Monte Carlo
A general overview of methods, theory and practice

Alexandros Sopasakis
Department of Mathematics
UNC Charlotte
Lectures

• Lecture 1- Background and theoretical/analytical development of the Monte Carlo method

• Lecture 2- Numerical simulation practices and common techniques used in modern modeling applications

• Lecture 3- A research project perspective: application to traffic flow
Historical background and general development

Monte Carlo simulation idea is older than the computer

MC method originally used to estimate pathological integrals.

Comte de Buffon showed in 1777 that: \( \frac{M}{N} \overset{N \to \infty}{\to} \frac{2l}{\pi d} \) for \( d > l \)

Laplace 1820, Calculating \( \pi \) .... the paper – needle experiment

Wolf 1850 (3.1596)

Smith 1855 (3.1553)

Lazzarini 1901 (3.14\textbf{15929})
Historical background and general development

William Thomson also describes an early 1901 MC method for the calculation of the motion of molecules undergoing collision in a gas (credit William Anderson).

It is believed that the first real application of the statistical sampling method was undertaken by Enrico Fermi in the 1930s.

Ulam, Metropolis and von Neumann reinvented Fermi’s statistical sampling methods around 1947.

Nicolas Metropolis named the statistical sampling method used at the time Monte Carlo in a paper published in 1949.

Dorrie in 1965 solves the equivalent of “Buffon’s needle” example using MC methods.
Beliefs / facts about Monte Carlo

“The only good Monte Carlo is a dead Monte Carlo” Trotter and Tukey 1954

“Anyone who considers arithmetic methods of producing random digits is, of course, in a state of sin” John Von Neumann, 1951

• Simple Monte Carlo – The direct modeling of a random process (queueing problems)

• Sophisticated Monte Carlo – Methods which recast deterministic problems in probabilistic terms
Monte Carlo Simulation: an archetypical example

Integrating pathological functions:

\[ f(x) = \sin^2\left(\frac{1}{x}\right) \]

Note: The integral \[ I(y) = \int_0^y \sin^2\left(\frac{1}{x}\right) dx \]

is bounded by \[ 0 \leq I(y) \leq y \]
Monte Carlo Simulation: an archetypical example

Approximating $I(y) = \int_0^y \sin^2(1/x) dx$

To use MC to approximate $I(y)$ we choose:
- random number $u$ distributed between 0 and $y$ and another
- random number $v$ between 0 and 1.

These will represent our horizontal and vertical coordinates on the plot of $f$.

The probability that this point $(u,v)$ will be below the line of the function $f$ is $I(y)/y$.

Thus for a large number $N$ of these random points we count the number $M$ of those bellow the line of the function $f$ and obtain,

$$I(y) = \lim_{N \to \infty} \frac{M}{N}$$

Alexandros Sopasakis
Some facts to put things into perspective

• One liter of air at standard temperature and pressure contains about $3 \times 10^{22}$ molecules (oxygen, nitrogen, carbon dioxide, etc...).

• The atmosphere of the earth contains $4 \times 10^{21}$ liters of air or about $1 \times 10^{44}$ molecules (all moving around and colliding).

• Clearly it is not feasible to solve Hamilton’s equations for these systems. Too many equations

• Surprisingly however the macroscopic properties of air or gas are well-behaved and many times predictable

• We can conclude that there must be something special about the behavior of the solutions of these many equations which “averages” out and gives us a predictable behavior for the system.

This is where statistical mechanics is employed!

Alexandros Sopasakis
Facts About Simulating Large Systems

Let’s consider a system which we wish to solve using a computational method. The most straightforward approach is to put this system into a lattice. Let’s assume a (very) small system of 10x10 (2 dimensional) lattice arrangement

-1 1 1 1 1 -1 -1 -1 -1 -1
-1 1 -1 -1 1 -1 -1 1 1 -1
1 1 1 -1 -1 -1 -1 -1 1 1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1
1 -1 1 -1 -1 1 1 1 -1 -1
1 -1 1 -1 -1 1 -1 1 -1 1
1 -1 1 -1 1 -1 1 1 -1 -1
1 1 1 1 1 1 1 1 1 1
1 -1 -1 -1 1 -1 -1 -1 -1 -1

