# Markov Chains and Random Walks

Petros Potikas

Randomized Algorithms 2010

# Outline

- 2-SAT algorithm
- Markov Chains
- Random Walks
- Electrical Networks
- Graph Connectivity
- Expanders
- Random Walks on Expanders
- Probability Amplification by Random Walks on Expanders

# A 2-SAT algorithm

*Def.* k-SAT is the special case of SAT where each clause has exactly k literals.

Example:

$$(x_1 \vee x_2) \wedge (\neg x_1 \vee x_3) \wedge (\neg x_2 \vee \neg x_3)$$

- For k>2, it is NP-hard.
- For k=1, k=2 it is in P.

# A 2-SAT algorithm

*2-SAT Algorithm:*

1. Start with an arbitrary truth assignment.
2. Repeat up to $2mn^2$ times, terminating if all clauses are satisfied:
   a) Choose an arbitrary clause that is not satisfied
   b) Choose uniformly at random one of the literals in the clause and switch its value
3. If a valid truth assignment has been found, return true.
4. Otherwise, return that the formula is unsatisfied.

# Analysis of the 2-SAT algorithm

*Th.* The expected number of steps of the above 2-SAT algorithm to find a satisfying assignment is $O(n^2)$.

*Analysis:* Let A be a particular satisfying assignment. The progress of the algorithm can be represented by {0,1,…,n}, where the i-th position indicates how many variables in the current solution have the correct values. With probability at least ½ we move from i to i+1 in an unsatisfied clause. Thus it resembles a random walk in the line.

# Markov Chains

*Def.* A Markov chain M is a discrete time stochastic process defined over a set of states S in terms of a matrix P of transition probabilities.

- S={1,…,n}: set of states
- $X_t$: random variable: state of the system in time step t. $X_0$ is chosen according to some probability distribution.
- $P_{ij}=Pr[X_{t+1} = j \mid X_t = i]$

# Markov Chains

- Memorylessness property:

$\Pr[X_{t+1}=j \mid X_0 = i_0, \ldots, X_t=i] = \Pr[X_{t+1}=j \mid X_t=i]=P_{ij}$

- $r_{ij}^{(t)} = \Pr[X_t = j$, and for $1 \leq s \leq t-1, X_s \neq j \mid X_0=i]$

$$f_{ij} = \sum_{t>0} r_{ij}^{(t)}$$

$$h_{ij} = \sum_{t>0} t r_{ij}^{(t)}$$ Hitting time

- If $f_{ij} < 1$, then $h_{ij} = \infty$ (but not the converse)

# Markov Chains

*Def.* A state i with $f_{ii} < 1$ is said to be *transient*, and one for which $f_{ii} = 1$ is said *persistent*. Those persistent states i for which $h_{ii}=\infty$ are said *null persistent*, while the other *non-null persistent*.

- Underlying directed graph: $(S,\{(i,j): P_{ij} > 0\})$

- Strong component: with some probability it reaches all other vertices, final strong component Pr=1

# Markov Chains

*Def.* A Markov chain is said to be *irreducible* whenever its underlying graph consists of a single strong component.

- The unique strong component in an irreducible Markov chain must be final, and hence all states are persistent.

*Def.* $q^{(t)} = (q_1^{(t)}, q_2^{(t)}, \ldots, q_n^{(t)})$: *state probability vector* (distribution of the chain at time t)

$$q^{(t+1)} = q^{(t)}P$$

# Markov Chains

*Def.* A *stationary distribution* for the Markov chain with matrix P is a probability distribution π s.t. $\pi = \pi P$.

- Aperiodic: $\forall T>1, q^{(0)}, a, i$: $\exists t$: $q_i^{(t)}>0$ and $t \bmod T \neq a$
- Periodic Markov chains do not converge to the stationary distribution!

*Def.* An *ergodic* state is one that is aperiodic and non-null persistent.

# Markov Chains

*Th.* (Fundamental Theorem of Markov chains): Any irreducible, finite and aperiodic Markov chain has the following properties.

