- Markov Chains & Random Walks. Ch 7, M-U
  - Definitions & representations
  - Applications: 2-SAT & 3-SAT Algos, s-t connectivity.

- A stochastic process $\mathbb{X} = \{ X(t) : t \in T \}$ is a collection of random variables.

The index t often represents time and $\mathbb{X}$ models the value of a RV X that changes over time.

$X(t)$ or X_t : state of the process at time t.

If $\forall t$, X_t takes values from countably infinite set, then $\mathbb{X}$ is discrete space process.

If T is a countably infinite set, then $\mathbb{X}$ is called a discrete time process.

§ Markov Chain : A discrete time stochastic process $X_0, X_1, \ldots$ is a Markov chain if

$$\mathbb{P}(X_t = a_t \mid X_{t-1} = a_{t-1}, X_{t-2} = a_{t-2}, \ldots, X_0 = a_0)$$
$$= \mathbb{P}(X_t = a_t \mid X_{t-1} = a_{t-1})$$

→ Markov property or memoryless property means state X_t only depends on prev state X_{t-1}, but is independent of history of how the process arrived at state X_{t-1}. Note that it does not say X_t is indep. of $X_0, \ldots, X_{t-2}$; just implies that any dependency of X_t on the past is captured in the value of X_{t-1}.

• We will assume discrete state space of the Markov chain is $\{0, 1, \dots, \}$ and the transition probability $P_{ij} := \mathbb{P}(X_t = j \mid X_{t-1} = i)$ is the probability that the process moves from i to j in one step.

Markov property implies that Markov chain is uniquely defined by a transition matrix.

$$M_P = \begin{bmatrix} P_{0,0} & P_{0,1} & \cdots & P_{0,j} & \cdots \\ \vdots & & \ddots & & \\ P_{i,0} & & & P_{i,j} & \cdots \\ \vdots & & & & \end{bmatrix}$$

Let $p_i(t)$ denote the probability that the process is at state i at time t.

$\bar{p}(t) := (p_0(t), p_1(t), \dots)$ be the vector representation of distribution of states of chain at time t.

$\therefore p_i(t) = \sum_j p_j(t-1) P_{j,i}$, or $\bar{p}(t) = \bar{p}(t-1) . M_P$

Define m-step transition probability:

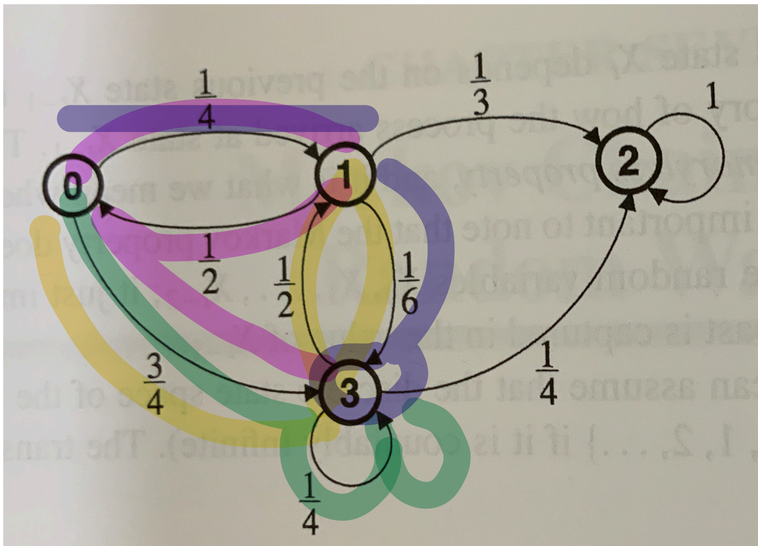
$$P_{i,j}^m := \mathbb{P}(X_{t+m} = j \mid X_t = i) = \sum_{k \geq 0} P_{i,k} P_{k,j}^{m-1}$$

Let $\mathbf{P}^{(m)}$ be matrix defined by m-step transition probabilities. Then, $M_P^{(m)} = M_P \cdot M_P^{(m-1)} \Rightarrow M_P^{(m)} = M_P^m$.