-1 1 1 1 1 -1 -1 -1 -1 -1
-1 1 -1 -1 1 -1 -1 1 1 -1
1 1 1 -1 -1 -1 -1 -1 1 1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1
1 -1 1 -1 -1 1 1 1 -1 -1
1 -1 1 -1 -1 1 -1 1 -1 1
1 -1 1 -1 1 -1 1 1 -1 -1
1 1 1 1 1 1 1 1 1 1
1 -1 -1 -1 1 -1 -1 -1 -1 -1

• Suppose that each spin on every lattice node is allowed to take only two values: +1, -1
• Thus this small system has a total $2^{100} = 33,554,432$ possible states
• **Fact:** even for a small problem it would be impossible to visit all states of the system!
Facts About Simulating Large Systems

Now imagine the situation even if we only wish to model just one liter of oxygen.

Furthermore, molecules change states at a rate of \(10^9\) collisions per second. At this rate it would take \((10^{10})^{23}\) times the lifetime of the universe for our liter of oxygen to move through all its possible states.

It is therefore impossible to just visit all states in order to solve such a system! **What do we do then?**

Alexandros Sopasakis
Lifetime of the universe: 13.7 billion years
A general modeling paradigm for large interacting systems

Our Hamiltonian system is fed by a thermal reservoir

The thermal reservoir is an external system which acts as a source and sink of heat, constantly exchanging energy with our system.

Alexandros Sopasakis
A general modeling paradigm for large interacting systems

• In general we think of the thermal reservoir as a weak perturbation of our Hamiltonian system which we ignore when calculating the energy levels of our system.

• Effects of the reservoir can be incorporated in our calculations by assigning the system a rule by which it can change from one state to another. This rule gives the system its dynamics

• There are several different types of dynamics – to be listed later – which can describe the physics for our system
Outline of what will follow

1. Markov Chain Monte Carlo (MCMC)
   
   **Task:** sample from a given probability distribution \( \pi = (\pi(\mu))_{\mu \in \Sigma} \) where \( \Sigma \) denotes our state space

   **Idea:** construct a discrete-time Markov Chain with a known probability matrix

   \[
   \mathbf{P} = \begin{pmatrix}
   P(\mu \rightarrow \nu) \end{pmatrix}_{\mu, \nu \in \Sigma}
   \]

   having \( \pi \) as a stationary (invariant) distribution.

   We will:
   
   - Define the **transition probabilities** \( P(\mu \rightarrow \nu) = g(\mu \rightarrow \nu) \ A(\mu \rightarrow \nu) \)
   - Pick the **selection probabilities** \( g(\mu \rightarrow \nu) \) for a proposed move from \( \mu \) to \( \nu \)
   - Discuss optimal **acceptance ratios** \( A(\mu \rightarrow \nu) \)
   - Use ergodicity to ensure that it is the invariant distribution \( \pi \) we sample
   - Condition the Markov Chain to satisfy **detailed balance**.
2. Continuous Time Monte Carlo (CTMC) – Kinetic Monte Carlo

We similarly can construct a continuous-time Monte Carlo Chain with invariant measure $\pi$

Note that

- The random jump time is known (exponentially distributed, etc.) and defines the time step $dt$ of the simulator
- No rejected moves!
- Drawback: algorithmically difficult to implement and computer memory intensive.
- CTMC is “real” dynamics
Markov Processes-building the Markov Chain

For purposes of MC methods a Markov Process is a mechanism by which a system is taken from one state to another in a random fashion. The probability of changing from state $\mu$ to state $\nu$ is called the transition probability and is denoted by $P(\mu \rightarrow \nu)$. Transition probabilities are responsible for building the Markov Process.
Markov Processes-building the Markov Chain

*Transition probabilities* must satisfy three conditions:

- Stay constant over time
- Depend only on the current states $\mu$, $\nu$ (*memoryless* property), using the Markov property: suppose a Markov sequence $X_n$ takes the values $a_1...a_N$:
  \[
P(x_n = a_{i_n} \mid x_{n-1} = a_{i_{n-1}}, ..., x_1 = a_{i_1}) = P(x_n = a_{i_n} \mid x_{n-1} = a_{i_{n-1}})
  \]
- $\sum_{\nu} P(\mu \rightarrow \nu) = 1$

Note that it is possible to go from state $\mu$ back to the same state $\mu$ with a non-zero probability.

