B.Tech. Seminar

Rapidly Mixing Markov Chains

Submitted in partial fulfillment of the requirements
for the degree of
Bachelor of Technology

by
Rahul T. Shah
93005004

under the guidance of
Dr. Ketan Mulmuley
and
Dr. Sundar Vishwanathan

Department of Computer Science and Engineering
Indian Institute of Technology
Bombay
1996

Acknowledgement

Abstract

The computational difficulty of the problems of counting the elements of a finite set of combinatorial structures has led to the development of randomized algorithms. The problem of generating at random an element of a set with some property is central for various randomized algorithms because of its connection with counting problem. A technique based on Markov chain is used to generate uniformly at random the element of set. This leads to polynomial time randomized algorithm if the Markov chain is rapidly mixing.

Contents

Chapter 1

Introduction

Counting the elements of finite set of combinatorial structures and generating them uniformly at random from some probability distribution are two classical computational problems. These problems occur in various branches of mathematics and physical sciences. A classic example is to count number of spanning trees in graph. What makes the counting problem particularly challenging is that the size of the finite set is typically exponential in the size of the its specification. The fundamental problem that has been studied extensively ,and is still open, is the problem of computing permanent. Other counting problems are integrating a given function and computing the volume of convex body.

The complexity of counting problems was studied systematically by Valiant, who introduced class $\#P$ for counting problems which is analogous to class NP for decision problems. Valiant proved the fundamental result that computing permanents is complete for the class $\#P$. The $\#P$-complete problems can be considered hopeless to compute in polynomial time. But on the positive side we have that these intractability only applies to exact counting.

1.1 Some Basic Concepts

The problem instances, such as specification of graph, can be encoded in binary in the form of bit string x. We also specify a relation R which associates the problem instance x to the set of its solutions $R(x)$.
So, if y is encoding of some solution then $y \in R(x)$, R may be seen as binary relation $\langle x, y \rangle$.

For example

$$R = \{\langle x, y \rangle : x \to \text{boolean formula in DNF}$$
$$y \to \text{satisfying assignments of the formula}\}$$

A relation $R \subseteq \Sigma^* \times \Sigma^*$ is a *p-relation* if

1. there exists a polynomial $p(n)$ such that $\langle x, y \rangle \in R \Rightarrow |y| \leq p(|x|)$,

2. the predicate $\langle x, y \rangle \in R$ can be tested in deterministic polynomial time.

Here $\Sigma = \{0, 1\}$.

Definition 1.1 *A counting problem is a function $f \in \{0, 1\}^* \to N \bigcup \{0\}$ which maps problem instance x to its number of solutions $f(x) = |R(x)|$*

The function f is said to be #P-complete iff

1. f belongs to class $\#P$ i.e. can be computed by Non Deterministic Turing Machine in polynomial time and

2. Any function g in $\#P$ can be computed by deterministic turing machine equipped with an oracle for f in polynomial time.

Typical members of this class $\#P$-complete are the counting problem for which the decision problems are NP-complete. But there are many combinatorial problems which are apparently unrelated to any NP-complete problems, but still the counting problems are $\#P$-complete. We use approximation algorithms for such problems. The model of computation for such algorithms is *probabilistic turing machine* (PTM). A probabilistic turing machine has a output tape and special coin-tossing states. All the transitions of the turing machine are deterministic except for coin tossing states where decisions are taken by tossing of fair coin.

An algorithm for estimating the answer to a counting problem is considered to be a good algorithm if given an error parameter ϵ with confidence parameter δ, it outputs an estimate with relative error at most ϵ with confidence $1 - \delta$, in time bounded by some polynomial in $1/\epsilon, \log 1/\delta$ and the length of input. Such an algorithm is called *fully polynomial randomized approximation scheme* (**fpras**).

A seemingly unrelated problem to the counting problem is the (uniform) generation problem : pick a random element of a finite set characterized by

by some property. But if we can generate the solutions almost uniformly the under some assumptions we may be able to count them approximately. This holds for self-reducible relations [Sin93]. For most of the problems of our interest we would be satisfied with uniform generation.

PTM's can be used as generators of solutions of a problem instance x. We say that a PTM M is a *uniform generator* for the relation $R \subseteq \Sigma^* \times \Sigma^*$ iff

1. there exists a function $\phi \in \Sigma^* \to (0,1]$ such that, for all $x, y \in \Sigma^*$,

$$\Pr(\text{given input } x, M \text{ outputs } y) = \begin{cases} 0 & \text{if } \langle x, y \rangle \notin R \\ \phi(x) & \text{if } \langle x, y \rangle \in R \end{cases}$$

2. for all inputs $x \in \Sigma^*$ such that $\{y \in \Sigma^* \mid xRy\}$ is nonempty,

$$\Pr(M \text{ accepts } x) \geq \frac{1}{2}.$$

We define here the notion of randomized approximate counting. Suppose $f \in \Sigma^* \to \mathbf{N}$. A *randomized approximation scheme* for f is a PTM M which, for all inputs $\langle x, \epsilon, \delta \rangle$ produces an output $M(x, \epsilon, \delta)$ (a random variable of the coin-tossing sequence of M) such that

$$\Pr\left(M(x, \epsilon) \text{ approximates } f(x) \text{ within ratio } (1 + \epsilon)\right) \geq 1 - \delta$$

A fully polynomial randomized approximation scheme is one whose execution time is bounded by polynomial in $|x|, 1/\epsilon, \log 1/\delta$

The Markov chain technique focuses on the problem of uniform generation. We simulate a Markov chain whose states are solutions of the problem. The Markov chain converges to stationary distribution on solution space. If the Markov chain mixes rapidly then we can get almost uniform generation in polynomial time. For getting rapid mixing property, we use the concept of conductance of chain , which in some sense measures worst bottleneck in the stationary chain. For this, we use the method of canonical path.

In chapter 3 we use the canonical path technique to bound the conductance of chain for perfect matchings in graph. We effectively use the Markov chain technique to approximate the value of permanent of dense 0-1 matrix.

A second set of techniques for lower-bounding the conductance are geometric and they make use of isoperimetric inequalities. This technique is used in chapter 4 to derive algorithm for approximating the volume of convex body.

3

Chapter 2

Markov Chains and Rapid Mixing

In this chapter, we introduce Markov chains, which can be used for almost uniform generation. The approach is based on simulating simple dynamic process called finite Markov chain, which moves around a space containing the structures of interest and converges to some desired distribution on them. Rapid mixing property, which is is required to get convergence in polynomial time will be introduced later. The material presented here if taken from [Sin93] and [Vaz91].

2.1 Introduction to Markov chains

In the theory of Markov Chains we consider a set of possible outcomes $E_1, E_2, \ldots$ (finite or infinite in number). We consider a sequence of trials. The outcome of any trial depends on the outcome of directly preceding trial. We define it as follows.

