CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 1: August 27

Lecturer: Prof. Alistair Sinclair *Scribes: Alistair Sinclair*

1.1 The Markov Chain Monte Carlo Paradigm

Assume that we have a very large but finite set Ω and a positive weight function $w : \Omega \to \Re^+$. Our goal is to sample $x \in \Omega$ with probability $\pi(x) = \frac{w(x)}{Z}$, where the normalizing factor $Z = \sum_{x\in\Omega} w(x)$, often called the "partition function", is usually unknown. (Indeed, in many applications our ultimate goal will be to estimate Z.)

Markov Chain Monte Carlo constructs a Markov Chain (X_t) on Ω that converges to π, ie $\Pr[X_t = y|X_0 = x] \to \pi(y)$ as $t \to \infty$, independent of x. Then we get a sampling algorithm by simulating the Markov chain, starting in an arbitrary state X_0, for sufficiently many steps and outputting the final state X_t. It is usually not hard to set up a Markov chain that converges to the desired stationary distribution; however, the key question is how many steps is "sufficiently many," or equivalently, how many steps are needed for the chain to get "close to" π. This is known as the "mixing time" of the chain. Obviously the mixing time determines the efficiency of the sampling algorithm.

In the remainder of this introductory lecture, we provide motivation for MCMC by sketching a number of application areas in which random sampling from large, complex combinatorial sets arises. We focus on applications in which rigorously justified algorithms can be achieved; for a more practically oriented focus, see the excellent book by Jun Liu [L02].

1.2 Applications

1.2.1 Combinatorics

Applications in combinatorics include:

- Examining typical members of a combinatorial set, which can be used, e.g., for fomulating and testing conjectures.
  For example, by sampling random 3-regular graphs on n vertices, we might formulate the conjecture that they are (almost) all Hamiltonian; this conjecture is actually now a theorem.
- Generating test data for algorithms.
  Often, testing an algorithm on completely random inputs (such as arbitrary random graphs) is uninformative. MCMC can be used to generate inputs from a more complex class (such as sparse connected graphs), which can form the basis of more convincing tests of the algorithm.
- Probabilistic constructions.
  The existence of certain objects (such as networks with desired connectivity properties) can be proven

by the probabilistic method, but in many cases the probabilistic construction required is quite complex and it is not obvious how to realize it algorithmically. For example, recent constructions of efficient Low Density Parity Check codes require a random bipartite graph with specified degrees for each vertex. It is not known how to generate such graphs directly, but they can be generated quite efficiently by MCMC.
A similar example is provided by certain models of the WWW, which are also based on random graphs with specified vertex degrees (sometimes with additional properties).

- Approximate counting.
A much less obvious and more far-reaching combinatorial application is to count the number of elements of the set Ω, which might be (e.g.) the set of cliques in a graph G, or the set of satisfying assignments of a boolean formula ϕ. Almost all such counting problems are *#P-complete* (which is the analog of NP-completeness for decision problems); however, in many cases MCMC provides an efficient *randomized approximation algorithm*.

  The general technique for reducing (approximate) counting to random sampling can be explained in the following folksy scenario for counting the number of people in a large crowd Ω:

  1. Partition the crowd Ω into two parts, B and $\overline{B} = \Omega - B$, according to some property (such as "having black hair").
  2. Estimate the proportion $|B|/|\Omega|$ of people with black hair by taking a small uniform sample from the crowd and letting p be the proportion of the sample that have black hair.
  3. Recursively estimate the size of B by applying the same technique to this smaller crowd but using some other property (such as "being male"). Let $\widehat{N_B}$ be this recursive estimate.
  4. Output the final estimate $\widehat{N} = \widehat{N_B} \cdot \frac{1}{p}$.

  Notice that the choice of properties at each level of the recursion is essentially arbitrary; in particular, we do not require them to be independent. The only thing we require is that the proportion of people in the remaining crowd who have the property is bounded away from 0 and 1: this ensures that (i) the number of levels of recursion (until we get down to a crowd of size 1) is small; and (ii) the sample size needed to get a good estimate p at each level is small.

