Markov Chains and Random Walks

Josep Díaz Maria J. Serna Conrado Martínez
U. Politècnica de Catalunya

RA-MIRI 2023–2024

Stochastic Process

- A stochastic process is a sequence of random variables $\{X_t\}_{t=0}^{n}$.
- Usually the subindex t refers to time steps and if $t \in \mathbb{N}$, the stochastic process is said to be discrete.
- The random variable X_t is called the state at time t.
- If $n < \infty$ the process is said to be finite, otherwise it is said infinite.
- A stochastic process is used as a model to study the probability of events associated to a random phenomena.

An example: Gambler's Ruin

Model used to evaluate insurance risks.

- You place bets of 1€. With probability p, you gain 1€, and with probability $q = 1 - p$ you loose your 1€ bet.
- You start with an initial amount of 100€.
- You keep playing until you loose all your money or you arrive to have 1000€.

- One goal is finding the probability of winning i.e. getting the 1000€.

Notice in this process, once we get 0€ or 1000€, the process stops.

An example: Gambler's Ruin

Model used to evaluate insurance risks.

- You place bets of 1€. With probability p, you gain 1€, and with probability $q = 1 - p$ you loose your 1€ bet.
- You start with an initial amount of 100€.
- You keep playing until you loose all your money or you arrive to have 1000€.

- One goal is finding the probability of winning i.e. getting the 1000€.

Notice in this process, once we get 0€ or 1000€, the process stops.

An example: Gambler's Ruin

Model used to evaluate insurance risks.

- You place bets of 1€. With probability p, you gain 1€, and with probability $q = 1 - p$ you loose your 1€ bet.
- You start with an initial amount of 100€.
- You keep playing until you loose all your money or you arrive to have 1000€.

- One goal is finding the probability of winning i.e. getting the 1000€.

Notice in this process, once we get 0€ or 1000€, the process stops.

Markov Chain

One simple model of stochastic process is the Markov Chain:

- Markov Chains are defined on a finite set of states (S), where at time t, X_t could be any state in S, together with by the matrix of transition probability for going from each state in S to any other state in S, including the case that the state X_t remains the same at $t + 1$.

- In a Markov Chain, at any given time t, the state X_t is determined only by X_{t-1}.
  memoryless: does not remember the history of past events,

Other memoryless stochastic processes are said to be Markovian.

An example: Gambler's Ruin

- You place bets of 1€. With probability p, you gain 1€, and with probability $q = 1 - p$ you loose your 1€ bet.
- You start with an initial amount of 100€.
- You keep playing until you loose all your money or you arrive to have 1000€.
- We have a state for each possible amount of money you can accumulate $S = \{0, 1, \dots, 1000\}$.
- The probability of losing/winning is independent on the state and the time, so this process is a Markov chain.
- Observe that the number of states is finite.

An example: Gambler's Ruin

- You place bets of 1€. With probability p, you gain 1€, and with probability $q = 1 - p$ you loose your 1€ bet.
- You start with an initial amount of 100€.
- You keep playing until you loose all your money or you arrive to have 1000€.
- We have a state for each possible amount of money you can accumulate $S = \{0, 1, \ldots, 1000\}$.
- The probability of losing/winning is independent on the state and the time, so this process is a Markov chain.
- Observe that the number of states is finite.

An example: Gambler's Ruin

- You place bets of 1€. With probability p, you gain 1€, and with probability $q = 1 - p$ you loose your 1€ bet.
- You start with an initial amount of 100€.
- You keep playing until you loose all your money or you arrive to have 1000€.
- We have a state for each possible amount of money you can accumulate $S = \{0, 1, \dots, 1000\}$.
- The probability of losing/winning is independent on the state and the time, so this process is a Markov chain.
- Observe that the number of states is finite.

An example: Gambler's Ruin

- You place bets of 1€. With probability p, you gain 1€, and with probability $q = 1 - p$ you loose your 1€ bet.
- You start with an initial amount of 100€.
- You keep playing until you loose all your money or you arrive to have 1000€.
- We have a state for each possible amount of money you can accumulate $S = \{0, 1, \ldots, 1000\}$.
- The probability of losing/winning is independent on the state and the time, so this process is a Markov chain.
- Observe that the number of states is finite.