Definition 2.1 *A sequence of trials with possible outcomes $E_1, E_2, \ldots$, is called a Markov Chain if probability distribution of random variable X_k, which gives outcome of kth trial, is given by*

- $Pr(X_0 = E_j) = a_j$

- $Pr(X_k = E_j | X_{k-1} = E_i) = p_{ij}, k \geq 1$

Here a_i is the initial distribution over possible outcomes (which we call states) and p_{ij} is the transition probability from state E_i to E_j. Note that both are independent of t.

4

Let the sequence of random variable $(X_t)_{t=0}^{\infty}$ be a Markov chain on a finite state space $[N] = \{0, 1, \dots, N-1\}$, $N \geq 1$, with *transition matrix* $P = (p_{ij})_{i,j=0}^{N-1}$. The matrix P is non-negative and *stochastic*, i.e. its row sums are all unity. For $s \in \mathbf{N}$, the *s-step transition matrix* is simply the power $P^s = (p_{ij}^{(s)})$. Thus $p_{ij}^{(s)} = \Pr(X_{t+s} = j \mid X_t = i)$, independent of t. We denote the distribution of X_t by the row vector $\boldsymbol{\pi}^{(t)} = (\pi_i^{(t)})_{i=0}^{N-1}$, so that $\pi_i^{(t)} = \Pr(X_t = i)$. Here $\boldsymbol{\pi}^{(0)}$ denotes the *initial distribution*, and $\boldsymbol{\pi}^{(t)} = \boldsymbol{\pi}^{(0)} P^t$ for all $t \in \mathbf{N}$. Usually, we will have $\pi_i^{(0)} = 1$ for some $i \in [N]$ (and 0 elsewhere); i is then called the *initial state*.

The chain is called *ergodic* if there exists a distribution $\boldsymbol{\pi}' = (\pi_i) > \mathbf{0}$ over $[N]$ such that

$$\lim_{s \to \infty} p_{ij}^{(s)} = \pi_j, \forall i, j \in [N].$$

In this case, we have that $\boldsymbol{\pi}^{(t)} = \boldsymbol{\pi}^{(0)} P^t \to \boldsymbol{\pi}'$ pointwise as $t \to \infty$, and the limit is independent of $\boldsymbol{\pi}^{(0)}$. The *stationary distribution* $\boldsymbol{\pi}'$ is the unique vector satisfying $\boldsymbol{\pi}' P = \boldsymbol{\pi}'$, $\sum_i \pi_i = 1$, i.e. the unique normalized left eigenvector of P with eigenvalue 1.

Necessary and sufficient conditions for ergodicity are that the chain should be

1. irreducible , i.e. for each pair of states $i, j \in [\mathbf{N}]$, there is an s such that $p_{ij}^{(s)} > 0$ (j can be reached from i in finite number of steps)

2. aperiodic, i.e. $\gcd\{s \mid p_{ij}^{(s)} > 0\} = 1$ for all $i, j \in [N]$.

2.2   Defining the Convergence

Suppose now that we wish to sample elements of the state space, assumed very large, according to the stationary distribution $\boldsymbol{\pi}'$. The desired distribution can be realized by picking an arbitrary initial state and simulating the transitions of the Markov Chain according to the probabilities p_{ij}, which we assume can be computed locally as required. As the number t of simulation steps increases, the distribution of the random variable X_t approaches $\boldsymbol{\pi}'$. Clearly, for this process to be effective it is necessary to know a priori how many steps are required to achieve a distribution sufficiently close to $\boldsymbol{\pi}'$ for our purposes, or in other words we have to bound the rate of convergence of the chain.

In order to formalize the notion of convergence, we define the the following time-dependent measure of deviation from the limit:

The *relative pointwise distance* (r.p.d.) over a non-empty subset $U \subseteq [N]$ after t steps by

$$\Delta_U(t) = \max_{i,j \in U} \frac{|p_{ij}^{(t)} - \pi_j|}{\pi_j}.$$

Thus $\Delta_U(t)$ is the largest relative difference between $\boldsymbol{\pi}^{(t)}$ and $\boldsymbol{\pi}'$ at any state $j \in U$, maximized over all possible initial states $i \in U$. The inclusion of the parameter U merely allows us to specify that certain portions of the state space are not relevant in the sampling process. The aim of the next few sections is to obtain useful upper bounds on Δ_U as a function of t. In particular, we want to find conditions under which convergence is *rapid* in the sense that $\Delta_{[N]}(t)$ becomes very close to 0 while $t \ll N$. This is referred to as the *rapid mixing* property.

2.3   Rapid Mixing

Here we give the precise definition of rapid mixing property which was mentioned informally earlier.

Let $\Delta(t)$ denote the r.p.d. of Markov chain $\mathcal{MC}$ over its entire space after t steps. We define the function $\tau : R \to N$ by

$$\tau(\epsilon) = \min \{t \in N : \Delta(t') \le \epsilon, \forall t' \ge t\}$$

The Markov chain is rapidly mixing iff there exists a polynomially bounded function $q : N \times R^+ \to N$ such that

$$\tau(\epsilon) \le q(\log N, \log \epsilon^{-1})$$

for all ϵ between 0 and 1.

Example: Consider the problem of generating uniformly at random the members of a set $R(x)$, i.e. different solutions for a problem instance defined by x. For this we start from some solution of the problem which belongs to the state space of Markov chain. Then we simulate t number of steps, where t is bounded by $q(|x|, log(2\epsilon)^{-1})$. Now we may get some solution. Thus we have almost uniform generator.

The conditions under which we get fully polynomial almost uniform generator are:

6

1. The construction problem can be solved in polynomial time to obtain the initial state.

2. Individual steps of Markov chains can be performed in polynomial time.

3. In the stationary distribution of chain the probability of being at some element of $R(x)$ (i.e. some solution) is bounded below by $1/p(|x|)$ for some polynomial p. This means that number of auxiliary states (which do not represent the solution) is not too numerous.

4. The function q is polynomially bounded.This means that Markov chain is rapid mixing.

In examples we see, the conditions (1) and (2) are easily satisfied. The conditions (3) and (4) would be interesting ones.

2.4   Underlying Graph

An ergodic Markov Chain is said to be time reversible iff

$$\forall i, j \in [N], p_{ij}\pi_i = p_{ji}\pi_j.$$

This informally means that expected number of transitions per unit time from state i to state j and from state j to state i are equal in stationary distribution.

We can identify an ergodic reversible chain with a weighted undirected graph called *underlying graph*.The vertex set of this graph is the *state space* [N] of the chain. For any pair i, j the weight of edge (i, j) is $w_{ij} = \pi_i p_{ij} = \pi_j p_{ji}$. The chain can be uniquely specified by underlying graph. Note that this graph contains self-loops.

