m_k^4}{\left(\frac{1}{K}\sum\limits_{k=1}^{K} m_k^2\right)^2} \tag{3.65}$$

while the K jackknife estimates $\widehat{f}_i^J$ of f are given by

$$\widehat{f}_i^J = \frac{x_i^J}{(y_i^J)^2} = \frac{\frac{1}{K-1}\sum\limits_{k\neq i} x_k}{\left(\frac{1}{K-1}\sum\limits_{k\neq i} y_k\right)^2} = \frac{\frac{1}{K-1}\sum\limits_{k\neq i} m_k^4}{\left(\frac{1}{K-1}\sum\limits_{k\neq i} m_k^2\right)^2} \tag{3.66}$$

The overall jackknife estimator $\widehat{f}^J$ of $f(\langle \boldsymbol{x}\rangle)$ is the the arithmetic average over all K jackknife averages $\widehat{f}_i^J$:

$$\widehat{f}^J = \frac{1}{K}\sum_{i=1}^{K} \widehat{f}_i^J \tag{3.67}$$

Jackknife bias reduction

To prepare for the assessment of bias, we again perform a Taylor expansion of the jackknife estimator around $\langle \boldsymbol{x}\rangle$. Trivially rewriting

$$\boldsymbol{x}_i^J = \langle \boldsymbol{x}\rangle + \sum_{k\neq i} \underbrace{\frac{\boldsymbol{x}_k - \langle \boldsymbol{x}\rangle}{K-1}}_{\equiv \boldsymbol{\delta}_k}, \tag{3.68}$$

and anticipating that all components $\delta_k^\mu = O(1/\sqrt{K})$ will be small, we expand

$$\widehat{f}_i^J = f\left(\langle \boldsymbol{x}\rangle + \sum_{k\neq i} \boldsymbol{\delta}_k\right) = f + \sum_{k\neq i}\sum_{\mu} f_\mu \delta_k^\mu + \frac{1}{2}\sum_{\mu\nu} f_{\mu\nu} \sum_{k,l\neq i} \delta_k^\mu \delta_l^\nu + \ldots \tag{3.69}$$

We will need the averages

$$\langle \delta_k^\mu \rangle = \left\langle \frac{x_k^\mu - \langle x^\mu\rangle}{K-1} \right\rangle = 0 \tag{3.70}$$

and, taking into account that $\boldsymbol{x}_k$ and $\boldsymbol{x}_l$ are only correlated for $k = l$

$$\begin{aligned} \langle \delta_k^\mu \delta_l^\nu \rangle &= \left\langle \frac{x_k^\mu - \langle x^\mu\rangle}{K-1} \frac{x_l^\nu - \langle x^\nu\rangle}{K-1} \right\rangle = \delta_{kl} \frac{\langle x_k^\mu x_k^\nu\rangle - \langle x^\mu\rangle\langle x^\nu\rangle}{(K-1)^2} \\ &= \delta_{kl} \frac{\mathrm{Cov}(x^\mu, x^\nu)}{(K-1)^2} \end{aligned} \tag{3.71}$$

Armed with these relations, we compute

$$\begin{aligned}
\langle \widehat{f}_i^J \rangle &= f + \sum_{\mu\nu} \frac{f_{\mu\nu}}{2} \sum_{k,l\neq i} \delta_{kl} \frac{\mathrm{Cov}(x^\mu, x^\nu)}{(K-1)^2} + \dots \\
&= f + \frac{1}{K-1} \sum_{\mu\nu} \frac{f_{\mu\nu}}{2} \mathrm{Cov}(x^\mu, x^\nu) + \dots \qquad (3.72)
\end{aligned}$$

such that also

$$\begin{aligned}
\langle \widehat{f}^J \rangle &= \left\langle \frac{1}{K} \sum_{i=1}^{K} \widehat{f}_i^J \right\rangle = \frac{1}{K} \sum_{i=1}^{K} \left\langle \widehat{f}_i^J \right\rangle \\
&= f + \frac{1}{K-1} \sum_{\mu\nu} \frac{f_{\mu\nu}}{2} \mathrm{Cov}(x^\mu, x^\nu) + \dots \qquad (3.73)
\end{aligned}$$

