

---

# Monte Carlo simulation in statistical physics

---

Mats Wallin

Theoretical Physics, KTH, SE-100 44 Stockholm, Sweden

September 23, 2005

# Preface

These lecture notes give a simple introduction to some main ideas and techniques used in Monte Carlo simulations. Both the principles and practicalities are discussed. The aim is to use simulations as a quick and effective tool to obtain nontrivial information in difficult problems, where solution with other methods may be difficult or impossible.

The focus is on applications in statistical physics. Examples such as spin models in classical statistical physics, and harmonic oscillators in quantum mechanics are explained in some detail. Various advanced techniques like finite size scaling and collective updating algorithms are considered, which can vastly improve performance of the simulation. Some orientation on open problems will be given.

These lecture notes are for a fourth year undergraduate course on computational physics. Assumed prerequisites are basic statistical mechanics and quantum mechanics. Elementary programming skills are assumed, preferably in a compiling language like FORTRAN or C, since interpreting languages are often too slow for simulations.

Simulation is best learned by simulation. A set of study projects accompany this text, and problems are included throughout the text. For more details on the subject some of the excellent textbooks in the reference list should be consulted. Please report suggestions, comments, and errors in this note to Mats Wallin, email: wallin@kth.se.

Enjoy!

Mats Wallin

# Chapter 1

## Introduction

*Picasso: "Computers are useless. They can only give you answers."*

In studies of complex phenomena we are interested in understanding and being able to predict the properties of interacting many body systems. When nature couples a large number of microscopic degrees of freedom, whose individual properties are often simple to understand, new emergent properties will appear that arise solely due to the various interactions between the coupled subsystems. It is these emergent properties that account for everything of interest in the universe, for example, formation of matter, electronic properties like superconductivity, and the possibility of life, and thus of you reading this. To study these systems is difficult and challenging, and requires powerful tools.

We need to start from some microscopic model. There are some standard models in statistical and quantum physics that are prototypes for understanding complicated collective behavior. Such models will be considered as examples here. The task is to use simulation to calculate useful information from such models. Generalizations to more complicated models are in principle often straightforward.

What kind of properties do we want to study? It is often enough to be able to understand the average properties of a complicated system, given by for example the thermal averages of statistical mechanics. How do we get such macroscopic properties out of a microscopic model? This is easy in principle but difficult in practice. The machinery of statistical physics tells us that this is done by just integrating over the microscopic degrees of freedom. The thermal average of some quantity  $A$  is

$$\langle A \rangle = \frac{1}{Z} \text{Tr} A e^{-H/T} \quad (1.1)$$

where  $Z = \text{Tr} \exp(-H/T)$  is the partition function.  $\text{Tr}$  means summation over all possible configurations or states in the case of discrete degrees of freedom, or integration in the case of continuous degrees of freedom.  $H$  is the Hamiltonian defining the system.  $T$  is the temperature (in units such that  $k_B = 1$ ). In classical systems the kinetic energy, which is typically a quadratic polynomial in the velocities, can be integrated out, and only the configuration potential energy remains in the problem. In quantum systems the kinetic and potential energies do not commute, and both have to be considered.

Exact analytical solution of interacting many body problems are exceptional. An exact solution means to calculate the trace over microscopic degrees of freedom in Eq. (1.1) analytically. There are some prominent cases where this is possible. Examples are the exact solution of the two dimensional Ising model by Onsager, and Bethe-ansatz solutions to one dimensional quantum problems. But these successful cases are few and they do often not seem possible to generalize much. Furthermore, these exact solutions are often very complicated and difficult to learn and understand. It is therefore important to consider other possibilities.

Approximate methods like perturbation theory and mean field theory are often very powerful and give valuable estimates and insights. However, it is not always so that such approximate

approaches can give all the required information, or that it is accurate enough, or in fact any accurate at all. A dramatic case where such methods break down is if the system undergoes a phase transition. Very complicated fluctuation calculations are required to analyze such behavior. The modern machinery using the renormalization group is designed to handle such cases, but usually does not give complete information, and is often very complicated to use. It is necessary to understand the background outlined above to fully appreciate the ease and speed at which a Monte Carlo simulation just solves a model for its average properties, also in the difficult case when the system has a phase transition. This is what will be discussed here. The solution of a model thus obtained can be useful in many ways, for example to check if a given model can actually explain an experiment. We will introduce Monte Carlo simulation by simple examples. However, we wish to stress its general character and that Monte Carlo methods are well suited for almost any computation involving complex behavior.

## Chapter 2

# Simple Monte Carlo methods

In this chapter the Monte Carlo method will be introduced. The Metropolis algorithm for applications in statistical physics is discussed. Practical considerations required to get it to work are presented. In the end various advanced methods like collective updating and finite size scaling are discussed.

### 2.0.1 Pi from random numbers: hit-and-miss method

As indicated by its name, Monte Carlo simulation involves calculation using random numbers. How can one calculate anything at all using random numbers? The following example illustrate the basic idea.

**Example** Let us calculate an approximation for  $\pi$  using simulation. The first thing to think about is how you would estimate  $\pi$  without using any other results. No formulas, no calculator. Can you do it?

**Solution:** Draw a circle of radius  $R$  inscribed in a square of side length  $2R$ . Throw many stones at random. Count the ratio of the number of hits inside the circle to the total number of hits inside the square. The ratio of these numbers approximates  $\pi/4$ . We will come back to this example later and make it more complicated.

Similarly, any “volume” integral can be done. Just inscribe the graph of the function to be integrated in a box, generate random points in the box, and compute the ratio of points under the graph to the total number of points. This will converge to the ratio of the integral of the function to the volume of the box.

**Problems.** Write computer programs using the hit-and-miss method to calculate: (a)  $\pi$  as above. (b)  $\int_0^{2\pi} \sin^2 x \, dx$ . (c) The volume of a 10-dimensional unit sphere. Can you obtain the answer by other methods? (Answer:  $\pi^5/120 \approx 2.550$ )

These simple and beautiful ideas are at the heart of the Monte Carlo method, which we will now explore in great detail. The next issue is to introduce an alternative to the hit and miss method, that can be generalized in useful ways.

## 2.1 Monte Carlo using simple sampling

The average of a function over an interval is defined to be its integral over an interval divided by the length of the interval:

$$\langle f \rangle = \frac{1}{b-a} \int_a^b f(x) \, dx \quad (2.1)$$

Hence, if we know the integral we can easily compute the average, and vice versa.

To introduce an element of randomness into the nonrandom concept of integration, we will view the average as taken with respect to a given, uniform probability distribution, defined as  $P(x) = 1/(b-a)$  for  $a \leq x \leq b$ , and  $P(x) = 0$  for other values of  $x$ . The same average as above

can now be written

$$\langle f \rangle = \int_a^b f(x)P(x) dx \quad (2.2)$$

Now the MC calculation consists of estimating the average over a set of samples  $x_i$  of a random variable  $x$  with a uniform probability distribution  $P$  on  $[a, b]$ :

$$\langle f \rangle = \frac{1}{N} \sum f(x_i) + \Delta f \quad (2.3)$$

where  $\Delta f$  is a statistical error present for  $N < \infty$ . This is called **Monte Carlo by simple sampling**. We will later in this course consider more complicated sampling schemes obtained with nonuniformly distributed random variables.

A simple sampling MC estimate of  $\pi$  goes as follows. Restrict the calculation to the first quadrant. Sample  $x$  uniformly in  $[0, 1]$ . Estimate  $\pi$  from

$$\frac{\pi}{4} \approx \frac{1}{N} \sum \sqrt{1 - x_i^2} \quad (2.4)$$

The direct sampling is more effective than the hit and miss sampling described in the previous section, since for each generated  $x$ , we use our knowledge of the corresponding smartest choice for  $y$ . This leads to a smaller statistical error.

Direct sampling means to that each  $x_i$  is randomly generated independent of all other points  $x_j$ . This is the way stones were thrown to compute  $\pi$ . Monte Carlo by direct sampling will work in a variety of situations to produce efficient approximate evaluation of integrals, but, as we will see, in statistical physics it is quite useless. However, before we worry about that, we will first discuss several points that have to be considered to turn the Monte Carlo into a powerful practical tool.

## 2.2 Statistical error

To know if our Monte Carlo method is any good we must be able to estimate the statistical error. This can be done systematically during the MC calculation. The fact that the error is under control is one of the good reasons for the usefulness of the method.

Suppose that we have a sequence of independent variables  $A_i$  and estimate the average from

$$\langle A \rangle = \frac{1}{N} \sum_{i=1}^N A_i + \Delta A \quad (2.5)$$

An estimate of the error in each individual variable,  $A_i - \langle A \rangle$ , is given by the variance  $\sigma$ :

$$\sigma^2 = \langle (A - \langle A \rangle)^2 \rangle = \langle A^2 \rangle - \langle A \rangle^2 \quad (2.6)$$

What is the error in our estimate of the average? Let us estimate this from the variance formula as follows:

$$\begin{aligned} (\Delta A)^2 &= \left\langle \left( \frac{1}{N} \sum_{i=1}^N A_i - \langle A \rangle \right)^2 \right\rangle = \frac{1}{N^2} \sum_{i=1}^N \underbrace{\langle (A_i - \langle A \rangle)^2 \rangle}_{=\sigma^2} + \underbrace{\langle \text{cross terms} \rangle}_{=0} \\ &= \frac{N\sigma^2}{N^2} = \frac{\sigma^2}{N} \end{aligned} \quad (2.7)$$

The estimates of the average and the error can thus be calculated from:

$$\langle A \rangle \approx \frac{1}{N} \sum_{i=1}^N A_i \pm \frac{\sigma}{\sqrt{N}} \quad (2.8)$$

$$\sigma^2 \approx \frac{1}{N} \sum_{i=1}^N A_i^2 - \left( \frac{1}{N} \sum_{i=1}^N A_i \right)^2 \quad (2.9)$$

### Comments:

1. The assumption that  $\langle \text{cross terms} \rangle = 0$  is only true for uncorrelated variables. For correlated variables the error is bigger. This difficulty will occur later when we do MC in statistical physics. We will then have to redo the preceding analysis, and analyze the contributions from the cross terms.
2. The formula (2.9) for the variance  $\sigma$  should in principle be multiplied by  $N/(N-1)$  to give an unbiased estimate. The reason is that when  $\langle A \rangle$  has been estimated from  $N$  variables, only  $N-1$  independent variables remain to estimate  $\sigma$ . However for the large values of  $N$  where the formulas make sense, the difference between  $N$  and  $N-1$  is irrelevant.
3. These estimates are put on solid ground by the **central limit theorem**. This will be discussed later.

**Problem:** Monitor the error in the calculation of  $\pi$  in Sec. 2.0.1. Verify the formula  $\sigma/\sqrt{N}$  by repeating the calculation many times for each  $N = 10, 100, 1000, 10000$  etc.

## 2.3 Multi Dimensional Integration

A  $d$ -dimensional integral over a volume  $V$  is given by exactly the same formulas:

$$\langle f \rangle = \frac{1}{V} \int_V d^d x f(x) = \frac{1}{N} \sum_{i=1}^N f(x_i) \pm \frac{\sigma}{\sqrt{N}} \quad (2.10)$$

where  $x_i$  are random points uniformly distributed in  $V$ . The key point is that the error goes down as  $1/\sqrt{N}$  in any  $d$ .