2.5   Conductance and Rate of Convergence

Here we shall define structural property called conductance of underlying graph. Essentially, a Markov chain will turn out to converge fast *if and only if* the conductance is not too small. The central point of the proof is relation between conductance and second eigen value of transition matrix of chain. To define conductance, it is useful to view our Markov Chain as a random

7

walk on a underlying graph. The conductance of Markov Chain, in some sense , measures the worst bottle-neck in the underlying graph.

Proposition 2.1 *Let P be the transition matrix of an ergodic time-reversible Markov Chain, $\boldsymbol{\pi}'$ its stationary distribution and $\{\lambda_i \mid 0 \le i \le N - 1\}$ its (necessarily real) eigenvalues, with $\lambda_0 = 1$. Then for any non-empty subset $U \subseteq [N]$ and all $t \in \mathbf{N}$, the relative pointwise distance $\Delta_U(t)$ satisfies*

$$\Delta_U(t) \le \frac{\lambda_{max}^t}{\min_{j \in U} \pi_j}$$

where $\lambda_{max} = max\{|\lambda_i| : 1 \le i \le N - 1\}$.

We shall not prove the above Proposition here. The reader is referred to [Sin93] for a proof.

Proposition 2.1 implies that chain will converge rapidly if the smallest π_i is not too small and λ_{max} is bounded away from 1. We shall discuss the issue of bounding λ_{max}. If P has eigenvalues $1 = \lambda_0 > \lambda_1 \ge \lambda_2 \ldots \ge \lambda_{N-1} > -1$ then $\lambda_{max} = max(\lambda_1, |\lambda_{N-1}|)$. The negative eigenvalues give oscillatory or "near periodic" behavior. In particular case, if the graph is bipartite, then the distribution depends on parity of number of steps. So, to avoid this problem, we add self-loop probability to each state. If we ensure that $min_j P_{jj} \ge 1/2$ then we get non-negative eigenvalues, all lying between 0 and 1.

Proposition 2.2 *Let P be the transition matrix of an ergodic time-reversible Markov Chain, and $1 = \lambda_0 > \lambda_1 \ge \cdots \ge \lambda_{N-1} > -1$ its eigenvalues. Then the modified chain with transition matrix $P' = \frac{1}{2}(I_N + P)$ is also ergodic and time-reversible with the same stationary distribution, and its eigenvalues $\{\lambda_i'\}$, similarly ordered, satisfy $\lambda_{N-1}' > 0$ and $\lambda_{max}' = \lambda_1' = \frac{1}{2}(1 + \lambda_1)$.*

From the above discussion, it is sufficient for rapid mixing to bound the *second eigenvalue* λ_1 away from 1. To do this we will relate λ_1 to a more easily computable structural property of the underlying graph.

Intuitively, we would expect an ergodic chain to converge rapidly if it is unlikely to 'get stuck' in any small subset S of the state space whose total stationary probability is quite small. This idea is formalized by demanding that the cut edges which separate S from the rest of the space must be capable of supporting a sufficiently large 'flow' in the graph, viewed as a network.

So we consider cut edges with separate S (some subset) from rest of the space of the underlying graph. We define the following:

$$
\begin{aligned}
C_s &= \textstyle\sum_{i \in S} \pi_i && \text{the } \textit{capacity} \text{ of } S \\
F_s &= \textstyle\sum_{i \in S, j \in \bar{S}} p_{ij} \pi_i && \text{the } \textit{ergodic flow} \text{ out of } S \\
\Phi_S &= F_S / C_S
\end{aligned}
$$

Φ_S may be viewed as the conditional probability that the chain crosses the cut from S to $\bar{S}$ in a single step, given that it starts in S. The *conductance* of the chain is hence defined as

$$
\Phi = \min_{C_S \leq 1/2} \Phi_S
$$

For time-reversible chain, the conductance may be viewed as structural property of the underlying graph G. It may be viewed as $\Phi = \Phi(G)$ and may be given by

$$
\Phi(G) = \min_{|S| \leq \frac{|N|}{2}} \left\{ \frac{\sum_{i \in S, j \in \bar{S}} w_{ij}}{\sum_{i \in S} \pi_i} \right\} \text{ where denominator} \geq 1/2
$$

In some sense it measures the minimum relative connection strength between "small" subsets S and rest of the space. It is related to rate of convergence of chain.

We state, without proof, the bound imposed by the conductance on the second eigenvalue λ_1.

Lemma 2.3 *For an ergodic time-reversible Markov Chain with underlying graph G, the second eigenvalue λ_1 of the transition matrix satisfies*

$$
\lambda_1 \leq 1 - \frac{\Phi(G)^2}{2}
$$

Combining Propositions 2.1 and 2.2 and Lemma 2.3, we arrive at the desired bound on the convergence of a Markov Chain.

Theorem 2.4 *Let G be the underlying graph of an ergodic time-reversible Markov Chain, modified if necessary as in Proposition 2.2 to ensure that $\min_j p_{jj} > \frac{1}{2}$, and $\boldsymbol{\pi}'$ its stationary distribution. Then for any non-empty subset $U \subseteq [N]$ and all $t \in \mathbf{N}$, the relative pointwise distance $\Delta_U(t)$ satisfies*

$$
\Delta_U(t) \leq \frac{(1 - \Phi(G)^2/2)^t}{\min_{i \in U} \pi_i}
$$

9

Thus, the number of steps required for an ergodic Markov Chain to approach stationarity is $O(\Phi(G)^{-2} \log(1/\pi_{\min}))$, where $\pi_{\min}$ is the minimum stationary probability of any state. So we can see that convergence is rapid if conductance $\Phi(G)$ is not too small.

2.6 Conductance and Edge Magnification

Many Markov Chains can be viewed as simple random walk on graph $H = (V, E)$. Here the transitions are made from any vertex v to its adjacent vertex with probability β/d, where d is the maximum vertex degree of H and $\beta \leq 1$ is a positive constant. In addition v has additional self loop probability $1 - \beta deg(v)/d$, where $deg(v)$ is the degree of v in H.

Note that in the underlying graph all edges of of H have equal non-zero weight and all other edges have weight zero.

For any subset $S \subseteq V$, let $E_{s,\bar{s}}$ denote the cut set in H defined by S then edge magnification $\mu(H)$ of the graph H is defined as

$$\mu(H) = \min_{0 < |S| \leq \frac{|V|}{2}} \frac{|E_{s,\bar{s}}|}{|S|}$$

Proposition 2.5 *Let G be the underlying graph of an ergodic random walk on graph H with minimum degree d and transition probabilities β/d between distinct adjacent states. Then the conductance of G is given by*

$$\Phi(G) = \beta\mu(H)/d$$

So, to show that conductance is not too small it may be sufficient to show that edge magnification is not too small.

