Monte Carlo Simulation technique

S. B. Santra
Department of Physics
Indian Institute of Technology Guwahati
Introduction

- What is Monte Carlo (MC)?

  Monte Carlo method is a common name for a wide variety of stochastic techniques. These techniques are based on the use of random numbers (sampling) and probability statistics to investigate problems in areas as diverse as economics, Material Science, Statistical Physics, Chemical and Bio Physics, nuclear physics, flow of traffic and many others. In general, to call something a "Monte Carlo" method, all you need to do is use random numbers to examine your problem.

Why MC? Difficult to solve analytically, may have too many particles in the system, may have complex interactions among the particles and or with the external field.

For example: A macroscopic system has atoms or molecules of the order of Avogadro number \(N_A \approx 6.022 \times 10^{23}\) per mole. Moreover, there exists complex interaction between them.

The complexity in the interaction may give rise to important qualitative features in the behaviour of a macroscopic system.
Can we determine the value of π using a Monte Carlo method?

Draw a square on the ground, then inscribe a circle within it. Uniformly scatter some objects of uniform size (grains of rice or sand) over the square. Count the number of objects inside the circle and the total number of objects. The ratio of the two counts is an estimate of the ratio of the two areas, which is π/4. Multiply the result by 4 to estimate π.
What have we done?

1. Defined a domain of possible inputs.
2. Generated inputs randomly from a probability distribution over the domain.
3. Performed a deterministic computation on the inputs.
4. Aggregate the results.

Domain of inputs is the square that circumscribes the circle.

**Common properties of Monte Carlo methods:**

The inputs should truly be random. If grains are purposefully dropped into only the center of the circle, they will not be uniformly distributed, and so our approximation will be poor.

There should be a large number of inputs. The approximation will generally be poor if only a few grains are randomly dropped into the whole square. On average, the approximation improves as more grains are dropped.
Evaluation of definite Integral by MC

Conventional approaches: rectangle, trapezoidal, Simpson’s methods …

Quadrature formula:
\[ I = \int_{a}^{b} f(x) \, dx \]
\[ I \approx \Delta x \sum_{i=1}^{n} f(x_i) = \frac{b-a}{n} \sum_{i=1}^{n} f(x_i) \]

Monte Carlo Integration: same quadrature formula, different selection of points

\[ I \approx \frac{b-a}{n} \sum_{i=1}^{n} f(x_i) \]

Very effective in evaluating higher dimensional integrations. Error is less in MC
How to make things random?

- You need to have Random Number Generator.
- RNG provides uniformly distributed numbers between 0 and 1.
- How to generate?
  - Congruential method: a simple and popular algorithm

\[ X_n = (cX_{n-1} + a_0) \text{MOD}(N_{\text{max}}), \]

provides numbers between 1 and \( N_{\text{max}} \).

In 32-bit linear congruential algorithm, \( a_0 \) is zero, \( X_0 \) is taken as an odd number, \( c=16807 \), \( N_{\text{max}}=2^{31}-1=2147483641 \).

\[ X_n = (cX_{n-1}) \text{MOD}(N_{\text{max}}) \]
Different types of MC

**Classical Monte Carlo**: samples are drawn from a probability distribution, often the classical Boltzmann distribution, to obtain thermodynamic properties or minimum-energy structures;

**Quantum Monte Carlo**: random walks are used to compute quantum-mechanical energies and wave functions, often to solve electronic structure problems, using Schrödinger's equation as a formal starting point;
Macroscopic system: Classical MC

Consider a macroscopic system described by a Hamiltonian $H$ in thermodynamic equilibrium with a heat bath at temperature $T$.

$$P_s = \frac{1}{Z} e^{-E_s/k_B T}$$

The expectation value of a physical quantity $A$ is given by

$$\langle A \rangle = \frac{1}{Z} \int \int A(p, q) \exp \{-\beta H(p, q)\} \, d^3N \, q \, d^3N \, p,$$