Note that this is directly analogous to the random walk result:  $\langle R_N^2 \rangle = N \Rightarrow \Delta = \sqrt{\langle R_N^2 \rangle}/N = 1/\sqrt{N}$ , independent of the dimension  $d$  of the walk.

**Problem:** Derive  $\langle R_N^2 \rangle = Na^2$  for a  $d$ -dimensional random walk with step length  $a$ .

There is one point special to volume integrals worth noting. Suppose that the integration volume  $V$  is complicated, so that it is difficult to generate random points uniformly in  $V$ . The way to fix this is to inscribe  $V$  in a simple region  $W$  and define  $f = 0$  outside  $V$ . Then the integral over  $V$  is given by  $\int_V f = V \langle f \rangle$ , where  $\langle f \rangle$  is computed by direct sampling of  $W$ . The region  $W$  should be chosen as small as possible to get few  $f_i = 0$  terms, and thus keep  $\sigma$  small.

## 2.4 Importance sampling and reduction of variance

The following idea is of central importance. If the integrand  $f(x)$  is strongly varying, the variance  $\sigma = \sqrt{\langle f^2 \rangle - \langle f \rangle^2} = \sqrt{\langle (f - \langle f \rangle)^2 \rangle}$  is large, and the convergence of the MC estimate for  $\langle f \rangle$  is slow. In such cases one should try to change integration variable such that the variance  $\sigma$  is reduced. This works as follows.

Make use of the freedom to divide and multiply by *any* function  $P(x)$  as follows:

$$\int_a^b f(x) dx = \int_a^b \frac{f(x)}{P(x)} P(x) dx = \int_{y(a)}^{y(b)} \frac{f(x(y))}{P(x(y))} dy \quad (2.11)$$

where we have set  $dy = P(x)dx$ . We should think of  $P(x)$  as a probability density and  $dy = P(x)dx$  as a probability. We assume that  $y(x)$  can be inverted to give  $x(y)$  explicitly. If we now sample  $y$  uniformly in  $[y(a), y(b)]$ , then  $x$  will be distributed according to  $P(x)$ , which we will write as  $x \in P$ . Monte Carlo calculation by sampling of  $y$  gives

$$\langle f \rangle = \frac{1}{N} \sum_{x_i \in P, i=1, \dots, N} \frac{f(x_i)}{P(x_i)} \pm \frac{\sigma_P}{\sqrt{N}} \quad (2.12)$$

where the variance is estimated by

$$\sigma_P^2 = \left\langle \left( \frac{f}{P} - \left\langle \frac{f}{P} \right\rangle \right)^2 \right\rangle \quad (2.13)$$

The idea is to choose the function  $P$  such that the integrand  $f/P$  is as nearly constant as possible, which reduces the variance. The best choice seems to be  $P = f$ . Then the integrand becomes constant and the variance  $\sigma_P = 0$ . The simulation converges after one single term to the exact answer! This is however hypothetical, because if we can solve the DE  $P = f = dy/dx$  for the transformed variable  $y$ , then we can also immediately calculate the original integral:  $\int f = y(x=b) - y(x=a)$ , and no simulation is necessary. The technique to use a suitable probability distribution for  $x$  to reduce the variance is called importance sampling. In practice, select  $P(x)$  to be a simple function which is easy to sample, and is as similar as possible to  $f(x)$ .

**Problem.** Study the integral  $\int_{-\infty}^{\infty} \exp(-x^2) = \sqrt{\pi}$  using (a) direct sampling, and (b) importance sampling choosing  $P = \exp(-x)$ . Monitor the statistical error.

## 2.5 Central limit theorem

The theoretical basis for the MC method is given by the central limit theorem, that will be stated here without proof.

Let  $I = \int d^d x f(x) P(x)$ , where  $x$  denotes a vector in  $d$  dimensions and  $P$  is a probability distribution satisfying  $P(x) \geq 0$ ,  $\int d^d x P(x) = 1$ . Approximate  $I$  as an average over  $N$  independent samples  $x_i \in P$  by

$$X = \frac{1}{N} \sum_{i=1}^N f(x_i) \quad (2.14)$$

What is the probability distribution  $\tilde{P}$  for  $X$ , and how does it behave for large  $N$ ? Answer:

$$\tilde{P}(X) \rightarrow \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(X - \langle f \rangle_P)^2 / 2\sigma^2} \quad (2.15)$$

when  $N \rightarrow \infty$ , where

$$\sigma = \sqrt{\frac{\langle f^2 \rangle_P - \langle f \rangle_P^2}{N}} \quad (2.16)$$

This result gives the theoretical foundation for the Monte Carlo method. For any choice of  $P(x)$  and any  $d$ , the average obeys this normal distribution for large  $N$ .

**Problem:** One way to generate Gaussian distributed random numbers is to sum up uniform random numbers. Explain how you can approximate Gaussian random numbers by summing  $N$  random numbers in  $[0, 1]$ . Write a computer program and test how it works. How can you generate a Gaussian distribution with given mean  $x$  and standard deviation  $\sigma$ ? Other methods to generate Gaussian distributions (or any other distribution!) will be explained later in this text.

## 2.6 When is MC better than deterministic methods?

As an example we will study **Simpson's rule in one dimension**:

$$\int_a^b f(x) dx \approx \frac{h}{3} [f_0 + 4(f_1 + f_3 + \dots + f_{N-1}) + 2(f_2 + f_4 + \dots + f_{N-2}) + f_N] \quad (2.17)$$

where  $h = 1/N$  is the step length and  $N$  is assumed to be even. The leading error in an interval, say  $[-h, h]$  is

$$\int_{-h}^h x^4 dx = O(h^5) \quad (2.18)$$



because the integrand is locally approximated quadratically and odd terms vanish. The total error is therefore  $Nh^5 = O(1/N^4)$ . (For a rectangular approximation the error is  $O(1/N)$ , and for the trapetiz method,  $O(1/N^2)$ .)

Similarly, for **Simpson's rule in  $d$  dimension**, the error per cell is  $O(h^{d+4})$ . If the integrand is sampled at  $N$  points, then  $h \sim 1/N^{1/d}$ , and the total error becomes  $Nh^{d+4} = O(1/N^{4/d})$ . In general, if the error goes as  $N^{-a}$  in one dimension, then it goes as  $N^{-a/d}$  in  $d$  dimensions. This shows that when  $d$  is large the error falls off slowly with  $N$ .  $N$  must vary as  $(const)^d$  to keep a fixed error, which just explodes when  $d$  is large.

In contrast, MC errors vary like  $1/\sqrt{N}$  independent of  $d$ . We can roughly estimate the dimension where MC gets better by setting  $\sigma/\sqrt{N} = N^{-a/d} \Rightarrow d = 2a$ . MC is therefore always better in high enough  $d$ . As a rule of thumb, MC is usually better in  $d > 3$ , and practically always better when  $d > 10$ .



## Chapter 3

# Monte Carlo in statistical physics

*Two Great Rules of Life: "1) Never tell everything at once."*

*Please keep in mind that the methods presented here are completely general and work for any integral, and give a way to sample any probability distribution.*

We want to calculate thermal averages of the form

$$\langle A \rangle = \frac{1}{Z} \text{Tr} A e^{-H/T} = \sum_x A(x) P(x) \quad (3.1)$$

Here  $\text{Tr} = \sum_x$  denotes a sum over all states  $x$ , where  $x$  denotes a state, i.e. a multidimensional vector in phase space containing all microscopic coordinates of all particles (we will usually suppress vector notation for states). The Hamiltonian  $H(x)$  is the energy of the system in the state  $x$ .  $T$  is the temperature. The Boltzmann distribution  $P(x) = (1/Z) \exp(-H(x)/T)$  is the probability of being in state  $x$ ,  $Z = \exp(-F/T) = \text{Tr} \exp(-H/T)$  is the partition function, and  $F = -T \ln Z$  is the free energy. We will also use the notation  $P(x) = e^{-S(x)}$ , where  $S$  is the "action".

One may immediately think of two possible ways to simulate  $\langle A \rangle$ :

**Method 1.** Try to sum over all states numerically by a deterministic method. If  $\text{Tr}$  is the integral over particle positions, the number of mesh points in a deterministic integration method explodes with increasing particle number. This just doesn't work.

**Method 2.** Generate configurations at random using direct sampling. In most cases this will not work because:

1. The factor  $\exp(-H/T)$  varies exponentially so almost all generated configurations will give negligible contributions to averages.
2. The normalization constant  $Z$  is unknown, and has to be calculated in the same way, which is another source of errors.
3. In complicated cases, methods to directly generate configurations are sometimes not available. (This is the case for example, in many-fermion systems, where it would be of great interest to be able to directly sample configurations.)

The way out of these difficulties is to use importance sampling and generate configurations distributed according to the Boltzmann distribution. MC estimates of thermal averages have the form

$$\langle A \rangle = \frac{1}{N} \sum_{x_i \in P, i=1, \dots, N} A(x_i) \pm \frac{\sigma}{\sqrt{N}} \quad (3.2)$$

$$\sigma^2 = \frac{1}{N} \sum_i A(x_i)^2 - \left[ \frac{1}{N} \sum_i A(x_i) \right]^2 \quad (3.3)$$

Note that we can always use any distribution  $P'$  here, instead of the Boltzmann distribution  $P$ . The averages then become

$$\langle A \rangle = \frac{\frac{1}{N} \sum_{x_i \in P', i=1, \dots, N} \frac{A(x_i) e^{-H(x_i)/T}}{P'(x_i)}}{\frac{1}{N} \sum_{x_i \in P', i=1, \dots, N} \frac{e^{-H(x_i)/T}}{P'(x_i)}} \quad (3.4)$$

and similarly for the error. This is more complicated to use, since the normalization factor in denominator has to be calculated. However, sometimes the Boltzmann distribution can be unsuitable for sampling of states, and then this form becomes useful.

The remaining problem is to sample states that are distributed according to the Boltzmann distribution. This is solved by the method of Metropolis et al. But before we describe this, we will again consider throwing stones.

### 3.1 Stones, Markov process sampling, and pi

Let us get back to the calculation of  $\pi$  by throwing stones. That method corresponds to direct sampling, since each throw is independent of all other throws. We are going to use another sampling, called Markov process sampling, that goes as follows. Assume the square is big, say  $100 \times 100$  meters. Throw one stone at random. Then walk to the place where it landed, and from there throw the next stone at random. This should also converge to  $\pi/4$ .

But wait! What shall we do if a stone happens to miss the square? Keep throwing and wait until we enter the square again? Discard the missing throw and try again until a stone hits? Both these ways are wrong and give incorrect answers! The correct method is to add one more stone on top of the one we were throwing from, and then try again. This will increase the number of stones near the edges of the square, but it is the correct way.

**Problem.** Modify your direct sampling program for  $\pi$  to do Markov process sampling, and verify the above statements.