The bias of the jackknife estimator $\widehat{f}^J$ is therefore

$$\langle \widehat{f}^J \rangle - f = \frac{1}{K-1} \sum_{\mu\nu} \frac{f_{\mu\nu}}{2} \mathrm{Cov}(x^\mu, x^\nu) + \dots + \dots \qquad (3.74)$$

The leading contribution in powers of $1/K$ of this result is completely analogous to that for the bias of $f(\overline{\boldsymbol{x}})$ derived in Eqn. (3.45), except that the factor $1/K$ has been replaced by $1/(K-1)$. It follows that

$$\left\langle K\widehat{f} - (K-1)\widehat{f}^J \right\rangle - f = K\left(\langle \widehat{f} \rangle - f\right) - (K-1)\left(\langle \widehat{f}^J \rangle - f\right) = O(1/K^2)(3.75)$$

from which we conclude that

$$K\widehat{f} - (K-1)\widehat{f}^J \qquad \text{is an estimator for } f \text{ with a bias of } O(1/K^2) \qquad (3.76)$$

Jackknife error estimation

To obtain a jackknife estimator of the error in f, we now perform a Taylor expansion around $\overline{\boldsymbol{x}}$. We write

$$\boldsymbol{x}_i^J = \overline{\boldsymbol{x}} + \sum_{k\neq i} \underbrace{\frac{\boldsymbol{x}_k - \overline{\boldsymbol{x}}}{K-1}}_{\equiv \widetilde{\boldsymbol{\delta}}_k} \qquad (3.77)$$

and expand

$$\widehat{f}_i^J = f\left(\overline{\boldsymbol{x}} + \sum_{k\neq i} \widetilde{\boldsymbol{\delta}}_k\right) = \widetilde{f} + \sum_{k\neq i} \sum_{\mu} \widetilde{f}_\mu \widetilde{\delta}_k^\mu + \sum_{k,l\neq i} \sum_{\mu\nu} \frac{\widetilde{f}_{\mu\nu}}{2} \widetilde{\delta}_k^\mu \widetilde{\delta}_l^\nu + \dots \qquad (3.78)$$

where $\widetilde{f} \equiv f(\overline{\boldsymbol{x}})$, $\widetilde{f}_\mu \equiv \frac{\partial f(\boldsymbol{x})}{\partial x^\mu}\Big|_{\boldsymbol{x}=\overline{\boldsymbol{x}}}$ and so on. We will need linear averages of type

$$\begin{aligned}
\frac{1}{K}\sum_{i=1}^{K}\sum_{k\neq i}\widetilde{\delta}_k^\mu &= \frac{1}{K(K-1)}\sum_{i=1}^{K}\sum_{k\neq i}\left(x_k^\mu-\overline{x^\mu}\right) = \frac{1}{K(K-1)}\sum_{i=1}^{K}\sum_{k\neq i}x_k^\mu-\overline{x^\mu} \\
&= \frac{1}{K(K-1)}\sum_{i=1}^{K}\left[\sum_{k=1}^{K}x_k^\mu-x_i^\mu\right]-\overline{x^\mu} = \left(\frac{K}{K-1}-\frac{1}{K-1}-1\right)\overline{x^\mu} \\
&= 0
\end{aligned} \tag{3.79}$$

