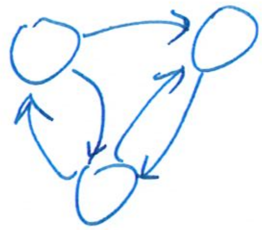
Lecture 6: Random Walks and Markov Chains

One major abstraction we use in randomized algos is that of

markov chains



"states" of algo

and "transitions" between states.

What properties do these have?

- Time to hit "final" node?
- Behavior in long term? (And "short" term)

etc.

So: first some general theorems about MCs. — plus matching also

then talk about random walks on graphs (undirected)

(a special case)

and connections to electrical networks!

— plus 2SAT also.

Arise in other applications too!!

- discuss later in course
- also want to give this as a "way of thinking" / "useful abstraction"

Markov Chain:

"states"

"arcs" or "transitions"

Directed graph $\vec{G} = (V, \vec{A})$ self loops are ok.

Each arc (i,j) has a weight $P_{ij} \geq 0$

such that $\sum_j P_{ij} = 1$.   } transition probabilities

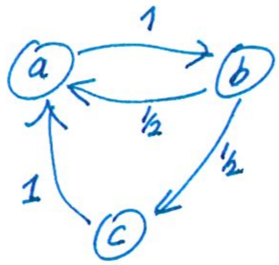
implies that each state has arc going out

Now: defines a process $X_0, X_1, X_2 \ldots X_t \ldots$

such that $Pr(X_{t+1} = j \mid X_t = i) = P_{ij}$ $\forall$ times t independently.

So depends only on the M.Chain and the start state X_0.

E.g.

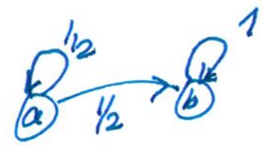


Suppose $X_0 = a$ $\Rightarrow$ $X_1 = b$ wp 1

$X_2 = \begin{cases} a \text{ wp } 1/2 \\ c \text{ wp } 1/2 \end{cases}$

$X_3 = \begin{cases} a \text{ wp } 1/2 \\ b \text{ wp } 1/2 \end{cases}$

$X_4 = \begin{cases} a \text{ wp } 1/4 \\ b \text{ wp } 1/2 \\ c \text{ wp } 1/4 \end{cases}$ ......



$X_0 = a$

$X_1 = \begin{cases} a \text{ wp } 1/2 \\ b \ 1/2 \end{cases}$

$X_2 = \begin{cases} a \text{ wp } 1/4 \\ b \text{ wp } 3/4 \end{cases}$

$\vdots$

$X_t = \begin{cases} a \text{ wp } 1/2^t \\ b \text{ wp } 1 - 1/2^t \end{cases}$

$X_0 = a$

$X_{2i} = a$ $X_{2i+1} = b$.

So at each time t, have $\boxed{\text{prob. distrib } \pi^{(t)} \text{ over states}}$.

Can talk about time average occupancy

$\boxed{a^{(t)} = \frac{1}{t}(\pi^{(1)} + \pi^{(2)} + \cdots + \pi^{(t)})}$.   (both depend on X_0)

$\pi^{(0)}$ = starting probability

{ if start at a single node $i \in [n]$
$\Rightarrow \pi^0(i) = 1 \qquad \pi^0(j) = 0 \ \forall j$
but could be any distribution.

P (matrix) $= (P_{ij})_{i,j=1}^{n}$

then $\pi^{(t+1)} = \pi^{(t)} P$

$$\boxed{\pi_j^{(t+1)} = \sum_i \pi_i^{(t)} P_{ij} = (\pi^{(t)}P)_j}$$

$\Rightarrow \pi^{(t)} = \pi^{(0)} P^t$

is the distribution at time t

—— x ——

<u>Def</u>: π is the stationary probability distribution for MC with transition matrix P
if $\pi = \pi P$ (it is a fixed point for $x \mapsto xP$)

<u>Thm</u>: For any strongly connected MC (i.e. $\exists$ non zero prob to get from i to j eventually)

$\exists$ unique π st $\pi = \pi P$.

<u>Moreover</u>: for any start distribution $\pi^{(0)}$, the limit $\lim_{t\to\infty} a^{(t)} \to \pi$

<u>Def</u>: Call a chain <u>aperiodic</u> if $\forall i$, if start at i, $\exists$ time t_0
one can return to i at any time $t \geq t_0$.