Thus, for any $t \geq 0$ and $m \geq 1$,

$$\bar{p}(t+m) = \bar{p}(t) M_P^m$$

• Graph Representation: Directed weighted graph, self loop allowed, weighted outdegree is 1.



$$P = \begin{bmatrix} 0 & 1/4 & 0 & 3/4 \\ 1/2 & 0 & 1/3 & 1/6 \\ 0 & 0 & 1 & 0 \\ 0 & 1/2 & 1/4 & 1/4 \end{bmatrix}$$

$$\frac{3}{4}\cdot\frac{1}{2}\cdot\frac{1}{6} + \frac{3}{4}\times\frac{1}{4}\times\frac{1}{4} + \frac{1}{4}\times\frac{1}{2}\times\frac{3}{4} + \frac{1}{4}\times\frac{1}{6}\times\frac{1}{4} = \frac{9+18+2+12}{192} = \frac{41}{192}$$

Say, we are interested in prob. of going from state 0 to state 3 in 3 steps. Just compute $M_{P^3_{0,3}}$.

$$P^3 = \begin{bmatrix} 3/16 & 7/48 & 29/64 & 41/192 \\ 5/48 & 5/24 & 79/144 & 5/36 \\ 0 & 0 & 1 & 0 \\ 1/16 & 13/96 & 107/192 & 47/192 \end{bmatrix}$$

§ Application: Algorithm for 2-SAT.

Satisfiability (SAT) problem:

Given: A Boolean formula ϕ given as conjunction (AND) of a set of clauses, where each clause is disjunction (OR) of literals, where a literal is a Boolean variable or its negation.

Goal: Assign True (T) / False (F) to the variables s.t. all clauses are satisfied.

K-SAT: Each clause has exactly K literals.

Example of a 2-SAT formula:

$$(x_1 \vee \overline{x_2}) \wedge (\overline{x_1} \vee \overline{x_3}) \wedge (x_1 \vee x_2) \wedge (x_4 \vee \overline{x_1})$$

One satisfying assignment: $x_1 = 1, x_2 = 0, x_3 = 0, x_4 = 1$.

- An example of unsatisfiable formula:

$$(x_1 \vee x_2) \wedge (x_1 \vee \overline{x_2}) \wedge (\overline{x_1} \vee \overline{x_2}) \wedge (\overline{x_1} \vee x_2).$$

- 2-SAT algo:

Comparison with LLL (Moser-Tardos):
- It is for SAT w/o any assumptions.
- Moser-Tardos # resamples $= \sum_{i=1}^{n} \frac{x_i}{1-x_i}$

Input: n: number of variables
m: f(probability of success).

Algo:

1. Start with an arbitrary truth assignment.
2. Repeat upto $2mn^2$ times steps, terminating if all clauses are satisfied:

(a) choose an arbitrary clause that is unsatisfied,
(b) choose uniformly at random, one of the variables in that cause & switch its value.

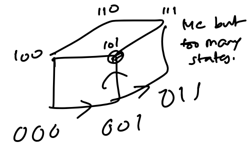
3. If a valid truth assignment has been found, return it.

4. Otherwise, return the formula to be unsatisfiable.

- As there n variables, a SAT formula has $O(n^2)$ distinct clauses.

Let S : a satisfying assignments for n variables.
A_i : variable assignment after i'th step.
X_i : number of variables having same value in A_i & S.

So, $X_i = n \Rightarrow$ satisfying assignment.



Q. How long does it take for X_i to reach n?

Observation : $\mathbb{P}(X_{i+1} = 1 \mid X_i = 0) = 1$.

Now suppose, $1 \leq X_i \leq n-1$.
Consider an unsatisfied clause C.
S must disagree with A_i on at least one the variables in C.

So, if we random switch one of ~~the~~ variables in C, prob. #matches increase is $\geq 1/2$.

$$\mathbb{P}(X_{i+1} = j+1 \mid X_i = j) \geq 1/2.$$ $1/k$ for k-SAT.
$$\mathbb{P}(X_{i+1} = j-1 \mid X_i = j) \leq 1/2.$$

	x_1	$\vee$	x_2
A_i :	0		0
S :	1		0.
	1		1.
	0		1.