and quadratic ones of type

$$\begin{aligned}
\frac{1}{K}\sum_{i=1}^{K}\sum_{k,l\neq i}\widetilde{\delta}_k^\mu\widetilde{\delta}_l^\nu &= \frac{1}{K(K-1)^2}\sum_{i=1}^{K}\sum_{k,l\neq i}\left(x_k^\mu-\overline{x^\mu}\right)\left(x_l^\nu-\overline{x^\nu}\right) \\
&= \frac{1}{K(K-1)^2}\sum_{i,k,l=1}^{K}(1-\delta_{ki})(1-\delta_{li})\left(x_k^\mu-\overline{x^\mu}\right)\left(x_l^\nu-\overline{x^\nu}\right) \\
&= \frac{1}{K(K-1)^2}\sum_{i,k,l=1}^{K}\left(1-\delta_{ki}-\delta_{li}+\delta_{ki}\delta_{li}\right)\left(x_k^\mu-\overline{x^\mu}\right)\left(x_l^\nu-\overline{x^\nu}\right) \\
&= \frac{1}{(K-1)^2}\underbrace{\sum_{k,l=1}^{K}\left(x_k^\mu-\overline{x^\mu}\right)\left(x_l^\nu-\overline{x^\nu}\right)}_{=0} \\
&\quad -\frac{2}{K(K-1)^2}\underbrace{\sum_{i,k=1}^{K}\left(x_k^\mu-\overline{x^\mu}\right)\left(x_i^\nu-\overline{x^\nu}\right)}_{=0} \\
&\quad +\frac{1}{(K-1)^2}\frac{1}{K}\sum_{i=1}^{K}\left(x_i^\mu-\overline{x^\mu}\right)\left(x_i^\nu-\overline{x^\nu}\right)
\end{aligned} \tag{3.80}$$

i.e.

$$\frac{1}{K}\sum_{i=1}^{K}\sum_{k,l\neq i}\widetilde{\delta}_k^\mu\widetilde{\delta}_l^\nu = \frac{\widetilde{\mathrm{Cov}}(x^\mu,x^\nu)}{(K-1)^2} \tag{3.81}$$

Armed with these relations, we can now compute

$$\begin{aligned}
\frac{1}{K}\sum_{i=1}^{K}\widehat{f}_i^J &= \frac{1}{K}\sum_{i=1}^{K}\left[\widetilde{f}+\sum_{k\neq i}\sum_{\mu}\widetilde{f}_\mu\widetilde{\delta}_k^\mu+\sum_{k,l\neq i}\sum_{\mu\nu}\frac{\widetilde{f}_{\mu\nu}}{2}\widetilde{\delta}_k^\mu\widetilde{\delta}_l^\nu+\dots\right] \\
&= \widetilde{f}+\sum_{\mu\nu}\frac{\widetilde{f}_{\mu\nu}}{2}\frac{\widetilde{\mathrm{Cov}}(x^\mu,x^\nu)}{(K-1)^2}+\dots
\end{aligned} \tag{3.82}$$

and

$$\begin{aligned}
\frac{1}{K}\sum_{i=1}^{K}(\widehat{f}_i^J)^2 &= \frac{1}{K}\sum_{i=1}^{K}\left[\widetilde{f}+\sum_{k\neq i}\sum_{\mu}\widetilde{f}_\mu\widetilde{\delta}_k^\mu+\sum_{k,l\neq i}\sum_{\mu\nu}\frac{\widetilde{f}_{\mu\nu}}{2}\widetilde{\delta}_k^\mu\widetilde{\delta}_l^\nu+\dots\right]^2 \\
&= \frac{1}{K}\sum_{i=1}^{K}\left[\widetilde{f}^2+\sum_{k,l\neq i}\sum_{\mu\nu}\left(\widetilde{f}_\mu\widetilde{f}_\nu+\widetilde{f}\widetilde{f}_{\mu\nu}\right)\widetilde{\delta}_k^\mu\widetilde{\delta}_l^\nu+\dots\right] \\
&= \widetilde{f}^2+\sum_{\mu\nu}\left(\widetilde{f}_\mu\widetilde{f}_\nu+\widetilde{f}\widetilde{f}_{\mu\nu}\right)\frac{\widetilde{\mathrm{Cov}}(x^\mu,x^\nu)}{(K-1)^2}+\dots
\end{aligned} \tag{3.83}$$

such that altogether

$$\frac{1}{K}\sum_{i=1}^{K}(\widehat{f}_i^J)^2 - \left(\frac{1}{K}\sum_{i=1}^{K}\widehat{f}_i^J\right)^2 = \sum_{\mu\nu}\left(\widetilde{f}_\mu\widetilde{f}_\nu + \widehat{f}\widetilde{f}_{\mu\nu}\right)\frac{\widetilde{\mathrm{Cov}}(x^\mu, x^\nu)}{(K-1)^2} + \dots \quad (3.84)$$