If the system is described by discrete energy states

$$\langle A \rangle = \frac{1}{Z} \sum_s A_s \exp(-E_s/k_B T)$$

• One needs to calculate $Z$, the partition function, which gives an exact description of the system.
• Evaluation of an infinite series or an integration over a $6N$ dimensional space.
• In most of the cases, it is not possible to evaluate $Z$ exactly.
• Difficulty is not only in evaluating the integral/infinite series but also handling the complex interaction appearing in the exponential.
Monte Carlo sampling

• In general, it is not possible to evaluate the summation over such a large number of state or the integral in such a high dimensional space.
• It is always possible to take average over a finite number of states, a subset of all possible states, if one knows the correct Boltzmann weight of the states and the probability with which they have to be picked up.
• Say, a small subset of all possible states are chosen at random with certain probability distribution $p_s$. So the best estimate of a physical quantity $A$ will be

$$A_N = \frac{\sum_{s=1}^{N} A_s p_s^{-1} \exp(-E_s/k_B T)}{\sum_{s=1}^{N} p_s^{-1} \exp(-E_s/k_B T)}.$$  

For a two state system: $N_s = 2^n$, $2^{100} \approx 10^{30}$

Usually, in a MC simulation $N$ is taken as $10^6$ to $10^8$. 

$N \rightarrow \infty$, $A_N \approx \langle A \rangle$. 

$N_s = 2^n$, $2^{100} \approx 10^{30}$
Simple sampling Monte Carlo

• The problem is how to specify $p_s$ so that the chosen $N$ states will lead to right $<A>$?
• A simple choice of $p_s=p$ for all state will lead the estimator

$$A_N = \frac{\sum_{s=1}^{N} A_s p_s^{-1} \exp(-E_s/k_B T)}{\sum_{s=1}^{N} p_s^{-1} \exp(-E_s/k_B T)} = \frac{\sum_{s=1}^{N} A_s p^{-1} \exp(-E_s/k_B T)}{\sum_{s=1}^{N} p^{-1} \exp(-E_s/k_B T)}$$

$$= \frac{\sum_{s=1}^{N} A_s \exp(-E_s/k_B T)}{\sum_{s=1}^{N} \exp(-E_s/k_B T)},$$

Such average is generally suitable for non-thermal, non-interacting systems. For these systems, $T$ is taken constant but high.

$$\langle A \rangle \approx A_N = \frac{1}{N} \sum_{i=1}^{N} A_i.$$
Example-1: Random walk - Diffusion

\[ Z_t = z^t \]  
For square lattice, \( z = 4 \).  
For \( t = 400 \), No. of configurations \( \approx 10^6 \).  

\[ R_t^2 = (x_t - x_0)^2 + (y_t - y_0)^2 \]  
\[ \langle R_t \rangle = \sqrt{\frac{1}{N} \sum_{s=1}^{N} \{R_t^2\}_s} \]  
\( \langle R_t \rangle \sim t^\nu \quad \nu = 1/2 \)
Example-2: Self-avoiding walk - Linear polymers in dilute solution

\[ Z_N \sim N^{\gamma-1} z_{eff}^N \quad \text{with} \quad z_{eff} \leq z - 1 \]

\[ \langle R_N^2 \rangle = \frac{1}{Z_N} \sum_r r^2 Z_N(r) = \frac{\sum_r r^2 Z_N(r)}{\sum_r Z_N(r)} \sim N^{2\nu} \]

\[ \langle R_N^2 \rangle_T = \frac{\sum_{n,r} r^2 Z_N(n,r)e^{n\epsilon/k_BT}}{\sum_{n,r} Z_N(n,r)e^{n\epsilon/k_BT}} \sim N^{2\nu_T} \]

For \( d = 2, \nu = \frac{3}{2+d} = 3/4 \)
Percolation—disordered systems

Composite material = mixture of metal and non-metal, C is the concentration of conducting elements, I is the current flowing in the circuit.