Is $X_0, X_1, \ldots$ a Markov chain?
- Not necessarily!

Whether X_i increases depend on whether A_i or S disagree on one or two variables in the chosen unsatisfied clause. This might depend on past history, which clauses have been considered in the past.

So we use the following Markov chain $Y_0, Y_1, \dots,$ which is a pessimistic version of the stochastic process $X_0, X_1, \dots$

$$Y_0 = X_0$$
$$\mathbb{P}(Y_{i+1} = 1 \mid Y_i = 0) = 1$$
$$\mathbb{P}(Y_{i+1} = j+1 \mid Y_i = j) = 1/2$$
$$\mathbb{P}(Y_{i+1} = j-1 \mid Y_i = j) = 1/2.$$

So, expected time to reach n starting from any point, is larger for Y than X.

This Markov chain models a random walk on an directed graph G.

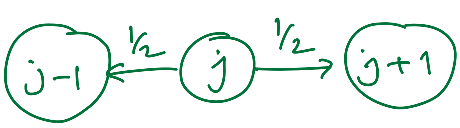
Vertices : $\{0, \dots, n\}$.

Edge : $(i, i+1)$ for $i = 0, n-1$.

Z_j : RV representing number of steps to reach n starting from state j. $h_j := \mathbb{E}[Z_j]$.

Observation : $h_n = 0$, $h_0 = h_1 + 1$. ← from h_0 we always move to h_1 in one step.

For 2-SAT, $h_j \geqslant \mathbb{E}$[number of steps to
(X) to fully match S when starting from A_0 that matches S in j locations]



w.p. 1/2 $Z_j = 1 + Z_{j+1}$
" $= 1 + Z_{j-1}$

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

$$\Rightarrow \quad h_j = \frac{h_{j-1}}{2} + \frac{h_{j+1}}{2} + 1, \quad \text{for } 1 \leq j \leq n-1.$$

Also, $h_n = 0$; $h_0 = h_1 + 1$.

We can show inductively, for $0 \leq j \leq n-1$,

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

Base : $j=0$: As $h_1 = h_0 - 1$, we get

$$h_0 = h_1 + 1 = h_1 + 2.0 + 1 . \checkmark$$

Induction:

We have, $h_j = \dfrac{h_{j-1}}{2} + \dfrac{h_{j+1}}{2} + 1$,

$$\Rightarrow h_{j+1} = 2h_j - h_{j-1} - 2.$$

$$= 2h_j - \overbrace{(h_j + 2(j-1) + 1)}^{\text{induction}} - 2$$

$$= h_j - 2j - 1.$$

Hence, $h_0 = h_1 + 1 = h_2 + 1 + 3 = \ldots = \sum_{i=0}^{n-1} (2i+1) = n^2.$

Hence, for a satisfying formula expected number of steps to find a satisfying formula is n^2.

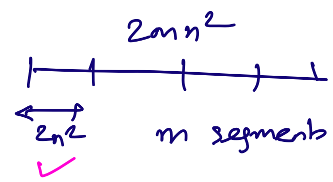
So, using Markov

$$\mathbb{P}(Z_0 > mn^2) \leq \frac{n^2}{mn^2} = \frac{1}{m}.$$

Using power of repetitions, we can improve further.

Theorem:

If ϕ is satisfiable,
Algo returns correct assignment w.p. $\geq 1 - 2^{-m}$.
If ϕ is unsatisfiable
Algo is always correct. trivial.



Proof: Let ϕ be satisfiable.
Divide the execution into segments of $2n^2$ steps each.

In each segment, prob of success $\geq 1/2$.
[using Markov]

So the prob. Algo fails after m segments is $\leq \left(\frac{1}{2}\right)^m$.

HW: Think about deterministic polytime algos for 2SAT.

Extending 2-SAT Algo we study 3-SAT.

§ 3-SAT Algorithm:

3-SAT Algorithm:

1. Start with an arbitrary truth assignment.
2. Repeat up to m times, terminating if all clauses are satisfied:
   (a) Choose an arbitrary clause that is not satisfied.
   (b) Choose one of the literals uniformly at random, and change the value of the variable in the current truth assignment.
3. If a valid truth assignment has been found, return it.
4. Otherwise, return that the formula is unsatisfiable.