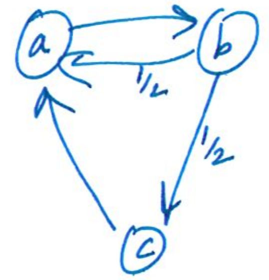
• if strongly connected <u>and</u> aperiodic
$\Rightarrow \pi^{(t)} \to \pi$ (and not just $a^{(t)} \to \pi$)

is not aperiodic
(it is periodic)

this later when argue about "mixing times"

<u>Jargon</u>: strongly connected = "recurrent"

strongly connected + aperiodic = "ergodic".

Eg: for 

we have

$\pi_b = \pi_a$

$\pi_a = \frac{1}{2}\pi_b + \pi_c$

$\pi_c = \pi_b \cdot \frac{1}{2}$ $\qquad \pi_a + \pi_b + \pi_c = 1$

$\Rightarrow \pi_a = \pi_b = 2/5$

$\pi_c = 1/5$.

Can find stationary distribution by solving the linear system

$$\pi = \pi P$$

$$\sum_i \pi_i = 1.$$

— x —

Useful Theorem: Suppose MC is strongly connected (then unique π exists)

Let $r_x = \mathbb{E}[\text{time to return to state } x \mid X_0 = x]$ "return time for x"

then $r_x = 1/\pi_x$.

Loosely: the markov chain spends π_x fraction of its time at x
so it should take $1/\pi_x$ time to return to x on average.

(See Probability texts for proof).

— x —

Let's use this to prove an algorithmic result

- give a fast and simple algo for matchings in bipartite graphs.

Perfect Matchings in Bipartite Regular Graphs

$G = (L, R, E)$ is a bipartite graph.
Each vertex is incident to d edges.
(d-regular graph).

Thm: G has a perfect Matching.

How do you find it?

- Ford Fulkerson? takes $O(mn)$ time naively $\Rightarrow O(n^2 d)$.

- Better analysis can show $O(m\sqrt{n}) = O(n^{1.5} d)$.

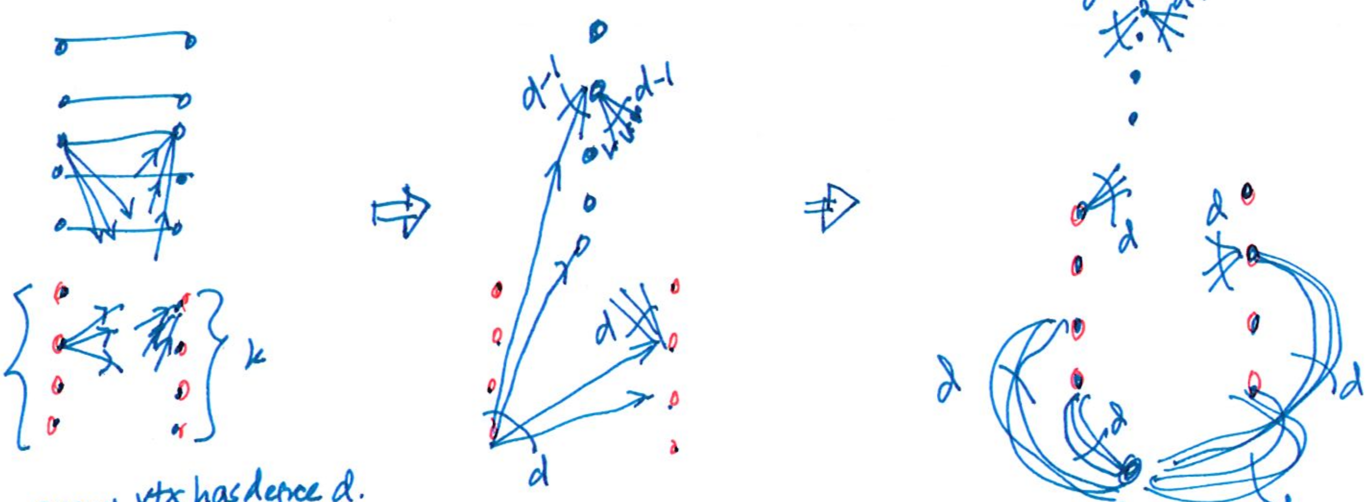
- **Much smarter algorithms:** Finally: Richard Cole + Ost + Schapira gave a linear time $O(m)$ also for this case. (deterministic)

Today: an $O(m + n \log n)$ time algorithm. using randomization [Goel Kapralov Khanna]. for regular bipartite graphs.

