Random walks on undirected graphs

$G=(V,E)$ undirected graph

consider simple random walk on this graph:

$p_{ij} = \frac{1}{d_i} \quad \forall (i,j) \in E$ [MC is periodic iff graph is bipartite]
[but if so, consider lazy r.w.]

$\boxed{\pi_i = \frac{d_i}{2m} \qquad m: \#\text{ of edges}}$

$\sum_i \frac{d_i}{2m} = 1$

$\pi_j = \sum_i \pi_i p_{ij}$

$\frac{d_j}{2m} = \sum_{\substack{i \text{ s.t.} \\ (i,j)\in E}} \frac{d_i}{2m} \cdot \frac{1}{d_i}$

Some key quantities:

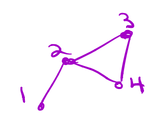
hitting time	$h_{ij} = E(T_{ij})$
commute time	$c_{ij} = h_{ij} + h_{ji}$
covertime	$C(G) =$ exp. time to visit all vertices

$h_{ii} = \frac{1}{\pi_i} = \frac{2m}{d_i}$

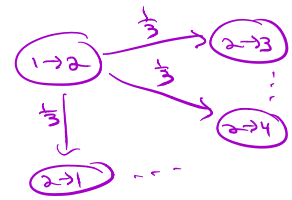
Lemma $\forall$ edge (i,j) $\quad h_{ij} + h_{ji} \leq 2m$

Pf: Consider corresponding random walk on directed edges (states are $i\to j$, $j\to i$ $\forall (i,j)\in E$) $2m$ states

transition probabilities: $q_{(i\to j,\, \ell\to r)} = \begin{cases} \frac{1}{d_j} & \ell = j \\ 0 & \text{o.w.} \end{cases}$



state at time t

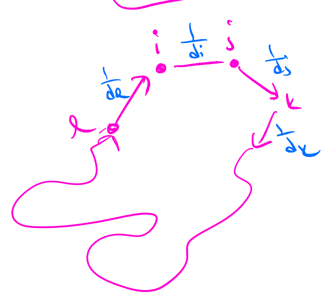


$Q = (q_{\vec{e},\vec{e}'})$ is doubly stochastic

col sum for $i\to j$ $= \sum_{k \text{ s.t. } (k,i)\in E} q_{k\to i,\, i\to j} = \sum_{k \text{ s.t. } (k,i)\in E} \frac{1}{d_i} = 1$

$\Rightarrow$ π is uniform, i.e. $\pi_{\vec{e}} = \frac{1}{2m}$ $\forall \vec{e}$

$\Rightarrow$ $h_{i\to j,\, i\to j} = 2m$



Corollary

$\forall\, G=(V,E)$ connected nonbipartite (or lazy) $\quad C(G)$ = expected cover time of random walk on G $\leq 2m(n-1)$

Pf: Let T be a spanning tree on G and let $\underset{(u_1,v_1)}{e_1},\dots,\underset{(u_n,v_n)}{e_{n-1}}$ be the edges in the tree.

Consider doubled tree T^*, where each edge is duplicated once in each direction.

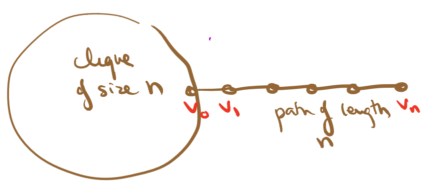
Every vertex has indegree = outdegree
$\Rightarrow$ has Euler tour say

$\underset{u_1\to v_1}{\vec{e}_1}\ \underset{v_1\to v_2}{\vec{e}_2} \cdots \underset{v_1\to u_1}{\overleftarrow{e}_1} \cdots$

$$E(\text{cover time}) \leq \sum_{i=1}^{n-1} E(T_{u_i v_i} + T_{v_i u_i}) = (n-1)2m$$

Examples

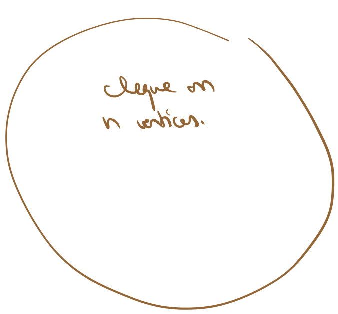
$C(G) = \Theta(n^2)$



$m = O(n^2)$

$C(G) = O(n^3)$

Fact: Starting from v_1, $Pr(\text{reach } v_n \text{ before returning to } v_0) = \frac{1}{n}$



Application: s-t connectivity

Given G undirected graph $s,t \in V$

determine if s & t are in same CC.