During a MC simulation we will use this mechanism to generate a **Markov Chain**.

*Main idea* will be to run this mechanism long enough so that new states will appear with probabilities given by their corresponding *Boltzmann distribution*. When that happens we say that the system has **equilibrated**.
Some useful definitions

1. We say that states $\nu, \mu \in \Sigma$ communicate if

$$P(\nu \rightarrow \mu) > 0, \quad P(\mu \rightarrow \nu) > 0$$

If all states communicate the Markov chain is called irreducible.

Alexandros Sopasakis
Some useful definitions

2. We define the **period** of a state \( \nu \) to be the greatest common divisor of all \( k \) greater than or equal to 1.

\[
P(X_{t+k} = \nu \mid X_t = \nu) > 0
\]

A Markov chain is called **aperiodic** if each state has period \( k=1 \).

3. If state \( \nu \in \Sigma \) is **revisited** with probability 1 at some finite time then it is called **recurrent**; otherwise the state is called **transient**.
System description

We assume that our systems starts at a given state \( \mu \) and **define the rates** \( P(\mu \rightarrow \nu) \). Thus \( P(\mu \rightarrow \nu) \Delta t \) is the probability that is will be in state \( \nu \) at time \( \Delta t \) later.

We also define a set of **weights** \( W_\mu(t) \) which represent the probability that the system will be in state \( \mu \) at time \( t \).

The system is described by the following **Master Equation** which is a rule for the evolution of the weights \( W_\mu(t) \) in terms of the rates \( P(\mu \rightarrow \nu) \):

\[
\frac{dW_\mu}{dt} = \sum_\nu \left[ w_\nu(t) P(\nu \rightarrow \mu) - w_\mu(t) P(\mu \rightarrow \nu) \right]
\]

subject to \( \sum_\mu W_\mu(t) = 1 \) for all \( t \)

Statistical mechanics deals with these weights and represent our entire knowledge of the state of the system.

Alexandros Sopasakis
Some statistical mechanics concepts.

Equilibrium:

Define the **equilibrium probabilities**:

$$ p_\mu = \lim_{t \to \infty} w_\mu(t) $$

**Boltzmann distribution** – [Gibbs, 1902]:

$$ p_\mu = \frac{1}{Z} e^{-E_\mu / kT} $$

$Z$ is known as the **partition function**.

$$ Z = \sum_\mu e^{-E_\mu / kT} = \sum_\mu e^{-\beta E_\mu} $$

Alexandros Sopasakis
In terms of measuring quantities of interest, once we obtain the solution for $W_\mu(t)$ we can also obtain information for quantities of interest, say $Q$,

$$\langle Q \rangle = \sum_\mu Q_\mu w_\mu(t)$$

Remember: at equilibrium $\lim_{t \to \infty} w_\mu(t) = p_\mu = \frac{1}{Z} e^{-\beta E_\mu}$ thus,

$$\langle Q \rangle = \frac{\sum_\mu Q_\mu e^{-\beta E_\mu}}{\sum_\mu e^{-\beta E_\mu}} \text{ for ALL states}$$

**Question:** Can we really calculate this quantity?  
**Answer:** Not really!
Example: suppose a small 3D system 10x10x10. This system is so small that is actually of no use towards realistic predictions. Still, this system would have a total of $2^{1000}=10^{300}$ states.

What is currently possible? It is possible, in a very fast computer, to sample $10^{10}$ states given a few hours. In other words we would only sample 1 in every $10^{290}$ states.

Clearly not possible to sample over all states that a system may visit...

Furthermore, to make things worst: in cases of low temperature very few states are responsible for the majority of the behavior of this system.
How Monte Carlo Works:

Monte Carlo techniques work by selectively sampling only a small finite subset of all possible states....

Question: is it enough to just sample a finite number, say M, of those states?

\[
\langle Q \rangle \approx Q_M := \frac{\sum_{\mu=1}^{M} Q_{\mu} e^{-\beta E_{\mu}}}{\sum_{\mu=1}^{M} e^{-\beta E_{\mu}}} \quad \text{BAD IDEA}
\]

Answer: Yes, but not exactly as given in the formula above!