3-SAT is NP-hard, so we don't expect m to be polynomial in n.

Similar to 2-SAT, we obtain following:

$$\mathbb{P}(X_{i+1} = j+1 \mid X_i = j) \geq 1/3$$
$$\mathbb{P}(X_{i+1} = j-1 \mid X_i = j) \leq 2/3$$

X_i : number of variables having same value in A_i & S.

$$\begin{array}{ccc} 0 & 0 & 1 \\ (x_1 \vee & x_2 \vee & x_3) \\ 0 & 0 & 0 \end{array}$$

more likely to go down than up!

MC:

$$\mathbb{P}(Y_{i+1} = 1 \mid Y_i = 0) = 1$$
$$\mathbb{P}(Y_{i+1} = j+1 \mid Y_i = j) = 1/3$$
$$\mathbb{P}(Y_{i+1} = j-1 \mid Y_i = j) = 2/3.$$

This gives set of equations:

$$h_n = 0$$
$$h_j = \frac{2h_{j-1}}{3} + \frac{h_{j+1}}{3} + 1, \quad 1 \leq j \leq n-1$$
$$h_0 = h_1 + 1.$$

HW: use induction to show

$$h_j = h_{j+1} + 2^{j+2} - 3.$$

which implies:

$$h_j = 2^{n+2} - 2^{j+2} - 3(n-j)$$
$$\Rightarrow h_0 = 2^{n+2} - 4 - 3n = \Theta(2^n).$$

- This is quite bad as there are only 2^n possible assignments.

Can we improve?

Observations:

1. If we choose an initial assignment uniformly at random, then the number of variables that match S has a binomial distribution with mean $n/2$.

2. We are more likely to go to 0 than n. So better to restart the process if we are not successful after some time.

Modified 3-SAT Algorithm: Schöning's Algorithm

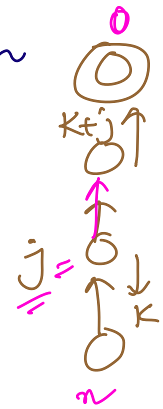
1. Repeat up to m times, terminating if all clauses are satisfied:
   (a) Start with a truth assignment chosen uniformly at random.
   (b) Repeat the following up to $3n$ times, terminating if a satisfying assignment is found:
      i. Choose an arbitrary clause that is not satisfied.
      ii. Choose one of the literals uniformly at random, and change the value of the variable in the current truth assignment.
2. If a valid truth assignment has been found, return it.
3. Otherwise, return that the formula is unsatisfiable.

Random restart hill-climbing

Let q be the prob. that after $3n$ steps we reach S, after starting from a random assign.

q_j be prob. that after $3n$ steps we reach S (or some other satisfying assignment), after starting from an assignment that has exactly j variables that do not agree with S.

Now, $\mathbb{P}$ of having exactly k moves down and $(k+j)$ moves up in a sequence of $(j+2k)$ moves $= \binom{j+2k}{k}\left(\frac{2}{3}\right)^k\left(\frac{1}{3}\right)^{j+k}$.



we are interested in $j+2k \le 3n$.

Now, consider special case $k=j$, then $3j \le 3n$.

So, $q_j \ge \binom{3j}{j}\left(\frac{2}{3}\right)^j\left(\frac{1}{3}\right)^{2j}$ (for $k=j$)

$$\overset{*}{\ge} \frac{c}{\sqrt{j}}\left(\frac{27}{4}\right)^j\left(\frac{2}{3}\right)^j\left(\frac{1}{3}\right)^{2j} \ge \frac{c}{\sqrt{j}}\cdot\frac{1}{2^j}.$$

[where $c = \sqrt{3}/8\sqrt{\pi} \approx 0.1222$

$*$ follows from Stirling formula: for $m>0$,

$$\sqrt{2\pi m}\left(\frac{m}{e}\right)^m \le m! \le 2\sqrt{2\pi m}\left(\frac{m}{e}\right)^m.$$