- **C < C***
  - Non-conducting
- **C > C***
  - Always conducting
- **C = C***
  - Just conducting/Percolation
Other Examples

**Dilute magnet:** Paramagnet disordered by non-magnetic ions. Concentration of magnetic ions is, say $p$.

<table>
<thead>
<tr>
<th>$p &lt; p_c$</th>
<th>$p &gt; p_c$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-magnetic</td>
<td>Magnetic</td>
</tr>
</tbody>
</table>

**Polymer gelation:** Concentration of monomers is $p$.

<table>
<thead>
<tr>
<th>$p &lt; p_c$</th>
<th>$p &gt; p_c$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solution</td>
<td>Gel</td>
</tr>
</tbody>
</table>

- Flow of liquid in porous media
- Spread of a disease in a population
- Stochastic star formation in spiral galaxies
- Quarks in nuclear matter
- Local / extended wetting
- Containment / epidemic
- Non-propagation / propagation
- Confinement / non-confinement
Percolation model

$p=0.4$

$p=0.6$

$p=0.8$
Percolation clusters at different occupation probability

- $p = 0.55$
- $p = 0.58$
- $p = 0.59274621$
- $p = 0.65$
Moments of cluster size distribution and Connectivity length:

\[ \chi \sim |p - p_c|^{-\nu} \]

\[ \xi \sim |p - p_c|^{-\nu} \]

Signature of a second order phase transition
Importance sampling MC:

The statistical average of a quantity $A$ is given by:

$$\langle A \rangle = \frac{1}{Z} \sum_s A_s \exp(-E_s/k_B T)$$

$$Z = \sum_s \exp(-E_s/k_B T)$$

Sum is over all possible states.

At high temperature, the Boltzmann probability goes to 1 and it leads to simple average.

However, in the low temperature limit, the system spends almost all its time sitting in the ground state or one of the lowest excited states.

An extremely restricted part of the phase space should contribute to the average and the sum should be over the few low energy states.

Picking up points randomly over the whole phase space is no good here.
Importance Sampling

It is a method that will lead us automatically in the important region of phase space.

How one could sample points preferentially from the region which is populated by the appropriate states? Such a method is realizable via the importance sampling MC scheme.

In such a scheme, a state \( s \) is picked up with a probability \( p_s \) proportional to the Boltzmann factor.

A MC estimator then can be defined for a subset of \( N \) microstates (much smaller than all possible states) as

\[
A_N = \frac{\sum_{s=1}^{N} A_s p_s^{-1} \exp(-E_s/k_BT)}{\sum_{s=1}^{N} p_s^{-1} \exp(-E_s/k_BT)}
\]

Note that \( p_s = p \) leads to simple sampling.
Importance Sampling

Instead of picking N states randomly, pick them with a probability $p_s$, the correct Boltzmann weight.

$$p_s = \exp\left(-\frac{E_s}{k_BT}\right)/Z$$

$$A_N = \frac{\sum_{s=1}^{N} A_s p_s^{-1} \exp\left(-\frac{E_s}{k_BT}\right)}{\sum_{s=1}^{N} p_s^{-1} \exp\left(-\frac{E_s}{k_BT}\right)}$$

$$A_N = \frac{1}{N} \sum_{s=1}^{N} A_s$$

This averaging is called ‘importance sampling’ average.

However, it is not an easy task to pick up states which appear with their correct Boltzmann weight.
Markov chain

It is possible to realize importance sampling with the help of Markov chain.

A Markov chain is a sequence of states each of which depends only on the preceding one.

\[ [n] \xrightarrow{\mathcal{H}} [m] \xrightarrow{\mathcal{H}} [o] \]

The transition probability \( W_{nm} \) from state \( n \) to \( m \) should not vary over time and depends only on the properties of current states \((n,m)\).

\[
W_{nm} \geq 0 \text{ and } \sum_m W_{nm} = 1
\]

The Markov chain has the property that average over a successive configurations

\[
\langle A \rangle_n = \langle A \rangle + O(1/\sqrt{n}) \quad \text{as } n \to \infty,
\]

if the states are weighted by Boltzmann factor.
Master equation:

At time $t$:

\[ p_n(t + 1) = p_n(t) + \sum_m [W_{mn}p_m(t) - W_{nm}p_n(t)] \]
Ergodicity

It should be possible in Markov process to reach any state of the system starting from any other state in long run.

This is necessary to achieve a state with its correct Boltzmann weight. Since each state $m$ appears with some non-zero probability $p_m$ in the Boltzmann distribution, and if that state is inaccessible from another state $n$, then the probability to find the state $m$ starting from state $n$ is zero. This is in contradiction with Boltzmann distribution which demands the state should appear with a probability $p_m$.