2.7 A Simple Example

Here we take the example of random walk on *n-dimensional hypercube.* n-dimensional hypercube consists of graph with 2^n vertices which are given by bit strings of length n.There is an edge between two vertices if they differ exactly one position.(if hamming distance is 1). So, it is an n-regular graph i.e. all vertices have degree 1.

Let $H(n)$ be n-dimensional hypercube. We add self loop probability of $1/2$ to each vertex.The transition probability from one vertex to its neighboring vertex is $1/2n$.So,here we see that $\beta = 1/2$.

10

Theorem 2.6 *The magnification of n-dimensional hypercube satisfies* $\mu(H(n)) \geq 1$.

Proof :
The proof uses canonical path technique. Consider a collection of paths P_{ij}, one between every pair of vertices. It is possible to specify a canonical simple path between each ordered pair of vertices.Let $N = 2^n$ be the number of states(vertices).

Congestion: The congestion of an edge $e \in E$ in graph $G(V, E)$ is the number of canonical paths that contain e.

Conductance and congestion are related.Intuitively, a bottleneck in a graph cause a high congestion and low conductance.

Claim: In directed graph $G(V, E)$,where $N = |V|$, if bN is maximum congestion through an edge then

$$\mu \geq 1/2b$$

Proof of the claim: If S is any subset of states with $0 < |S| \leq N/2$ then number of paths crossing S to $\bar{S}$ is

$$|S||\bar{S}| \geq |S|N/2$$

for any such S the number of cut edges is bounded below by

$$|S|N/2bN = |S|/2b$$

To prove that conductance is large it is sufficient to find canonical paths with small congestion. so now what remains to be proved is that $b \leq 1/2$.

For this we introduce injective mapping technique.
We define the canonical path from $u = (u_i)_0^{n-1}$ to $v = (v_i)_0^{n-1}$ as follows: Let u and v differ in positions $i_1 < .. < i_l$.Then for $1 \leq j \leq l$, the jth edge of canonical path from u to v corresponds to transition in which i_jth bit is flipped from u_{i_j} to v_{i_j}.Consider now an arbitrary transition t of the $\mathrm{MC}(n)$ from w to w'. Our aim is to bound the number of canonical paths containing t.Let $P(t)$ denote the set of canonical paths containing t.
We define an injective mapping σ_t from $P(t)$ to state space $B(n)$. The mapping $\sigma_t : P(t) \rightarrow B(n)$ is defined as follows: given an ordered pair $\langle u, v \rangle \in P(t)$,set $\sigma_t(u, v) = (s_i)$, where

$$s_i = \begin{cases} u_i & 0 \leq i \leq k; \\ v_i & k < i < n \end{cases}$$

Here t is flipping kth bit.

s_i is also called complementary point.

u and v can be obtained from the knowledge of w, w' and s. So, $|P(t)| \leq N$.But we observe that s_k is always equal to w_k.Hence we get $|P(t)| \leq N/2$. Hence we get $b \leq 1/2$.

Example

Let $n = 7$ and consider canonical path from $u = 1011001$ and $v = 0001111$. path has the vertices :

$$1011001$$
$$0011001$$
$$0001001$$
$$0001101$$
$$0001111$$

Now consider transition t from vertex $0011001 \rightarrow 0001001$

This transition flips the 3rd bit so $k = 3$. So earlier bits in w match to v and later to u. Therefore, we construct complementary point which remembers the lost bits of u and v.

Hence $s = 1011111$.

We can get back u and v from t and s.

The canonical paths should be designed so that the amount of information about the initial vertex plus the final vertex remains constant along the path. In some cases we may require some additional bits of information to reconstruct the end points.

2.8   Summary

In this chapter, we showed how Markov chain approach can be used to generate almost uniformly the different solutions of some combinatorial problem (each solution can be seen as an element of state space). To generate almost uniformly in polynomial time we introduced rapid mixing property of Markov chain.We viewed the Markov chain as random walk on graph. Through various theorems we showed that

high magnification $\Rightarrow$ lower $\lambda_{max} \Rightarrow$ faster convergence

$\Rightarrow$ rapid mixing $\Rightarrow$ f.p. almost uniform generator.

In the end we illustrated the technique by simple example.We shall take more complicated example in the next chapter.

Chapter 3

Approximating the Permanent

In this chapter we present the approximation algorithm for computation of permanent of 0-1 matrices, which is a $\#P$ complete problem.It is same as counting the number of perfect matchings in a bipartite graph.

The algorithm uses Markov chain which converges to uniform distribution over the space of perfect matchings for any given bipartite graph. We show it converges in polynomial time for the *dense* bipartite graph. But the problem is still an open problem for non dense graphs. Material presented here is from [Sin93] and [Bro86].

3.1 Introduction

The permanent of an $n \times n$ matrix A with 0-1 entries a_{ij} is defined by

$$per(A) = \sum_{\sigma} \prod_{i=0}^{n-1} a_{i\sigma(i)}$$

Where the sum is over all permutations σ of set $[N]$.

The permanent can be seen as determinant without *minus* sign. The permanent function arises naturally in number of fields like algebra, combinatorial enumeration,physical sciences, statistical physics etc.

Evaluating permanent of A is equivalent to counting perfect matchings in a bipartite graph $G(V_1, V_2, E)$ with $|V_1| = |V_2| = n$ and $E \subseteq V_1 \times V_2$.

$$a_{ij} = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$$

The best known algorithm for exact computation of permanent,given by Karmarkar, Karp *et al* has the complexity of $O(n2^n)$.

3.2 The Markov Chain

Let $G(V_1, V_2, E)$ denote bipartite graph with $|V_1| = |V_2| = n$ and for $k \in N$ let $M_k(G)$ denote the set of matchings of size k in G.We are interested in counting $M_n(G)$.Note that construction problem for perfect matching can be solved in polynomial time.

We define $\mathcal{MC}_{pm}(G)$ with state space $\mathcal{N} = M_n(G) \cup M_{n-1}(G)$.We note that $\mathcal{N}$ includes some extra states $M_{n-1}(G)$ which are called *near perfect* matchings.These are required to permit free movements between perfect matchings.

The transitions in chain are specified as follows:
In any state $M \in \mathcal{N}$, choose an edge $e = (u,v) \in E$ uniformly at random and then

1. if $M \in M_n(G)$ and $e \in M$,move to state $M' = M - e$ *(type 1 transition)*.

2. if $M \in M_{n-1}(G)$ and u, v unmatched in M then move to $M' = M + e$ *(type 2 transition)*.

