Random walks

Example: Randomized algorithm for 2-SAT

2-SAT: Given a 2-CNF formula Φ, i.e. a formula of $\underline{m}$ clauses on $\underline{n}$ variables of the form

$$(x_2 \vee \sim x_3) \wedge (\sim x_1 \vee \sim x_4) \wedge (x_3 \vee x_8) \wedge \cdots$$

– is Φ satisfiable?

Rand. 2-SAT Algorithm

– Start with an arbitrary assignment $x \in \{0,1\}^n$

– while there is any unsat clause do:

– Pick an unsat clause c

– Pick u.r. one of two var's of c and flip it's value.

– Suppose Φ is satisfiable and let $\underline{A}$ be a satisfying assignment.

– Values of var's as in A: correct value

n: # of var's.

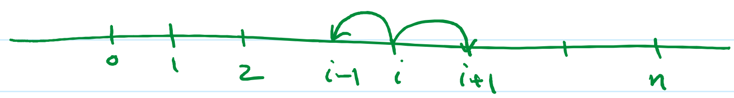
X_i: # of var's in the current iteration that are correct (w.r.t. A).

– Algorithm stops when $X_i = n$; all X_i's ≥ 0.

- Algorithm stops when $X_i = n$; all X_i's ≥ 0, each iteration X_i changes by 1:

$$|X_i - X_{i+1}| = 1$$

- It is like a random walk on a line with a barrier at 0; each step we move one step to left/right.



Goal: time to reach n

$$\Pr[X_{i+1} = 1 \mid X_i = 0] = 1$$

- let $1 \leq X_i \leq n-1$ and consider a random unsat clause $\underline{C}$. It disagrees with A in at least one var.

- We increase # of agreements with probabability $\geq \frac{1}{2}$

$$\begin{cases} \text{both disagree} \longrightarrow \text{inc. by 1 w.p. } 1 \\ \text{one disagree} \longrightarrow \text{inc. by 1 w.p. } \frac{1}{2} \end{cases}$$

$\longrightarrow$ $\Pr[X_{i+1} = j+1 \mid X_i = j] \geq \frac{1}{2}$ (and $\Pr[X_{i+1} = j-1 \mid X_i = j] \leq \frac{1}{2}$)

- Is this a Markov's chain? No, because the probability of going from one state to another depends on the assignment of clauses (and so prev.

states).

- Trick: We can define another random walk that is a Markov's chain & is a pessimistic estimator for X_i.

- $Y_0, Y_1, \ldots$ where

$$Y_0 = X_0, \quad \Pr[Y_{i+1} = 1 \mid Y_i = 0] = 1$$
$$\Pr[Y_{i+1} = j+1 \mid Y_i = j] = \frac{1}{2}$$
$$\Pr[Y_{i+1} = j-1 \mid Y_i = j] = \frac{1}{2}$$

- Easy to see: Expected time for Y to reach n is $\geq$ expected time for X.

- Z_j : # of steps to reach $\underline{n}$ from $\underline{j}$

h_j : $E[Z_j]$.

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

- We also have: $h_0 = h_1 + 1$, $h_n = 0$

- We use induction to show: $h_i = h_{i+1} + 2i + 1$.

- Base : $i=0$. Now induction step:

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

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

$$= 2h_j - (h_j + 2(j-1)+1) - 2$$

$$= h_j - 2j - 1$$

$$\rightarrow h_0 = \sum_{i=0}^{n-1} (2i+1) = n^2$$

- So if Φ is satisfiable $\longrightarrow$ Expected # of steps $\leq n^2$.
- So if we run the algorithm for $2n^2$ steps:

  find a sol with Prob. $\geq \frac{1}{2}$ if it exists

- repeat this α times: find a sol in $2\alpha n^2$ steps

  w.p. $1 - 2^{-\alpha}$.

Random walks on Graphs

- Applications:
  - approximation/estimation (perm, Volume)
  - rand. Selection
  - derandomizing rand. algorithms
  - Probability amplification; pseudo-rand generation.

* Start from a vertex $s \in V$
* repeat for T steps
  * choose a neighbor v u.r.

$* \; u \leftarrow v$

- Interesting qualities to look at:

H_{uv}: Hitting time, from u to v is

$$E[\# \text{ of steps to reach } v \mid \text{start at } u]$$

C_{uv}: Commute time from u to v and back

$$C_{uv} = H_{uv} + H_{vu}$$

C_u: Cover time =

$$E[\# \text{ of steps to visit all vertices} \mid \text{start at } u]$$

$$C_G = \max_u C_u$$

Application: Undirected s-t connectivity (USTConn)

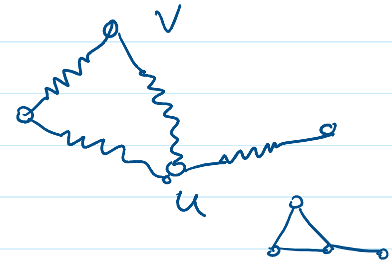
Given undirected $G(V,E)$ and nodes s,t, are they connected?

