CSP Project

Justus Grabowsky

March 2024

1 Summary/Notes Papers

1.1 Problems of MC methods close to critical points

Near critical points (points where phase transitions occur), systems exhibit a phenomenon called critical slowing down. This means the system takes increasingly longer to reach equilibrium after being perturbed. In the context of Monte Carlo simulations, this translates to extremely long autocorrelation times. Autocorrelation refers to the degree of similarity between data points separated by a certain time interval. When autocorrelation times are long, successive measurements are highly correlated, leading to inefficiencies in sampling the system's configuration space.

1.2 Collective Monte Carlo Updating for Spin Systems

Monte Carlo algorithm that updates large clusters of spins simultaneously in systems at and near criticality. The elementary cluster update step consists of the following sequence of operations:

1. choose random reflection $r \epsilon S_{n-1}$ and a random lattice site $x \epsilon \Lambda$ as the first point of a cluster $c \epsilon \Lambda$
2. flip spin σ_x to $R(r)\sigma_x$ and mark x
3. go over all links connecting $x \epsilon c$ to its nearest neighbors y and activate $\langle xy \rangle$ with probability

   $$P(\sigma_x, \sigma_y) = 1 - exp\left(min\left[0, \beta\sigma_X\left[1 - R(r)\right]\sigma_y\right]\right) = 1 - exp\left(min\left[0, 2\beta(r \cdot \sigma_x)(r \cdot \sigma_y)\right]\right),$$

   if this happens, flip σ_y and mark y as adjoined to c. Intuitively the probability of σ_y being flipped is higher when the vectors σ_x and σ_y have components in opposite directions relative to the vector r.
4. Continue iteratively in the same way for all bonds leading to unmarked neighbors of newly adjoined sites until the process stops.

Ergodicity (all possible states of the system are accessible) and detailed balance ($\pi_i \cdot p_{ij} = \pi_j \cdot p_{ji}$ are both fulfilled. No sign of critical slowing down is visible with autocorrelation times of 1-2 steps per spin for estimators of long-range quantities.

1.3 Critical exponents of the classical three-dimensional Heisenberg model: A single-cluster Monte Carlo study

Simulation of the three-dimensional Heisenberg model in simple cubic lattices, using single-cluster MC update algorithm. The expected pronounced reduction of critical slowing down at the phase transition is verified (why do we expect that??).
Before the precision of MC methods was poor since they were plagued by both systematic and statistical errors. For spin systems, cluster algorithms turned out to be successful. For three dimensions of the Ising and XY model the single-cluster algorithm is superior and is therefore chosen in this paper for the 3D Heisenberg simulation. One main reason for this is that the autocorrelation time τ_0 is almost independent of the linear lattice size L, while for the standard heat-bath or Metropolis algorithms τ_0 rapidly increases with L.
The algorithm used is the one described above in 1.2. The single-cluster algorithm outperforms the Metropolis algorithm; for the largest lattice size $L = 48$ a reduction of the autocorrelation time by about three orders of magnitude was found.

1.4 Optimal phase space sampling for Monte Carlo simulations of Heisenberg spin systems

Cluster update algorithms used to simulate classical spin models, such as the classical 3D Heisenberg model, are useful to reduce the critical slowing down generated close to the critical temperature. However, their applicability is limited to Hamiltonians that just include exchange interaction contribution. If other contributions (e.g. magnetic anisotropy or externally applied magnetic field) are considered, the cluster algorithms must be modified accordingly, increasing the complexity of the algorithms and their implementation. Furthermore, for small systems, local update algorithms have shown to be more efficient and show great flexibility allowing their application to a great variety of systems. Therefore, much effort has been made to enhance the performance of these algorithms, which relies on their efficiency to sample the phase space and avoid long correlation times that slow the convergence of the thermal averages to equilibrium.
In MC Metropolis algorithms new states of the system are generated from a previous state by randomly selecting a spin and updating it to a new trail position and are rejected or accepted according to a transition probability, that depends on the energy difference between initial and trial state. There are different types of trial moves, that directly influence the efficiency of the algorithm, including

the spin-flip, the random, and the small-step moves. The efficiency of these moves varies with the anisotropy of the system.
To deal with this problem, a Gaussian move is chosen, that is based on the small-step move. The new adaptive algorithm that is presented keeps the acceptance rate close to 50 %, which is ideal for efficiently sampling the phase space of the system. The Gaussian trail move takes the initial spin direction and moves the spin to a point on the unit sphere in the vicinity of the initial position according to the expression

$$S_i' \;=\; \frac{S_i + \sigma\Gamma}{|S_i + \sigma\Gamma|},$$

where Γ is a Gaussian distributed random vector and σ is proportional to the width of a cone around the initial spin direction. When using this Gaussian move, the acceptance rate can be adjusted by varying the value of σ. To keep the acceptance at around 50% at all temperatures for a given system, σ should be a function of parameters such as the temperature, the exchange coupling, the anisotropy, and the field intensity. At higher temperatures, it is necessary to use a higher cone width.
Because the optimal cone width σ_{opt} is a characteristic of every system at a given temperature and Hamiltonian parameters, finding a general equation for σ_{opt} is too complex. Therefore, an adaptive algorithm for the Gaussian move is used that changes the cone width adaptively to keep an acceptance rate close to 50%. The adaptive move is done in the following way: at each temperature, the simulation starts with a high cone width ($\sigma = 60$) in the first Monte Carlo Step (MCS), and from then on, is recalculated every step, by multiplying the current cone width by a factor obtained according to the acceptance rate in the previous MCS. The factor is for all temperatures of the form

$$f \;=\; \frac{0.5}{1 \;-\; R},$$

where R is the acceptance rate of the previous step. Therefore, when the acceptance rate is high, the cone width is multiplied by a large factor and the cone width increases to reach the optimal acceptance rate. When the acceptance rate is low, the cone width is multiplied by a small factor and the cone width decreases. Some observations:

1. The acceptance rate converges to a specific value
2. Far below T_c the initial acceptance rate is very low when the system is initially ordered, when the system initially is disordered, the acceptance rate is higher (large angular changes take place to order the system)
3. At a higher temperature below T_c the adaptive move requires more MCS to reach equilibrium and the cone width stabilizes at a higher value than before
4. Above T_c an acceptance rate of 50% is not possible. When the system is ordered, the initial acceptance rate is low, since the exchange energy

due to the order prevents large angular changes. The cone width initially decreases but since an acceptance rate below 50% is not possible, the width increases again after the rate has grown past 50%. When the initial state is disordered the acceptance rate initially is high and stabilizes at a lower value, but above 50%. Independently of the initial state, the cone width increases indefinitely when the acceptance rate stabilizes above 50%. Therefore, it is reset to a width of 60 every time it reaches a higher value. We conclude, that above T_c the adaptive move has the same efficiency as the random move at temperatures above T_c where the cone width stabilizes at 60.