This means that there must be at least one path of non-zero transition probabilities between any two states.

One should take care in implementing MC algorithms that it should not violate ergodicity.
Detailed balance

In the steady state, the probability flux out of a state $n$ is equal to the probability flux into the state $n$ and the steady state condition is given by the "global balance" equation

$$\sum_m W_{nm} p_n(t) = \sum_m W_{mn} p_m(t).$$

Since $\sum_m W_{nm} = 1$, one has

$$p_n(t) = \sum_m W_{mn} p_m(t).$$

However, the above condition is not sufficient to guarantee that the probability distribution will tend to $p_n$ after a long evolution of a Markov chain starting from any state.

Detailed balance:

$$W_{nm} p_n(t) = W_{mn} p_m(t).$$
Metropolis algorithm

\[ p_n = \frac{1}{Z} e^{-E_n/k_BT} \quad \text{and} \quad p_m = \frac{1}{Z} e^{-E_m/k_BT} \quad \text{where} \quad Z = \sum_s e^{-E_s/k_BT} \]

Since the canonical partition function \( Z \) is extremely difficult to calculate exactly, \( p_n \) or \( p_m \) is not known exactly.

\[ \frac{W_{nm}}{W_{mn}} = \frac{p_m}{p_n} = e^{-E_{nm}/k_BT} \quad \text{where} \quad E_{nm} = E_m - E_n \]

Algorithm:

\[ p_{nm} = \begin{cases} e^{-E_{nm}/k_BT} & \text{if} \ E_{nm} > 0 \\ 1 & \text{if} \ E_{nm} \leq 0 \end{cases} \]

Using this transition probability one can now generate a series of states with their right Boltzmann weight.
How to implement? \[ P_{nm} = e^{-E_{nm}/k_BT} \text{ for } E_{nm} > 0? \]

Call a random number \( r = [0,1] \) uniformly distributed between 0 and 1.

If \( r \leq P_{nm} \), change the state otherwise do not change the state.
Application to Ising Model:

\[ \mathcal{H} = -J \sum_{i,j} s_i s_j - \mu B \sum_i s_i \]

Set \( B = 0 \).

The critical temperature:

\[ 2 \tanh^2 \frac{2J}{k_B T_c} = 1, \quad k_B T_c / J = 2.269185. \]

Scaling:

\[ t = \frac{T - T_c}{T_c} \]

\[ M \sim (-t) \beta, \quad \chi \sim |t|^{-\gamma}, \quad C_B \sim |t|^{-\alpha}, \quad \xi \sim |t|^{-\nu} \]

\[ M \sim B^{1/\delta} \]

Critical exponents:

\[ \alpha = 0 \text{ (logarithmic), } \beta = 1/8, \gamma = 7/4, \nu = 1. \]

\[ \beta(\delta - 1) = \gamma \quad \Rightarrow \quad \delta = 15 \]

Applications:

- Para to ferromagnetic transition
- Liquid gas transition
- Binary alloys
Metropolis algorithm for spin-1/2 Ising model

1. Choice of initial state: Take a \((L \times L)\) square lattice. A random spin configuration is taken as initial state assigning \(s = +1\) (or \(-1\)) to the lattice sites randomly with probability 0.5 corresponding to up spin (or down spin) respectively.

2. Calculation of energy change: Choose a site \(j\) randomly from the sites with \(s = +1\) or \(-1\) which correspond to the initial state of the \(j\)th spin. The final state is the state with the overturned spin at the \(j\)th site. The change in energy \(\Delta E = E_f - E_i\) for the sing Hamiltonian without external field is given by

\[
\Delta E = 2J s_j \sum_{k=1}^{4} s_k
\]

where \(s_j\) is the initial spin state and \(s_k\)s are the nearest neighbour spins.

3. Accept or reject the spin flip: If \(\Delta E < 0\) flip the spin, else call a random number \(r\) uniformly distributed between 0 and 1. If \(r \leq \exp(-\Delta E/k_B T)\) flip the spin, otherwise do not flip it.