1. All states are ergodic.
2. There is a unique stationary distribution $\pi$ s.t., for $1 \leq i \leq n$, $\pi_i > 0$.
3. For $1 \leq i \leq n$, $f_{ii}=1$ and $h_{ii}=1/\pi_i$
4. Let $N(i,t)$ be the number of times the Markov chain visits state i in t steps. Then,

$$\lim_{t \to \infty} N(i,t)/t = \pi_i$$

# Random walks

- Let G=(V,E) be a non-bipartite, connected, undirected graph, n=|V|, m = |E|.
- A random walk on G induces the Markov chain $M_G$ with $P_{ij}=1/d(i)$, for $(i,j) \in E$ and $P_{ij}=0$ otherwise.

*Lemma*: For all $v \in V$, $\pi_v = d(v)/2m$ defines a unique stationary distribution.

*Proof*: The j-component of πP is

$$\sum_{i=1}^{n} \pi_i P_{ij} = \sum_{i \in (i,j) \in E} \frac{1}{2m} = \frac{d(j)}{2m}$$

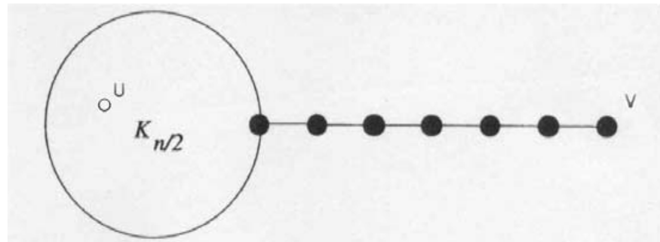
- G is aperiodic because it is non-bipartite, so π is unique

# Random walks

*Lemma*: For all $v \in V$, $h_{vv} = 1/\pi_v = 2m/d(v)$

- $h_{uv}$: expected number of steps in a random walk starting at u and first reaching v
- Commute Time $C_{uv} = h_{uv} + h_{vu}$
- $C_u(G)$: expected length of a walk starting at u and visiting all $v \in V$
- Cover time $C(G) = \max_u C_u(G)$

# Random walks

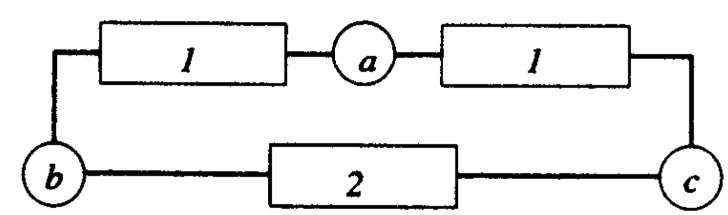



- Lollipop graph:
  - –A clique on $n/2$ vertices
  - –A path on $n/2$ vertices
  - –Let $u,v \in V$, $u$ is in the clique, $v$ is at the far end of the path.

  1. Surprisingly, $h_{uv} \neq h_{vu}$ ($h_{uv}$ is $\Theta(n^3)$ $h_{vu}$ is $\Theta(n^2)$)
  2. Cover time is not monotone in the number of edges (chain has cover time $\Theta(n^2)$)

# Electrical networks

- Resistive electrical network: an undirected graph, each edge has a positive real branch resistance
- Kirchhoff's Law, Ohm's Law
- *Effective resistance* $R_{uv}$ *between any two vertices u,v: the voltage between u and v, when one ampere is injected into u and removed from v*



# Electrical networks

- Given an undirected graph G, let N(G) be the electrical network: a node for every vertex in V and for every edge in E, one ohm resistance between the corresponding nodes.

*Th*. For any $u,v \in V$, $C_{uv} = 2mR_{uv}$

- Effective resistance $\leq$ shortest path $\leq$ diameter

*Corollary*: In any n-vertex graph, and for all $u,v \in V$,

$$C_{uv} < n^3$$

# Cover Time

*Th.* $C(G) \leq 2m(n-1)$

- Lollipop: $C(L_n)=O(n^3)$, $C(K_n)=\Theta(n\log n)$

- Refine the upper bound.

Let $R(G)=\max_{u,v\in V}R_{uv}$ , called *resistance* of G.