Markov-Chains: An important tool for CS

- One of the simplest forms of stochastic dynamics.
- Allows to model stochastic temporal dependencies
- Applications in many areas
  - Surfing the web
  - Design of randomizes algorithms
  - Random walks
  - Machine Learning (Markov Decision Processes)
  - Computer Vision (Markov Random Fields)
  - etc. etc.

Formal definition of Markov Chains

Definition

A finite, time-discrete Markov Chain, with finite state $S = \{1, 2, \ldots, k\}$ is a stochastic process $\{X_t\}$ s.t. for all $i, j \in S$, and for all $t \geqslant 0$,

$$\mathbb{P}[X_{t+1} = j \mid X_0 = i_0, X_1 = i_1, \ldots, X_t = i] = \mathbb{P}[X_{t+1} = j \mid X_t = i] .$$

We can abstract the time and consider only the probability of moving from state i to state j, as $\mathbb{P}[X_{t+1} = j \mid X_t = i]$

Formal definition of Markov Chains

Definition

A finite, time-discrete Markov Chain, with finite state $S = \{1, 2, \ldots, k\}$ is a stochastic process $\{X_t\}$ s.t. for all $i, j \in S$, and for all $t \geqslant 0$,

$$\mathbb{P}[X_{t+1} = j \,|\, X_0 = i_0, X_1 = i_1, \ldots, X_t = i] = \mathbb{P}[X_{t+1} = j \,|\, X_t = i] \,.$$

We can abstract the time and consider only the probability of moving from state i to state j, as $\mathbb{P}[X_{t+1} = j \,|\, X_t = i]$

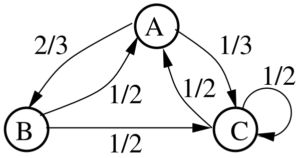
MC: Transition probability matrix

For $v, u \in S$, let $p_{u,v}$ be the probability of going from $u \rightsquigarrow v$ in 1 step i.e. $p_{u,v} = \mathbb{P}[X_{s+1} = v \,|\, X_s = u]$.

$P = (p_{u,v})_{u,v \in S}$ is a matrix describing the transition probabilities of the MC

P is called the transition matrix

P also defines digraph, possibly with loops.



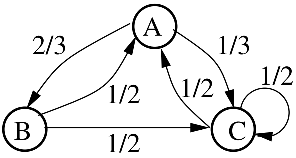
MC: Transition probability matrix

For $v, u \in S$, let $p_{u,v}$ be the probability of going from $u \rightsquigarrow v$ in 1 step i.e. $p_{u,v} = \mathbb{P}[X_{s+1} = v | X_s = u]$.

$P = (p_{u,v})_{u,v \in S}$ is a matrix describing the transition probabilities of the MC

P is called the transition matrix

P also defines digraph, possibly with loops.

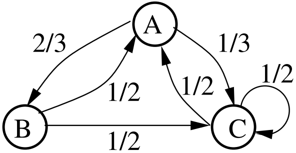


MC: Transition probability matrix

For $v, u \in S$, let $p_{u,v}$ be the probability of going from $u \rightsquigarrow v$ in 1 step i.e. $p_{u,v} = \mathbb{P}[X_{s+1} = v \,|\, X_s = u]$.

$P = (p_{u,v})_{u,v \in S}$ is a matrix describing the transition probabilities of the MC
P is called the transition matrix

P also defines digraph, possibly with loops.

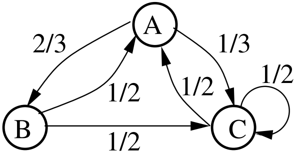


MC: Transition probability matrix

For $v, u \in S$, let $p_{u,v}$ be the probability of going from $u \rightsquigarrow v$ in 1 step i.e. $p_{u,v} = \mathbb{P}[X_{s+1} = v \,|\, X_s = u]$.

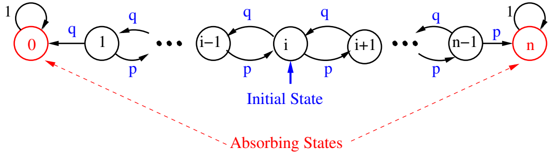
$P = (p_{u,v})_{u,v \in S}$ is a matrix describing the transition probabilities of the MC
P is called the transition matrix

P also defines digraph, possibly with loops.



