

1 Introduction

The model which we will simulate in this course is the ϕ^4 theory in $D = 3$ dimension. The aim is to make a complete calculation, starting from writing the code and debugging it, running and data analysis. We are going to use the Hybrid Monte Carlo algorithm.

1.1 Definition of the model

We immediately define the model on a D dimensional L^D lattice with lattice spacing a and size $L = Na$. Thus the lattice points are given by

$$x = a(n_0, \dots, n_{D-1}) ; \quad n_i \in \mathbb{N}_0, \quad 0 \leq n_i < N$$

The real valued fields ϕ are living on the sites, $\phi_x \in \mathbb{R}$ and we adopt periodic boundary conditions. So if $\hat{\mu}$ is the unit vector in μ direction, then $\phi_{x+L\hat{\mu}} = \phi_x$. The action S is given by

$$S(\phi) = \sum_x \left[-2\kappa \sum_{\mu=0}^{D-1} \phi_x \phi_{x+\hat{\mu}} + \phi_x^2 + \lambda(\phi_x^2 - 1)^2 \right] \quad (1)$$

Expectation values can be computed using the path integral method

$$\langle A \rangle = \frac{1}{Z} \int \prod_x d\phi_x \exp(-S(\phi)) A(\phi)$$

with the partition function $Z = \int \prod_x d\phi_x \exp(-S(\phi))$.

The model has two well known limits: $\lambda = 0$ gives a Gaussian model, in the limit $\lambda \rightarrow \infty$ the Ising model is recovered. It has a second order critical line in the space spanned by the two coupling constants λ and κ .

1.2 Observables

We will be looking just at a few observables. The most important ones are powers of the magnetization m

$$m = \sum_x \phi_x . \quad (2)$$

Note that on a finite lattice, $\langle m \rangle = 0$, so interesting quantities are the magnetic susceptibility

$$\chi = \frac{1}{V} \langle m^2 \rangle$$

and the Binder cumulant

$$U = \frac{\langle m^4 \rangle}{(\langle m^2 \rangle)^2} .$$

We will also look at derivatives of observables with respect to κ . If A does not depend on κ , we can get them from correlations with the interaction term $W = 2 \sum_x \sum_{\mu=0}^{D-1} \phi_x \phi_{x+\hat{\mu}}$

$$\frac{\partial}{\partial \kappa} \langle A \rangle = \langle WA \rangle - \langle W \rangle \langle A \rangle . \quad (3)$$

1.3 Algorithm

The purpose of these exercises is to get hands on experience implementing and using the HMC algorithm. For this purpose, momenta π_x conjugate to the field variables ϕ_x are introduced and the partition function is extended as follows

$$Z = \int \prod_x d\pi_x \prod_x d\phi_x e^{-H(\pi, \phi)}$$

with Hamiltonian $H(\pi, \phi) = \frac{1}{2} \sum_x \pi_x^2 + S(\phi)$. Expectation values, which are functions of ϕ only, are unaltered.

As the “hybrid” already indicates, it has several parts: momentum heat bath, molecular dynamics evolution and Metropolis step

1. Momentum heat-bath: Choose new random momenta according to the distribution $P(\pi_i) \propto \exp(-\pi_i^2/2)$
2. Molecular dynamics evolution: Numerically solve the Hamiltonian equations of motion

$$\begin{aligned} \frac{d}{d\tau} \phi_x^\tau &= \frac{\partial}{\partial \pi_x} H(\pi^\tau, \phi^\tau) \\ \frac{d}{d\tau} \pi_x^\tau &= -\frac{\partial}{\partial \phi_x} H(\pi^\tau, \phi^\tau) \end{aligned}$$

for some fictitious time interval τ . This moves the fields from some initial (π, ϕ) to (π', ϕ') .

3. Acceptance step: Calculate the change in the Hamiltonian $\Delta H = H(\pi', \phi') - H(\pi, \phi)$ and accept the proposed new field ϕ' with probability $P_{\text{acc}} = \min[1, \exp(-\Delta H)]$.

2 Implementation

Getting all this will take some time, so let's start simple. Some basic routines are provided. They are written in C and are just propositions to make life easier. If you are unfamiliar with this language, you can (re-)implement them in any other language of your choice. You can get them at <http://www-com.physik.hu-berlin.de/~sschaef/LH>

Most of it is in file `phi4.c`. It contains the main program, which at the moment does read in the basic parameters of the action, initializes the hopping field, fills the field ϕ with random numbers and computes the action S on a given field configuration ϕ .

2.1 Layout of the lattice

The ϕ field and the hopping field are global in `lattice.h`, where also the lattice size L , the dimension D and the volume V are defined.

The lattice is ordered lexicographically, each point with coordinate $(n_0, n_1, \dots, n_{D-1})$ gets assigned a unique index j . It can be computed by

$$j = \sum_{i=0}^{D-1} n_i L^i.$$

This way, the field ϕ_x of the Lagrangian is realized as the field `phi[j]` of the computer program.

Fortunately, the specific ordering of the points needs rarely to be understood. However, the hopping field `hop[V][2*D]` is important to understand: it is used to navigate the D dimensional lattice. The index of the neighbor of the point with index i in direction μ is `hop[i][mu]` in the forward direction $0 \leq \mu < D$ and `hop[i][D+mu]` in the backward direction. If this point has coordinates (n_0, n_1, n_2) , then the point with coordinates $(n_0, n_1 + 1, n_2)$ has index `jp=hop[i][1]`. Analogously, if we want to increase n_2 by one, then `jp=hop[i][2]`. The field also takes care of the periodic boundary conditions. If you want to go into the negative direction, the corresponding indices are in the upper D entries of the `hop` field. So $(n_0, n_1 - 1, n_2)$ has index `jp=hop[i][D+1]`.

2.2 Tasks

To get familiar with the code here are a few simple exercises

1. Get the code, type `make phi4` to compile and then run it `./phi4 infile`. `infile` is the input file from which the parameters are read.
2. Understand the `phi4.c` file. First `main()` then `action()` which computes the action S for the global field `phi[]`, in particular how the hopping field is used.
3. Write a routine to measure the magnetization m defined in Eq. 2. Test this routine on field configurations ϕ of which you know the result.
4. Write a routine to measure the other quantities necessary for the κ derivative of $\langle m^2 \rangle$ and $\langle m^4 \rangle$ using Eq. 3.

3 Random Numbers

Gaussian random numbers: As random number generator we use `ranlux` by M. Lüscher. It is available from the author's web-page:

<http://luscher.web.cern.ch/luscher/ranlux/index.html>

It generates pseudo-random numbers, equally distributed in the range $[0, 1)$. One method to get Gaussian random numbers is the so-called Box-Muller procedure. Given x_1 and x_2 which are drawn from a flat distribution in $(0, 1]$ the following transformation gives Gaussian random numbers y_1 and y_2 , distributed according to $P(y) \propto \exp(-y^2/2)$

$$y_1 = \sqrt{-2 \ln x_1} \cos(2\pi x_2)$$

$$y_2 = \sqrt{-2 \ln x_1} \sin(2\pi x_2)$$

3.1 Tasks

1. Write a routine which fills a vector of length n with double precision Gaussian random numbers distributed according to $P(x) \propto \exp(-x^2/2)$. Test this routine by filling many such vectors and histogramming. Does the distribution match your expectation? (Do only if time allows, otherwise get the routine from me.)
2. Now introduce the global momentum field `mom[V]` which will hold the momenta π conjugate to the field variables `phi[V]` and fill it with normally distributed random numbers. Write a routine which, given the `phi` and `mom` fields, computes the molecular dynamics Hamiltonian H .