*Th*. $mR(G) \leq C(G) \leq 2e^3mR(G)\ln n+n$

# Graph Connectivity

- STCON: Given a graph G and two vertices s and t in G, decide whether s and t are in the same connected component.
- Important in space-bounded complexity classes and a natural abstraction of a number of graph search problems.

DFS: $O(m)$ time and $\Omega(n)$ space.

# Graph Connectivity

- A probabilistic log-space Turing machine (TM) for a language L is a probabilistic Turing machine using space O(logn) on instances of size n, and running in time polynomial time in n.

- A language A is in RLP if there exists a probabilistic log-space TM M s.t. on any input x,

$$\Pr[M \ \textbf{accepts}\ x] = \begin{cases} \geq 1/2 & x \in A \\ 0 & x \notin A \end{cases}$$

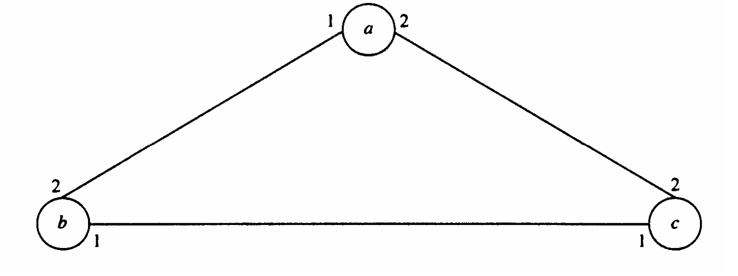
# USTCON $\in$ RLP

*T*h. USTCON $\in$ RLP.

*Pr*. The log-space probabilistic TM simulates a random walk of length $2n^3$.If it encounters t in the walk it outputs YES, o.w. NO. The probability that t is in the same component with s and outputs NO is at most ½, by the Markov inequality and $h_{st}=n^3$. It uses O(logn) space.

# USTCON

- Turn this randomized algorithm to a deterministic algorithm: use the class of non-uniform, deterministic, log-space algorithms called *universal traversal sequences*.




σ=(1,2,1,1,2)

- A sequence σ is said to be *universal traversal sequence* for a class of labeled graphs if it traverses every labeled graph in the class.

# USTCON

- A universal traversal sequence with polynomial length in n can be used by a deterministic log-space TM as follows: the sequence is stored in the finite-state control of the TM and is used to traverse G starting from s on an instance of USTCON.

Let $\mathcal{G}$ be a family of connected, labeled, d-regular graphs on n vertices.

- U($\mathcal{G}$): length of the shortest universal traversal sequence for all the labeled graphs in $\mathcal{G}$
- R($\mathcal{G}$): maximum resistance between any pair of vertices.

*Th.* $U(\mathcal{G}) \leq 5mR(\mathcal{G})\log(n|\mathcal{G}|)$

# USTCON

- U(n,d): length of the shortest universal traversal sequence for connected, labeled, d-regular, n-vertex graphs

*Corollary:* U(d,n) = $O(n^3 d \log n)$

- So there is a deterministic log-space TM that decides USTCON. We proof that there exists a universal traversal sequence (by probabilistic method), but we cannot constructed it.

# Directed Graphs

## STCON Algorithm

1. Starting at s, simulate a random walk of n-1 steps. Each step consists of choosing an edge leaving the current vertex uniformly at random. It t is reached, output YES. If the walk reaches a vertex with no outgoing edge, or a vertex other than t after n-1 steps, return to s. It needs O(logn) bits.
2. Flip $\log n^{n}$ unbiased coins. If they all come up HEADS, halt and output NO. This needs a counter for the coins, so O(logn) bits.

# Directed Graphs

*Th.* The above algorithm, will given an instance of STCON

1. Always output NO, if there is no path from s to t.
2. Output YES with probability at least ½ if there is a path from s to t.

Requires O(logn) space.

# Expanders

- Informally, an expander is a graph that any set of vertices S has neighborhood large relative to S.
- Sparse expander graphs: linear number of edges
- Deciding whether a given graph is an expander is co-NP-complete.
- We use algebraic graph theory to describe its properties.

# Expanders and Eigenvalues

Let G=(V,E) be an undirected, multipgraph, |V|=n.