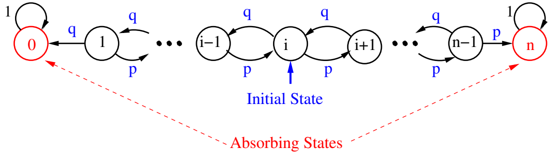
Gambler's Ruin: MC digraph

- You place bets of 1€. With probability p, you gain 1€, and with probability $q = 1 - p$ you loose your 1€ bet.
- You start with an initial amount of i € and keep playing until you loose all your money or you arrive to have n €.
- We have a state for each possible amount of money you can accumulate $S = \{0, 1, \ldots, n\}$.

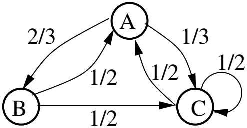


Gambler's Ruin: MC digraph

- You place bets of 1€. With probability p, you gain 1€, and with probability $q = 1 - p$ you loose your 1€ bet.
- You start with an initial amount of i € and keep playing until you loose all your money or you arrive to have n €.
- We have a state for each possible amount of money you can accumulate $S = \{0, 1, \ldots, n\}$.



Transition matrix: Example



$$\begin{array}{cc} & \begin{array}{ccc} A & B & C \end{array} \\ \begin{array}{c} A \\ B \\ C \end{array} & \begin{pmatrix} 0 & 2/3 & 1/3 \\ 1/2 & 0 & 1/2 \\ 1/2 & 0 & 1/2 \end{pmatrix} \end{array} = P$$

Notice the entry (u, v) in P denotes the probability of going from $u \to v$ in one step.

Notice, in a MC the transition matrix is stochastic, so sum of transitions out of any state must be 1 = sum of the elements of any row of the transition matrix must be 1

Longer transition probabilities

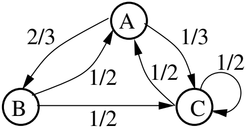
For $v, u \in S$, let $p_{u,v}^{(t)}$ be the probability of going from $u \rightsquigarrow v$ in exactly t steps i.e. $p_{u,v}^{(t)} = \mathbb{P}[X_{s+t} = v \,|\, X_s = u]$.

Formally for $s \geqslant 0$ and $t > 1$, $p_{u,v}^{(t)} = \mathbb{P}[X_{s+t} = v \,|\, X_s = u]$.

Notice that $p_{u,v} = p_{u,v}^{(1)}$; we shall use $P^{(t)}$ for the matrix whose entries are the values $p_{u,v}^{(t)}$, and $P^{(1)} = P$.

How can we relate $P^{(t)}$ with P?

The powers of the transition matrix



$$\begin{array}{c c} & \begin{array}{ccc} A & B & C \end{array} \\ \begin{array}{c} A \\ B \\ C \end{array} & \left(\begin{array}{ccc} 0 & 2/3 & 1/3 \\ 1/2 & 0 & 1/2 \\ 1/2 & 0 & 1/2 \end{array} \right) \end{array} = P$$

In ex. $\mathbb{P}[X_1 = C | X_0 = A] = P^{(1)}_{A,C} = 1/3$.

$\mathbb{P}[X_2 = C | X_0 = A] = P^{(1)}_{AB} P^{(1)}_{BC} + P^{(1)}_{AC} P^{(1)}_{CC} = 1/3 + 1/6 = P^{(2)}_{A,C}$

In general, assume a MC with k states and transition matrix P, let $u, v \in S$:

- What is the $\mathbb{P}[X_1 = u | X_0 = v]$, i.e. $= P_{v,u}$?
- What is the $\mathbb{P}[X_2 = u | X_0 = v] = P^{(2)}_{v,u}$?

The powers of the transition matrix

Use Law Total Probability+ Markov property:

$$P^{(2)}_{v,u} = \mathbb{P}[X_2 = u | X_0 = v] = \sum_{w=1}^{m} \mathbb{P}[X_1 = w | X_0 = v]\, \mathbb{P}[X_2 = u | X_1 = w]$$

$$= \sum_{w=1}^{m} P_{v,w} P_{w,u}.$$

The powers of the transition matrix

In general

$$\begin{aligned} p^{(t)}_{v,u} &= \mathbb{P}[X_t = u | X_0 = v] \\ &= \sum_{w=1}^{m} \mathbb{P}[X_{t-1} = w | X_0 = v]\, \mathbb{P}[X_t = u | X_{t-1} = w] \\ &= \sum_{w=1}^{m} P^{(t-1)}_{v,w} P_{w,u}. \end{aligned}$$

Lemma

Given the transition matrix P *of a MC, then for any* $t > 1$*,*

$$P^{(t)} = P^{(t-1)} \cdot P$$

With the convention $P^{(0)} = \mathbf{I}$ *(the identity matrix), we have*

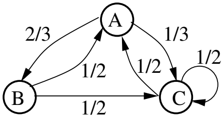
$$P^{(t)} = P^{t},$$

for any $t \geqslant 0$*.*

Distributions at time t

To fix the initial state, we consider a random variable X_0, assigning to S an initial distribution π_0, which is a row vector indicating at $t = 0$ the probability of being in the corresponding state.