Hence, q

$$\ge \sum_{j=0}^{n} \mathbb{P}\left[\begin{array}{l}\text{a random assignment has}\\ j \text{ mismatches with } S\end{array}\right]\cdot q_j$$

$$\ge \underbrace{\frac{1}{2^n}}_{\text{handles } 0.} + \sum_{j=1}^{n}\binom{n}{j}\left(\frac{1}{2}\right)^n\cdot\frac{c}{\sqrt{j}}\cdot\frac{1}{2^j}$$

$$\ge \frac{c}{\sqrt{n}}\cdot\left(\frac{1}{2}\right)^n\sum_{j=0}^{n}\binom{n}{j}\left(\frac{1}{2}\right)^j(1)^{n-j} \qquad \left[\because j\le n \ \& \ 1 \ge \frac{c}{\sqrt{n}}\right]$$

$$= \frac{c}{\sqrt{n}}\left(\frac{1}{2}\right)^n\left(\frac{3}{2}\right)^n \qquad \left[\because \sum_{j=0}^{n}\binom{n}{j}\left(\frac{1}{2}\right)^j(1)^{n-j} = \left(1+\frac{1}{2}\right)^n\right]$$

$$= \frac{c}{\sqrt{n}}\left(3/4\right)^n.$$

Assuming ϕ is satisfiable, the number of random assignments the process tries before finding a satisfying assignment, is a geometric RV with parameter q.

Expected number of assignments tried $= 1/q$.
Each assignment uses $3n$ steps.

$\therefore$ $\mathbb{E}$[number of steps until a solution is found]

$= \frac{1}{q} \cdot 3n$

$= \left(\frac{4}{3}\right)^n \cdot \frac{\sqrt{n}}{c} \cdot 3n = O\left(n^{3/2} \cdot (4/3)^n\right) =: \alpha.$

Exponential-time hypothesis

ETH

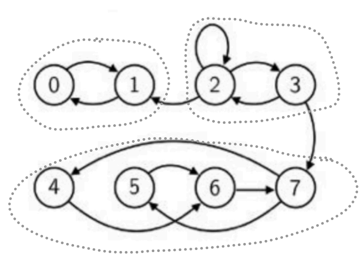
SETH

This is a Monte Carlo algorithm.
setting $m = 2\alpha\beta$, the prob that no assignment is found is $\leq 2^{-\beta}$, using similar arguments as in 2SAT analysis.

for k-SAT, same analysis leads to $\Omega(2^n)$ as $k \to \infty$.

§ Classification of states:

Definition 7.2: *State j is accessible from state i if, for some integer n* ≥ 0, $P^n_{i,j} > 0$. *If two states i and j are accessible from each other, we say that they communicate and we write* $i \leftrightarrow j$.



In the graph representation of a chain, $i \leftrightarrow j$ if and only if there are directed paths connecting i to j and j to i.

The communicating relation defines an equivalence relation. That is, the communicating relation is

1. *reflexive* – for any state i, $i \leftrightarrow i$;
2. *symmetric* – if $i \leftrightarrow j$ then $j \leftrightarrow i$; and
3. *transitive* – if $i \leftrightarrow j$ and $j \leftrightarrow k$, then $i \leftrightarrow k$.

Proving this is left as Exercise 7.4. Thus, the communication relation partitions the states into disjoint equivalence classes, which we refer to as *communicating classes*. It might be possible to move from one class to another, but in that case it is impossible to return to the first class.

- A Markov chain is irreducible if all states belong to one communicating class.

[Its graph representation is a strongly connected]

- Recurrent & transient states.

Let $r_{i,j}^{t}$ denote prob. that starting at state i, the first transition to state j occurs at time t.

$$r_{i,j}^{t} = \mathbb{P}\left(X_t = j \text{ and, for } 1 \leq s \leq t-1,\ X_s \neq j \mid X_0 = i\right).$$

– A state i is recurrent if $\sum_{t \geq 1} r_{i,i}^{t} = 1$,

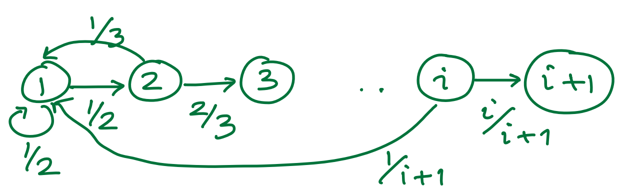
" transient if $\sum_{t \geq 1} r_{i,i}^{t} < 1$.