When G is bipartite: two independent sets of vertices $X=\{v_1,\ldots,v_n\}$ and $Y=\{v_{n/2+1},\ldots,v_n\}$.

A(G): adjacency matrix

$$A(G) = \begin{bmatrix} 0 & B \\ B^T & 0 \end{bmatrix}$$

B: edges between X, Y

A(G) is symmetric, eigenvalues $\lambda_1 \geq \ldots \geq \lambda_n$, and eigenvectors $e_1,\ldots,e_n$ (orthonormal basis)

# Expanders

*Th.* (Fundamental Th. of Algebraic Graph Theory): Let G=(V,E) be an n-vertex, undirected multigraph with maximum degree d. Then, under the canonical labeling of eigenvalues $\lambda_i$ and orthonormal eigenvectors $e_i$ for A(G),

1. If G is connected, then $\lambda_2 < \lambda_1$.
2. For $1 \leq i \leq n$, $|\lambda_i| \leq d$.
3. d is an eigenvalue iff G is regular.
4. If G is d-regular, then $\lambda_1 = d$ has $e_1 = 1/\sqrt{n}(1,1,\dots,1)$.
5. The graph G is bipartite iff for every eigenvalue $\lambda$ there is an eigenvalue $-\lambda$ of the same multiplicity.
6. Suppose G is connected. Then, G is bipartite iff $-\lambda_1$ is an eigenvalue.
7. If G is d-regular and bipartite, $\lambda_n = -d$ and $e_n = 1/\sqrt{n}(1,1,\dots,-1,\dots,-1)$.

# Special Class of Expanders

*Def*. An (n,d,c)-expander is a d-regular bipartite multigraph $G(X,Y,E)$ with $|X|=|Y|=n/2$ s.t. for any $S \subseteq X$,

$$|\Gamma(S)| \geq (1+c(1-2|S|/n))|S|$$

- Gabber-Galil: For $m \in N$, let $n=2m^2$.
- Vertex: a distinct label (a,b), for $a,b \in Z_m$.
- Edges: a vertex (x,y) in X is connected to vertices (x,y), (x,x+y), (x,x+y+1), (x+y,y), (x+y+1,y).
- It gives family of (n,5,a)-expanders, with $a=(2-\sqrt{3})/4$
- Similar, (n,7,2a)-expander: edges (x,y), (x,2x+y), (x,2x+y+1), (x,2x+y+2), (x+2y,y), (x+2y+1,y), (2x+y+2,y).

# Special Class of Expanders

- There is polynomial time neighborhood algorithm of any given vertex

*Th.* If G is an (n,d,c)-expander, then A(G) has

$$|\lambda_2| \leq d - (c^2/1024+2c^2)$$

*Th.* If A(G) has $|\lambda_2| \leq d-\varepsilon$, then G is an (n,d,c)-expander with

$$c \geq (2d\varepsilon - \varepsilon^2)/d^2$$

# Expanders

- For a d-regular graph $G=(V,E)$, define

$$\text{split}(G)=\min_{\emptyset\subset X\subset V}(|e(X,V\backslash X)|/|X||V\backslash X|)$$

where $e(A,B)$: multiset of edges of G between sets A,B.

*Th*. If G is d-regular, then

$$\text{split}(G)\geq(d-\lambda_2)/n$$

*Corollary*: If G is d-regular then for any $W\subset V$,

$|W \cup\Gamma(W)| \geq[1+(1-\lambda_2/d)/2]|W|$

# Random Walks on Expanders

- Let G be a (n,d,c)-expander. Random walk: for k edges between u and w, the probability that a random walk goes from v to w is k/d(v). This corresponds to a Markov chain with P=A(G)/d, and the eigenvalues are $\lambda_i$/d same eigenvectors.

- G is periodic: Overcome this, adding a self loop at each vertex with probability ½ (reduce all transition probabilities by 2). Then, Q=(I+P)/2, $\lambda_i' = (1+\lambda_i/d)/2$
Thus, $1=\lambda_1' \geq \lambda_2' \geq \ldots \geq \lambda_n'$ and assuming $\lambda_2 = d-\varepsilon$, we have $\lambda_2' = 1-\varepsilon/2d$.