For example, in the MC:



we may consider,

$$\begin{array}{ccc} A & B & C \\ \end{array}$$
$$\begin{pmatrix} 0 & 0.3 & 0.6 \end{pmatrix} = \pi_0$$

Distributions at time t

Starting with an initial distribution π_0, we can compute the state distribution π_t (on S) at time t,

For a state v,

$$
\begin{aligned}
\pi_t[v] &= \mathbb{P}[X_t = v] \\
&= \sum_{u \in S} \mathbb{P}[X_0 = u]\, \mathbb{P}[X_t = v | X_0 = u] \\
&= \sum_{u \in S} \pi_0[u] P^{(t)}_{v,u}.
\end{aligned}
$$

where $\pi_t[y]$ is the probability at step t the system is in state y.

Therefore, $\pi_t = \pi_0 P^t$ and $\pi_{s+t} = \pi_s P^t$.

Gambler's Ruin: Exercise

- You place bets of 1€. With probability p, you gain 1€, and with probability $q = 1 - p$ you loose your 1€ bet.
- You start with an initial amount of i € and keep playing until you loose all your money or you arrive to have n €.
- We have a state for each possible amount of money you can accumulate $S = \{0, 1, \ldots, n\}$.
- Which is the initial distribution π_0?
- And, the state distribution at time $t = 3$?

Gambler's Ruin: Exercise

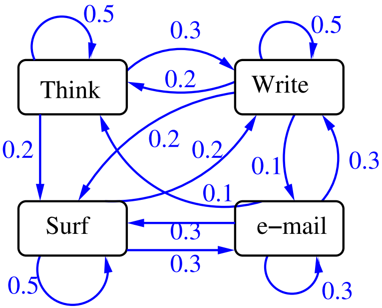
- You place bets of 1€. With probability p, you gain 1€, and with probability $q = 1 - p$ you loose your 1€ bet.
- You start with an initial amount of i € and keep playing until you loose all your money or you arrive to have n €.
- We have a state for each possible amount of money you can accumulate $S = \{0, 1, \ldots, n\}$.

- Which is the initial distribution π_0?
- And, the state distribution at time $t = 3$?

Example MC: Writing a research paper

Recall that Markov Chains are given either by a weighted digraph, where the edge weights are the transition probabilities, or by the $|S| \times |S|$ transition probability matrix P,

Example: Writing a paper $S = \{r, w, e, s\}$



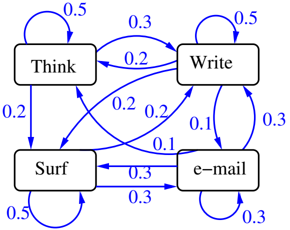
$$
\begin{array}{c|cccc}
 & r & w & e & s \\
\hline
r & 0.5 & 0.3 & 0 & 0.2 \\
w & 0.2 & 0.5 & 0.1 & 0.2 \\
e & 0.1 & 0.3 & 0.3 & 0.3 \\
s & 0 & 0.2 & 0.3 & 0.5
\end{array}
$$

More on the Markovian property

Notice the memoryless property does not mean that X_{t+1} is independent from $X_0, X_1, \ldots, X_{t-1}$.

(For instance notice that intuitively we have:
$\mathbb{P}[\text{Thinking at } t+1] < \mathbb{P}[\text{Thinking at } t \,|\, \text{Thinking at } t-1]$).

But, the dependencies of X_t on $X_0, \ldots, X_{t-1}$, are all captured by X_{t-1}.