Lemma: Suppose we have a matching of size $n-k$ for some $1 \le k \le n$. Then can find an augmenting path using a random walk in $O(\frac{n}{k})$ steps.

$\Rightarrow$ total length of augmenting paths $= O\left(\frac{n}{n} + \frac{n}{n-1} + \cdots + \frac{n}{1}\right) = O(n \log n)$.

Pf:



every vtx has degree d.

Take the graph G

Direct edges from L to R.

Contract all matched edges.

Add a new "source" s

attach it using d parallel edges to each unmatched node, in $L \cup R$

edges from s to L directed out of s

edge from R to s directed into s.

Every unmatched node (except s) has in-degree d

out-degree d.

Every matched node has indegree $d-1$

out-degree $d-1$

node s has indegree dK = outdegree dk.

Let degree of node d_v = indegree = outdegree

<u>Fact</u>: stationary distribution:

$$\pi_v = \frac{d_v}{\sum_w d_w}.$$

<u>Pf</u>: $$\sum_{\substack{u:u\to v\\ \text{edge}}} \pi_u \cdot \Pr(u\to v) = \sum_{u:u\to v} \frac{d_u}{\sum_w d_w}\cdot\frac{1}{d_u}$$

$$= \sum_{u:u\to v} \frac{1}{\sum_w d_w} = \frac{d_u}{\sum_w d_w}$$ ☺

$\Rightarrow E[\text{return time from } s \text{ to itself}] = \frac{1}{\pi_s} \le \frac{dK+nd}{dK} = 1+\frac{n}{K}$ ☺☺

$\Rightarrow$ find augmenting path using a random walk

of length (expected) $\le 1+\frac{n}{K}$.

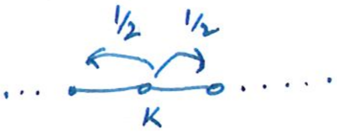
Random Walks, on Graphs

- Consider the simplest setting. Path of length n.

- Each timestep at some location $X_t \in [0,n]$

  then at the next timestep, move to a uniformly random neighbor

  So if $X_t = k$, $X_{t+1} = \begin{cases} k+1 & \text{wp } 1/2 \\ k-1 & \text{wp } 1/2. \end{cases}$ for $k \neq 0$ or n.



  $X_t = 0 \Rightarrow X_{t+1} = 1$ wp 1.

  $X_t = n \Rightarrow X_{t+1} = n-1$ wp 1.

- Suppose start at some location k.

  Want to understand:

  today [(Q1) What is the chance that reach 0 before n? "gambler's ruin"

  (Q2) How long to reach one of these special locations? "hitting time"

  maybe? [(Q3) In the long run, what is the probability of being at j? (ie what fraction of time spent at j) "stationary probability"

  later [(Q4) How quickly do we reach close to this stationary probability? "mixing time"

All these have algorithmic applications ....

$\boxed{\text{Q1}}$ is easy enough: Say p_j = $\underbrace{\text{prob of hitting } n \text{ before } 0}_{}$. starting at j

good event: get ~~jackpot~~ rich before going bankrupt.

$p_n = 1$

$p_0 = 0$

$p_j = \frac{1}{2} p_{j-1} + \frac{1}{2} p_{j+1}$

Simple induction: $p_j = j/n$. ← unique soln (solving a system of linear equations).

$\boxed{\text{Q2:}}$ also easy: say T_j = expected time to hit one of the endpoints starting at j.

$\Rightarrow T_0 = T_n = 0$. ← $n+1$ equations (lin. ind.) in $n+1$ vars

$T_j = \frac{1}{2}(1+T_{j-1}) + \frac{1}{2}(1+T_{j+1})$

Again: check that $T_j = j(n-j)$ is the unique solution

$\Rightarrow$ starting on the line, takes time $\leq n^2/4$ to hit an endpoint.

Great. Let's solve a slightly related problem —

Starting at j what is the expected time H_j, that we hit 0?