4. Iteration: Go to the next site and start from (2). Repeat the process for all spins.
Calculation of change in energy:  

\[ E = -J \sum_{\langle ij \rangle} s_i s_j \]

<table>
<thead>
<tr>
<th>Energy</th>
<th>Spin Configuration</th>
<th>Change</th>
<th>New Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>( E = -4J )</td>
<td>↑ ↑ ↑ ↑</td>
<td>( \Rightarrow )</td>
<td>( E = 4J )</td>
</tr>
<tr>
<td>( \Delta E = 8J )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( E = -2J )</td>
<td>↓ ↑ ↑ ↑</td>
<td>( \Rightarrow )</td>
<td>( E = 2J )</td>
</tr>
<tr>
<td>( \Delta E = 4J )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( E = 0 )</td>
<td>↓ ↑ ↑ ↑</td>
<td>( \Rightarrow )</td>
<td>( E = 0 )</td>
</tr>
<tr>
<td>( \Delta E = 0 )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( E = 2J )</td>
<td>↓ ↑ ↑ ↑</td>
<td>( \Rightarrow )</td>
<td>( E = -2J )</td>
</tr>
<tr>
<td>( \Delta E = -4J )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( E = 4J )</td>
<td>↓ ↑ ↓ ↓</td>
<td>( \Rightarrow )</td>
<td>( E = -4J )</td>
</tr>
<tr>
<td>( \Delta E = -8J )</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Spin configuration:

Figure 1.13: Spin configuration at temperature $k_B T/J = 2.0$, corresponds to $T < T_c$ ($k_B T_c/J \approx 2.27$). $t$ is the MC time step per spin.
Spin configuration:

Figure 1.14: Spin configuration at $k_B T/J = 3.0$, corresponds to $T > T_c$ ($k_B T_c/J \approx 2.27$). $t$ is the MC time step per spin.
Equilibrium:
Equilibrium configurations

Figure 1.16: Equilibrium spin configuration at different temperatures around $T_c$ after 8000 Monte Carlo time step per spin.
Magnetization and Susceptibility:

The magnetization $m$ per spin in a state $n$ is given by

$$M_n = \sum_{i=1}^{L^2} s_i^n$$

Since only one spin $(k)$ flips at a time in the Metropolis algorithm, so the change in magnetization is given by

$$\Delta M = M_m - M_n = \sum_{i=1}^{L^2} s_i^m - \sum_{i=1}^{L^2} s_i^n = s_k^m - s_k^n = 2s_k^m \quad s_k^n = -s_k^m$$

One can then calculate the magnetization at the beginning of the simulation and then use the following equation

$$M_m = M_n + \Delta M = M_n + 2s_k^m$$

One could calculate the per spin susceptibility as

$$\chi_T = \frac{1}{k_B T} [\langle m^2 \rangle - \langle m \rangle^2]$$
Magnetization and Susceptibility:

On a 2d square lattice of size $100 \times 100$. 

\[ K_B T/J = 2.269 \]
Energy and specific heat:

\[ E_n = -J \sum_{\langle ij \rangle} s_i^n s_j^n \]  

Set \( J = 1 \)

The energy difference in going from state \( n \) to state \( m \) is:

\[ \Delta E = E_m - E_n \]

\[ E_m = E_n + \Delta E. \]

One may calculate the per spin specific heat as:

\[ c_V = \beta^2 (\langle E^2 \rangle - \langle E \rangle^2) \]
Energy and specific heat:
Time auto correlation

In order to take average of a physical quantity, we have presumed that the states over which the average has been made are independent. Thus to make sure that the states are independent, one needs to measure the “correlation time” $\tau$ of the simulation. The time auto correlation $C_m(t)$ of magnetization is defined as

$$C_m(t) = \sum_{t'} [m(t') - \langle m \rangle][m(t' + t) - \langle m \rangle] = \sum_{t'} m(t')m(t' + t) - \langle m \rangle^2.$$  

The auto correlation $C_m(t)$ is expected to fall off exponentially at long time

$$C_m(t) \sim e^{-t/\tau}.$$  

Thus, at $t = \tau$, $C_m(t)$ drops by a factor of $1/e$ from the maximum value at $t = 0$. For independent samples, one should draw them at an interval greater than $\tau$. In most of the definitions of statistical independence, the interval turns out to be $2\tau$. 
Time auto correlation

Critical slowing down
Cluster update algorithms

Cluster update algorithms are the most successful global update methods in use. These methods update the variables globally, in one step, whereas the standard local methods operate on one variable at a time.