Example of writing a paper

$\mathbb{P}[X_2 = s | X_0 = r]$ is the probability that, at $t = 2$, we are in state s, starting in state r.

$$\begin{pmatrix} 0.5 & 0.3 & 0 & 0.2 \\ 0.2 & 0.5 & 0.1 & 0.2 \\ 0.1 & 0.3 & 0.3 & 0.3 \\ 0 & 0.2 & 0.3 & 0.5 \end{pmatrix} \begin{pmatrix} 0.5 & 0.3 & 0 & 0.2 \\ 0.2 & 0.5 & 0.1 & 0.2 \\ 0.1 & 0.3 & 0.3 & 0.3 \\ 0 & 0.2 & 0.3 & 0.5 \end{pmatrix} = \begin{array}{c c} \begin{pmatrix} 0.31 & 0.34 & 0.09 & 0.26 \\ 0.21 & 0.38 & 0.14 & 0.27 \\ 0.14 & 0.33 & 0.21 & 0.32 \\ 0.07 & 0.29 & 0.26 & 0.38 \end{pmatrix} & \begin{array}{c} r \\ w \\ e \\ s \end{array} \end{array}$$

$\mathbb{P}[X_1 = s | X_0 = r] = 0.07.$

Distribution on states

Recall π_t is the prob. distribution at time t over S.

For our example of writing a paper, if $t = 0$ (after waking up):

$$\pi_0 = \begin{array}{cccc} r & w & e & s \\ (0.2 & 0 & 0.3 & 0.5) \end{array}$$

$$\begin{pmatrix}0.2 & 0 & 0.3 & 0.5\end{pmatrix}\begin{pmatrix}0.5 & 0.3 & 0 & 0.2\\ 0.2 & 0.5 & 0.1 & 0.2\\ 0.1 & 0.3 & 0.3 & 0.3\\ 0 & 0.2 & 0.3 & 0.5\end{pmatrix} = \begin{pmatrix}0.13 & 0.25 & 0.24 & 0.38\end{pmatrix} = \pi_1$$

Therefore, we have $\pi_t = \pi_0 \times P^t$ and $\pi_{k+t} = \pi_k \times P^t$

Notice $\pi_t = (\pi_t[r], \pi_t[w], \pi_t[e], \pi_t[s])$

An Example of MC analysis: The 2-SAT problem

Section 7.1 of [MU].

Given a Boolean formula ϕ, on
- a set X of n Boolean variables,
- defined by m clauses $C_1, \dots C_m$, where each clause is the disjunction of exactly 2 literals, (x_i or $\bar{x}_i$), on different variables.
- $\phi =$ conjunction of the m clauses.

The 2-SAT problem is to find an assignment $A^* : X \to \{0, 1\}$, which satisfies ϕ,
i.e, to find an A^* s.t. $A^*(\phi) = 1$.

Notice that if $|X| = n$, then $m \leqslant \binom{2n}{2} = \mathcal{O}(n^2)$.

In general k-SAT$\in$ NP-complete, for $k \geqslant 3$. But 2-SAT$\in$ P.

A randomized algorithm for 2-SAT

Given a n variable 2-SAT formula ϕ, $\{C_j\}_{j=1}^{m}$
for $1 \leqslant i \leqslant n$ **do**
    $A(x_i) := 1$
end for
$t := 0$
while $t \leqslant 2cn^2$ and some clause is unsatisfied **do**
    Pick and unsatisfied clause C_j
    Choose u.a.r. one of the 2 variables in C_j and flip its value
    if ϕ is satisfied **then**
        return A
    end if
end while
return ϕ is unsatisfiable

An example: unsat formula

If $\phi = (x_1 \vee x_2) \wedge (\bar{x}_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee x_2) \wedge (x_1 \vee \bar{x}_2)$
does not has a $A^* \models \phi$.

t	x_1	x_2	sel clause
1	1	1	2
2	1	0	3
3	0	0	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$

ϕ is unsat eventually the algorithm will stop after reaching the maximum number of steps.

An example: unsat formula

If $\phi = (x_1 \vee x_2) \wedge (\bar{x}_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee x_2) \wedge (x_1 \vee \bar{x}_2)$
does not has a $A^* \models \phi$.

t	x_1	x_2	sel clause
1	1	1	2
2	1	0	3
3	0	0	1
⋮	⋮	⋮	⋮

ϕ is unsat eventually the algorithm will stop after reaching the maximum number of steps.

An example: unsat formula

If $\phi = (x_1 \vee x_2) \wedge (\bar{x}_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee x_2) \wedge (x_1 \vee \bar{x}_2)$
does not has a $A^* \models \phi$.

t	x_1	x_2	sel clause
1	1	1	2
2	1	0	3
3	0	0	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$

ϕ is unsat eventually the algorithm will stop after reaching the maximum number of steps.