Questions:
- But which M states should we choose?
- Do any M states work?
- How do we choose the important few states which are responsible for the majority of the behavior for this system?

Answer: Importance Sampling. That is how Monte Carlo works!

Alexandros Sopasakis
Importance Sampling  
Which states are the most important?

What is importance sampling?
An idea: pick each state $\mu$ based on its Boltzmann probability  

$$p_{\mu} = \frac{e^{-\beta E_{\mu}}}{Z}$$

Remember we wish to obtain  

$$\langle Q \rangle = \sum_{\mu} Q_{\mu} w_{\mu}(t) = \frac{\sum_{\mu} Q_{\mu} e^{-\beta E_{\mu}}}{\sum_{\mu} e^{-\beta E_{\mu}}}$$

The idea: propose the estimator  

$$Q_M = \frac{\sum_{\mu=1}^{M} Q_{\mu} p_{\mu}^{-1} e^{-\beta E_{\mu}}}{\sum_{\mu=1}^{M} p_{\mu}^{-1} e^{-\beta E_{\mu}}} = \frac{1}{M} \sum_{\mu=1}^{M} Q_{\mu}$$

Note that $Q_M$ is a good estimator for $\langle Q \rangle$ since in fact  

$$Q_M \to \langle Q \rangle \text{ as } M \to \text{ all states}$$

Question: still... how do we really pick these states with their correct Boltzmann probabilities?
Ergodicity-Reaching all states

Note that states do not all need to have a non-zero probability assigned to them. In practice most transition probabilities are set to zero as long as there are exist paths which connect any state to any other one.

Our Markov Chain must be able to reach all states of the system – otherwise we will not be able to produce new states with their correct Boltzmann probabilities. Reaching all states from any starting state is the property of ergodicity.
Stationary distributions, long-time behavior and ergodicity

Let \( \pi = \pi(x) \geq 0, \sum_{x \in \Omega} \pi(x) = 1 \) be a probability distribution.

Stationary (invariant) if \( \pi P^n = \pi \) for all \( n \); true if \( \pi P = \pi \)

- If \( X_t \) is a \textit{aperiodic} and \textit{irreducible} then
  \[
  \lim_{n \to \infty} p^n(x, y) = \pi(y) \text{ for all } x \quad (\text{Ergodicity})
  \]

- If \( X_t \) is just \textit{irreducible} then we have:
  \[
  \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} p^n(x, y) = \pi(y) \text{ for all } x \quad (\text{Weak Ergodicity})
  \]
Detailed Balance-ensuring equilibrium distribution

Detailed balance is responsible for ensuring that it is the Boltzmann probability which we generate, instead of any other distribution. This is equivalent in fact to saying that the system is in equilibrium.

We can achieve this by simply allowing the rates by which the system transitions into and out of state $\mu$ to be equal:

$$\sum_\nu p_\mu P(\mu \rightarrow \nu) = \sum_\nu p_\nu P(\nu \rightarrow \mu)$$

which, since $\sum_\nu P(\mu \rightarrow \nu) = 1$ simplifies to

$$p_\mu = \sum_\nu p_\nu P(\nu \rightarrow \mu)$$

This condition however is not sufficient to guarantee that we attain the desired probability distribution if we run the process long enough.
Example:
The probability $W_{\nu}(t+1)$ of being in state $\nu$ at time $t+1$ is given by,

$$w_{\nu}(t+1) = \sum_{\mu} P(\mu \to \nu)w_{\mu}(t)$$

(Exercise: derive the above from the master equation)

In matrix notation this becomes $w(t + 1) = P \cdot w(t)$. It is possible to obtain either

**Simple Equilibrium:** $w(\infty) = P \cdot w(\infty)$

**Dynamic Equilibrium:** $w(\infty) = P'' \cdot w(\infty)$ (limit cycle)

Additional condition guaranteeing the desired probability distribution is achieved:

$$p_{\mu}P(\mu \to \nu) = p_{\nu}P(\nu \to \mu)$$

Note:
- this also enforces **Detailed Balance** (Exercise)
- this condition also forbids the appearance of limit cycles (Exercise).
Once the limit cycles are removed we can prove that as $t \rightarrow \infty$ the $w(t) \rightarrow$ exponentially towards the eigenvector corresponding to the largest eigenvalue of the stochastic matrix $P$.