3. $M \in M_n(G), u$ is matched to w in M and v is unmatched then move to $M' = (M + e) - (u,v)$ *(type 0 transition)*

4. In all other cases do nothing.

For the sake of convenience we also introduce an additional self loop probability of $1/2$ to each state. The conditions (1) and (2) for existence of fully polynomial almost uniform generator *(fpaug)* hold because transitions can be carried out in $O(|E|)$. For condition 3 to hold we need to ensure that number of near perfect matching is not too large as compared to number of perfect matching. Let us call G dense if its minimum vertex degree is atleast $n/2$,it can be shown that $|M_n(G)|/|\mathcal{N}| \geq 1/n^2$. The condition 4 is satisfied as a consequence of following theorem.

14

3.3 Uniform Generation of Perfect Matching

Theorem 3.1 *For dense bipartite graphs G, the conductance of underlying graph of the Markov chain $\mathcal{MC}_{pm}(G)$ is at least $1/12n^6$.*

Proof :
Here $\beta/d = 1/2|E| \geq 1/2n^2$ so by it is sufficient to show that graph H defining random walk performed by $\mathcal{MC}_{pm}(G)$ has magnification

$$\mu(H) \geq 1/6n^4$$

We shall again use canonical path technique.If no transition occurs in more than $b|\mathcal{N}|$ of these then we gets bound on magnification in terms of *bottleneck parameter b*.

Specifying the nearest perfect matching for near perfect matchings: For each $M \in \mathcal{N}$ we specify canonical path from M to unique closest perfect matching $\bar{M}$ as follows:

1. if $M \in M_n(G)$ then $\bar{M} = M$ and the path is empty;

2. if $M \in M_{n-1}(G)$ and $(u,v) \in E$, then $\bar{M} = M + e$ and path consists of single type 2 transition.

3. if $M \in M_{n-1}(G)$ and $(u,v) \notin E$, then fix some (u',v') such that $(u,v'),(u',v) \in E$.Then $\bar{M} = (M - (u',v') + (u,v') + (u',v)$ and the path consists of type 0 transition followed by type 2 transition.

The canonical path from $\bar{M}$ to M consists of same edges in reverse order. We note that no perfect matching is *close to* too many near perfect matchings. We define for $M \in M_n(G)$

$$\mathcal{K}(M) = \{M' \in \mathcal{N} \ : \ \bar{M}' = M\}$$

Now each matching in $\mathcal{K}(M)$ has at least $n-2$ edges common with M. We can easily see that $|\mathcal{K}(M)| \leq n^2$. Note the sets $\mathcal{K}(M)$ partition $\mathcal{N}$. So $|\mathcal{N}| \leq n^2|M_n(G)|$ and condition 3 is satisfied.

Defining canonical path between two perfect matchings: Here we first assume a fixed ordering of all even cycles of graph G and a fixed start vertex in cycle. Now consider symmetric difference $I \oplus F$ where I is initial state and F is final state. $I \oplus F$ contains those edges which are present in

exactly one of I and F. It just consists of even length cycles which can be order $C_1, .., C_r$ where indices respect the initial ordering. Now from going from I to F we just have to unwind the cycles in symmetric difference one by one in increasing order of vertices. The unwinding of each cycle starts from the specified start vertex u_0 of that cycle. If the cycle consists vertices $(u_0, v_0, u_1, v_1, .., u_l, v_l)$ where (u_j, v_j) is edge in I then unwinding of cycle consists of removing (u_0, v_0) (type 1),replacing edge (u_j, v_j) by (u_j, v_{j-1}) for $1 < j \leq l$ and adding the edge (u_0, v_l) The canonical path between any pair of matching I, F consists of three segments

1. *initial segment:* canonical path from I to $\bar{I}$

2. *main segment:* canonical path from $\bar{I}$ to $\bar{F}$

3. *final segment:* canonical path from $\bar{F}$ to F

Determination of bottleneck parameter b: Now consider arbitrary oriented edge of H corresponding to transition t in Markov chain. t can occur in some path in three ways corresponding to each segment. i.e. number of path through t is sum of number of paths having t in initial segment , no of paths containing t in main segment and no of paths having t in final segment.

From the definition of initial segment the perfect matching $\bar{I}$ is uniquely determined by t. We already know that $\mathcal{K}(\bar{I}) \leq n^2$ and since $I \in \mathcal{K}(\bar{I})$ the number of paths which contain t in initial segment is atmost $n^2 |\mathcal{N}|$.By symmetry number of paths having t in final segment is also bounded by $n^2 |\mathcal{N}|$.

For main segment, we use injective mapping technique. Let t be the transition from states M to M' and $P(t)$ be the set of ordered pair $\langle I, F \rangle$ of perfect matchings such that t is contained in the canonical path from I to F.

Let the symmetric difference $I \oplus F$ consists of cycles $C_1, ..C_r$ and let t be the transition in unwinding of cycle C_i. Then we define the mapping $\sigma_t(I, F)$ as follows:

- $\sigma_t(I, F)$ matches with I for cycles $C_1, .., C_{i-1}$

- $\sigma_t(I, F)$ matches with F for cycles $C_{i+1}, .., C_r$

16

- for cycle C_i, $\sigma_t(I, F)$ contains all edges that are not in $M \cup M'$ except $e_{I,t}$ when t is of type 0. $e_{I,t}$ denotes edge in I which is incident on u_0 the start vertex.

So we can define

$$\sigma_t(I, F) = \begin{cases} (I \oplus F \oplus (M \cup M')) - e_{I,t}, & \text{if } t \text{ is type 0} \\ I \oplus F \oplus (M \cup M') & \text{otherwise} \end{cases}$$

Now from this complementary point $\sigma_t(I, F)$ we can obtain I and F. We first observe that $I \oplus F$ can be immediately recovered by

$$I \oplus F = \begin{cases} (\sigma_t(I, F) \oplus (M \cup M')) + e_{I,t}, & \text{if } t \text{ is type 0} \\ \sigma_t(I, F) \oplus (M \cup M') & \text{otherwise} \end{cases}$$

Then $I \cap F = M \backslash (I \oplus F)$. and from this I and F can be recovered.So, the mapping is injective.

Therefore, we get $|P(t)| \leq |\mathcal{N}|$. Since $|\mathcal{K}(M)| \leq n^2$ t is contained in atmost $n^4 |\mathcal{N}|$ paths. So the total number of paths containing t is bounded by

$$(n^4 + n^2 + n^2)|\mathcal{N}| \leq 3n^4 |\mathcal{N}|$$

So we get $b = 3n^4$.