### 3.2 The Metropolis method

A Markov process is a sequence of states  $x_1, x_2, x_3, \dots$  such that the probability distribution for  $x_{i+1}$  is specified from the previous state  $x_i$  by a transition probability  $w(x_i \rightarrow x_{i+1})$ . It will be very useful to think of the index  $i$  as a time step  $t$ , so the Markov process will represent a dynamic process in phase space. The time evolution of the probability distribution is then given by the Master equation

$$P(x, t+1) - P(x, t) = \sum_{y \neq x} [P(y, t)w(y \rightarrow x) - P(x, t)w(x \rightarrow y)] \quad (3.5)$$

In words this equation means the following: The change during time  $t$  to  $t+1$  in the probability  $P(x, t)$  of being in state  $x$  is  $P(x, t+1) - P(x, t)$ . This change must be given by the sum of all the transitions into  $x$  from other states  $y$ , that happens with probability  $P(y, t)w(y \rightarrow x)$ , minus the sum of transitions out of  $x$  to all other states  $y$ , whose probability is  $P(x, t)w(x \rightarrow y)$ . The Master equation holds generally, but we are interested in constructing  $w$  such that the Master equation gives us the desired  $P$ . In other words, we want to construct the Markov process, i.e. choose  $w$ , such that after sufficiently many time steps the probability distribution for the states approaches the desired distribution  $P(x) = e^{-S(x)}$ .

The following two conditions turn out to be sufficient to make the Markov process approach the desired distribution  $P(x) = e^{-S(x)}$  after a big number of steps:

**1. Ergodicity:** Any state in the system is reachable by the transition probability in the Markov process in a finite number of steps.

**2. Detailed balance (or micro-reversibility):** The probability  $w(x \rightarrow y)$  to go from state  $x$  to  $y$  obeys

$$e^{-S(x)}w(x \rightarrow y) = e^{-S(y)}w(y \rightarrow x) \quad (3.6)$$

The proof is straightforward and has two steps:

**STEP 1:** Show that  $e^{-S(x)}$  is an equilibrium solution. This means that if  $P(x) = e^{-S(x)}$  gives the distribution for state  $x$ , then in the next step of the Markov process, the distribution is the same:

$$P(y) = \int dx e^{-S(x)}w(x \rightarrow y) = e^{-S(y)} \quad (3.7)$$

This follows from detailed balance and the normalization condition  $\int dx w(x \rightarrow y) = 1$ :

$$P(y) = \int dx e^{-S(x)}w(x \rightarrow y) = e^{-S(y)} \int dx w(y \rightarrow x) = e^{-S(y)} \quad (3.8)$$

This proves that the desired distribution  $P(x)$  is a stationary distribution. Note that this corresponds to setting all terms in the Master equation to zero! This is clearly a sufficient but not necessary condition for a stationary distribution.

**STEP 2:** Show that the Markov process approaches  $e^{-S(x)}$ . As a measure of the distance between two distributions  $P_A(x)$  and  $P_B(x)$  we use  $D = \int dx |P_A(x) - P_B(x)|$ . If the distribution at one step is  $M(x)$ , the deviation from equilibrium is

$$D_{\text{old}} = \int dx |M(x) - e^{-S(x)}| \quad (3.9)$$

The distribution at the next step,  $P(y) = \int dx M(x)w(x \rightarrow y)$ , has deviation from equilibrium:

$$D_{\text{new}} = \int dy \left| \int dx M(x)w(x \rightarrow y) - e^{-S(y)} \right| \quad (3.10)$$

$$= \int dy \left| \int dx [M(x) - e^{-S(x)}]w(x \rightarrow y) \right| \quad (3.11)$$

$$\leq \int dy \int dx |M(x) - e^{-S(x)}|w(x \rightarrow y) \quad (3.12)$$

$$= \int dx |M(x) - e^{-S(x)}| = D_{\text{old}} \quad (3.13)$$

where we used the normalization condition  $\int w = 1$  and detailed balance. This shows that  $D \rightarrow 0$  when  $N \rightarrow \infty$ . Strict equality above holds in equilibrium:  $M(x) = e^{-S(x)}$ , or if some states are not accessible to  $w(x \rightarrow y)$ , which is excluded by hypothesis.

This completes the derivation of the convergence of the Markov process towards the Boltzmann distribution. Note that there are no statements about how quickly the generated distribution approaches the desired equilibrium distribution.

The next task is to construct the transition probability  $w(x \rightarrow y)$ . We set  $w(x \rightarrow y) = t(x \rightarrow y)a(x \rightarrow y)$  where  $t(x \rightarrow y)$  is the trial probability, i.e. the probability to try to go from  $x$  to  $y$  in the Markov process, and  $a(x \rightarrow y)$  is the acceptance probability, i.e. the probability that the state  $y$  is accepted if it is attempted from state  $x$ . Detailed balance requires:

$$\frac{w(x \rightarrow y)}{w(y \rightarrow x)} = \frac{t(x \rightarrow y)}{t(y \rightarrow x)} \frac{a(x \rightarrow y)}{a(y \rightarrow x)} = \frac{e^{-S(y)}}{e^{-S(x)}} \quad (3.14)$$

Here there is apparently an infinite freedom in choosing  $t$  and  $a$  such that the desired distribution  $e^{-S(x)}$  is obtained. One usually (but not always) takes the trial probability to be symmetric:  $t(x \rightarrow y) = t(y \rightarrow x)$ . Common choices for the acceptance probability are:

1. The original Metropolis choice is

$$a(x \rightarrow y) = \min \left\{ 1, \frac{e^{-S(y)}}{e^{-S(x)}} \right\} \quad (3.15)$$

2. In practice the following acceptance probability is often to be preferred (since this is a smooth function)

$$a(x \rightarrow y) = \frac{1}{1 + \frac{e^{S(y)}}{e^{S(x)}}} \quad (3.16)$$

The following algorithm satisfies detailed balance. Generate a trial state  $x_T$  from  $x$  with probability  $t(x \rightarrow x_T)$ . This state is accepted with probability  $a(x \rightarrow x_T)$ . If the state is accepted, set the new state in the Markov process to  $y = x_T$ , otherwise set the new state equal to the previous state,  $y = x$ . (This corresponds to placing a second stone on top of the first if the square is missed.)

A main difficulty with Markov process sampling is that the successive states are correlated. This means that it can be highly nontrivial to decide how well the simulation converges and to estimate the error. This will be considered in later sections.

### 3.2.1 MC algorithm using the Metropolis method

1. Specify some initial state  $x_0$ .
2. Generate a trial state  $y$  from  $x_t$  from a trial probability distribution, and accept the trial state with some acceptance probability distribution based on detailed balance. If accepted, set the new state in the Markov process to  $x_{t+1} = y$ , otherwise set the new state equal to the old state:  $x_{t+1} = x_t$
3. Repeat from step 2 until enough states have been generated in equilibrium to form accurate thermal averages. Discard initial states to approach the equilibrium distribution.

### 3.2.2 MC algorithm for the Ising model

The Ising model is defined by the Hamiltonian

$$H = -J \sum_{\langle i,j \rangle} S_i S_j \quad (3.17)$$

where  $J$  is a coupling constant,  $S_i = \pm 1$  is a "spin" degree of freedom on site  $i$  of a simple cubic lattice in  $d$  dimensions, and  $\sum_{\langle i,j \rangle}$  denotes summation over all nearest-neighbor pair of sites. Thermal averages have the form

$$M = \frac{\text{Tr} \sum_i S_i e^{-H/T}}{\text{Tr} e^{-H/T}} \quad (3.18)$$

where  $M$  is the magnetization, i.e. the average spin, and  $\text{Tr}$  denotes summation over all states, i.e. sum over all possible combinations of values of spins, ( $S_1 = \pm 1, S_2 = \pm 1, \dots$ ).

Monte Carlo simulation is easily implemented by the following algorithm:

1. Start with all spins up:  $S_i = 1$  for all  $i$ .
2. Select a spin  $S_i$  at random. The trial move is to flip this spin  $S_i \rightarrow -S_i$ . Compute  $w$  by e.g.  $w = \exp(-\Delta H/T)$  where  $-\Delta H = J \Delta S_i \sum_j S_j$ , where  $j$  are the nearest neighbors to  $i$ . If  $w > r$ , where  $r \in [0, 1]$  is a random number, then accept the new state, otherwise, keep the old state. Advance MC "time" to  $t + \Delta t$ . If the trial move was rejected, let the old state be the new state at  $t + \Delta t$ .
3. Measure variables of interest on the generated states and add to averages at "suitable" MC time intervals, but do not start to form averages until a "suitable" number of initial states have been skipped to approach the equilibrium distribution. Repeat from step 2 until enough data has been obtained. (The meaning of "suitable" will be analyzed later.)

### 3.3 How to get it to work

- To simulate bulk properties in a large system with macroscopic number of degrees of freedom, whose linear dimension is  $L$ , one often uses *periodic boundary conditions*:  $x+L \rightarrow x$ . This eliminates surface effects.
- A number of MC steps where on average every spin is attempted to be updated once, is called a MC sweep through the system. The unit of time in a MC simulation is one MC sweep ( $\Delta t = 1$ ). More on the concept of MC time later.
- Selecting spins sequentially rather than at random perhaps seems more natural, but it is not. With random selection, each move is precisely equivalent, whereas with sequential updates they are not. A limit where sequential selection fails completely is for an Ising model at high enough temperature where all moves are accepted. Starting from a uniform initial state, a sequential sweep simply inverts the configuration.

The next 3 comments are related to relaxation times and will be discussed in more detail later:

- Discard  $t_{\text{eq}} \sim 10^3$  sweeps before data collection starts to allow the system to approach equilibrium. Otherwise the computed averages will be biased with systematic errors from the initial nonequilibrium states.
- Skip a certain number of sweeps ( $\sim 2$ ) between measuring data for the averages. The reason is that successive configurations are correlated, and do not give independent information. If the data collection is expensive (takes much computer time) more sweeps can be taken in between, if it is cheap data can be taken after every sweep. It makes no sense to skip more than one relaxation MC time, since this would mean throwing away uncorrelated data.
- Collect data during  $t_{\text{sample}} \sim 10^5$  sweeps. Sometimes less is enough, sometimes much more is necessary. An individual run should not take a very long time: we are not patient, the computer might crash, and the results may look bad and require some change in the program, etc. Thus one should not take a very large number of steps (see next point instead).
- Repeat the simulation  $N_{\text{runs}}$  times, using different random numbers. This ensures  $N_{\text{runs}}$  independent samples, that is, the averages from the independent runs. The error estimate is then

$$\Delta = \pm \frac{\sigma}{\sqrt{N_{\text{runs}}}} \quad (3.19)$$

where  $\sigma$  is the variance of the *averages*. Rerun until  $\Delta$  becomes small enough! Usually  $N_{\text{runs}} \sim 10 - 1000$ .

- Use lookup tables for time consuming calculations that are performed in the MC loop. For example, define a table  $w(\Delta H) = \exp(-\Delta H/T)$ ,  $\Delta H = -J\Delta s_i \sum_j s_j = 0, \pm 4J, \dots$ . For models with continuous spin (XY, Heisenberg), discretize the spin in e.g.  $\sim 1000$  steps, and the same method works.
- It is important to realize that also the rejected MC moves are counted as new configurations in the simulation. It is not correct to keep trying different moves until something is accepted; also the old configuration when a trial move is rejected contribute to the averages. Physically, it is to be expected that almost no moves are accepted at low temperatures, corresponding to few fluctuations out of the ground state. At high temperatures most moves should be accepted, corresponding to fast fluctuations. The number of accepted and rejected moves has important physical meaning!