(Exercise: prove the statement above.)

Note also that from $p_\mu P(\mu \rightarrow \nu) = p_\nu P(\nu \rightarrow \mu)$ we can get

$$\frac{P(\mu \rightarrow \nu)}{P(\nu \rightarrow \mu)} = \frac{p_\nu}{p_\mu}$$

Which, given that $p_\mu = Z^{-1} e^{-\beta E_\mu}$ gives

$$\frac{P(\mu \rightarrow \nu)}{P(\nu \rightarrow \mu)} = e^{-\beta (E_\mu - E_\nu)}$$

As a result, one possible choice for the transition probabilities would be

$$P(\mu \rightarrow \nu) \propto e^{-\beta (E_\mu - E_\nu)/2}$$

(Exercise: verify that this is indeed a possible choice which satisfies (*) above)

This however is not really a very good choice....

Alexandros Sopasakis
Acceptance Ratios or
How to construct an **efficient** MC algorithm

Let’s start by breaking up and *re-defining* the transition probability as follows,

\[ P(\mu \to \nu) = g(\mu \to \nu) A(\mu \to \nu) \]

where

- \( g(\mu \to \nu) \) represents the **selection probability** \((0 \leq g \leq 1)\)
- \( A(\mu \to \nu) \) is the **acceptance ratio** \((0 \leq A \leq 1)\)

Our task will be to choose new states with higher acceptance ratios:

\[ \frac{P(\mu \to \nu)}{P(\nu \to \mu)} = \frac{g(\mu \to \nu) A(\mu \to \nu)}{g(\nu \to \mu) A(\nu \to \mu)} \]

Note that the ratio \( 0 \leq \frac{A(\mu \to \nu)}{A(\nu \to \mu)} < \infty \)

while \( g(\mu \to \nu), g(\nu \to \mu) \) can take any values

Alexandros Sopasakis
Good MC algorithm practices

We wish to maximize our chance of accepting a new state $A(\mu \rightarrow \nu)$ (near 1) while at the same time satisfying the previous equation. This will ensure that at least in one direction the acceptance ratio will be as high as possible.

**Ideal algorithm:** new states selected with the correct transition probabilities while the acceptance ratio is always equal to 1.

**Good algorithm:** again correct transition probabilities and acceptance ratios close to 1.
An example on picking acceptance ratios and selections probabilities: The Ising Model

An Ising model consists of micromagnets (dipoles) on a lattice. The magnetic spins are only allowed to take two values -1,1 which represent the magnetic charge of the dipole. If the lattice has N nodes then the system can be in any of $2^N$ possible states at any given time.
The Ising Model

An Ising model consists of micromagnets (dipoles) on a lattice. The magnetic spins are only allowed to take two values -1,1 which represent the magnetic charge of the dipole. If the lattice has N nodes then the system can be in any of $2^N$ possible states at any given time.

The energy of this system is given by its Hamiltonian:

$$H = -J \sum_{<i,j>} s_i s_j - h \sum_i s_i$$

where J is the interaction energy

<i,j> represent nearest neighbour spins

h is a given external field (for now we will assume h=0)

We typically wish to examine the average magnetization m as well as the specific heat c for this system

Alexandros Sopasakis
The Metropolis algorithm for single spin-flip Ising Model dynamics

In the Metropolis algorithm all selection probabilities are chosen to be equal. In fact one such natural choice is to let them be $1/N$ where $N$ represents the total number of nodes in the lattice

$$g(\mu \rightarrow \nu) = \frac{1}{N}$$

Revisiting now our ratio of transition probabilities we obtain,

$$\frac{P(\mu \rightarrow \nu)}{P(\nu \rightarrow \mu)} = \frac{g(\mu \rightarrow \nu)A(\mu \rightarrow \nu)}{g(\nu \rightarrow \mu)A(\nu \rightarrow \mu)} = \frac{A(\mu \rightarrow \nu)}{A(\nu \rightarrow \mu)}$$