# Rapidly mixing

- The Markov chain defined by Q, is “rapidly mixing”, i.e. converges to the stationary distribution in a small number of steps, starting from an initial distribution.

*Def.* Let $q^{(t)}$ be the state probability vector of a Markov chain defined by Q at time $t>0$, given an initial distribution $q^{(0)}$. Let $\pi$ denote the stationary distribution of Q. The *relative pointwise distance* of the Markov chain at time t is a measure of deviation from the limit and is defined as

$$\Delta(t)=\max_i|q_i^{(t)}-\pi_i|/\pi_i$$

# Rapidly mixing

*Th*. Let Q be the transition matrix of the aperiodic random walk on a (n,d,c)-expander G with $\lambda_2 \leq d-\varepsilon$. Then, for any initial distribution $q^{(0)}$, the relative pointwise distance is bounded by:

$$\Delta(t) \leq n^{1.5}(\lambda_2')^t \leq n^{1.5}(1-\varepsilon/2d)^t$$

This show that the relative pointwise distance of the random walk on an expander converges to zero at an exponential rate.

# Random Walks on Expanders

- For any $0<\delta<1$, let $T(\delta)$ denote the time at which the relative pointwise distance of the random walk defined by Q first falls below $\delta$. Then

$$T(\delta) \leq (\log n^{1.5}/\delta)/(-\log\lambda_2')$$

- So, to get a relative pointwise distance that is bounded from above by an inverse polynomial in n run the random walk only a logarithmic number of steps. Best possible, since the length of the random walk must at least the diameter of the graph. In expander graphs, the diameter is $\Omega(\log n)$.

# Probability Amplification by Random Walks on Expanders

Probability amplification: Randomness-efficient error reduction of randomized algorithms. From a constant error rate randomized algorithm, to a randomized algorithm with exponentially small error with a small number of random bits.

# Probability Amplification by Random Walks on Expanders

The class BPP (Bounder-error Probabilistic Polynomial time) consists of all languages L that have a randomized polynomial time algorithm A s.t. for any $x \in \Sigma^*$, given a suitably long random string r,

- $x \in L \Rightarrow \Pr[A(x,r) \text{ rejects}] \leq 1/100$
- $x \notin L \Rightarrow \Pr[A(x,r) \text{ accepts}] \leq 1/100$

Fix an input x, and consider a BPP algorithm A that uses n random bits on inputs of length |x|. Suppose we choose k independent n-bit random strings $r_1,\ldots,r_k$ and compute $A(x,r_1),\ldots,A(x,r_k)$. By Chernoff bounds, Pr[majority is incorrect] = $1/2^{\Omega(k)}$ .

nk random bits used. What is the minimum number of random bits?

# Probability Amplification by Random Walks on Expanders

Idea: Take Gabber-Galil expanders, with vertices from $\{0,1\}^n$. Start from a uniform random vertex a random walk of constant length. For every vertex on the random walk, we will run the algorithm with this vertex as its random string. Output as final decision the majority of these decisions. This will give an error that is exponentially small using as few random bits as possible (derandomization).

Sampling from an expander walk is as good as sampling independently.

# Probability Amplification by Random Walks on Expanders

- Consider the (N,7,2a)-expander. Choose $m=2^{(n-1)/2}$ and $N=2m^2=2^n$, label each vertex with a distinct sequence of $\{0,1\}^n$. Let A be the adjacency matrix of the expander.
- Let $Q=(I+A/7)/2$ be the probability transition of the ergodic Markov chain. Denote by $X_0$, $X_1$,… the states of the Markov chain.
- Choose a positive integer β, s.t. $\lambda_2^{\beta} \leq 1/10$. β=O(1)
- Scheme: For $0 \leq i \leq 7k$, let $r_i = X_{i\beta}$ $(X_0, X_\beta, \ldots)$. Run the algorithm A(x,*) using these 7kβ different choices of random inputs. The majority of these 7k YES/NO decisions is the final decision. n bits are needed to choose an initial vertex and 4 bits for each of the 7kβ steps of the random walk.