Comparing this expression with (3.60), we see that

$$\frac{1}{K}\sum_{i=1}^{K}(\widehat{f}_i^J)^2 - \left(\frac{1}{K}\sum_{i=1}^{K}\widehat{f}_i^J\right)^2 = \frac{(\sigma_f^G)^2}{K-1} + O(1/(K-1)^2) \quad (3.85)$$

The *jackknife error bar* estimator $\widehat{\sigma}_f^J$ for σ_f is therefore given by

$$\widehat{\sigma}_f^J = \sqrt{(K-1)\left[\overline{(f^J)^2} - (\overline{f^J})^2\right]} \quad (3.86)$$

where, of course, $\overline{(f^J)^2} = \frac{1}{K}\sum_{i=1}^{K}(f_i^J)^2$. This is a major improvement over (3.60), since it only uses the data points without making any explicite reference to f and its derivatives or to the cross-correlations between different components of $\boldsymbol{x}$.

- Notice that in Eq. (3.86) we *multiply* (instead of divide) by $K-1$. This is totally reasonable: since the jackknife averages f_i^J are calculated using all K but one datapoint, their variance will certainly be extremely small.

- For large K, roundoff errors may severyly hamper the numerical precision of evaluating of (3.76) and (3.86). A simple remedy to overcome such a numerical instability is to use a larger block size $M^{(B)}$ for block-avaraging the raw data.

3.5.3 Bootstrap

The jackknife works very well for the task of estimating bias-reduced averages and error bars for simple quantities like combinations of moments of observables measured during a simulation. Sometimes, however, we also need to deal with derived quantities, whose computation involves much more complex procedures than just forming such simple functions of the elementary measurements. For instance, we may need e.g. to calculate error bars of least-squares fits or parameters of a minimization. Also, some observables may be subject to underlying probability distributions which are heavily skewed, such that a simpler error bar is not adequate to describe deviations from the mean and should be replaced by a confidence interval. In such situations, a so-called *bootstrap resampling* strategy, whose name was coined after the famous “Baron Münchhausen”, is a valuable alternative to the jackknife which is more general and yet conceptually somewhat simpler. On the other hand, it usually requires much greater computational resources than the jackknife does.
The bootstrap consists of the following steps:

1. From the K original blocked data $D \equiv \{x_1, \ldots, x_K\}$, we form N_B new data set $D_\alpha = \{x^B_{1,\alpha}, \ldots, x^B_{K,\alpha}\}$, $\alpha = 1, \ldots, N_B$ by selecting at random K data points, *irrespective of whether we have already selected it before (in statistics, this strategy is called "with replacement").* To condense our notation, let us denote by $n_{k,\alpha}$ the number of times x_k occurs in D_α ("occupation number"). In this way, we are trying to "resample" the unknown underlying statistical distribution in a Monte Carlo- like fashion, using our finite dataset D as a device to generate an approximation to this distribution. Below we will therefore use the notation $[\ldots]_{MC}$ to denote "exact" averages

$$[\ldots]_{MC} := \lim_{N_B \to \infty} \frac{1}{N_B} \sum_{\alpha=1}^{N_B} (\ldots) \tag{3.87}$$

over infinitely many bootstrap samples for fixed given dataset D. Since the probability to select one particular data point x_k is $p = 1/K$, the probability $P(n_k)$ that a data point x_k occurs n_k times in a bootstrap sample follows the binomial distribution

$$P(n_k) = \frac{K!}{n_k!(K-n_k)!} p^{n_k} (1-p)^{K-n_k} \tag{3.88}$$

with mean and standard deviation

$$\begin{aligned} E[n_k] &= Kp = 1 \\ \mathrm{Var}[n_k] &= E[n_k^2] - E^2[n_k] = Kp(1-p) = 1 - \frac{1}{K} \end{aligned} \tag{3.89}$$

- In particular, in the limit $K \to \infty$ the probability of a data point x_k to be completely absent from a particular bootstrap sample is

$$P(n_k = 0) = (1-p)^K = \left(1 - \frac{1}{K}\right)^K \overset{(K\to\infty)}{\longrightarrow} e^{-1} = 0.367879\ldots \tag{3.90}$$

  which is quite appreciable.