- A Markov chain is recurrent if every state in the chain is recurrent.

- Denote $h_{i,j} := \sum_{t \geq 1} t \cdot r_{i,j}^{t}$, the expected time to first reach j from state i.

- Is $h_{i,i}$ finite in a recurrent chain? → NO!
(Note state i is visited infinitely often)

- A recurrent state i is positive recurrent if $h_{i,i} < \infty$, otherwise it is null recurrent.



Pr [not returning state 1 after t steps]

$$= \prod_{j=1}^{t} \frac{j}{j+1} = \frac{1}{t+1}.$$

Hence, state 1 is recurrent.

$$r_{1,1}^{t} = \Pr\left[\begin{array}{c}\text{not returning}\\ \text{after } t-1 \text{ steps}\end{array}\right] \cdot \Pr\left[\begin{array}{c}\text{returning}\\ \text{on } t\text{ th step}\end{array}\right]$$

$$= \frac{1}{t} \cdot \frac{1}{t+1}.$$

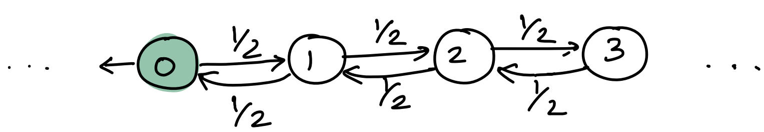
$\therefore h_{1,1} = \sum_{t=1}^{\infty} t \cdot r_{1,1}^{t} = \sum_{t=1}^{\infty} \frac{1}{t+1}$, which is unbounded.

For null recurrent states, it is necessary to have infinite number of states.

Lemma: In a finite Markov chain:

1. at least one state is recurrent, and
2. all recurrent states are positive recurrent

- Periodicity:



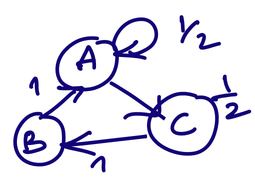
If the chain starts at 0, after odd (resp. even) number of moves it is at odd (resp. even) states.

$$\mathbb{P}(X_{t+s} = j \mid X_t = j) = 0, \text{ unless } 2 \mid s.$$

- Defn: A state j in a discrete time Markov chain is periodic if there exists an integer $\Delta > 1$ s.t. $\mathbb{P}(X_{t+s} = j \mid X_t = j) = 0$ unless $\Delta \mid s$.

→ a chain is periodic if any of its states is periodic.

- Aperiodic = not periodic. $(\Delta = 1)$



Defn: An aperiodic, positive recurrent state is an ergodic state.

A Markov chain is ergodic if all its states are ergodic.

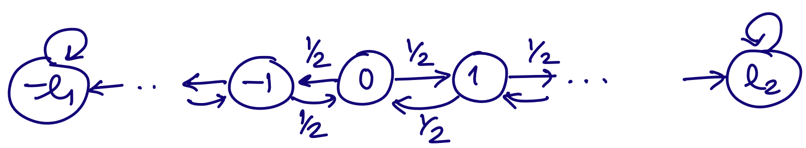
Lemma:

⊙ Any $\overbrace{\text{finite, irreducible}}^{\text{+ve recurrent}}$, aperiodic Markov chain is an ergodic chain.

§ Example: The Gambler's Ruin.

- Fair gambling game between two players.
- In each round, a player wins 1 ₹ w.p. ½ and looses 1 ₹ w.p. ½.
- The state of the system at time t is the number of ₹'s won by player one.

The game stops when player 1 wins ℓ_2 ₹ or looses ℓ_1 ₹. What is the prob. player 1 wins ℓ_2 ₹?



Then $-\ell_1$ and ℓ_2 are recurrent states.

All other states are transient, as there is a nonzero prob of reaching ℓ_1 and ℓ_2.