  To apply this technique in a more mathematical context, let Ω be the set of all cliques of a given graph G. (This is a very large and complex set, whose size is in general exponential in the size of G.) We can partition Ω into those cliques that contain some vertex v, and those that do not. But the first of these sets is equivalent to the set of all cliques in the graph G_v (the subgraph of G consisting of v with all its neigbors); and the second set is equivalent to the set of all cliques in the graph $G - v$ (the subgraph of G with v removed). So both subsets correspond to instances of the same counting problem applied to smaller graphs; hence they can be estimated recursively, as required by the method.

1.2.2 Volume and Integration

Given as input a convex body K in $\Re^n$, the goal is to estimate the volume of K. While in low dimensions exact methods may be used, they become computationally intractable in higher dimensions. (In fact, the problem is #P-complete when the dimension n is treated as part of the input.)

The volume can be estimated by the following reduction to random sampling:

Construct concentric balls $B_0 \subset B_1 \subset B_2 \ldots \subset B_r$, such that $B_0 \subset K \subset B_r$. By a non-trivial geometric result, it can be assumed w.l.o.g. that B_0 is the unit ball, and that B_r has radius $O(n \log n)$, where n is

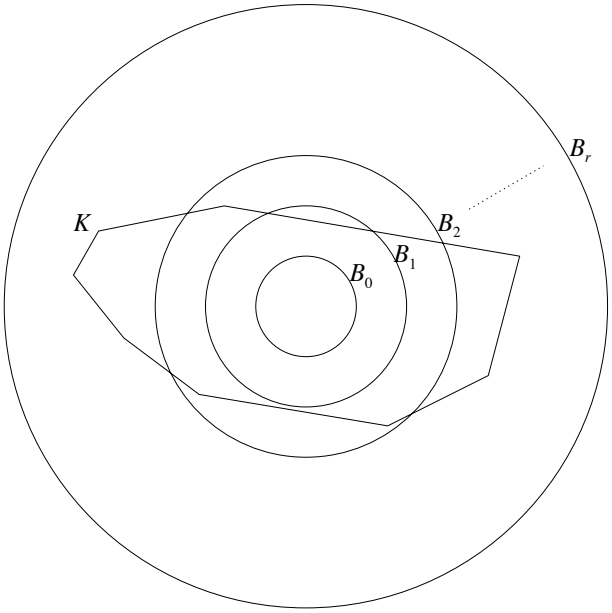


Figure 1.1: Estimating the volume of a convex set by choosing a sequence of increasing balls and computing the ratio $\frac{Vol(K\cap B_i)}{Vol(K\cap B_{i-1})}$

the dimension. (This can be achieved by simple transformations of K.) The construction of the balls is illustrated by Figure 1.1. Then we have

$$Vol(K) = \frac{Vol(K\cap B_r)}{Vol(K\cap B_{r-1})}\times\frac{Vol(K\cap B_{r-1})}{Vol(K\cap B_{r-2})}\times\cdots\times\frac{Vol(K\cap B_1)}{Vol(K\cap B_0)}\times Vol(K\cap B_0).$$

Since $Vol(K\cap B_0) = Vol(B_0)$ is trivial, we can estimate $Vol(K)$ by estimating each of the ratios in the above equation. The ratio $\frac{Vol(K\cap B_i)}{Vol(K\cap B_{i-1})}$ can be estimated by sampling uniformly at random from $Vol(K\cap B_i)$ (which is the intersection of two convex bodies and hence also convex) and counting the proportion of samples falling in B_{i-1}. In this scheme, to ensure that the number of samples needed is small we need to ensure that the ratio $\frac{Vol(K\cap B_i)}{Vol(K\cap B_{i-1})}$ is bounded by a constant. But for this it is enough to make the radius of the balls grow slowly enough, namely $\text{rad}(B_i) = (1+\frac{1}{n})\text{rad}(B_{i-1})$. Notice that this implies that the number of balls is only $r = O(n\log n)$. The random sampling within each $K\cap B_i$ can be done by MCMC.

Observe that the sequence of balls above is necessary because, for a general convex body K in $\Re^n$ that contains the unit ball, the volume of the smallest ball B containing K may be exponentially larger than $Vol(K)$. Hence the "naive Monte Carlo" approach of sampling randomly from B and observing how many samples fall in K is hopeless: we will need exponentially many samples before we even see one that falls in K. The sequence of balls with slowly growing radii ensures that our random sampling always produces an estimator with bounded variance.