$H_0 = 0$ $\qquad H_j = \frac{1}{2}(1+H_{j+1}) + \frac{1}{2}(1+H_{j-1})$

Again, solve to get $H_j = n^2 - (n-j)^2$ ← slightly more tricky.

$H_n = n^2$

$\Rightarrow$ hit 0 in at most n^2 timesteps.

- Try small examples for values of n. Extrapolate the pattern.
- Or use the linear system.

All this is great, but where are we going?

Let's use this to solve an algorithmic task —

$\boxed{\text{2SAT}}$: $\varphi = (x_1 \vee \bar{x}_5) \wedge (x_2 \vee x_9) \wedge \cdots\cdots \wedge (x_3 \vee x_{17})$

What's an algo to find a satisfying assignment, if any?

Algos 101: can solve in poly time.

But let's give a randomized Algo.

Algo [Papadimitriou]

Start with any assignment x_0

While $\exists$ an unsatisfied clause C, pick say the first such clause

not important which clause we pick.

Pick a random var in that clause, flip its value.

Thm: if φ is satisfiable, then find satisfying assignment in $O(n^2)$ timesteps.

Proof: Let x_t be the assignment we have after t steps.

Let x^* be a satisfying assignment ($\exists$ one by assumption)

Let D_t = # bits that differ between x_t & x^* $\qquad D_t = \|x_t - x_0\|_{\text{Hamming}}$

Say $D_0 = k$.

Observe that at each step, we must differ from x^* in at least one of 2 vars in Clause C.

So with prob $\geq 1/2$ we flip that var and reduce $D_t \to D_t - 1$

with prob $\leq 1/2$ we flip the other var, which may increase $D_t \to D_{t+1}$.

Want E[#steps to hit 0] $\geq E$[#steps to find some sat assignment]

So it's almost the problem we studied except

(a) We may stop earlier in case we hit another satisfying assignment $\neq x^*$

(b) Some clauses differ from x^* on both vars, so may actually get $D_t \to D_t - 1$ w.p. 1.

But both these help us (intuitively, at least)

— Should only decrease the time until we stop. (This is true)

Can be formalized — See HW!

so expected time to find sat assignment $\leq O(n^2)$!

——— X ———

(Almost) memoryless algorithm

And very simple.

Slower than $O(m)$ algo but has some advantages.

Suggests ways to extend to 3SAT and get fast (but still exponential) algos

See HW / Exercises

——— X ———

So let's understand these questions some more. (even more generally)

- Given graph G, vertices s & t, what is Pr(hits before t) if we walk "randomly" in the graph?

- What is the expected time to hit t starting from s?

- What is the expected time to see all vertices of the graph?

Random Walks on Graphs

$G = (V, E)$. Say undirected for now; talk about directed later.

- at each time, sitting at $X_t \in V$. (say $X_t = v$)

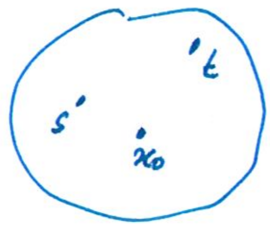
  Pick a uniformly random neighbor u of v and move to it (so that $\forall u \in \partial v$,

  $$\Pr(X_{t+1} = u \mid X_t = v) = \frac{1}{\deg(v)} \cdot \mathbb{1}(u \text{ adjacent to } v))$$

- so if we start $X_0 = x_0$ (say)

  ask the same questions.

  - What is Prob (hit s before t)?



say $p_v = \Pr(\text{hit } s \text{ before } t \text{ starting at } v)$

$$\boxed{\text{then } p_s = 1 \quad p_t = 0. \qquad p_v = \Big(\sum_{u \in \partial v} p_u\Big) \cdot \frac{1}{d_v}}$$

degree of v

- Let's consider an electrical network.

  - Every edge is a 1-ohm resistor.
  - Current follows Ohm's and Kirchoff's Laws.

    - Ohm's Law: if $u \overset{R_{uv}}{\text{—WW—}} v$

      the voltages at u and v are ϕ_u and ϕ_v,

      then the current from u to v is $\dfrac{\phi_u - \phi_v}{R_{uv}}$. (negative means current from v to u)

    - Kirchoff's Law:

      total current flow into a node

      = total current flow out of a node

- Attach a 1 volt battery to s and t.