Corollary 3.2 *There exists a fully polynomial almost uniform generator for perfect matchings in dense bipartite graphs.*

3.4   Approximately Counting Perfect Matchings

For this we first state the following lemma from [Bro86]

Lemma 3.3 *Given (ϵ, δ)-approximation schemes for the ratios $x_1/x_2, x_2/x_3, .., x_{n-1}/x_n$, we can construct (ϵ, δ)-approximation scheme for the ratio x_1/x_n.*

Corollary 3.4 *There exists a f.p. randomized approximate counter for perfect matching in dense bipartite graphs, and hence fully polynomial randomized approximation scheme(fpras) for the permanent of dense square 0-1 matrices.*

Proof :

Consider a dense bipartite graph $G(V_1, V_2, E)$ with $|V_1| = |V_2| = n$. Let us consider family of graphs G_l for $l = 0, .., n-1$ constructed by appending l vertices to each part of bipartition and connecting each new vertex to all vertices of G in opposite part of bipartition. So,$G_l(V_1', V_2', E)$ is a graph with

$$
\begin{aligned}
V_1' &= V_1 \cup \{x_0, .., x_{l-1}\} \\
V_2' &= V_2 \cup \{y_0, .., y_{l-1}\} \\
E' &= E \cup \{(u, y_i) : u \in V_1\} \cup \{(u, x_i) : u \in V_2\}
\end{aligned}
$$

Clearly graph G_l is dense (minimum vertex degree is atleast $n/2 + l$). Now for set $\mathcal{N} = M_{n+l}(G_l) \cup M_{n+l-1}(G_l)$ we can set up bijections and show that

$$|M_{n+l}(G_l)| = (l!)^2 |M_{n-l}(G)|$$

$$|M_{n+l-1}(G_l)| = (l!)^2 (2l|M_{n-l}(G)| + |M_{n-l+1}(G)| + (l+1)^2 |M_{n-l-1}(G)|)$$

Now let $p_1 = |M_{n-l}(G)|/|\mathcal{N}|$ and $p_2 = |M_{n-l-1}(G)|/|\mathcal{N}|$. By simulating the Markov chain $\mathcal{MC}_{pm}(G_l)$ we generate almost uniformly some number of elements of $\mathcal{N}$.

Now it can be shown that if s_1 and s_2 are proportions of the sample which correspond to matchings in G of size $n-l$ and $n-l-1$ respectively, then the quantity $(l+1)^2 s_1/(2l+1)s_2$ may be taken as approximate value of the ratio p_1/p_2 i.e. $|M_{n-l}(G)|/|M_{n-l-1}(G)|$

Now we have (ϵ, δ)-approximation scheme for the ratio $|M_{n-l}(G)|/|M_{n-l-1}(G)|$ for all l, so by lemma 3.3 we get (ϵ, δ)-approximation scheme for $|M_n(G)|$.

Note that to ensure that necessary accuracy is achieved in polynomial time we have

$$\frac{1}{n^2} \leq \frac{|M_k(G)|}{|M_{k+1}(G)|}, \leq n^2$$

So from above bijections the proportion of matching in $\mathcal{N}$ corresponding to $(n-l)$ matchings in G is atleast

$$\frac{(l!)^2 (2l+1)|M_{n-l}(G)|}{|\mathcal{N}|} \geq \frac{2l+1}{n^2((l+1)^2 + 1) + 2l + 1} \geq \frac{1}{3n^3}$$

a similar bound holds for proportion corresponding to $(n-l-1)$ matchings in G.

$\square$

18

A chain with better conductance is obtained if we change the transitions of $\mathcal{MC}_{pm}(G)$.Instead of picking edge at random, we choose a vertex at random. This gives random walk with transition probabilities $1/2n$ rather than $1/2|E|$ and same bound on the magnification.

3.5 Role of Density Assumption

The density assumption plays an important role in proving that Markov chain is rapid mixing. It is a very important criteria because for nondense graphs it is still an open problem. No better approximation algorithm is known for that.

The roles played by density assumption are

1. we used density assumption to prove polynomial upper bound on ratio $|M_k(G)|/|M_{k+1}|$

2. In the proof of theorem 3.1 we use strong property of dense graph, namely $M_{n-1}(G)$ can be partitioned into classes of polynomial size, corresponding to each element of $M_n(G)$.

Instead of taking strong density assumption that minimum vertex degree is atleast $n/2$, we may have weaker density condition that

$$|M_{n-1}(G)| \leq q(n)|M_n(G)|$$

for some fixed polynomial q.

3.6 Perfect Matchings in arbitrary dense graph

We may generalize the problem problem of counting perfect matching in dense bipartite graph as problem of counting perfect matchings in arbitrary dense graph $G(V, E)$ with $|V| = 2n$.The bipartite graph is the special case of this,which is of interest because of its connection with permanent.We may use the same technique without making any essential modifications.

We call G to be dense if minimum vertex degree is atleast n.We may define $\mathcal{K}(M)$ for $M \in M_n(G)$ as before and $|\mathcal{K}(M)| \leq 2n^2$. This gives $b = 8n^4$ and $\mu(H) \geq 1/64n^6$.Hence we get fully polynomial almost uniform generator and fully polynomial randomized approximation scheme.

3.7 Perfect Matchings under Weaker Density Assumption

Here we state without proof certain important theorems to that there exists a fully polynomial randomized approximation scheme for counting no of perfect matchings in arbitrary dense graphs where density assumption means

$$\frac{|M_{n-1}(G)|}{|M_n(G)|} \leq q(n)$$

for some fixed polynomial q.

Lemma 3.5 *For any graph G and positive integer k,*

$$|M_{k+1}(G)||M_{k-1}(G)| \leq |M_k(G)|^2$$

This property is known as *log-concavity.*Because of this property we get similar polynomial bounds on ratios $|M_{k-1}(G)|/|M_k(G)|$.

Next we state the following theorem to show that this weak density assumption is sufficient to ensure that the Markov chins $\mathcal{MC}_{pm}(G)$ is rapid mixing.

Theorem 3.6 *For any graph $G = (V, E)$ with $|V| = 2n$ and $|M_n(G)| > 0$ the conductance of underlying graph of $|\mathcal{MC}_{pm}(G)|$ is bounded below by*

$$\frac{1}{16|E|} \left(\frac{|M_n(G)|}{|M_{n-1}(G)|} \right)^2.$$

For this we use similar technique as before and *log-concavity* to show that the magnification is bounded below by

$$\mu(H) \geq \frac{1}{8} \left(\frac{|M_n(G)|}{|M_{n-1}(G)|} \right)^2.$$

Again here we can improve the transition probability ,which is $1/2|E|$ by using intelligent implementation.

3.8   Conclusions

We saw in this chapter how Markov chain approach provides randomized polynomial time algorithm to generate uniformly the perfect matching in bipartite graph and count the number of perfect matchings in the same. Hence we get a polynomial time randomized algorithm for approximating the value of *permanent* of dense 0-1 matrices.