- How are averages actually computed, for example, the average energy per spin? For the first  $t_{\text{eq}}$  sweeps no measurements are taken. For the following  $t_{\text{sample}}$  sweeps e.g. the energy is measured after, say, every 2 sweeps, and the current value of the energy  $E(t)$  is added to the average  $E_{\text{sum}} \rightarrow E_{\text{sum}} + E(t)$ . Remember to initialize:  $E_{\text{sum}} = 0$ . The current energy  $E(t)$  does not really have to be computed: it should be recorded by adding  $\Delta H$  to  $E$  every time a move is accepted. The desired expectation values including the final normalization become:

$$\text{energy density: } e = \frac{E_{\text{sum}}}{N_{\text{samples}} N_{\text{spins}}} \quad (3.20)$$

$$\text{specific heat: } c = \frac{E_{\text{sum}}^2 / N_{\text{samples}} - (E_{\text{sum}} / N_{\text{samples}})^2}{T^2 N_{\text{spins}}} \quad (3.21)$$

$$\text{magnetization: } m = \frac{M_{\text{sum}}}{N_{\text{samples}} N_{\text{spins}}} \quad (3.22)$$

$$\text{susceptibility: } \chi = \frac{M_{\text{sum}}^2 / N_{\text{samples}} - (M_{\text{sum}} / N_{\text{samples}})^2}{T N_{\text{spins}}} \quad (3.23)$$

**Problem.** Derive these formulas.

### 3.4 Random numbers

Can you write a program that gives something random as output? Think about it. It is not easy! Actually, this is an unsolved problem! This is bad news, because the MC program requires lots of random numbers, both to select which spin to update and to test for acceptance. However, there are several algorithms to compute approximate random numbers, giving a sequence  $r_1, r_2, r_3, \dots$  from some formula. Such numbers are not really random: if you know the first number and the formula, you know all consequent generated numbers! Such numbers are called *pseudo random numbers*.

How good are pseudo random numbers? To mimic "real" random numbers, the pseudo random numbers should be independent and perfectly uniformly distributed. This turns out to be very difficult to accomplish, and is an active area of research! Pseudo random numbers have two types of defects:

1. Correlations between  $x_i$  and  $x_{i+j}$ .
2. Finite period:  $x_i = x_{i+\text{period}}$ .

It turns out that MC is sensitive to the quality of random numbers. To converge to the correct equilibrium distribution requires unbiased random numbers, otherwise some biased distribution is generated, giving biased averages. This might be detected in high precision simulations. Collective updating routines (these are described in Sec. 3.9) are especially sensitive. In recent research the quality of random number generators have been tested by comparing MC data for the 2D Ising model with the exact solution for finite systems.

**WARNING:** Machine provided random number generators are often poor and should be avoided (compare this with having the  $\sin(x)$  function correct to only 3 decimal places, which would be equally unacceptable!).

If high precision results are needed, you are forced to use a very good random number generator. You normally have to provide your own random number generator. Take, for example, the best one from Numerical Recipes. This is a so called lagged Fibonacci generator: a new random numbers is generated as the decimal part of the sum of two old random numbers. This has very small correlations and practically infinite period.



## 3.5 Correlation functions

Usually in a simulation it is desirable to calculate correlation functions. Spatial and temporal correlations give important information about ordering, the approach to equilibrium, and relaxation of excitations.

### 3.5.1 Spatial correlations

The spin correlation function is defined by

$$G(r) = \langle S(r)S(0) \rangle \quad (3.24)$$

The correlation function measures the ordering properties of the system. In an ordered phase the correlation function approaches a constant at large distance:

$$G(r) = \langle S(r)S(0) \rangle \rightarrow m^2, \quad r \rightarrow \infty \quad (3.25)$$

where  $m = \langle S \rangle$ .

In the disordered phase, at temperatures above  $T_c$ , relaxation is usually approximately exponential:

$$G(r) \sim e^{-r/\xi} \quad (3.26)$$

which defines the **correlation length**  $\xi(T)$ . At the transition temperature  $T_c$  the correlation length diverges, and the correlation function typically becomes a power law:

$$G(r) \sim \frac{1}{r^{d-2-\eta}} \quad (3.27)$$

In the simulation, it may be enough to compute the correlation function in the  $x$  and  $y$  directions and average the results. The exponents etc. can be computed in suitable logarithmic plots by fitting a straight line to determine the slope.

The correlation function contains important information about the properties of the system, and these properties influence the convergence properties of the simulation. Away from a phase transition, the correlation length is small. In the Ising model, below  $T_c$ , the correlation of fluctuations out of the ordered state is small, and an effective averaging over such fluctuations is accomplished effectively in a limited number of sweeps. Similarly, above  $T_c$  spins are fluctuating and have short correlation length, so each sample effectively averages over many fluctuations. This is not true close to  $T = T_c$ , where the correlations grow big. Consequently each sample gives limited averaging over different fluctuations, and the convergence becomes slow. The specific heat, Eq. (3.21), measures fluctuations in the energy, and the order parameter susceptibility, Eq. (3.23), measures fluctuations in the order parameter. These can be directly measured in the simulation and typically diverge at  $T_c$ .

### 3.5.2 Temporal correlations

The **correlation time**  $\tau(T)$  is a very useful quantity to calculate in a MC simulation. It is defined via the autocorrelation function

$$C_{AA}(t) = \langle A(t)A(0) \rangle - \langle A \rangle^2 \quad (3.28)$$

The correlation time  $\tau$  is the time it takes for the autocorrelation function to drop to a fraction of its value  $C_{AA}(0)$  at  $t = 0$ . A convenient definition of  $\tau$  is given by the integral of the normalized autocorrelation function,

$$\tau = 1 + 2 \sum_{t=1}^{\infty} \frac{C_{AA}(t)}{C_{AA}(0)} \quad (3.29)$$

The sum can in practice be terminated at the first negative value:  $C_{AA}(t) \leq 0 \Rightarrow t_{\max} = t$ .

### 3.5.3 Statistical error of correlated samples

We can now estimate the expected statistical error in the simulation using correlated Markov chain sampling. The number of independent samples obtained in the simulation is approximately  $N/\tau$ , where  $N$  is the total number of samples and  $\tau$  is the relaxation time. The statistical error of the average can be estimated by

$$\sigma \approx \sqrt{\frac{C_{AA}(0)\tau}{N}} \quad (3.30)$$

(note that  $C_{AA}(0) = \langle A^2 \rangle - \langle A \rangle^2$  is the variance of each term in the average). This relation is instructive to derive:

$$\begin{aligned} \sigma^2 &= \left\langle \left( \frac{1}{N} \sum_{t=1}^N (A(t) - \langle A \rangle) \right)^2 \right\rangle \\ &= \frac{1}{N^2} \sum_{t=1}^N \underbrace{\langle (A(t) - \langle A \rangle)^2 \rangle}_{=C_{AA}(0)} + \frac{2}{N^2} \sum_{t'=1}^N \sum_{t=1+t'}^N \langle (A(t') - \langle A \rangle)(A(t) - \langle A \rangle) \rangle \end{aligned}$$

Now let us approximate by changing the upper limit of the last sum from  $N$  to  $N + t'$ , which is a good approximation if  $N \gg \tau$ . Using  $\tau$  from Eq. (3.29) in this expression leads to

$$\sigma^2 = \frac{C_{AA}(0)}{N} + \frac{2}{N^2} \sum_{t'=1}^N \sum_{t=1+t'}^{N+t'} \underbrace{\langle (A(t') - \langle A \rangle)(A(t) - \langle A \rangle) \rangle}_{C_{AA}(t-t')} = C_{AA}(0)\tau/N$$

### 3.5.4 Calculation of the equilibration time

Related to the relaxation time is the *equilibration time*  $t_{\text{eq}}$ , which is the time to approach equilibrium starting from an initial non-equilibrium state.  $t_{\text{eq}}$  can be obtained by the same procedure as described above for the relaxation time, where in this case  $t = 0$  denotes the starting time in the simulation. This is the way to calculate how many sweeps must be discarded initially in the simulation before data collection starts.

**WARNING!** Actually, it is often more important to ensure that the transient from the initial state has decayed, than it is to generate a huge number of independent configurations near equilibrium. Numerous incorrect papers have been published that suffer from this. It is useful to do a few very long runs to check that the approach to equilibrium has been achieved.

### 3.5.5 Calculation of the relaxation time

There are two important points to note when the correlation time is computed:

1. The term  $\langle A \rangle^2$  is a potential source of systematic error. Suppose that we get in the simulation  $\langle A \rangle = X \pm \Delta a$ . Squaring gives  $\langle A \rangle^2 = X^2 + 2X\Delta X + (\Delta X)^2$ . The last term is not uniformly distributed around zero, and gives a bias if it is evaluated in this way. A fix-up is to evaluate  $\langle f \rangle$  independently twice and set  $\langle A \rangle^2 = X_1 X_2 + X_1 \Delta X_2 + X_2 \Delta X_1 + \Delta X_1 \Delta X_2$ , which is unbiased. If this point is not noticed, the procedure of repeating the simulation many times and averaging over the results will have systematic errors.
2. The calculation of the sum of the autocorrelation function over time is best organized as follows. Rewrite the sum involved in calculating  $\sum_t \langle A(t)A(0) \rangle$  in Eq. (3.29) in the following way:

$$\sum_t \langle A(t)A(0) \rangle = \sum_{t=0}^{t_{\text{max}}} \frac{1}{N} \sum_{t'=1}^N A(t+t')A(t') \quad (3.31)$$

Change summation order and define  $F(t') = \sum_{t=0}^{t_{\max}} A(t+t')$ , which gives

$$\sum_t \langle A(t)A(0) \rangle = \frac{1}{N} \sum_{t'=1}^N F(t')A(t') \quad (3.32)$$

The advantage is that this contains no double time summation, but instead only involves updating the sum  $F(t') = \sum_{t=0}^{t_{\max}} A(t+t')$  at every time step, which requires storing all the terms  $A(t+t'), t=0, \dots, t_{\max}$ . For this calculation to be reliable,  $t_{\max} \gg \tau$  must be fulfilled.

### 3.6 Free energy methods

A MC simulation does not directly give information about the free energy  $F$  since we avoided to calculate the partition function  $Z = \exp(-F/T)$  which is the normalization factor in the expectation values, i.e. derivatives of  $F$ .

However, we can integrate the expectation values calculated by MC, and thereby obtain information about  $F$ . For example,

$$Z = e^{-F/T} = \text{Tr} e^{-H/T} \quad (3.33)$$

$$\frac{1}{Z} \frac{\partial Z}{\partial T} = \frac{\partial -F/T}{\partial T} = \frac{1}{Z} \text{Tr} \frac{H}{T^2} e^{-H/T} = \frac{E}{T^2} \quad (3.34)$$

$$\left. \frac{F}{T} \right|_{T_1}^{T_2} = - \int_{T_1}^{T_2} \frac{E}{T^2} dT \quad (3.35)$$

Hence we get information about the free energy difference. This allows us to determine also the entropy difference:

$$F = E - TS \Rightarrow \Delta S = \Delta \frac{E - F}{T} \quad (3.36)$$

### 3.7 Pedestrian account on finite size scaling

*Kilgore Trout: "The universe is a big place, perhaps the biggest."*

This introduction to finite size scaling demonstrates how to operate the finite size scaling ansatz to make it into a practical tool. Intimately tied to scaling properties is the concept of universality. Universality roughly means independence of details of the model. For example, adding a next nearest neighbor coupling to the Ising model shifts the critical temperature (which is non-universal) but preserves the critical exponents because they are universal. A deeper understanding of these concepts is given by renormalization group theory.