An example: unsat formula

If $\phi = (x_1 \vee x_2) \wedge (\bar{x}_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee x_2) \wedge (x_1 \vee \bar{x}_2)$
does not has a $A^* \models \phi$.

t	x_1	x_2	sel clause
1	1	1	2
2	1	0	3
3	0	0	1
⋮	⋮	⋮	⋮

ϕ is unsat eventually the algorithm will stop after reaching the maximum number of steps.

An example: unsat formula

If $\phi = (x_1 \vee x_2) \wedge (\bar{x}_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee x_2) \wedge (x_1 \vee \bar{x}_2)$
does not has a $A^* \models \phi$.

t	x_1	x_2	sel clause
1	1	1	2
2	1	0	3
3	0	0	1
⋮	⋮	⋮	⋮

ϕ is unsat eventually the algorithm will stop after reaching the maximum number of steps.

An example: unsat formula

If $\phi = (x_1 \vee x_2) \wedge (\bar{x}_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee x_2) \wedge (x_1 \vee \bar{x}_2)$
does not has a $A^* \models \phi$.

t	x_1	x_2	sel clause
1	1	1	2
2	1	0	3
3	0	0	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$

ϕ is unsat eventually the algorithm will stop after reaching the maximum number of steps.

An example: sat formula

If $\phi = (x_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee \bar{x}_3) \wedge (\bar{x}_1 \vee x_2) \wedge (\bar{x}_4 \vee \bar{x}_3) \wedge (x_4 \vee \bar{x}_1)$

t	x_1	x_2	x_3	x_4	sel clause
1	1	1	1	1	2
2	0	1	1	1	1
3	0	0	1	1	4
4	0	0	1	0	–

$(0, 0, 1, 0)$ satisfies ϕ

An example: sat formula

If $\phi = (x_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee \bar{x}_3) \wedge (\bar{x}_1 \vee x_2) \wedge (\bar{x}_4 \vee \bar{x}_3) \wedge (x_4 \vee \bar{x}_1)$

t	x_1	x_2	x_3	x_4	sel clause
1	1	1	1	1	2
2	0	1	1	1	1
3	0	0	1	1	4
4	0	0	1	0	–

$(0, 0, 1, 0)$ satisfies ϕ

An example: sat formula

If $\phi = (x_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee \bar{x}_3) \wedge (\bar{x}_1 \vee x_2) \wedge (\bar{x}_4 \vee \bar{x}_3) \wedge (x_4 \vee \bar{x}_1)$

t	x_1	x_2	x_3	x_4	sel clause
1	1	1	1	1	2
2	0	1	1	1	1
3	0	0	1	1	4
4	0	0	1	0	–

$(0, 0, 1, 0)$ satisfies ϕ

An example: sat formula

If $\phi = (x_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee \bar{x}_3) \wedge (\bar{x}_1 \vee x_2) \wedge (\bar{x}_4 \vee \bar{x}_3) \wedge (x_4 \vee \bar{x}_1)$

t	x_1	x_2	x_3	x_4	sel clause
1	1	1	1	1	2
2	0	1	1	1	1
3	0	0	1	1	4
4	0	0	1	0	–

$(0, 0, 1, 0)$ satisfies ϕ

An example: sat formula

If $\phi = (x_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee \bar{x}_3) \wedge (\bar{x}_1 \vee x_2) \wedge (\bar{x}_4 \vee \bar{x}_3) \wedge (x_4 \vee \bar{x}_1)$

t	x_1	x_2	x_3	x_4	sel clause
1	1	1	1	1	2
2	0	1	1	1	1
3	0	0	1	1	4
4	0	0	1	0	–

$(0, 0, 1, 0)$ satisfies ϕ

An example: sat formula

If $\phi = (x_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee \bar{x}_3) \wedge (\bar{x}_1 \vee x_2) \wedge (\bar{x}_4 \vee \bar{x}_3) \wedge (x_4 \vee \bar{x}_1)$

t	x_1	x_2	x_3	x_4	sel clause
1	1	1	1	1	2
2	0	1	1	1	1
3	0	0	1	1	4
4	0	0	1	0	–

$(0, 0, 1, 0)$ satisfies ϕ

An example: sat formula

If $\phi = (x_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee \bar{x}_3) \wedge (\bar{x}_1 \vee x_2) \wedge (\bar{x}_4 \vee \bar{x}_3) \wedge (x_4 \vee \bar{x}_1)$