- Can easily solve in $O(m+n)$ using DFS/BFS
- What if the graph is very large?
- We can solve this using randomness in $O(\log n)$ space! (i.e. randomized log-space; so USTConn $\in$ RL. Reingold in 2005 in a breakthrough showed this can be done in deterministic time!

USTConn $\in$ L

Analysis of random walks using resistence graph

- View G as electrical network
- Each edge has resistence 1

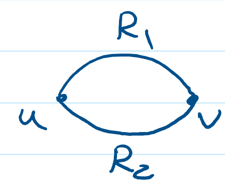


Kirchoff's law: total current going into a point = total current going out

Ohm's law: $V = R \cdot I$

Series: $R = R_1 + R_2$

Parallel: $\frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2} \rightarrow R = \frac{R_1 R_2}{R_1 + R_2}$



Definition: Effective resistence between u & v is the potential difference (i.e. V) between u & v when one unit of flow is going from u to v.

Theorem: Given G, think of each edge as a unit resistence. Let R_{uv} be effective resistence between u & v

$$C_{uv} = 2m R_{uv}$$

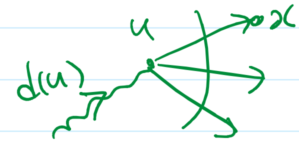
proof: Suppose we take a battery an inject deg(x) units of flow into each node x s.t. we get 2m units of flow out of v.

let ϕ_{uv}: diff. of voltage between u & v

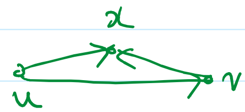
u $\rightarrow$ x

let ϕ_{uv} : diff. of voltage between u & v

$$\deg(u) = \sum_{ux \in E} (\text{current from } u \text{ to } x)$$



$$= \sum_{ux \in E} \phi_{ux}$$



$$= \sum_{ux \in E} (\phi_{uv} - \phi_{xv})$$

$$= \deg(u) \cdot \phi_{uv} - \underline{\sum_{ux \in E} \phi_{xv}} \longrightarrow$$

$$\phi_{uv} = 1 + \sum_{ux \in E} \frac{\phi_{xv}}{\deg(u)} \qquad (1)$$

- let $d(u)$ denote degree of u & consider the first step of a random walk. Then:

$$H_{uv} = 1 + \sum_{ux \in E} \frac{H_{xv}}{d(u)} \qquad (2)$$

- Note that (1) & (2) are the same linear equations for all the nodes. $\longrightarrow$ $H_{uv} = \phi_{uv}$

- Doing the same thing in reverse (injecting from each node & getting $2m$ from u): $H_{vu} = \phi'_{vu}$

- Now in this second experiment reverse all flow. So $2m$ going into v & $d(x)$ going out of all nodes.

- Combine the two:

    - $2m$ units going into u.

- $2m$ units going into u
- $2m$ units going out of v
- in/out flow from every other node $= 0$

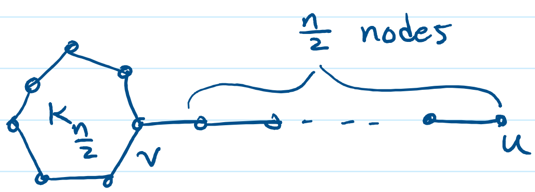
$$\longrightarrow \phi_{uv} + \phi'_{vu} = H_{uv} + H_{vu} = 2m \cdot R_{uv} = C_{uv}$$

Example: Random walk on a path P_n

$R_{ij} = j - i \qquad H_{ij} + H_{ji} = 2(n-1)(j-i)$

$H_{1n} + H_{n1} = 2(n-1)^2$

Example: Lollipop graph L_n



$m = \frac{n}{2} + \binom{\frac{n}{2}}{2} \in \Theta(n^2)$

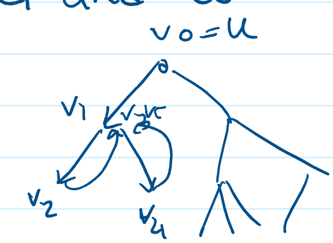
$R_{uv} = \frac{n}{2} \qquad H_{uv} + H_{vu} = 2m R_{uv} \in \Theta(n^3)$

$H_{uv} = \left(\frac{n}{2}\right)^2 \in \Theta(n^2)$

$\longrightarrow H_{vu} \in \Theta(n^3)$

Theorem: $C(G) \leq 2m(n-1)$

Proof: let T be some Spanning Tree of G and do a DFS: $u = v_0, v_1, v_2, \dots, v_{2n-2} = u$



$$C_u \leq H_{v_0 v_1} + H_{v_1 v_2} + \dots + H_{v_{2n-3} v_{2n-2}}$$

$$= \sum C_{vw} = \sum 2m R_{vw} \leq 2m(n-1)$$

$$= \sum_{vw \in T} c_{vw} = \sum_{vw \in T} 2m \underbrace{R_{vw}}_{\leq 1} \leq 2m(n-1)$$

Example: For line/Path $c_G \in O(n^2)$

For K_n, $c_G \in O(n^3)$