Simulations always have to be done on small finite systems, because these are the only ones the computer can converge in reasonable time. However, one is normally interested in the result of a big system, with a very large number of lattice points. Hence it is crucial to understand how to extrapolate from finite systems with  $N$  lattice points to the thermodynamic limit  $N \rightarrow \infty$ . This can be illustrated by studying the Ising phase transition.

Singular behavior at a phase transition is possible only in the limit of an infinite system. This is evident: The Hamiltonian is a polynomial in the  $s_i$ , which is a smooth function. The partition function is a sum of exponentials of that polynomial, which is therefore a smooth function of  $T$ . The only way to get something singular is to have an infinite polynomial in the exponent, and the exact solution verifies this.

All singular behavior at the transition is rounded off in finite systems. The approach to singular behavior is often slow in  $L$ . We are in practice limited to rather small systems:  $L = 4, 6, 8, 10, 12, \dots, \leq \sim 100$ . Extrapolation to the thermodynamic limit by finite size scaling can often successfully capture the critical behavior of the transition.

Assume we have a second order phase transition at  $T = T_c$  where the correlation length diverges like

$$\xi \sim |T - T_c|^{-\nu} \quad (3.37)$$

where  $\nu$  is the **correlation length exponent**. The relaxation time, i.e. typical time for a fluctuation to relax, also diverge at  $T_c$ :

$$\tau \sim \xi^z \quad (3.38)$$

where  $z$  is the **dynamical critical exponent**. The divergence of the relaxation time at the transition is called critical slowing down. The diverging length and time scales control the behavior of the averages close to the transition. This can be taken advantage of in the following way.

The order parameter in the Ising model is the magnetization, which vanishes at  $T_c$  like

$$m \sim |T - T_c|^\beta \sim \xi^{-\beta/\nu} \quad (3.39)$$

Other thermodynamic quantities also behave singularly, for example,

$$\chi \sim |T - T_c|^{-\gamma}, \quad c \sim |T - T_c|^{-\alpha} \quad (3.40)$$

In a finite system of size  $L$ , (a somewhat simplified argument says that) the diverging correlation length must be replaced by the largest length available in the finite system, which is the system size:

$$\xi \rightarrow \infty \quad \text{becomes} \quad \xi = L \quad \text{at} \quad T = T_c \quad (3.41)$$

Thus:

$$m \sim L^{-\beta/\nu} \quad \text{at} \quad T = T_c \quad (3.42)$$

which implies

$$L^{\beta/\nu} m = \text{constant} \quad \text{at} \quad T = T_c \quad (3.43)$$

From the exact solution of the 2D Ising model it is known that  $\beta = 1/8$  and  $\nu = 1$ . Hence a plot of data for  $L^{1/8}m(T, L)$  vs.  $T$  for different system sizes  $L$  is expected to give curves that become size independent and intersect at  $T = T_c$ , whose exact value in two dimensions is  $T_c = 2/\ln(1 + \sqrt{2}) \approx 2.269$ . This allows accurate determination of  $T_c$  if  $\beta/\nu$  happens to be known. Of course, it is exceptional that the exponents are known. In fact, it is often the exponents we want to compute! Now, take the logarithm of the ratio of Eq. (3.42) for two different lattice sizes,  $L$  and  $L'$ :

$$\frac{\ln(m(L)/m(L'))}{\ln(L/L')} = -\beta/\nu \quad \text{at} \quad T = T_c \quad (3.44)$$

A plot of this quantity for different sizes and temperatures gives curves that intersect at  $T_c$  and the y-value at the intersection is the estimate for  $-\beta/\nu$ . This is called the *intersection method* and determines  $T_c$  and the combination  $-\beta/\nu$ .

All three quantities  $T_c, \beta/\nu$  and  $\nu$  can be determined from the following finite-size scaling procedure. Make the ansatz that for any  $T$  (close enough to the transition) and  $L$  (big enough) we have

$$m(T, L) = L^{-\beta/\nu} f_{\pm}(L/\xi) = L^{-\beta/\nu} \tilde{f}(L^{1/\nu}(T - T_c)) \quad (3.45)$$

where  $f_{\pm}, \tilde{f}$  are universal scaling functions ( $\pm$  refers to the cases  $T > T_c$  and  $T < T_c$  respectively). This instructs us to plot  $L^{\beta/\nu}m(T, L)$  vs.  $L^{1/\nu}(T - T_c)$ . Now, adjust  $T_c, \beta/\nu$  and  $\nu$  (or alternatively, only  $\nu$  if the others are already known) until data for different sizes  $L$  and temperatures  $T$  collapse onto the common curve  $y = \tilde{f}(x)$ . This procedure determines  $T_c, \beta/\nu, \nu$  and also the scaling function  $\tilde{f}$ .

### 3.7.1 Binder cumulant

A variation on this theme is what is known as the Binder cumulant method. Here one calculates the quantity

$$R = \frac{\langle m^4 \rangle}{\langle m^2 \rangle^2} = \tilde{R}(L^{1/\nu}(T - T_c)) \quad (3.46)$$

so that the RHS is  $\propto L^0$ . The reason for the  $L^0$ -dependence is that both the denominator and numerator of the fraction must vanish with the same power of  $L$ , namely  $L^{-4\beta/\nu}$ . Here  $\tilde{R}$  is a universal scaling function. At  $T_c$ , we have  $R = \text{const}$  so data for different system sizes intersect. By adjusting  $\nu$  until data away from  $T_c$  collapse onto the common curve,  $y = \tilde{R}(x)$ , it is possible to determine  $\nu$  and  $\tilde{R}$ .

### 3.7.2 Discussion

We note that the universal scaling function is universal at  $T_c$ , where its argument vanishes. Thus a suitable power times the left hand side of the scaling relation which provides the dimensionless combination given by the scaling function, if it is finite at the critical point, is also universal. This universality property of certain critical amplitudes is called *two-scale-factor universality*. The finite size scaling analysis presented here is the secret in the art of computing macroscopic quantities in simulations. It often permits calculation of critical temperatures and exponents with an accuracy of better than 10%. Critical exponents are often determined in experiments from similar scaling analysis (but usually not involving the system size which is practically infinite in most experiments). This is certainly not the end of the story. The ideas touched upon in this section form parts of the basis for the modern foundations of the very meaning of physical models and approximations. This picture provides understanding of why physical quantities and phenomena are at all possible to study using highly simplified models.

## 3.8 Histograms

Histograms are very useful and can be instruments to decide the order and critical exponents of a phase transition. They can also allow us to extrapolate to temperatures close to the ones where the simulation was done. Both these techniques will now be described.

### 3.8.1 Extrapolation to nearby temperatures

Simulate the system at inverse temperature  $\beta_0$ , which is near the values  $\beta$  of interest. Split the accessible energy range in a fine grid of equally spaced intervals. Store the energy of the system after every sweep by incrementing a counter that measures the number states  $N_0(E)$  obtained in an interval centered around  $E$ . The measured probability that the system has energy  $E$  is then

$$P(E, \beta_0) = \frac{N_0(E)}{\sum_E N_0(E)} \quad (3.47)$$

The histogram  $N_0(E)$  is the number of states with energy  $E$ , and the denominator is the number of MC sweeps. According to the Boltzmann distribution we have

$$P(E, \beta) = \frac{W(E)e^{-\beta E}}{\sum_E W(E)e^{-\beta E}} \quad (3.48)$$

$W(E)$  is the number of states with energy  $E$ , generally called the density of states. Comparing (3.47) and (3.48), we have

$$W(E) = a_0 N_0(E) e^{-\beta_0 E} \quad (3.49)$$

since  $W(E)$  is independent of  $\beta$ , where  $a_0$  is a proportionality constant that depends on  $\beta_0$ . Eliminating  $W(E)$  gives

$$P(E, \beta) = \frac{N_0(E)e^{-(\beta-\beta_0)E}}{\sum_E N_0(E)e^{-(\beta-\beta_0)E}} \quad (3.50)$$

This accomplishes the desired task: We have expressed the probability at  $\beta$  in terms of  $N_0(E)$ , the energy histogram at  $\beta_0$ .

We can now estimate the  $\beta$  dependence of the mean value of any function that depends on  $E$ :

$$\langle A(\beta) \rangle = \sum_E A(E)P(E, \beta) \quad (3.51)$$

If the quantity also depends on another quantity  $M$ , e.g. the magnetization, we instead form

$$\langle A(\beta) \rangle = \sum_{E, M} A(E, M)P(E, M, \beta) \quad (3.52)$$

where

$$P(E, M, \beta) = \frac{N_0(E, M)e^{-(\beta-\beta_0)E}}{\sum_{E, M} N_0(E, M)e^{-(\beta-\beta_0)E}} \quad (3.53)$$

Hence we can calculate the temperature dependence of  $\langle A \rangle$  in a single run. The drawbacks of this method are:

1. The temperature difference  $|T - T_0|$  must be small. Otherwise states important at  $T$  may be unimportant at  $T_0$  and thus have poor statistics.
2. This method does not easily allow for error estimates.

Generally this method works well for small systems, but the allowed temperature range shrinks quickly as the system size is increased.

### 3.8.2 The Lee-Kosterlitz method

The following method was invented recently by Lee and Kosterlitz to determine the nature of a phase transition. The energy histogram  $N(E)$  described in the previous section can help us to define a restricted free energy that only includes configurations at a particular energy  $E$ :

$$F(E) = -T \ln \left( W(E)e^{-\beta E} \right) \quad (3.54)$$

For a system with a first-order transition, a plot of  $F(E)$  vs.  $E$  will show two local minima, corresponding to configurations that are characteristic of the high and low temperature phases. At the transition these two minima will have the same value of  $F(E)$ . Equivalently, the probability distribution  $P(E)$  will have two local maxima (such a curve is called a **camel**). For a system with no phase transition in the thermodynamic limit there will only be one minimum of  $F(E)$  for all  $T$ .

From the finite size scaling properties of  $F(E)$ , the order of the transition can be determined after the following considerations. The free energy barrier  $\Delta F(L)$  between the two minima is expected to grow with system size  $L$  at a first order transition, since the surface energy of the coexisting states grows with size. It is expected to be size independent at a critical point, since there are domains at all sizes, that will not change  $F(E)$  as  $L$  grows. If there is no transition, there might be a spurious minimum at small sizes, but this will disappear for larger systems. Hence to categorize the order of a phase transition, Lee and Kosterlitz proposed the following method:

1. Simulate at a temperature close to the suspected transition temperature and calculate  $N(E)$ . Usually the temperature of the specific heat peak is chosen as the simulation temperature.

2. Use the histogram method to compute  $F(E) \propto -\ln N_0(E) + (\beta - \beta_0)E$  at neighboring values of  $T$ . If there are two minima in  $F(E)$  vary  $\beta$  until the minima are equal. This gives an estimate of the possible transition temperature  $T_c$ .
3. Measure the difference  $\Delta F$  between  $F(E)$  at the minima and at the maximum between the minima.
4. Repeat steps 1-3 for larger systems. If  $\Delta F$  increases with size, the transition is first order. If  $\Delta F$  is independent of size, the transition is continuous. If  $\Delta F$  decreases with size, there is no thermodynamic transition.