t	x_1	x_2	x_3	x_4	sel clause
1	1	1	1	1	2
2	0	1	1	1	1
3	0	0	1	1	4
4	0	0	1	0	–

$(0, 0, 1, 0)$ satisfies ϕ

An example: sat formula

If $\phi = (x_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee \bar{x}_3) \wedge (\bar{x}_1 \vee x_2) \wedge (\bar{x}_4 \vee \bar{x}_3) \wedge (x_4 \vee \bar{x}_1)$

t	x_1	x_2	x_3	x_4	sel clause
1	1	1	1	1	2
2	0	1	1	1	1
3	0	0	1	1	4
4	0	0	1	0	–

$(0, 0, 1, 0)$ satisfies ϕ

Analysis for 2-SAT algorithm

Given $\phi, |X| = n, \{C_j\}_{i=1}^{m}$

assume that there is A^* such that $\phi(A^*) = 1$

- Let A_i be the assignment at the i-th iteration.
- Let $X_i = |\{x_j \in X \mid A_i(x_j) = A^*(x_j)\}$.
- Notice $0 \leqslant X_i \leqslant n$. Moreover, when $X_i = n$, we found A^*.
- Analysis: Starting from $X_i < n$, how long to get $X_i = n$?
- Note that $\mathbb{P}[X_{i+1} = 1 \mid X_i = 0] = 1$.

Analysis for 2-SAT algorithm

- As A^* satisfies ϕ and A_i no, there is a clause C_j that A^* satisfies but A_i not.
- So A^* and A_i disagree in the value of at least one variable.
- It is also possible to flip the value of a variable in C_j in which A and A^* agree.
- Therefore,

  For $1 \leqslant k \leqslant n-1$, $\mathbb{P}[X_{i+1} = k+1 \mid X_i = k] \geqslant 1/2$ and $\mathbb{P}[X_{i+1} = k-1 \mid X_i = k] \leqslant 1/2$.

Analysis for 2-SAT

The process $X_0, X_1, \ldots$ is not necessarily a MC,

- The probability that $X_{i+1} > X_i$ depends on whether A_i and A^* disagree in 1 or 2 variables in the selected unsatisfied clause C.
- If A^* makes true both literals in C, $\mathbb{P}[X_{i+1} = k+1 \,|\, X_i = k] = 1$, otherwise $\mathbb{P}[X_{i+1} = k+1 \,|\, X_i = k] = 1/2$
- This difference might depend on the clauses and variables selected in the past, so the transition probabilities are not memoryless.
- X_t is not a Markov chain. Can we bound the process by a MC?.

Analysis for 2-SAT

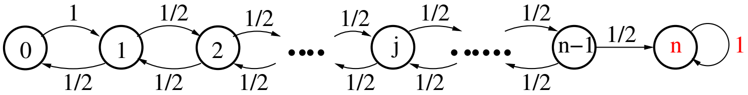
The process $X_0, X_1, \ldots$ is not necessarily a MC,

- The probability that $X_{i+1} > X_i$ depends on whether A_i and A^* disagree in 1 or 2 variables in the selected unsatisfied clause C.
- If A^* makes true both literals in C, $\mathbb{P}[X_{i+1} = k+1 \,|\, X_i = k] = 1$, otherwise $\mathbb{P}[X_{i+1} = k+1 \,|\, X_i = k] = 1/2$
- This difference might depend on the clauses and variables selected in the past, so the transition probabilities are not memoryless.
- X_t is not a Markov chain. Can we bound the process by a MC?.

Analysis for 2-SAT

Define a MC $Y_0, Y_1, Y_2, \ldots$ which is a pessimistic version of process $X_0, X_1, \ldots$, in the sense that Y_i measures exactly the same quantity than X_i but the probability of change (up or down) will be exactly $1/2$.

- $Y_0 = X_0$ and $\mathbb{P}[Y_{i+1} = 1 \mid Y_i = 0] = 1$;
- For $1 \leqslant k \leqslant n-1$, $\mathbb{P}[Y_{i+1} = k+1 \mid Y_i = k] = 1/2$;
- $\mathbb{P}[Y_{i+1} = k-1 \mid Y_i = k] = 1/2$.



MC for 2−SAT

The time to reach n from $j \geqslant 0$ in $\{Y_i\}_{i=0}^{n}$ is $\geqslant$ that in $\{X_i\}_{i=0}^{n}$.