Recall that

$$\frac{P(\mu \rightarrow \nu)}{P(\nu \rightarrow \mu)} = e^{-\beta(E_\mu - E_\nu)}$$

and

$$A(\mu \rightarrow \nu) = A_0 e^{\frac{1}{2} \beta (E_\mu - E_\nu)}$$

**Exercise:** how big a constant $A_0$ can we choose to maximize the acceptance ratio?

Alexandros Sopasakis
Alexandros Sopasakis
The Metropolis algorithm (continuous)

\[ A(\mu \to \nu) = A_0 e^{-\frac{1}{2} \beta (E_\mu - E_\nu)} \]

Thus for a lattice node with four neighbors the difference of $E_\mu - E_\nu$ can be as big as $8J$. 

Alexandros Sopasakis
The Metropolis algorithm (continuous)

Thus the acceptance ratio

\[ A(\mu \rightarrow \nu) = A_0 e^{\frac{1}{2} \beta (E_\mu - E_\nu)} \]

can be as big as

\[ A(\mu \rightarrow \nu) \leq A_0 e^{4\beta J} \]

Therefore in order to maximize the acceptance ratio \( A(\mu \diamond \nu) \) (close to 1) we choose

\[ A_0 \leq e^{-4\beta J} \]

Thus the best possible acceptance ratio for this type of lattice interactions is

\[ A(\mu \rightarrow \nu) = e^{\frac{1}{2} \beta (E_\mu - E_\nu + 8J)} \]

Alexandros Sopasakis
What is wrong with this picture?

Plot of $A(\mu \rightarrow \nu) = e^{-\frac{1}{2} \beta (E_\mu - E_\nu + 8J)}$
To make this more efficient we choose a different strategy:

$$A(\mu \rightarrow \nu) = \begin{cases} 
e^{-\beta(E_\nu - E_\mu)} & \text{if } E_\nu > E_\mu \\ 1 & \text{otherwise} \end{cases}$$

This is the Metropolis algorithm!

Alexandros Sopasakis
Equilibrium and Autocorrelation

One way to obtain reliable information, such as calculating an expected value, about quantities of interest is to obtain

$$Corr(Q)(t^*) = \int (Q(t + t^*) - \langle Q \rangle)(Q(t) - \langle Q \rangle) dt$$

This also reveals the correct MC time $\tau$ necessary for equilibration.

As a rule of thumb we can use the resulting MC simulation and collect independent statistics for our parameters of interest every $2\tau$. 

Alexandros Sopasakis
Books:

More General Books:
E. Segre, 1980, From X-Rays to Quarks, Freeman, San Fransisco

Most Important Scientific Papers:
Books and other useful references

Books:

More General Books:
E. Segre, 1980, From X-Rays to Quarks, Freeman, San Francisco

Most Important Scientific Papers:
The estimator for $\langle Q \rangle$

The true quantity is,

$$\langle Q \rangle = \sum_{\mu} Q_{\mu} w_{\mu}(t) = \frac{\sum_{\mu} Q_{\mu} e^{-\beta E_{\mu}}}{\sum_{\mu} e^{-\beta E_{\mu}}}$$

The estimator is,
Stochastic systems are described by a master equation

• Master Eq.: ....
  – ... is the probability that the system is in configuration
  – Wba is the transition probability per unit time going from configuration b to a

• Detailed balance at equilibrium: ....

• Solving the master eq.
  – ODE solvers could be used but the number of states is huge
    • For the simple network
  – Only some of these states are important
    • MC simulations take advantage of this fact and provide the exact, stochastic solution to a time-dependent master equation

Alexandros Sopasakis
Monte Carlo Methods

• Equilibrium (metropolis) MC algorithms
  – rate function $W_{\alpha\beta}$ has no connection to dynamics
  – applicable only to systems at equilibrium/well-mixed systems.