The above procedure is also applicable for field driven transitions, i.e. transitions that occur by varying the pressure or a magnetic field. For example, for the magnetic field driven Ising transition a histogram  $N(E, M)$  is computed at zero magnetic field, and  $F(M) \propto -\ln \sum_E N(E, M) \exp(-(\beta - \beta_0)E)$  is calculated. Because the states with positive and negative magnetization are equally likely to occur in zero magnetic field, we should get a double minimum in  $F(M)$ . As we increase the system size,  $\Delta F$  should grow for a first order transition and remain the same for a continuous transition.

### 3.9 Collective sampling methods

Near a continuous phase transition, the most important convergence limiting factor of a Monte Carlo simulation is critical slowing down. There exist domains of all sizes, and to flip such a domain with random single spin flips takes longer and longer time the larger these domains are. Hence the convergence time grows rapidly with system size. The relaxation time using a local flip algorithm grows as  $\tau \sim \xi^z \rightarrow L^z$ , where  $z$  is the dynamical critical exponent, which is typically  $z \approx 2$  for local spin flip dynamics. The CPU time to visit all spins is  $L^2$  in two dimensions. Hence the CPU time to generate an approximately independent configuration is  $\tau L^2 \approx L^4$ . Hence an increase of  $L$  by a factor 10 requires  $10^4$  more computer time! Hence critical slowing down seriously limits the accessible sizes that can be converged.

Critical slowing down can in many cases be completely eliminated by the following extremely useful method. Instead of using single spin flip dynamics, we switch nonlocal dynamics where whole clusters of spins are flipped in each MC trial move. This is implemented in the following algorithm:

1. Choose a seed spin at random and flip it. Its four neighbor sites (assuming a square lattice) are the perimeter spins. Form an array of the perimeter spins and define a counter for the total number of perimeter spins.
2. Choose the first spin in the perimeter array. Remove it from the array and replace it by the last spin in the array. If the bond probability  $p = 1 - e^{-2JS_i S_j / T}$  is  $p \geq r$ , where  $r$  is a random number, the spin is added to the cluster and flipped.
3. If the spin is added to the cluster, inspect its perimeter spins. If such a spin is not already in the cluster, add it to the end of the perimeter spin array.
4. Repeat steps 2 and 3 until no perimeter spins remain.

This algorithm is called the **Wolff algorithm**. It performs very well for Ising, XY, Heisenberg and similar models. It effectively beats critical slowing down and is therefore very valuable for studying static properties at continuous phase transitions. However, it is not useful for studying dynamic critical phenomena, since there we want to know the dynamic exponent, which is altered upon changing from local to nonlocal dynamics. In other words, these methods belong to different dynamic universality classes.

The main drawbacks with the Wolff algorithm is that: (1) it can be highly sensitive to defect random numbers, and (2) it will in general not work for so called frustrated systems, because here it violates detailed balance. Frustration means that is it not simultaneously possible to minimize all terms in the Hamiltonian. Example: The antiferromagnetic Ising model on a triangular lattice.

### 3.10 Other ensembles

Above we have been working only with the canonical ensemble. It is obviously possible to work in the other ensembles as well. Monte Carlo methods exist for the canonical (the one we have discussed), microcanonical (constant energy-this is accomplished by employing a so called demon), grand canonical ensembles, PVT-ensemble, etc. See e.g. [Gould-Tobochnik].

### 3.11 Simulation accepting all moves

*Jean-Luc Picard: "Things are only impossible until they're not."*

It is easy to construct a simulation algorithm where all trial moves are accepted. Calculate the probabilities for all possible trial moves, and construct the probability distribution for these moves. Then select one move according to this distribution and do it. Advance the time by an interval given by the probability of the move. The reason that this method is not much used is that calculation of all possible probabilities at every step is usually very expensive. Therefore this method is good only when the acceptance rate using ordinary methods is small.

### 3.12 Dynamic aspects of MC simulation

Since we were able to calculate the correlation time in the simulation, it is tempting to interpret Monte Carlo simulation dynamically. In MC dynamics we want the opposite to the case of static simulations. There it was desirable to use special methods to remove correlations between successive configurations, to get quick convergence. Quite to the contrary, in the dynamic calculation it is those correlations we want to calculate.

Under some conditions it may be useful to interpret Monte Carlo dynamically. One requirement is that the motion of the physical system should be overdamped, that is, purely relaxational without any inertial motion. This is often fulfilled near a phase transition, where the dynamics slows down due to critical slowing down. This slowing down can then be studied by Monte Carlo simulations.



## Chapter 4

# Quantum Monte Carlo

As we understand it today, quantum mechanics gives a precise description of how nature really behaves, and enables us to in principle exactly calculate all the information that exists about a physical system. Unfortunately, such calculations quickly become very difficult in complicated cases. It is hence well motivated to consider simulations. This chapter will introduce quantum Monte Carlo simulations in simple cases. Only systems with one single degree of freedom will be considered. The extensions and difficulties involved in doing quantum many body problems will thus only be mentioned in this introductory chapter.

In classical statistical mechanics we can solve problems by molecular dynamics calculations. Then we determine the position and momentum of a particle,  $r(t), p(t)$ . The properties of the system can then be determined from these trajectories. However, in quantum systems, i.e. in systems where the classical approximation is poor, this is not possible. This is because  $r(t)$  and  $p(t)$  cannot be specified simultaneously according to the uncertainty principle:

$$\Delta x \Delta p > \hbar \quad (4.1)$$

As a consequence, we can not study quantum systems by molecular dynamics: It is simply not meaningful to speak about the path of a quantum particle. But we can solve the Schrödinger equation numerically. This is the subject of this chapter.

There are efficient algorithms for solving the time independent or time dependent Schrödinger equation by standard discretization methods. These methods are however not practical in many body problems. We will therefore not go into these methods here. We will concentrate on methods that are possible to generalize to complex systems involving many interacting quantum particles. This generalization is still a very active field of research. For example, it is presently not known how to effectively simulate many fermion problems. Inventing an efficient Monte Carlo algorithm would be a crucial breakthrough, which would have many very important applications.

We will discuss a sequence of methods: variational methods, random walk Monte Carlo, diffusion Monte Carlo. These methods are suitable for finding the ground state (zero temperature). Finally we will discuss path integral Monte Carlo, which allows study of finite temperatures.

### 4.1 Variational Method

In the section below on diffusion quantum Monte Carlo, we will use trial wave functions as one ingredient. We will therefore first discuss the variational method.

The variational method is based on the inequality

$$\langle H \rangle = \frac{\int dx \psi_T^*(x) H \psi_T(x)}{\int dx \psi_T^*(x) \psi_T(x)} \geq E_0 \quad (4.2)$$

where  $H$  is the Hamiltonian operator,  $\psi_T$  is any trial wave function, and  $E_0$  is the ground state energy. This is because any function can be expanded in terms of eigenstates to  $H$ :

$\psi_T = \sum c_n \psi_n$  where  $H\psi_n = E_n \psi_n, E_0 < E_1 < \dots$ . If  $\psi_T$  is the exact ground state we get  $\langle H \rangle = E_0$ .

In practice, one makes a guess for  $\psi_T(x)$  that contains variational parameters. Calculate  $\langle H \rangle$  and minimize by tuning the variational parameters. A typical variational ansatz is the following. If  $H = T + V(x)$  where  $T$  is the kinetic and  $V$  potential energy, guess  $\psi_{T\alpha} = \exp(\alpha V(x))$ . This gives the exact answer for the harmonic oscillator:  $V(x) = kx^2/2, \psi_0 = \exp(-\sqrt{mk}x^2/2\hbar), E = \hbar\omega/2, \omega = \sqrt{k/m}$ .

## 4.2 Monte Carlo Evaluation of Quantum Expectation Values

Calculation of expectation values in a *given* state, like the variational energy estimate in Eq. (4.2), often require numerical evaluation.

Monte Carlo is often a useful tool for evaluating expectation values in any given state  $\psi$ , for example, a variational state as above. The idea is to sample the degrees of freedom (in this simplest example just the position  $x$ ), with probability distribution defined by the given quantum state  $\psi$ :  $P(x) \sim |\psi(x)|^2$ . The Monte Carlo evaluation of the expectation value for, for example, the energy is then done by

$$E[\psi] = \frac{\int dx \psi^* H \psi}{\int dx \psi^* \psi} = \frac{\int dx \psi^* \psi \frac{H\psi}{\psi}}{\int dx \psi^* \psi} = \frac{1}{N_{smp}} \sum_{i=1}^{N_{smp}} E(x_i) \quad (4.3)$$

where  $E(x) = \frac{H\psi(x)}{\psi(x)}$  will be called the local energy in the state  $\psi(x)$ .

Amusingly, the Monte Carlo simulation methods of classical statistical physics can in this way often be very useful for evaluating quantum expectation values.

## 4.3 Random Walk Monte Carlo (RWMC)

Now, let us face the time dependent Schrödinger equation (TD SE)

$$-\frac{\hbar^2}{2m} \frac{\partial^2 \psi(x, t)}{\partial x^2} + V(x) \psi(x, t) = i\hbar \frac{\partial \psi(x, t)}{\partial t} \quad (4.4)$$

We will only consider time independent potentials  $V(x)$ .

Now we allow ourselves to do a strange but extremely serviceable change of variables. Change from real time to imaginary time  $\tau = it/\hbar$ , which gives

$$\frac{\hbar^2}{2m} \frac{\partial^2 \psi}{\partial x^2} - V(x) \psi = \frac{\partial \psi}{\partial \tau} \quad (4.5)$$

If we neglect the potential term  $V\psi$  and only keep the free part, we get the well known diffusion equation  $D\partial^2 \psi / \partial x^2 = \partial \psi / \partial \tau, D = \hbar^2/2m$ . This suggests that we can simulate quantum mechanics as a diffusion process, or simply as a random walk!

On the other hand, if we ignore the diffusion term we have a decay or growth process:

$$\frac{\partial \psi}{\partial \tau} = -V\psi \Rightarrow \psi = e^{-V\tau} \quad (4.6)$$

Hence  $\psi(x, \tau)$ , which corresponds to the density of random walkers at  $(x, \tau)$ , grows if  $V < 0$  and decays if  $V > 0$ . Note that the random walkers are non-interacting since the SE is linear in  $\psi$ .

### 4.3.1 Ground State

To determine the ground state we note that the general solution to the TDSE is, if the eigenstates are given by  $H\phi_n = E_n\phi_n$ ,

$$\psi(x, \tau) = \sum_n c_n \phi_n(x) e^{-E_n \tau} \rightarrow c_0 \phi_0(x) e^{-E_0 \tau}, \quad \tau \rightarrow \infty \quad (4.7)$$

To avoid  $\psi(x, \tau \rightarrow \infty) = 0$ , we can measure  $E_0$  wrt an arbitrary reference energy  $V_0$ , which is adjusted so that an approximate steady state distribution of the random walkers is obtained.

### 4.3.2 Ground State Energy

To compute the ground state energy from  $\psi(x, \tau \rightarrow \infty)$  is not accurate. Instead we use

$$E_0(\tau) = \langle V \rangle = \frac{\sum n_i V(x_i)}{\sum n_i} \quad (4.8)$$

where  $n_i$  is the number of walkers at  $(x_i, \tau)$ , averaged over  $\tau$ , once steady state has been reached.