2. Averaging over bootstrap samples, (3.89) yields the "diagonal variance"

$$[n_k]_{MC} = 1, \qquad [n_k^2]_{MC} - [n_k]^2_{MC} = 1 - \frac{1}{K} \tag{3.91}$$

3. We will also need "covariances" $[n_k n_l]_{MC} - [n_k]_{MC}[n_l]_{MC}$. For $k \neq l$, the occupation numbers $n_{k\alpha}$ and $n_{l\alpha}$ are, however, not independent. Since our bootstrap datasets D_α are constructed to hold K datapoints each, they have to satisfy the constraints

$$\sum_{k=1}^{K} n_{k,\alpha} \equiv K, \qquad \alpha = 1, \ldots, N_B \tag{3.92}$$

from which we obtain

$$[n_k n_l]_{MC} - [n_k]_{MC}[n_l]_{MC} = -\frac{1}{K}, \qquad k \neq l \tag{3.93}$$

- From the squared constraint

$$K^2 \equiv \sum_{k,l=1}^{K} n_{k,\alpha} n_{l,\alpha} = \sum_{k=1}^{K} n_{k,\alpha}^2 + \sum_{k\neq l} n_{k,\alpha} n_{l,\alpha} \tag{3.94}$$

  we get

$$K^2 \equiv \sum_{k=1}^{K} [n_k^2]_{MC} + \sum_{k\neq l} [n_k n_l]_{MC} = \underbrace{\sum_{k=1}^{K} \left(1 - \frac{1}{K} + 1\right)}_{K(2-1/K)=2K-1} + \sum_{k\neq l} [n_k n_l]_{MC} \tag{3.95}$$

  i.e.

$$K^2 - 2K + 1 = (K-1)^2 \equiv \sum_{k\neq l} [n_k n_l]_{MC} = K(K-1)[n_k n_l]_{MC} \tag{3.96}$$

  and thus

$$[n_k n_l]_{MC} - [n_k]_{MC}[n_l]_{MC} = (K-1)/K - 1 = \cancel{1} - 1/K - \cancel{1} \tag{3.97}$$

  □

- For $K \gg 1$, the binomial distribution is well approximated by the Poisson distribution, for which indeed $[n_k n_l]_{MC} - [n_k]_{MC}[n_l]_{MC} \to \delta_{kl}$.

From (3.91) and (3.93) we immediately see that

$$[n_k n_l]_{MC} = \delta_{kl} + 1 - \frac{1}{K} \tag{3.98}$$

4. For calculating the simple average $\langle x \rangle$, the standard method is, of course, sufficient. However, estimating $\langle x \rangle$ from bootstrap will illustrate how to obtain bias reduction and errorbars for this simple case. Generalization to more complicated nonlinear functions will then be straightforward.

   We now assume that N_B is large enough to allow the approximation

$$[\ldots]_{MC} \approx \overline{(\ldots)} \equiv \frac{1}{N_B} \sum_{\alpha=1}^{N_B} (\ldots) \tag{3.99}$$

   as an average over a finite number N_B of bootstrap samples. Let

$$x_\alpha^B \equiv \frac{1}{K} \sum_{k=1}^{K} n_{k,\alpha} x_k, \qquad \alpha = 1, \ldots, N_B \tag{3.100}$$

   denote the average of x over the bootstrap dataset D_α. We calculate

$$\begin{aligned} \overline{x^B} \quad &= \quad \frac{1}{N_B} \sum_{\alpha=1}^{N_B} x_\alpha^B = \frac{1}{K} \sum_{k=1}^{K} \overline{n_{k,\alpha}}\, x_k \overset{(3.99)}{\approx} \frac{1}{K} \sum_{k=1}^{K} [n_{k,\alpha}]_{MC} x_k \\ &\overset{(3.91)}{=} \quad \frac{1}{K} \sum_{k=1}^{K} x_k \end{aligned} \tag{3.101}$$

i.e.