• Kinetic or dynamic MC algorithms
  – $W_{\alpha\beta}$ describe the microscopic dynamics and are obtained from experiments or lower length scale simulation (MD, DFT, TST)
  – Spatially well-mixed
  – Spatially distributed
  – Arrhenius, Glauber, Kawasaki, ... dynamics
Challenges in conducting well-mixed Monte Carlo simulations

Alexandros Sopasakis
Background: Markov processes and Monte Carlo Simulation

- **Discrete-time Markov Chains:**
  Stochastic process \( \{X_t : t = 1,2,3,...,n,....\} \)
  taking finitely many values on the state space \( \Sigma : X_t = x \in \Sigma \)
  \[ P(X_{t+1} = y \mid X_t = x, X_{t-1} = x_{t-1},..., X_1 = x_1) = \]
  Markov property:
  \[ P = \{p(x, y)\}_{x,y \in \Sigma} \]

  Transition probability matrix:
  \[ p(x, y) := P(X_{t+1} = x \mid X_t = y) \]

  \( p(x, y) \geq 0 \), and \( \sum_{y \in \Sigma} p(x, y) = 1 \)

  Note:
  \[ P^n = \{p^n(x, y)\}_{x,y \in \Sigma} \]

  n-step transition probability matrix:
  \[ p^n(x, y) := P(X_{t+n} = y \mid X_t = x) \]
Evolution of the probability distribution:

\[ p^{2}(x, y) = P(X_2 = y \mid X_0 = x) \]
\[ = \sum_{z} P(X_2 = y, X_1 = z \mid X_0 = x) \]
\[ = \sum_{z} P(X_2 = y \mid X_1 = z, X_0 = x) P(X_1 = z \mid X_0 = x) \]
\[ = \sum_{z} p(x, z) p(z, y) \]

Chapman-Kolmogorov equation:  \( P^{n} = P x P x P x \ldots x P \) (n-fold product) and \( P^{n+m} = P^{n} P^{m} \)

\[ p^{n+m}(x, y) = \sum_{z \in \Omega} p^{n}(x, z) p^{m}(z, y) \]

Alexandros Sopasakis
Construction of sample paths

Assume $\Sigma = \{x_1, x_2, \ldots, x_m\}$; if $X_t = x$

$$X_{t+1} = \begin{cases} 
    x_1 & \text{if } U \leq p(x, x_1) \\
    x_2 & \text{if } p(x, x_1) < U \leq p(x, x_1) + p(x, x_2) \\
    \vdots & \vdots \\
    x_m & \text{if } p(x, x_1) + p(x, x_2) + \ldots + p(x, x_{m-1}) < U \leq 1 
\end{cases}$$

where $U$ is a uniformly distributed random variable in $(0,1)$.

Example 1: random walk on a lattice $\{\xi_i : i = 1, 2, \ldots\}$ i.i.d. random variables with $P(\xi_i = \pm 1) = p^\pm$, $p^+ + p^- = 1$

Define:

$$X_n = \sum_{i=1}^{n} \xi_i$$

Then $X_n$ is a Markov chain, i.e. satisfies the Markov property (Exercise 1)
Continuous-time Markov Chains

Stochastic process ... taking finitely many values on the state space ....

Markov property:

Stationary transition probability matrix ...

Note ....

Chapman-Kolmogorov equation: ...
Continuous vs Discrete time Markov Chains

**Discretization**: The C-K relation implies that .... Is a discrete-time Markov Chain with transition probability matrix ..... Hence ... is a discretization of the continuous time Markov Chain ...

**Residence time**: ... time spend by the process ... ; random waiting time between consecutive jumps.

Markov Property implies:

i.e. ... is a memoryless distribution! Hence

thus

Alexandros Sopasakis
Generators

Observables: ....

Then

Where $L$ is the \textit{generator} of the process

The generator $L$ completely defined the Markov process...
Example 2. Continuous-time random walk on a lattice... i.i.d. random variables with

...i.i.d. exponentially distributed non-negative random variables (... also independent of ....) with

Consider the sequence of pairs

(... is a Markov Chain – see Example 1 – also the pair of a Markov Chain) and define

Then ... is a continuous Markov chain, i.e. satisfies the Markov property (Exercise 2).

Generator:

Alexandros Sopasakis
Example 3. Birth-death processes

Markov process with random data, ....

Invariant (stationary) distribution: .... Hence

A stronger condition is detailed balance:

Ergodicity: unique ... such that .... For all ...
Bistability

Noise does not increase monotonically with increasing temperature; it is larger within the bistable regime.

Example: Fluctuations in molecular electronics
Example: Effect of substrate roughness
Example: Cell interactions, understanding complex receptor dynamics