**Derivation.** The TDSE with the reference potential is

$$\frac{\partial \psi}{\partial \tau} = \frac{\hbar^2}{2m} \frac{\partial^2 \psi}{\partial x^2} - [V(x) - V_0] \quad (4.9)$$

Integration wrt  $x$ , using  $\partial \psi(|x| \rightarrow \infty) / \partial x \rightarrow 0 \Rightarrow \int dx \partial^2 \psi / \partial x^2 = 0$ , gives

$$\int dx \frac{\partial \psi}{\partial \tau} = - \int V(x) \psi dx + V_0 \int \psi dx \quad (4.10)$$

Differentiate

$$\psi(x, \tau) \approx c_0 \phi_0(x) e^{-(E_0 - V_0)\tau} \quad (4.11)$$

wrt  $\tau$ :

$$\frac{\partial \psi}{\partial \tau} = (V_0 - E_0) \psi \quad (4.12)$$

Eqs. (4.10-4.12) now give  $\int (V_0 - E_0) \psi dx = - \int V(x) \psi dx + V_0 \int \psi dx$ , and

$$E_0 = \frac{\int V(x) \psi(x, \tau) dx}{\int \psi(x, \tau) dx} = \frac{\sum n_i V(x_i)}{\sum n_i} \quad (4.13)$$

The last equality follows from identifying the number of walkers as  $\sum n_i = \int \psi dx$ .

### 4.3.3 Algorithm for RWMC

1. Place  $N_0$  walkers at initial positions  $x_i$  (need not be on a grid).
2. Compute the reference potential as  $V_0 = \sum V_i / N_0$ .
3. Randomly move a walker to the left or right by a fixed step length  $\Delta s$ , related to the time step  $\Delta \tau$  by  $(\Delta s)^2 = 2D\Delta \tau$  (with  $D = 1/2$  in units such that  $\hbar = m = 1$ ).
4. Compute  $\Delta V = [V(x) - V_0]$  and a random number  $0 < r < 1$ .
  - Decay: if  $\Delta V > 0$  and  $r < \Delta V \Delta \tau$ : remove the walker.
  - Growth: if  $\Delta V < 0$  and  $r < -\Delta V \Delta \tau$ : add a new walker at  $x$ .
  - Otherwise: leave the walker at  $x$ .

This procedure is accurate only in the limit  $\Delta \tau \ll 1$ .

5. Repeat 3-4 for all  $N_0$  walkers. Compute the mean potential energy  $\langle V \rangle = \sum n_i V(x_i) / \sum n_i$  and the new number of walkers  $N$ . The new reference potential is  $V_0 = \langle V \rangle - (a/N_0 \Delta \tau)(N - N_0)$ . The parameter  $a$  can be adjusted so that  $N \approx \text{constant}$ .
6. Repeat 3-5 until the estimate of the ground state energy  $\langle V \rangle$  reaches a steady state without fluctuations. Update the reference potential for example every 100 sweeps, to keep the simulation stable.

Typical parameters:  $N_0 \geq 50$ ,  $\Delta s = 0.1$ , number of MC steps per walker  $\geq 500$ . The reference potential should be updated every 100 sweeps, say, to keep the simulation stable.  $V_0$  will, once steady state has been reached, average to the ground state energy.

## 4.4 Diffusion Monte Carlo (DMC)

The main shortcoming in the RWMC is the finite time step. The DMC method improves this approximation. Instead of the discrete random walk we are now going to explore algorithms based on a continuous diffusion process.

Define the **Green function** or **propagator**  $G$  by

$$\psi(x', \tau) = \int G(x', x, \tau) \psi(x, 0) dx \quad (4.14)$$

$G$  "propagates"  $\psi(x, 0)$  to  $\psi(x', \tau)$ . One can think of  $G(x', x, \tau)$  as the probability density for inserting a particle at  $(x, 0)$  and removing it at  $(x', \tau)$ . Differentiation gives

$$\frac{\partial \psi(x', \tau)}{\partial \tau} = \int \frac{\partial G(x', x, \tau)}{\partial \tau} \psi(x, 0) dx \quad (4.15)$$

Using the TDSE on the LHS gives

$$\frac{\partial \psi(x', \tau)}{\partial \tau} = -(H(x') - V_0) \psi(x', \tau) = - \int (H(x') - V_0) G(x', x, \tau) \psi(x, 0) dx \quad (4.16)$$

$$\Rightarrow \frac{\partial G(x', x, \tau)}{\partial \tau} = -(H(x') - V_0) G(x', x, \tau) \quad (4.17)$$

This shows that the Green function obeys the TDS.  $G$  is spatially symmetric:

$$G(x', x, \tau) = G(x, x', \tau) \quad (4.18)$$

The formal operator solution  $G(\tau) = \exp(-[H - V_0]\tau)$  to the TDSE is defined by the Taylor series

$$G(\Delta \tau) = 1 - (H - V_0) \Delta \tau + \frac{1}{2} (H - V_0)^2 \Delta \tau^2 + \text{higher order terms that we neglect} \quad (4.19)$$

We now evaluate, remembering that the kinetic energy  $T$  and potential energy  $V$  do not commute,  $(H - V_0)^2 = H^2 - HV_0 - V_0H + V_0^2$  and  $H^2 = (T + V)^2 = T^2 + TV + VT + V^2$ . To order  $\Delta \tau^2$  we thus get

$$\begin{aligned} G(\Delta \tau) &= G_{\text{branch}/2} G_{\text{diff}} G_{\text{branch}/2} = e^{-\frac{1}{2}[V - V_0]\Delta \tau} e^{-T\Delta \tau} e^{-\frac{1}{2}[V - V_0]\Delta \tau} \\ &= \left(1 - \frac{1}{2}[V - V_0]\Delta \tau\right) \left(1 - T\Delta \tau + \frac{1}{2}T^2\Delta \tau^2\right) \left(1 - \frac{1}{2}[V - V_0]\Delta \tau\right) \\ &= 1 - [H - V]\Delta \tau + \frac{1}{2}[V - V_0]T\Delta \tau^2 + T\frac{1}{2}[V - V_0]\Delta \tau^2 \\ &\quad + \frac{1}{4}[V - V_0]^2\Delta \tau^2 - \frac{1}{2}T^2\Delta \tau^2 \end{aligned} \quad (4.20)$$

since this is equal to Eq. (4.19)

$G_{\text{diff}}$  corresponds to diffusion, and  $G_{\text{branch}/2}$  to branching (changing the number of walkers). They are given by

$$\begin{aligned}\frac{\partial G_{\text{diff}}}{\partial \Delta\tau} &= -TG_{\text{diff}} = \frac{\hbar^2}{2m} \frac{\partial^2 G_{\text{diff}}}{\partial x^2} \Rightarrow \\ G_{\text{diff}}(x', x, \Delta\tau) &= \frac{1}{(4\pi D \Delta\tau)^{1/2}} e^{-(x'-x)^2/4D\Delta\tau}, \quad D = \frac{\hbar^2}{2m}\end{aligned}\quad (4.21)$$

and

$$\begin{aligned}\frac{\partial G_{\text{branch}/2}}{\partial \Delta\tau} &= \frac{1}{2}[V_0 - V]G_{\text{branch}/2} \Rightarrow \\ G_{\text{branch}/2}(x', x, \Delta\tau) &= e^{-(\frac{1}{2}[V(x)+V(x')]-V_0)\Delta\tau}\end{aligned}\quad (4.22)$$

This assumes that half the branching takes place before and half after, which gives the wanted symmetry:  $G(x', x, \tau) = G(x, x', \tau)$ .

#### 4.4.1 DIMC algorithm

1. Begin with  $N_0$  random walkers (no grid). Place the walkers in the regions of space where  $\psi$  is large.
2. **Diffusion step:** Pick a walker at  $x$  and move it to  $x'$ .  $x' - x$  is chosen from a Gaussian distribution<sup>1</sup> with zero mean and variance  $\sigma^2 = 2D\Delta\tau$ .
3. **Branching step:** Add or remove walkers from  $x'$  so that the number of walkers at  $x'$  becomes approximately equal to  $w = \exp(-(\frac{1}{2}[V(x) + V(x')] - V_0)\Delta\tau)$ . Set the number of walkers at  $x'$  equal to the integer part of  $w + r$ , where  $0 < r < 1$  is a random number. The number of walkers can thus be  $0, 1, 2, \dots$  (0 means the walker terminates).
4. Repeat 2 and 3 for all walkers. This corresponds to one time step  $\Delta\tau$ . This corresponds to the integration  $\psi(x', \tau) = \int G(x', x, \Delta\tau)\psi(x, \tau - \Delta\tau) dx$ .
5. The quantity of interest  $\psi(x', \tau)$  becomes independent of the original configuration  $\psi(x, 0)$  after sufficiently many MC time steps. Make sure that the number of walkers at time  $\tau$ ,  $N(\tau)$  is close to the desired number  $N_0$ .

**Typical parameters:**  $N_0 \sim 1000$  walkers,  $\Delta\tau \sim 0.01$ ,  $a \sim 0.1$ . Use at least 3 different values of  $\Delta\tau$  and extrapolate to  $\Delta\tau = 0$ . Adjust the reference energy  $V_0$  every  $\sim 20$  MCS.

In practice an improved version of this algorithm is used. This will be described below.

#### 4.4.2 How is DIMC related to RWMC and to Green function MC?

1. The Gaussian distribution gives the exact distribution for displacements of a walker in time  $\Delta\tau$ . The fixed step size in RWMC gives the average displacement. Hence, there are no systematic errors due to finite step size in DIMC.
2. If the exponential in  $G_{\text{branch}/2}$  is expanded to first order in  $\Delta\tau$ , and we set  $V(x') = V(x)$ , then we get the same branching rule as in RWMC.
3. DIMC is not exact since the branching is independent of  $x$ , which is only true for  $\Delta\tau \rightarrow 0$ . This is overcome in Green function MC, where the Green function is simulated directly, without a short time approximation. This is briefly mentioned in the next section.

---

<sup>1</sup>This can be done by the *Box-Muller method*: Set  $\rho = -\ln r$ ,  $x = (2\rho)^{1/2}\sigma \cos(2\pi s)$ , where  $0 < r, s < 1$  are uniformly distributed random numbers. Then  $x$  will be Gaussian distributed with zero mean and variance  $\sigma$ :  $p(x) = \exp(-x^2/2\sigma^2)/(2\pi\sigma^2)^{1/2}$ .

### 4.4.3 Weighting

*Miss Piggy: "Never eat more than you can lift."*

The DIMC described above is not yet very efficient. To avoid branching into a large number of walkers we can use two methods: weighting and importance sampling.

The idea is to work with a constant number of walkers, and replace branching with weighting of the walkers. At each weighting step each walker is weighted by  $G_{\text{branch}/2}$ , keeping a constant number of walkers. After  $n$  steps walker  $k$  gets weight

$$w_k = \prod_{i=1}^n G_{\text{branch}/2}(i, k)$$

where  $G_{\text{branch}/2}(i, k)$  is the branching factor at the  $i$ :th time step. The contribution to any average quantity of the  $k$ :th walker is then weighted by  $w_k$ .

The weighting method can however also become inefficient, because it does not terminate walkers that are located in unfavorable regions. We therefore rather recommend another method that we will now turn to.

### 4.4.4 Importance Sampling

The importance sampling method, due to Kalos and coworkers, can avoid these problems and often gives very satisfactory performance. We use a trial wave function  $\psi_T(x)$  to guide walkers towards important regions of  $V(x)$ . Thus  $\psi_T$  should be small where  $V$  is big.