$$\lim_{N_B\to\infty} \overline{x^B} = \overline{x} \tag{3.102}$$

from which it follows that – since this is tue for $\overline{x}$ – *for sufficiently large N_B the bootstrap average $\overline{x^B}$ is an unbiased estimator of the exact average $\langle x\rangle$*. Moreover

$$\begin{aligned}\overline{(x^B)^2} &= \frac{1}{K^2}\sum_{k,l=1}^{K} \overline{n_{k,\alpha}n_{l,\alpha}}x_k x_l \overset{(3.99)}{\approx} \frac{1}{K^2}\sum_{k,l=1}^{K}[n_{k,\alpha}n_{l,\alpha}]_{MC}x_k x_l \\ &\overset{(3.98)}{=} \frac{1}{K^2}\sum_{k,l=1}^{K}\left[\delta_{kl}+1-\frac{1}{K}\right]x_k x_l = \frac{\overline{x^2}}{K} + \left(1-\frac{1}{K}\right)\overline{x}^2 \end{aligned} \tag{3.103}$$

from which it follows immediately that

$$\overline{(x^B)^2} - (\overline{x^B})^2 = \frac{1}{K}\left(\overline{x^2}-\overline{x}^2\right) = \frac{\widetilde{\sigma}_x^2}{K} = \frac{K-1}{K^2}\widehat{\sigma}_x^2 = \frac{K-1}{K}\widehat{\sigma}_{\overline{x}}^2 \tag{3.104}$$

Since $\widehat{\sigma}_{\overline{x}}^2$ is an estimator of the error in the sample mean, we conclude that the corresponding *bootstrap estimator of the error in the sample mean* is

$$\widehat{\sigma}_{\overline{x}} = \sqrt{\frac{K}{K-1}\left[\overline{(x^B)^2} - (\overline{x^B})^2\right]} \tag{3.105}$$

- For large K, the factor $\sqrt{\frac{K}{K-1}}$ may be safely replaced by 1. Then the bootstrap error bar is just the standard deviation of the bootstrap averages over the bootstrap data sets.

The above strategy can be generalized to reducing bias and calculating error bars for a (possibly nonlinear) function $f(\langle \boldsymbol{x}\rangle)$:

1. As above, we generate N_B bootstrap samples D_α, $\alpha = 1, \dots N_B$.

2. From the datasets D_α we compute N_B bootstrap average vectors $\boldsymbol{x}_\alpha^B$, whose components are given by (3.100).

3. The bootstrap average of f is defined as

$$\overline{f^B} \equiv \frac{1}{N_B} \sum_{\alpha=1}^{N^B} \tilde{f}_\alpha^B, \qquad \text{where} \quad \tilde{f}_\alpha^B \equiv f(\boldsymbol{x}_\alpha^B) \tag{3.106}$$

4. A bias-corrected estimator $\widehat{f}^B$ for f with an estimated bias of order $1/K^2$ is

$$\widehat{f}^B = 2f(\overline{\boldsymbol{x}}) - \overline{f^B} \tag{3.107}$$

5. The bootstrap estimator $\widehat{\sigma}_f^B$ of the error bar σ_f of f is the square root of the bias-corrected variance

$$\widehat{\sigma}_f^B = \sqrt{\frac{1}{N_B - 1} \sum_{\alpha=1}^{N^B} \left(\tilde{f}_\alpha^B - \overline{f^B} \right)^2} \tag{3.108}$$

- In the literature, one finds the advice to choose at least $N_B > 100$, with values $N_B \sim 500$ recommended.
- For quite asymmetric probability distributions, the bootstrap method even allows to calculate asymmetric confidence intervals.