Chapter 4

Computing the Volume

The task of computing volume of convex body is very common but it becomes extremely difficult in higher dimensions. On the negative side we have that the computation of volume is #P-hard i.e. it takes exponential number of steps in number of dimensions. On the positive side we have randomized polynomial time algorithm to approximate the volume of a convex body K in R^n. Again the key point in this algorithm is to generate a random point in the convex body. This is achieved by making a random walk on the lattice points inside the body. So we again use the Markov chain technique, but here the conductance of chain is defined in the way slightly different from that in previous chapter. For showing that conductance is not very small we make use of isoperimetric inequality for the subsets of convex body.

The computation of volume of various convex body is related to numerical integration in higher dimensions. It has profound applications in various fields like geometry, statistics, theoretical physics etc.

Recently Dyer, Frieze and Kannan designed a randomized polynomial time algorithm to the volume approximately. The random point is generated in the body. The original algorithm needed $O(n^{27})$ convex programs to solve which is totally impractical. But it has been improved ultimately to $O(n^7)$ membership tests. Here we give a simple way to do this. The technicalities are suppressed.

4.1 How to define a convex body?

The convex body has no natural way of encoding. Mathematically, convex body is compact and full dimensional convex set in R^n. We can describe

a convex body (polytope) solution set of system of linear inequalities. The convex body (polytope) can be defined as convex hull of its vertices. But convex body in general is described by membership oracle.

4.1.1 Membership Oracle

Definition 4.1 Weak Membership Oracle: *For any $y \in Q^n$ we may ask the oracle whether y belongs to body K or not; together with this query we also include an error tolerance $\delta > 0$. The answer will be "YES" or "NO". The "YES" answer means that the distance of y from K is atmost δ; "NO" answer means that the distance of y from $R^n \backslash K$ is less than δ.*

We may have strong membership oracle which can test membership without any error.

All this definitions are equivalent upto polynomial time reductions.

4.2 Relation with Poset

We show that number of linear extension is related to volume of polytope. counting the number of linear extensions of partially ordered set $P = (E, \leq)$ is #P complete (this was proved by Brightwell and Winkler).

Here for any partially ordered set P on n vertices, we associate n-dimensional polytope defined by

$$K(P) = \{x \in [0,1]^n | x_i \leq x_j \ whenever \ i \leq j \ in \ P\}$$

It can be shown that
The number of linear extensions of P equals $n!$ times volume of $K(P)$.

$K(P)$ is simply the intersection of at most n^2 half-spaces specified by the inequalities of the form $x_i - x_j \leq 0$.

4.3 Löwner-John ellipsoids

For each convex body K, there exists unique ellipsoid E with minimum volume containing it, called Löwner-John ellipsoid. The important property of the ellipsoid is:

Theorem 4.1 *If we shrink the Löwner-John ellipsoid of a convex body K from its center by a factor of n, we obtain an ellipsoid that is contained in K.*

The Löwner-John ellipsoid itself may be difficult to compute. However we may can compute the ellipsoid with somewhat weaker property in polynomial time. We call an ellipsoid E a weak Löwner-John ellipsoid for K, if E contains K and if we shrink it by factor of $n^{3/2}$ we get an ellipsoid contained in K.

Theorem 4.2 *For convex body, a weak Löwner-John ellipsoid can be computed using polynomial number of operations and using polynomial number of digits.*

The theorem above assumes that convex body is given by weak membership oracle and also an inscribed and circumscribed balls for body are specified.

If R and r are the radii of circumscribing ball and inscribed ball respectively, then it can be computed using $O(n^4 \log (R/r))$ operations and $O(n^2(|\log R| + |\log r|))$ number of digits.

If we have K as the solution set of the system of linear inequalities or convex hull of vertices, then factor of $n^{3/2}$ can be improved to $2n$.

Computing a weak Löwner-John ellipsoid of the body yields a rough approximation on its volume.

4.4 The difficulty of computing the volume

Theorem 4.3 *Consider any polynomial time algorithm which assigns to every convex body K, given by a membership oracle, an upper bound $w(K)$ on $vol(K)$. Then there exists a constant $c > 0$ such that in every dimension n there exists a body K for which $w(K) > n^{cn}vol(K)$*

This theorem depends on the following lemma:

Lemma 4.4 *There exists a constant $c > 0$ such that the volume of the convex hull of any $p > 0$ points in the unit ball B is less than $pn^{-cn}vol(B)$.*

Let us give an argument giving a somewhat weaker bound of $p2^{-n}$. If $v_1, ..v_p$ are the given points then for each ball construct a ball B_i which has center $v_i/2$ and radius $|v_i|/2$. Then these balls cover the convex hull of these points. So the volume of the convex hull is atmost $\sum_{i=0}^{p} vol(B_i) \leq p2^{-n}vol(B)$.

24

Now to prove the theorem, first apply the algorithm to unit ball B. The algorithm runs and asks a polynomial number of points $v_1, .., v_p$ from oracle. It ultimately gives upper bound $w(B)$ of the volume. Now apply algorithm to the convex hull K of $\{e_1, .., e_n, v_1, .., v_p\} \cap B$. The algorithm runs exactly same as before and get $w(K) = w(B)$. So, there is a large relative error.

4.5 The volume computation is NP-hard

Here we shall assume that subset-sum problem is NP-complete.
Subset-sum problem:
Given $n + 1$ positive integers $a_1, ..a_n$ and b find a subset $J \subseteq \{1, .., n\}$ such that $\sum_{i \in J} a_i = b$
Consider the polytope P defined by following system of linear inequalities

$$0 \leq x_i \leq 1,$$

$$\sum_i a_i x_i \leq t.$$

where t is some positive real number.
We see that P is obtained by cutting a part of unit cube by a hyperplane. This hyperplane passes through a vertex of hypercube iff t is an integer and subset sum problem has a solution for $b = t$.
Let $f(t)$ be the volume of P(as a function of t). We see that $f(t)$ is polynomial of degree $n - 1$ in all intervals where the hyperplane doesn't pass through any vertices of cube.
So, if we can compute the volume of polytope in polynomial time then for positive integer b we can interpolate this polynomial in intervals $[b - 1, b]$ and $[b, b + 1]$, and check whether it gives the same polynomial. This will give polynomial time solution to subset-sum problem.

$\square$

4.6 Randomized Volume Computation

Here we sketch the randomized algorithm by Dyer,Frieze and Kannan for estimating the volume of a convex body $K \subseteq R^n$.
We have already made an assumption that K is contained in unit ball B.