A global update can reduce the autocorrelation time of the update and thus greatly reduce the statistical errors.

Two famous algorithms: Swendsen-Wang and Wolff

Beginning with an arbitrary configuration $s_i$, one SW cluster update cycle is:

1. Inspect all nn-states $s_i, s_j$. If $s_i = s_j$, create a bond between sites $i, j$ with probability $p = 1 - \exp(-2\beta)$ (otherwise no bond).
2. Construct clusters = sets of points connected by bonds.
3. Set each cluster to a random value $\pm 1$. 

Principle: do the cluster decomposition as in S-W, but invert (‘flip’) only one randomly chosen cluster! In practice:

1. Choose random site $i$.
2. Study neighbouring sites $j$. If $s_j = s_i$, join site $j$ to cluster with probability $p = 1 - \exp(-2\beta)$.
3. Repeat step 2 for site $j$, if it was joined to the cluster. Keep on doing this as long as the cluster grows.
4. When the cluster is finished, invert the spins which belong to it.
Auto correlation in cluster algorithm
Effect of finite size

Critical temperature:

\[ T_c(L) = T_c + aL^{-1/\nu} \]

Thermodynamic quantities:

\[ M = L^{-\beta/\nu} \phi(\epsilon L^{1/\nu}) \]
\[ \chi = L^{\gamma/\nu} \tilde{\chi}(\epsilon L^{1/\nu}) \]
\[ C_v = L^{\alpha/\nu} \tilde{C}_v(\epsilon L^{1/\nu}) \]

At \( T = T_c \):

\[ M = L^{-\beta/\nu} \]
\[ \chi = L^{\gamma/\nu} \]
\[ C_v = L^{\alpha/\nu}. \]
Wang-Landau

Instead of configuration sum, one may sum over the energy levels with right \( g(E) \).

\[
Z = \sum_{\text{configurations}} e^{-E/k_B T} = \sum_{E} g(E) e^{-E/k_B T},
\]

Again it is difficult to calculate value of \( g(E) \) for large systems.

In conventional MC, one directly generate the canonical distribution:

\[
P(E, T) = g(E) e^{-E/k_B T}
\]

In WL, \( g(E) \) is estimated directly and accurately by performing a random in energy space. Change the states with a probability that depends on the density of states as given below:

\[
p(E_1 \rightarrow E_2) = \min \left( \frac{g(E_1)}{g(E_2)}, 1 \right)
\]
Kinetic Monte Carlo

With MD we can only reproduce the dynamics of the system for \( \leq 100 \) ns. Slow thermally activated processes, such as diffusion, cannot be modeled. Metropolis Monte Carlo samples configurational space and generates configurations according to the desired statistical-mechanics distribution. However, there is no time in Metropolis MC and the method cannot be used to study evolution of the system or kinetics. An alternative computational technique that can be used to study kinetics of slow processes is the kinetic Monte Carlo (kMC) method.

As compared to the Metropolis Monte Carlo method, kinetic Monte Carlo method has a different scheme for the generation of the next state. The main idea behind kMC is to use transition rates that depend on the energy barrier between the states, with time increments formulated so that they relate

In Metropolis MC methods we decide whether to accept a move by considering the energy difference between the states. In kMC methods we use rates that depend on the energy barrier between the states.
Hybrid Monte Carlo

A hybrid Monte Carlo move involves running a molecular dynamics simulation for a fixed length of time. The configuration at the end of this molecular dynamics run is accepted or rejected based on the Metropolis criterion, where $E(n)$ is the potential energy before the molecular dynamics run, and $E(m)$ is the potential energy at the end of the run. Thus, molecular dynamics can be used to propose a Monte Carlo move that involves the collective motion of many particles. An advantage that this method has over conventional molecular dynamics simulations is that large time steps may be used in the hybrid Monte Carlo move.

![Diagram showing hybrid Monte Carlo move with states and energy conversion](Image)
Thank you