Let P_i^t be prob. that, after t steps, the chain is at state i. Then for $-\ell_1 < i < \ell_2$, $\lim_{t\to\infty} P_i^t = 0$.

recurrent   detail of   ∴ ⊛

calculation

Let q be the prob. that the game ends with player 1 winning ℓ_2 ₤, we call the chain was absorbed into state ℓ_2. Then

$$\lim_{t\to\infty} P^t_{\ell_2} = q, \quad \lim_{t\to\infty} P^t_{\ell_1} = 1-q. \quad \cdots (**)$$

Let W^t be the gain of player 1 after t steps. Then $\mathbb{E}[W^t]=0$ for any t by induction. ($\because$ fair game)

Thus, $\mathbb{E}[W^t] = \sum_{i=-\ell_1}^{\ell_2} i P_i^t = 0.$

(*) & (**)

and $\lim_{t\to\infty} \mathbb{E}[W^t] = \ell_2 q - \ell_1(1-q) = 0$

$$\Rightarrow q = \frac{\ell_1}{\ell_1+\ell_2}.$$

§ Stationary Distributions:

A stationary distribution (or equilibrium distr.) of a Markov chain is a prob. distr. $\bar{\pi}$ s.t. $\bar{\pi} = \bar{\pi} \cdot M_p$ ← transition matrix

Fundamental thm of MC:

aperiodic

Theorem 7.7: *Any finite, irreducible, and ~~ergodic~~ Markov chain has the following properties:*

0. All states are ergodic.

1. *the chain has a unique stationary distribution* $\bar{\pi} = (\pi_0, \pi_1, \ldots, \pi_n)$;
2. *for all j and i, the limit* $\lim_{t\to\infty} P^t_{j,i}$ *exists and it is independent of j;*
3. $\pi_i = \lim_{t\to\infty} P^t_{j,i} = 1/h_{i,i}$.

→ If we run the chain long enough, the initial state is not important and prob of being in state i converges to π_i.

Note: for bipartite (periodic) case there is no stationary distribution. It toggles between two partitions in consecutive steps.

Lemma 7.8: *For any irreducible, ergodic Markov chain and for any state i, the limit* $\lim_{t\to\infty} P^t_{i,i}$ *exists and*

$$\lim_{t\to\infty} P^t_{i,i} = \frac{1}{h_{i,i}}.$$

Theorem 7.9: *Let S be a set of states of a finite, irreducible, aperiodic Markov chain. In the stationary distribution, the probability that the chain leaves the set S equals the probability that it enters S.*

Theorem 7.10: *Consider a finite, irreducible, and ergodic Markov chain with transition matrix* **P**. *If there are nonnegative numbers* $\bar{\pi} = (\pi_0, \ldots, \pi_n)$ *such that* $\sum_{i=0}^{n} \pi_i = 1$ *and if, for any pair of states i, j,*

$$\pi_i P_{i,j} = \pi_j P_{j,i},$$

then $\bar{\pi}$ *is the stationary distribution corresponding to* **P**.

Theorem 7.11: *Any irreducible aperiodic Markov chain belongs to one of the following two categories:*

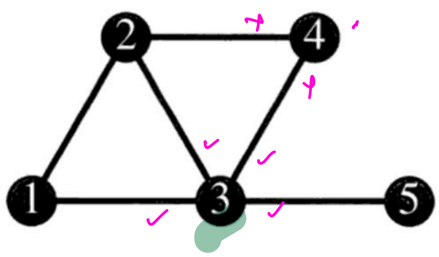
1. *the chain is ergodic – for any pair of states i and j, the limit* $\lim_{t\to\infty} P^t_{j,i}$ *exists and is independent of j, and the chain has a unique stationary distribution* $\pi_i = \lim_{t\to\infty} P^t_{j,i} > 0$*; or*
2. *no state is positive recurrent – for all i and j,* $\lim_{t\to\infty} P^t_{j,i} = 0$*, and the chain has no stationary distribution.*

Cut-sets and the property of time reversibility can also be used to find the stationary distribution for Markov chains with countably infinite state spaces.