So consider node $v \neq s, t$,

since all unit-resistances, total flow into v

$$\sum_{u \in \partial v} f_{uv} = 0 \quad \longleftarrow \text{net zero flow}$$

$$\Rightarrow \sum_{u \in \partial v} (\phi_u - \phi_v) = 0 \quad \Rightarrow \quad \sum_{u \in \partial v} \phi_u = d_v \cdot \phi_v$$

$$\Rightarrow \phi_v = \frac{1}{d_v} \cdot \sum_{u \in \partial v} \phi_u \quad \forall v \neq s, t$$

suppose we set $\phi_s = 1$, $\phi_t = 0$.

and we have $\phi_v = \frac{1}{d_v} \sum_{u \in \partial v} \phi_u \quad \forall v \notin s, t.$

Same as hitting probabilities! (p_v)

Indeed satisfy same equations (which has unique solution)

Why unique? Spse has two solutions $\bar{p}$ & $\bar{q}$.

$$\text{So } \underbrace{(p-q)_s = 0 \quad (p-q)_t = 0}_{\text{"boundary conditions } = 0\text{"}} \qquad (p-q)_v = \left(\sum_{u \in \partial v} (p-q)_u \right) \cdot \frac{1}{d_v}$$

↑ each node is the average of its neighbors

Only solution to this $= (p-q)_u = 0 \; \forall u \Rightarrow p = q$.

—— x ——

This electric network analogy is quite powerful.

Consider the hitting time H_{uv} = expected time to get to v starting at u

and commute time $C_{uv} = H_{uv} + H_{vu}$.

<u>Thm</u>: $C_{uv} = 2m \cdot R_{eff}(u,v)$

↑ effective resistance between u/v !

Lemma: assign potential $\begin{cases} +H_{vt} & \text{at all } v \neq t \\ 0 & \text{at } t. \end{cases}$

current flow out of $v = \sum_{u \in \partial v} \frac{\phi_v - \phi_u}{R_{vu}} = \sum_{u \in \partial v} (H_{vt} - H_{ut})$

$= d_v.$

$$\left(\begin{array}{l} \text{but } H_{vt} = 1 + \frac{1}{d_v} \sum_{u \in \partial v} H_{ut} \\ \Rightarrow \sum_{u \in \partial v} (H_{ut} - H_{vt}) = d_v. \end{array} \right)$$

$\Rightarrow$ d_v current leaves each $v \neq t$.

All of it must reach t $\Rightarrow$ t gets $\sum_{v \neq t} d_v = 2m - d_t$ current.

☺

Similarly: assign potential $\begin{cases} -H_{vs} & \text{at all } v \neq s \\ 0 & \text{at } s \end{cases}$

an identical calculation shows.

d_v current enters each $v \neq s$

$2m - d_s$ current leaves s.

Summing them up

gives 0 current entering or leaving $v \neq s, t$

$2m = (2m - d_s) + d_s$ current leaves s

$2m = d_t + (2m - d_t)$ current enters t.

And net potential/voltage difference between s & t is $H_{st} + H_{ts}$

$= C_{st}$

So this potential diff causes $2m$ current flow.

thus effective resistance between s & t must be $\frac{C_{st}}{2m} = R_{eff}(s, t)$ ∎

E.g.: 

$\mathrm{Reff}(0,n) = n$

$\Rightarrow C_{on} = \underset{\substack{\| \\ n}}{2m} \cdot \mathrm{Reff}(0,n) = 2n \cdot n = 2n^2$

By symmetry: $\boxed{H_{on} = \frac{1}{2} C_{on} = n^2.}$

$\boxed{\text{Q: What about } H_{io}\text{?}}$

$C_{oi} = 2n \cdot i = H_{oi} + H_{io}$

But H_{oi} in this graph is the same as H_{oi} in the graph 
(path of length i)

$= i^2$

$\Rightarrow C_{oi} = i^2 + H_{io} = 2n \cdot i$

$\Rightarrow H_{io} = (2n-i)\,i = n^2 - (n-i)^2$ ☺

Can use this to show:-

- Start from any vertex in an connected n vertex m-edge graph G
  - Do random walk

  $E[\text{time to see all vertices}] \le 2m(n-1)$.

- Gives memoryless connectivity checker for graphs!