Then we can random points inside B to estimate the ratio of volumes of two bodies. But this may **fail**. As we have shown in section on difficulty of computing the volume we may have the volume of K to be exponentially (in n) than B.For example if K is the cube inscribed in B then the ratio of volume is $(2/e\pi)^{n/2}$. But we have from the results of previous chapters that this works only if the ratio is not exponentially small.

So we use the following trick:
Connect K and B by a sequence of convex bodies $K = K_0 \subseteq K_1 \subseteq ... \subseteq K_m = B$ such that $vol(K_i)/vol(K_{i+1}) \geq 1/2$ (m is polynomial in n). Then we can use the Monte-Carlo Method to estimate each ratio and their product gives the estimate of $vol(K)$. Note that such a sequence can always be created.e.g. $K_i = B \cap (1 + 1/2n)^i K$.

But now the key point comes: **how to generate a random point with a uniform distribution in a convex body?**

4.7 The Markov Chain

In this section we shall describe how the Markov chain is constructed for generating a random point in a convex body K. We shall show without going into technical details that the conductance of the chain is not very low.

The graph to which Dyer,Frieze and Kannan apply this technique(of random walk) consists of lattice points i.e. the points in body K whose coordinates are multiples of fixed δ (with, say, $\delta \approx n^{-5/2}$). The two points are adjacent if the distance between them is δ. The transition is possible only between the neighboring points.

But unfortunately conductance may be 0. There may be some points in "corners" which are unaccessible. Also there can be subsets having very small number of edges leaving it. This makes the conductance exponentially small.

The remedy suggested by Dyer,Frieze and Kannan for this is to assume the surface of body to be sufficiently smooth. This can be done by replacing K with $K' = K + (\epsilon/n^2)B$ where "+" is Minkowski sum as defined in [LS90].

As the remedy of this Lovász and Simonovits defined a weaker notion of conductance called μ-conductance Φ_μ.

Definition 4.2 *Let $0 \leq \mu \leq 1/2$.The μ-conductance of the graph G is the largest number Φ_μ such that for every set $S \subseteq V$, with $|S| \leq |V|/2$, the number of edges of G connecting S to $V \backslash S$ is atleast $d\Phi_\mu(|S| - \mu|V|)$.*

26

So, Φ_μ is the least number such that the probability that if a node is drawn from the uniform distribution on V, and a transition is made, the next node belongs to other part of the partition $S, V \backslash S$ is atleast $\Phi_\mu \left(\frac{|S|}{|V|} - \mu \right)$.

So we have following theorem

Theorem 4.5 *Let $G = (V, E)$ be a finite graph, $0 \leq \mu \leq 1/2$, and first element of random walk $(v_0, v_1, ..)$ on G drawn from a distribution so that*

$$\left| Pr(v_0 \in S) - \frac{|S|}{|V|} \right| \leq H$$

for every $S \subseteq V$ with $|S| \leq \mu|V|$ or $|S| \geq (1 - \mu)|V|$. Then for every $t \geq 0$ and every $S \subseteq V$,

$$\left| Pr(v_t \in S) - \frac{|S|}{|V|} \right| \leq H + \left(1 - \frac{1}{2} \Phi_\mu \right)^t \frac{H}{\mu}.$$

So if Φ_μ is larger than n^{const} and we start from the distribution that is not very concentrated then after polynomial number of steps we get a lattice point almost uniformly at random.

4.8   Isoperimetric Inequality

From the above discussion we , for $S \subseteq V$, if K_1 is the set of points in K nearer to S than $V \backslash S$, then we expect that

1. the volume of K_1 is about $\delta^n |S|$,

2. the volume of $K \backslash K_1$ is about $\delta^n |V \backslash S|$,

3. the surface area of the boundary of K_1 inside K is about δ^{n-1} times the number of adjacent pairs of lattice points (u, v) with $u \in S$ and $v \in V \backslash S$.

We suppress the technical details here, the problem now reduces to following isoperimetric inequality:

Theorem 4.6 *Let K be a convex body in R^n with diameter d. Let F be a surface with $(n - 1)$ measure f, cutting K into two parts with volumes v_1 and v_2 Then*

$$min\{v_1, v_2\} \leq fd$$

27

The slightly stronger version of this is:

Theorem 4.7 *Let K be a convex set in R^n with diameter d. Let $K_1 \cup B \cup K_2$ be a partition of K into three closed parts such that distance of K_1 and K_2 is atleast t. Then*

$$vol(B) \geq \frac{t}{d} \min\{vol(K_1), vol(K_2)\}.$$

Proof of above theorem can be obtained in [LS90].

To show that the the chain is rapid mixing we define m-conductance of graph G as the μ-conductance of the random walk on G, where $\mu = m/|V|$. Then from the above isoperimetric inequalities we can prove the following theorem, which we just state without proof.

Theorem 4.8 *Assume that the convex body $K \subseteq R^n$ contains the ball with radius r and contained ball of radius R. Let $G = (V, E)$ be a lattice graph associated with K and $m \geq 4n^{3/2} vol(K)/r$. Then the m-conductance of G is at least $1/(4Rn)$.*

This theorem shows that the Markov chain is rapid mixing and hence we get polynomiality of volume algorithm.

4.9   Conclusions

Unfortunately, even with these improvements the algorithm is practically useless; its running time grows with 16th power of n. The improvement in this algorithm was made by Applegate and Kannan. By introducing faster Markov chains. This had running time bound in 10th power of n. Combining the idea with other improvements, like transformation to an integration problem, the running time can be pushed down to $O(n^7)$.

Bibliography

[Bro86]   A.Z. Broder. How hard it is to marry at random? (on the approximation of the permanent). In *Proceedings of the 18st ACM symposium on Theory of Computing*, pages 50–58, 1986.

[Cha94]   M. Charikar. Uniform generation and approximate counting. In *B.Tech Seminar*, 1994.

[DFK91] M. Dyer, A. Frieze, and R. Kannan. A random polynomial time algorithm for approximating the volume of convex bodies. In *Proceedings of the 21st ACM symposium on Theory of Computing*, pages 375–381, 1991.

[Fel68]   W. Feller. *Introduction to probability theory and its applications*. Wiley, iii edition, 1968.

[Lov91]   L. Lovász. How to compute the volume? In *DIMACS technical report*, pages 1–14, 1991.

[LS90]   L. Lováz and M. Simonovits. The mixing rate of markov chains, an isoperimetric inequality, and computing the volume. In *Proceedings of the 20st ACM symposium on Theory of Computing*, pages 346–354, 1990.

[Sin93]   A. Sinclair. *Algorithms for Random Generation and Countiing*. Birkhauser, 1993.

[Vaz91]   U.V. Vazirani. Rapidly mixing markov chains. In *Proceedings of symposia in applied mathematics vol. 44*, pages 99–121, 1991.