The above algorithm was first discovered by Dyer, Frieze and Kannan [DFK89]. The original version had a time complexity of about $O(n^{23})$ (which was a breakthrough because it is polynomial in n). A long sequence of deep and clever improvements has since brought this down to $\tilde{O}(n^4)$ [LV03] (where the $\tilde{O}$ hides logarithmic factors).

It is possible to generalize this approach to the integration of log-concave functions in $\Re^n$.

1.2.3 Combinatorial Optimization

In this setting Ω is the set of feasible solutions to a combinatorial optimization problem (e.g., the set of all cliques in a graph), and there is an *objective function* $f : \Omega \to \Re$ (e.g., the size of the clique). Our goal is to maximize f, i.e., to find an element $x \in \Omega$, such that $f(x) \geq f(y) \quad \forall y \in \Omega$.

Strategy

- Let G be any positive monotone increasing function.
- Sample from the distribution $\pi(x) = \frac{G(f(x))}{Z}$ using MCMC.
  - A popular choice is $G(y) = \lambda^y$, where $\lambda \geq 1$, so that the distribution is $\pi(x) \propto G(f(x))$.
  - We would like to sample using a large λ to get a good approximation to the maximum; however, on the other hand we want to keep λ fairly small (close to 1) because then the distribution π is close to uniform and thus presumably easier to sample from. (When λ gets large, the MCMC algorithm on Ω will tend to get trapped in local maxima.) Thus there is a tension between large and small values of λ.

A simple strategy is to simply choose some intermediate value of λ and hope that it achieves a good compromise between the above two concerns; this is often called the "Metropolis algorithm." A more sophisticated strategy is to start with $\lambda = 1$ and gradually increase λ according to some scheme (often called a "cooling schedule") until λ becomes very large and the algorithm "freezes" into a local maximum (which we hope is good). This latter heuristic is known as "simulated annealing."

1.2.4 Statistical Physics

In statistical physics Ω is the set of configurations of a physical system. Each configuration x has an energy $H(x)$. The probability of finding the system in configuration x is given by the "Gibbs distribution"

$$\pi(x) = \frac{\exp(-\beta H(x))}{Z},$$

where $\beta = \frac{1}{\text{Temp}} > 0$ is the inverse temperature. It can be observed that this Gibbs distribution favors configurations with low energy. Moreover, this effect increases with β: as $\beta \to 0$ (i.e., at high temperature) the Gibbs distribution is close to uniform, while at high β (low temperature) the system is almost certain to be in very low energy states. (Note the similarity with the combinatorial optimization application above, under the correspondence $\lambda = \exp(\beta)$ and $f = -H$. The energy minima can be thought of as local optima.)

The most famous and widely studied model in statistical physics is the *Ising model* of ferromagnetism. Here there is an underlying graph $G = (V, E)$, usually a finite portion of the d-dimensional square lattice, at each vertex of which there is a *spin*, which can be either $+$ or $-$. (The spins correspond to atomic magnets whose magnetization is either "up" or "down".) Thus the set of configurations is $\Omega = \{+, -\}^V$. An example of a configuration is depicted in Figure 1.2.

The energy of an Ising configuration σ is given by

$$H(\sigma) = -\sum_{i \sim j} \sigma_i \sigma_j. \tag{1.1}$$

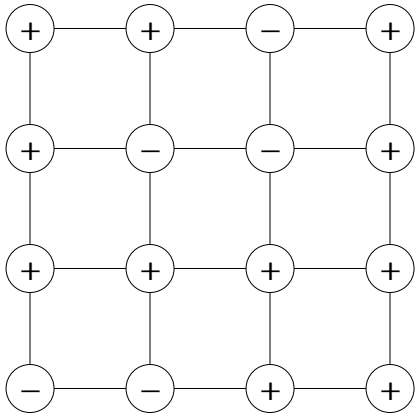


Figure 1.2: Example configuration of a 2-dimensional Ising model.