Upper Bound on the time to arrive state n

Lemma

If a 2-CNF ϕ on n variables has a satisfying assignment A^*, the 2-SAT algorithm finds one in expected time $\leqslant n^2$.

Proof

- Let h_j be the expected time, for process Y, to go from state j to state n.
- It suffices to prove that, when Y starts in state j the time to arrives to n is $\leqslant 2cn^2$.
- We devise a recurrence to bound h

Upper Bound on the time to arrive state n

Proof (cont'd)

- $h_n = 0$ and $h_1 = h_0 + 1$;
- We want a general recurrence on h_j, for $1 \leqslant j < n$
- Define a rv Z_j counting the steps to go from state $j \to n$ in Y.
- With probability $1/2$, $Z_j = Z_{j-1} + 1$ and, with probability $1/2$, $Z_j = Z_{j+1} + 1$.
- So $h_j = \mathbb{E}[Z_j]$.

$$\mathbb{E}[Z_j] = \mathbb{E}\left[\frac{Z_{j-1}+1}{2} + \frac{Z_{j+1}+1}{2}\right] = \frac{\mathbb{E}[Z_{j-1}]+1}{2} + \frac{\mathbb{E}[Z_{j+1}]+1}{2}.$$

So, $h_j = \frac{h_{j-1}}{2} + \frac{h_{j+1}}{2} + 1$.

Upper Bound on the time to arrive state n

Proof (cont'd)

From the previous bound we get $h_j = \frac{h_{j-1}}{2} + \frac{h_{j+1}}{2} + 1$.

The recurrence has the $n+1$ equations,

$$
\begin{aligned}
h_n &= 0 \\
h_0 &= h_1 + 1 \\
h_j &= \frac{h_{j-1}}{2} + \frac{h_{j+1}}{2} + 1 \qquad 0 \leqslant j \leqslant n-1
\end{aligned}
$$

Let us prove, by induction that

$$h_j = h_{j+1} + 2j + 1.$$

Upper Bound on the time to arrive state n

Proposition

For $0 \leqslant j \leqslant n-1$, $h_j = h_{j+1} + 2j + 1$.

Proof (of Proposition)

Base case: If $j = 0$, $2j + 1 = 1$, and we were given $h_0 = h_1 + 1$.

Upper Bound on the time to arrive state n

Proposition

For $0 \leqslant j \leqslant n-1$, $h_j = h_{j+1} + 2j + 1$.

Proof of Proposition (cont'd)

IH: for$j = k-1$, $h_{k-1} = h_k + 2(k-1) + 1$.
Now consider $j = k$. By the "middle case" of our system of equations,

$$
\begin{aligned}
h_k &= \frac{h_{k-1} + h_{k+1}}{2} + 1 \\
&= \frac{h_k + 2(k-1) + 1}{2} + \frac{h_{k+1}}{2} + 1 && \text{by IH} \\
&= \frac{h_k}{2} + \frac{h_{k+1}}{2} + \frac{2k+1}{2}
\end{aligned}
$$

Subtracting $\frac{h_k}{2}$ from each side, we get the result. $\square$

Upper Bound on the time to arrive state n

Proof (cont'd)

As

$$h_j = h_{j+1} + 2j + 1.$$

$$\begin{aligned} h_0 &= h_1 + 1 = h_2 + 3 + 1 = h_3 + 5 + 3 + 1 \cdots \\ &= \underbrace{h_n}_{=0} + \sum_{i=0}^{n-1} (2i + 1) = n^2. \end{aligned}$$

$\square$

Error probability for 2-SAT algorithm

Theorem

The 2-SAT algorithm gives the correct answer NO if ϕ is not satisfiable. Otherwise, with probability $\geqslant 1 - \frac{1}{2^c}$ the algorithm returns a satisfying assignment.

Error probability for 2-SAT algorithm

Proof

- Let ϕ be satisfiable (otherwise the theorem holds).
- Break the $2cn^2$ iterations into c blocks of $2n^2$ iterations.
- For each block i, define a r.v. $Z =$ number of iterations from the start of the i-block until a solution is found.
- Using Markov's inequality:

$$\mathbb{P}\left[Z > 2n^2\right] \leqslant \frac{n^2}{2n^2} = \frac{1}{2}.$$

- Therefore, the probability that the algorithm fails to find a satisfying assignment after c segments (no block includes a solution) is at most $\frac{1}{2^c}$.

$\square$