DFS / BFS $\begin{cases} O(m) \text{ time} \\ O(n) \text{ space} \end{cases}$ keep track of all vertices search has visited so far

Observation: very simple randomized alg using logspace

(input on separate read-only tape

Simulate r.w. of length $4mn$ on G starting from s

$\Pr($r.w. doesn't reach t when $\exists$ path$) \le \frac{1}{2}$

Both algorithms $\tilde{O}(mn)$ space·time

Can we interpolate?

Monte Carlo Methods

collection of tools for estimating values thru sampling & estimation

(ε, δ) Approximation

A randomized alg gives an (ε,δ) approx for value V
if the output X of the alg satisfies

$$\Pr(|X - V| > \varepsilon |V|) \le \delta$$

Example

Sample indep random vars whose mean is quantity we want
to estimate

Let $X_1, X_2, \ldots, X_m$ iid. Bernoulli with $E(X_i) = \mu$

If $m \geq \frac{3 \ln(\frac{2}{\delta})}{\varepsilon^2 \mu}$ then

$$\Pr\left(\left| \frac{1}{m} \sum_{i=1}^{m} X_i - \mu \right| \geq \varepsilon\mu \right) \le \delta$$

Pf: Chernoff bounds

Monte Carlo Thm

DNF Counting

Suppose want to know # satisfying assignments

$$(\bar{x}_1 \wedge x_2) \vee (x_2 \wedge x_3) \vee (x_1 \wedge x_2 \wedge \bar{x}_3 \wedge x_4) \vee (x_3 \wedge \bar{x}_4)$$

Obviously satisfying such a formula is easy
Counting # satisfying assignments is hard
If could do this, could solve 3SAT

Problem actually #P-complete $\longrightarrow$ strong intractability
#P is counting analogue of NP

problems of form:
compute $f(x)$ where $f(x)$
is # solutions to problem x in NP

Approximate DNF Counting?

Obvious approach: sample random assignments, indep m times

$X_i = \begin{cases} 1 & i^{th} \text{ random assignment satisfies } \varphi \\ 0 & \text{o.w.} \end{cases}$

return $\dfrac{\sum_{i=1}^{m} X_i}{m} \cdot 2^n$ as estimate

$\boxed{\exists \text{ FPRAS for DNF counting}}$

Fully polynomial randomized approx scheme

aka: a randomized alg for which, given an input x and any parameters ε & δ with $0 < \varepsilon, \delta < 1$, the alg outputs an (ε, δ) approx to $V(x)$ in time poly in $\frac{1}{\varepsilon}$, $\ln \frac{1}{\delta}$ & size of input

in this example x is DNF formula, $V(x)$ is # satisfying assignments

$\boxed{\bar{V} \text{ is } (\varepsilon,\delta) \text{ approx to } V \equiv \Pr\left(|\bar{V} - V| > \varepsilon V\right) \leq \delta}$

DNF counting illustrates fundamental connection

sampling $\leftrightarrow$ counting

From approx sampling $\longrightarrow$ approx counting

Counting Independent Sets

Input: x, ε $\longrightarrow$ **FPAUS** Fully polynomial almost uniform sampler $\longrightarrow$ ε uniform sample ω from $\Omega(x)$

$\forall S \quad \left| \Pr(\omega \in S) - \frac{|S|}{|\Omega|} \right| \le \varepsilon$

runs in time polynomial in size of x & $\ln(\frac{1}{\varepsilon})$

Want to show:

Given FPAUS for indep sets $\Longrightarrow$ can construct FPRAS for ISs

FPRAS

makes sequence of calls to FPAUS$(x_1,\varepsilon_1)(x_2,\varepsilon_2)\cdots$

runs in time poly in $\frac{1}{\varepsilon}$, size of input.

$G=(V,E)$ $\qquad e_1, e_2, \ldots, e_m$ arbitrary ordering of edges

$E_i = \{e_1, e_2, \ldots, e_i\}$ $\qquad G_i = (V, E_i)$ $\qquad G_m = G$ $\qquad G_0 = (V, \phi)$

$\Omega(G_i)$: set of ISs in G_i

$$|\Omega(G)| = \frac{|\Omega(G_m)|}{|\Omega(G_{m-1})|} \times \frac{|\Omega(G_{m-1})|}{|\Omega(G_{m-2})|} \times \cdots \times \frac{|\Omega(G_1)|}{|\Omega(G_0)|} \times |\Omega(G_0)|$$

So need to bound $\quad R = \prod_{i=1}^{m} \frac{\tilde{r}_i}{r_i}$

$\boxed{\text{Claim:}}$ If $\tilde{r}_i$ is $\left(\frac{\varepsilon}{2m}, \frac{\delta}{m}\right)$ approx to r_i $\forall i$
then X is (ε, δ) approx to $|\Omega(G)|$