# Probability Amplification by Random Walks on Expanders

Intuition: The random strings are constructed by the states of the random walk on the expander. The random walk on an expander is rapidly mixing.

Composition of 7k different random walks, starting resp. from $X_0, X_\beta, \ldots, X_{7k\beta}$, each generating a different random string $r_i$. Each random walk has length $\beta = O(1)$, instead of $O(\log N)$. On the other hand, we choose the initial vertex according to the stationary distribution, and this works for us.

# Probability Amplification by Random Walks on Expanders

$p^{(i)}$: the probability distribution vector for $r_i=X_{i\beta}$.
$B=Q^{\beta}$: transition matrix for the Markov chain to the sequence of $r_i$'s.

$W=\{r\in\{0,1\}^n \mid A(x,r\}$ is correct$\}$   $|W|\geq 0.99N$

0-1 NxN diagonal matrix **W**, s.t. $W_{ii}=1$ iff the i-th vertex corresponds to a string that is a witness for x and **W**'=I-**W**.

Event sequence $S=(S_1,S_2,\ldots,S_{7k})\in\{\mathbf{W},\mathbf{W}'\}^{7k}$ be s.t. $S_i=\mathbf{W}$ iff $r_i\in W$

S: encodes the pattern of errors in the executions of the algorithm

# Probability Amplification by Random Walks on Expanders

*Lemma*: For any fixed event sequence S,

$$\Pr[S \text{ occurs}] = ||p^{(0)}(BS_1)(BS_2)\ldots(BS_{7k})||_1$$

*Lemma*: For all vectors $p \in R^N$

$||pB\mathbf{W}|| \leq ||p||$

$||pB\mathbf{W}'|| \leq 1/5||p||$

# Probability Amplification by Random Walks on Expanders

*Th*. The probability that the majority of the outputs $A(x,r_1),\ldots,A(x,r_{7k})$ is incorrect is at most $1/2^k$.

*Proof:*

By a Chernoff Bound for expander walks. Half of the elements in S must equal to **W**'. Fix any particular S whose elements contain a majority of **W**'s, $\kappa \geq 7k/2$ of them.

$$\Pr[S \text{ occurs}] = \|p^{(0)}(BS_1)(BS_2)\ldots(BS_{7k})\|_1$$

# Probability Amplification by Random Walks on Expanders

*Proof (Cont):*

$$\begin{aligned}\Pr[\text{S occurs}] &= \|p^{(0)}(BS_1)(BS_2)\ldots(BS_{7k})\|_1 \\ &\leq \sqrt{N}\|p^{(0)}(BS_1)(BS_2)\ldots(BS_{7k})\| \\ &\leq \sqrt{N}(1/5)^{\kappa}\|p^{(0)}\| \\ &\leq \sqrt{N}(1/5)^{7k/2}\|p^{(0)}\|\end{aligned}$$

We chose $p^{(0)}$ uniform on the N vertices, so $\|p^{(0)}\| = 1/\sqrt{N}$.

The number of sequences S with a majority of **W**' is at most $2^{7k}$, so

$$\begin{aligned}\Pr[\text{Majority vote is incorrect}] &\leq 2^{7k}\sqrt{N}(1/5)^{7k/2}\|p^{(0)}\| \\ &\leq 2^{7k}(1/5)^{7k/2} \leq 1/2^{k}\end{aligned}$$

# Probability Amplification by Random Walks on Expanders

| | Number of repetitions | Number of random bits |
|---|---|---|
| Independent Repetitions | O(k) | O(km) |
| Pairwise Independent Repetitions | $O(2^k)$ | O(k+m) |
| Expander Walks | O(k) | m+O(k) |

# References

1. Randomized Algorithms, By R. Motwani and P. Raghavan.
2. Probability and Computing - Randomized Algorithms and Probabilistic Analysis By Michael Mitzenmacher, Eli Upfal
3. http://people.seas.harvard.edu/~salil/cs225/lecnotes/lec8.ps
4. A Chernoff Bound for Random Walks on Expander Graphs, Gillman D., SIAM J. Comput., 1998.