Set  $f(x, \tau) = \psi(x, \tau)\psi_T(x, \tau)$ . The TDSE becomes

$$\partial f \tau = D \frac{\partial^2 f}{\partial x^2} - \underbrace{D \frac{\partial f F(x)}{\partial x}}_{\text{drift towards important regions}} - [E(x) - V_0] = 0$$

where

$$F(x) = \frac{2}{\psi_T} \frac{\partial \psi_T}{\partial x}, \quad E(x) = \frac{H\psi_T}{\psi_T} = V(x) - \frac{D}{\psi_T} \frac{\partial^2 \psi_T}{\partial x^2}$$

Now, incorporate the drift term in  $G_{\text{diff}}$ . Replace

$$(x' - x)^2 \rightarrow (x' - x - D\Delta\tau F(x))^2 \Rightarrow \quad (4.23)$$

$$G_{\text{diff}}(x', x, \Delta\tau) = \frac{1}{(4\pi D\Delta\tau)^{1/2}} e^{-(x' - x - D\Delta\tau F(x))^2 / 4D\Delta\tau} \quad (4.24)$$

This however destroys the symmetry between  $x$  and  $x'$ . We restore the symmetry by using the Metropolis algorithm to accept the new position of the walker with probability

$$P = \frac{|\psi_T(x')|^2 G_{\text{diff}}(x, x', \Delta\tau)}{|\psi_T(x)|^2 G_{\text{diff}}(x', x, \Delta\tau)}$$

Accept the move if  $r \leq P$ , otherwise reject.

The branching step is replaced by:

$$G_{\text{branch}}(x', x, \Delta\tau) = e^{-(\frac{1}{2}[V(x)+V(x')]-V_0)\Delta\tau} \rightarrow e^{-(\frac{1}{2}[E(x)+E(x')]-V_0)\Delta\tau^*}$$

where  $\Delta\tau^*$  is the time interval since the last accepted diffusion move, thus counting also the time for rejected moves.

## 4.5 Path integral Monte Carlo (PIMC)

*Isaac Asimov: "The most exciting phrase to hear in science, the one that heralds new discoveries, is not 'Eureka!' (I found it!) but 'That's funny ...'"*

All previously described QMC methods are most suitable for finding ground state properties, and possibly the first few excited states. PIMC is of particular interest for finding thermal properties.

First we need to reformulate quantum mechanics. Classical mechanics is ruled by the principle of least action: A classical particle follows the path between two given space-time points that minimizes the action

$$S = \int_{x_0, t_0=0}^{x, t} L dt \quad (4.25)$$

where the Lagrangian is  $L = T - V$ . Quantum mechanics can also be formulated in terms of the action. The propagator is given by the Dirac-Feynman path integral

$$G(x, x_0, t) = \sum_{\text{all paths}} e^{\frac{i}{\hbar} S} = \int_{x_0, t_0=0}^{x, t} D[x(t)] e^{\frac{i}{\hbar} S} \quad (4.26)$$

(recall that  $G(x, x_0, t)$  is the probability amplitude to be at  $(x, t)$ , starting from  $(x_0, 0)$ ). Hence a quantum particle takes *all* possible paths between a starting and a final point!

The path integral has several highly attractive features. First, the transition to classical mechanics is directly transparent in this formulation of quantum mechanics: In the limit  $\hbar \rightarrow 0$  only the path with least action contributes to the path integral. Second, it allows thinking about quantum systems in a nearly mechanical picture, which is perhaps less elusive than trying to grasp wave functions. Thirdly, it unravels a deep connection between quantum mechanics and statistical mechanics, which is what we will make use of here.

The Green function  $G$  satisfies

$$\begin{aligned} \underbrace{\psi(x, t)}_{=\sum_n c_n \phi_n(x) e^{\frac{i}{\hbar} E_n t}} &= \int G(x, x_0, t) \underbrace{\psi(x_0, 0)}_{=\sum_n c_n \phi_n(x_0)} dx_0 \Rightarrow \\ G(x, x_0, t) &= \sum_n \phi_n(x) \phi_n(x_0) e^{\frac{i}{\hbar} E_n t} \end{aligned} \quad (4.27)$$

where  $H\phi_n = E_n\phi_n$ ,  $\int \phi_m \phi_n dx = \delta_{m,n}$ .

Now go to imaginary time:  $\tau = \frac{i}{\hbar} t$  (set  $\hbar = 1$ ). We find

$$G(x, x_0, \tau) = \sum_n \phi_n(x) \phi_n(x_0) e^{-E_n \tau} \quad (4.28)$$

First consider the ground state:

$$G(x, x, \tau \rightarrow \infty) = \phi_0(x)^2 e^{-E_0 \tau} \quad (4.29)$$

Hence we need to compute  $G$  and therefore  $S$  to get the ground state  $\phi_0$ .

To compute the action  $S = \int_{t_0}^t L dt$ , we first convert to a sum. For a single particle (with  $m = 1$ )

$$L = -\frac{1}{2} \left( \frac{dx}{d\tau} \right)^2 - V(x) = -E \quad (4.30)$$

Divide the  $\tau$  interval into  $N$  steps  $\Delta\tau$ :

$$E(x_j, \tau_j) = \frac{1}{2} \left( \frac{x_{j+1} - x_j}{\Delta\tau} \right)^2 + V(x_j) \quad (4.31)$$

where  $\tau_j = j\Delta\tau$ . This gives

$$iS = -\Delta\tau \sum_{j=0}^{N-1} E(x_j, \tau_j) = -\Delta\tau \sum_j \left[ \frac{1}{2} \left( \frac{x_{j+1} - x_j}{\Delta\tau} \right)^2 + V(x_j) \right] \quad (4.32)$$

The probability of the path becomes

$$e^{iS} = \exp -\Delta\tau \left[ \frac{1}{2} \left( \frac{x_{j+1} - x_j}{\Delta\tau} \right)^2 + V(x_j) \right] \quad (4.33)$$

The propagator then becomes

$$G(x = x_N, x_0, N\Delta\tau) = A \int dx_1 \cdots dx_{N-1} \exp -\Delta\tau \sum_j \left[ \frac{1}{2} \left( \frac{x_{j+1} - x_j}{\Delta\tau} \right)^2 + V(x_j) \right] \quad (4.34)$$

where  $A$  is an unimportant constant.

To get to the ground state we must set  $x = x_0$  according to Eq. (4.29). The propagator then becomes the same as the partition function of a system of weights on a ring, with transverse displacements  $x_j$ , connected with springs with spring constant  $k = 1/\Delta\tau$  to its neighbors.

#### 4.5.1 PIMC algorithm

1. Chose  $N$  and  $\Delta\tau$  ( $N\Delta\tau \gg 1$  for the  $T = 0$  limit). Chose  $\delta$  as the maximum trial change movement of a particle. Chose initial configuration for the positions  $x_j$ . Try to get close to the shape of the ground state probability amplitude.
2. Chose  $j$  randomly and generate a trial displacement  $x_{\text{trial}} = x_j + (2r - 1)\delta$ , where  $0 < r < 1$  is a random number. Compute

$$\begin{aligned} \Delta E &= \frac{1}{2} \left( \frac{x_{j+1} - x_{\text{trial}}}{\Delta\tau} \right)^2 - \frac{1}{2} \left( \frac{x_{j+1} - x_j}{\Delta\tau} \right)^2 + V(x_{\text{trial}}) \\ &\quad - \frac{1}{2} \left( \frac{x_j - x_{\text{trial}}}{\Delta\tau} \right)^2 + \frac{1}{2} \left( \frac{x_j - x_{j-1}}{\Delta\tau} \right)^2 - V(x_j) \end{aligned} \quad (4.35)$$

Accept if  $\Delta E < 0$ . Otherwise compute  $w = \exp(-\Delta\tau\Delta E)$  and accept if  $p > r$ , otherwise reject.

3. Update a probability density array element  $P(x)$ :  $P(x = x_j) \rightarrow P(x = x_j) + 1$  (even if the move was rejected!). Divide the possible  $x$  values into equal size bins of width  $\Delta x$ .
4. Repeat 3-4 until a sufficient number of MC sweeps have been taken. Skip initial sweeps to approach equilibrium.

**Typical choices of parameters to get to the ground state:**

$$N\Delta\tau > 15, N = 10, 20, 30, 40, 80 \quad (4.36)$$

Equilibrate for  $> 2000$  sweeps. Collect data for  $> 5000$  sweeps.

#### 4.5.2 Ground state

The probability density  $P(x)$  must be normalized by dividing  $P(x)$  by  $N \times N_{\text{sweep}}$ .

**The ground state energy** is obtained from

$$E_0 = \sum_x P(x) [T(x) + V(x)] \quad (4.37)$$

Here  $V(x)$  is the potential energy, and the kinetic energy is obtained from the virial theorem:  $2T(x) = x dV/dx$  (this is more accurate than to form averages of  $(x_{j+1} - x_j)$ ).

**The ground state** is obtained from Eq. (4.29)

$$\phi_0(x) = \sqrt{P(x)} \quad (4.38)$$



### 4.5.3 Thermal properties

The imaginary time path integral is equivalent to a Boltzmann distribution for a single particle, since it contains  $\exp(\tau E_n)$ , which becomes  $\exp(-\beta E_n)$  if we identify  $\beta = \tau = N\Delta\tau = (i/\hbar)t$ . Hence, to simulate at a temperature  $T > 0$ , we just don't take the limit  $\beta = N\Delta\tau \rightarrow \infty$  needed to get the ground state. For finite  $T$  simulations a smaller value of  $N$  is obtained to reach the same accuracy.

### 4.5.4 Generalization of PIMC to higher dimensions

It is possible to generalize the PIMC algorithm to higher dimensions and many interacting particles. In 3D,  $r_j$  is a 3D vector displacement. Each particle is represented by a ring of  $N$  “particles” with a spring potential connecting each particle within the ring. To make this algorithm efficient requires some work.



# Bibliography

- [1] H. Gould and J. Tobochnik, An Introduction to Computer Simulation Methods, 2nd edition (Addison-Wesley 1996). Introduction to classical and quantum Monte Carlo, as well as to many other topics in computational physics.
- [2] Two very good textbooks on MC simulation:  
M. E. J. Newman and G. T. Barkema, Monte Carlo methods in Statistical Physics, corrected edition, Oxford, 2001.  
D. P. Landau and K. Binder, Monte Carlo simulation in statistical physics, Cambridge 2000.
- [3] W. H. Press, S. A. Teukolsky, W. T. Wetterling, and B. P. Flannery, Numerical Recipes, Cambridge University Press. All you want to know about standard numerical algorithms. Comes with computer programs. Available both in FORTRAN and C.
- [4] Guide to Available Mathematical Software, WWW site at address  
<http://gams.nist.gov/>  
Contains a huge set of very useful free numerical software.
- [5] A good text on statistical mechanics is: M. Plischke and B. Bergersen, Equilibrium statistical mechanics, 2ed, World Scientific 1994. Contains modern methods and applications of statistical physics, and a chapter on Monte Carlo simulation and data analysis.
- [6] Two excellent texts on quantum mechanics:  
R. Shankar, Principles of Quantum Mechanics, 2 edition, Plenum, 1994.  
J. J. Sakurai, Modern quantum mechanics, revised edition, Addison-Wesley, 1994.