Pf: $\quad \Pr\left(|\tilde{r}_i - r_i| \le \frac{\varepsilon}{2m} r_i \right) \ge 1 - \frac{\delta}{m} \quad \forall i$

$\Rightarrow r_i\left(1 - \frac{\varepsilon}{2m}\right) \le \tilde{r}_i \le r_i\left(1 + \frac{\varepsilon}{2m}\right) \quad$ w.p. $\ge 1 - \frac{\delta}{m}$

$\Rightarrow \left(1 - \frac{\varepsilon}{2m}\right) \le \frac{\tilde{r}_i}{r_i} \le \left(1 + \frac{\varepsilon}{2m}\right)$

$1 - \varepsilon \le \left(1 - \frac{\varepsilon}{2m}\right)^m \le \prod_{i=1}^{m} \frac{\tilde{r}_i}{r_i} \le \left(1 + \frac{\varepsilon}{2m}\right)^m \le 1 + \varepsilon \quad$ w.p. $\ge 1 - \delta$

$(1-\varepsilon) \prod_{i=1}^{m} r_i \le \prod_{i=1}^{n} \tilde{r}_i \le (1+\varepsilon) \prod_{i=1}^{m} r_i \quad$ w.p. $1 - \delta$

$\equiv \Pr\left(\left| \prod_{i=1}^{m} \tilde{r}_i - \prod_{i=1}^{m} r_i \right| \ge \varepsilon \prod_{i=1}^{m} r_i \right) \le \delta$

To get $\tilde{r}_i$ $\left(\frac{\varepsilon}{2m}, \frac{\delta}{m}\right)$-approx for r_i,

use FPAUS for ISs

Idea: (approx) sample indep sets in G_{i-1} &
compute fraction of these that are indep in G_i

Input: $G = (V, E)$, ε $\longrightarrow$ $\boxed{\text{FPAUS: Fully polynomial almost uniform sampler}}$ $\longrightarrow$ ε uniform sample ω from Ω: $\left| \Pr(\omega \in S) - \frac{|S|}{|\Omega|} \right| \le \varepsilon$

runs in time polynomial in input size (G) & $\ln\left(\frac{1}{\varepsilon}\right)$

For this to work, need $r_i = \dfrac{|\Omega(G_i)|}{|\Omega(G_{i-1})|}$ not too small (no needle in haystack)

Claim: $r_i \ge \frac{1}{2}$

$\Rightarrow$ with polynomially many calls to FPAUS, we can get good approx $\tilde{r}_i$ to r_i using Monte Carlo Thm

Two errors we need to bound:

1. FPAUS $\neq$ exact sampler so exp value $\neq r_i$

   But using, say, $\frac{\varepsilon}{6m}$ sampler $|E(\tilde{r}_i) - r_i| \leq \frac{\varepsilon}{6m}$

2. With samples we get approx to $E(\tilde{r}_i)$

   since r_i big ($\geq \frac{1}{2}$), $E(\tilde{r}_i)$ big $\Rightarrow$

   Monte Carlo Thm $\Rightarrow$ $O\left(\frac{m^2}{\varepsilon^2} \ln\left(\frac{1}{\delta}\right)\right)$ samples suffice

__Thm:__ Given a FPAUS for sampling ISs, one can construct a FPRAS for ISs

Approach works for many "self-reducible" problems

Another example: counting # matchings in a graph

Again: $E = \{e_1, \ldots, e_m\}$

$G_i = (V, E_i)$ where $E_i = \{e_1, \ldots, e_i\}$

$$|M(G)| = \frac{|M(G_m)|}{|M(G_{m-1})|} \frac{|M(G_{m-1})|}{|M(G_{m-2})|} \cdots \frac{|M(G_2)|}{|M(G_1)|} |M(G_1)|$$

Like before: $\frac{|M(G_i)|}{|M(G_{i+1})|} \geq \frac{1}{2}$

Big question remains: how to construct approx sampler?

How to sample elts from a universe Ω according to some distn π?

Cool idea! design MC whose state space is Ω that has stationary distn π

- simulate MC until it "mixes"
- use state at that time as sample

2 key questions:

1. how to design chain with right π?
2. how to bound mixing time?

Example: Sampling indep sets uniformly from $G=(V,E)$

States: indep sets

X_t: some indep set

MC: in each step
choose vertex v u.a.r. from V
if $v \in X_t$, then $X_{t+1} := X_t \setminus v$
if $v \notin X_t$ & can be added w/ violating independence, then $X_{t+1} := X_t \cup v$
Otherwise $X_{t+1} := X_t$