§ Random walks on undirected graphs: $G=(V,E)$

finite, undirected, connected

Definition 7.9: *A* random walk *on G is a Markov chain defined by the sequence of moves of a particle between vertices of G. In this process, the place of the particle at a given time step is the state of the system. If the particle is at vertex i and if i has $d(i)$ outgoing edges, then the probability that the particle follows the edge (i, j) and moves to a neighbor j is $1/d(i)$.*



$$P = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} & 0 \\ \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}.$$

- Lemma 1. A random walk on an undirected graph G is aperiodic iff G is not bipartite.

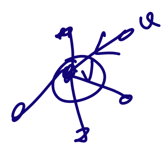
— For the remainder of the section, we assume G to be not bipartite i.e. aperiodic, to use theorem 7.7.

- Lemma 2. A random walk on a non-bipartite G converges to a stationary distribution π, where

$$\pi_v = \frac{d(v)}{2|E|}.$$

- $h_{u,v}$ (hitting time from u to v): Expected time to reach state v, starting at u.

commute time $:= h_{u,v} + h_{v,u}$.



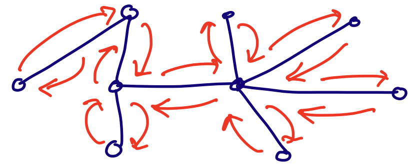
• Defn: Cover time: The cover time of G is the maximum over all vertices $v \in V$ of the expected time to visit all of the nodes in the graph by a random walk starting from v.

• Lemma 3. If $(u,v) \in E$, the commute time
$$h_{u,v} + h_{v,u} \leq 2|E|.$$

• Lemma 4. The cover time of G is bounded above by $2|E|(|V|-1)$.

Proof: choose T, a spanning tree of G.

Starting from any vertex v, consider an Eulerian tour on the spanning tree in which every edge is traversed once in each direction.

follow DFS: 

let $v_0, v_1, \cdots, v_{2|V|-2}$ be the sequence of the vertices starting from $v_0 = v$. ($2(n-1)$ edges)

$$\therefore \text{ cover time } \leq \sum_{i=0}^{2|V|-3} h_{v_i, v_{i+1}}$$
$$= \sum_{(x,y)\in T} (h_{x,y} + h_{y,x})$$
$$\leq 2|E|(|V|-1).$$

lem 3

edges in T.

∎

relates cover time & hitting time

• Matthew's theorem: The cover time C_G of graph G with n vertices is bounded by:

$$H(n-1) \min_{\substack{u,v \in V: \\ u \neq v}} h_{u,v} \leq C_G \leq H(n-1) \max_{\substack{u,v \in V: \\ u \neq v}} h_{u,v},$$

where $H(n) = \sum_{i=1}^{n} 1/i \approx \ln n$, the harmonic number.

§ Application: An s-t connectivity algorithm.

Given: $G = (V, E)$, $n = |V|$, $m = |E|$, $s, t \in V$.

Goal: Determine if there is path connecting s and t.

→ Can be done by BFS, DFS in $O(m+n)$ time, but needs $\Omega(n)$ space.

s–t Connectivity Algorithm:

1. Start a random walk from s.
2. If the walk reaches t within $2n^3$ steps, return that there is a path. Otherwise, return that there is no path.

steps $\leq 2n^3$. Counter $= O(\log n)$ bits,

present vertex, read corr. adj. list to get degree & then read to go to neighbor

– This only needs $O(\log n)$ bits, which is necessary to store vertex indices.

– For technical simplicity, assume G has no bipartite connected components. (to apply lem 2). But can be extended to general graphs.

Theorem: If there is no s-t path, ALGO is correct. Else if there exists an s-t path, ALGO fails w.p. $\leq 1/2$.

Proof: From Lemma 4, cover time $\leq 2mn < n^3$. $\left(\because m \leq \frac{n(n-1)}{2}\right)$

So, expected time T to reach t from s is atmost n^3.

Thus by Markov, prob. of failure:

$$\mathbb{P}(T \geq 2n^3) \leq \frac{\mathbb{E}(T)}{2n^3} = \frac{1}{2}. \quad \blacksquare$$

- This can be thought of a distributed algo. If we give some local memory to each vertex, it can store the parent the first time u reach the vertex. Then the s-t path can be reconstructed.