Thus low energy configurations are those in which many pairs of neighboring spins are aligned with one another. A famous classical fact about the Ising model is that there is a well-defined "critical (inverse) temperature" β_c at which the macroscopic behavior of the system changes dramatically: for $\beta < \beta_c$, the system will be in a "disordered" configuration (consisting of an essentially random sea of $+$ and $-$); and for $\beta > \beta_c$, the system will exhibit long-range order, so that there is likely to be a large region of $+$ (or of $-$) of size comparable to that of the entire graph. This phenomenon is referred to as a "phase transition", and corresponds to the onset of spontaneous magnetization in the ferromagnetic material.

Applications of random sampling from the Gibbs distribution include the following:

1. Examine typical configurations at a given value of β.
2. Compute expectations w.r.t. π (e.g., for the Ising model, the mean magnetization, specific heat, susceptibility).
3. Estimate Z (the partition function). Typically Z carries information about all the thermodynamic properties of the system.

The Glauber Dynamics

A standard approach to MCMC sampling from the Gibbs distribution is known as the Glauber dynamics. We illustrate this for the Ising model, though it is much more general. This is a Markov chain which at each step selects a vertex $v \in V$ at random and sets its spin to $+$ or $-$ with the appropriate probability *conditional on the spins of its neighbors*. Thus the probability of setting the spin to $+$ is $\frac{e^{\beta(n-p)}}{e^{\beta(n-p)}+e^{\beta(p-n)}} = \frac{1}{1+e^{\beta(2n-2p)}}$, where p is the number of neighbors with spin $+$ and n is the number of neighbors with spin $-$. It is not hard to see that this Markov chain converges to the Gibbs distribution on Ω.

The Glauber dynamics is of interest for two reasons:

- It provides an MCMC algorithm for sampling from the Gibbs distribution.
- It is a plausible model for the actual evolution of the physical system (so is interesting as a random process in its own right).

The following remarkable fact about the Glauber dynamics has been proved relatively recently for such a classical model [MO94]:

Theorem 1.1 *The mixing time of the Glauber dynamics for the Ising model on a $\sqrt{n} \times \sqrt{n}$ box in the 2-dimensional square lattice is:*

$$\begin{cases} O(n \log n) & \text{if } \beta < \beta_c; \\ e^{\Omega(\sqrt{n})} & \text{if } \beta > \beta_c, \end{cases}$$

where β_c is the critical point (i.e., phase transition).

This theorem is of algorithmic interest because it says that the physical phase transition has a *computational* manifestation, in the form of a sudden switch in the mixing time from linear to exponential. This gives us an additional motivation for studying MCMC in the context of statistical physics, because of its apparent connection with phase transitions.

1.2.5 Statistical Inference

Consider a statistical model with parameters Θ and a set of observed data X. The aim is to obtain Θ based on the observed data X; one way to formulate this problem is that we should *sample* Θ from the distribution $\Pr(\Theta \mid X)$. Using Bayes' rule, $\Pr(\Theta \mid X)$ translates to

$$\Pr(\Theta \mid X) = \frac{\Pr(X \mid \Theta) \Pr(\Theta)}{\Pr(X)},$$

where $\Pr(\Theta)$ is the *prior* distribution and refers to the information previously known about Θ, $\Pr(X \mid \Theta)$ is the probability that X is obtained with the assumed model, and $\Pr(X)$ is the unconditioned probability that X is observed. $\Pr(\Theta \mid X)$ is commonly called the *posterior* distribution and can be written in the form $\pi(\Theta) = w(\Theta)/Z$, where the weight $w(\Theta) = \Pr(X \mid \Theta) \Pr(\Theta)$ is easy to compute but the normalizing factor $Z = \Pr(X)$ is unknown. MCMC can then be used to sample from $\Pr(\Theta \mid X)$. We can further use the sampling in the following applications:

- Prediction: obtain the probability $\Pr(Y \mid X)$ that some future data Y is observed given X. $\Pr(Y \mid X)$ clearly can be written as $\sum_\Theta \Pr(Y \mid \Theta) \Pr(\Theta \mid X) = \mathrm{E}_\pi \Pr(Y \mid \Theta)$. Therefore we can use sampling to predict $\Pr(Y \mid X)$.
- Model comparison: perform sampling to estimate the normalizing factor $Z = \Pr(X)$ (using methods like those above for approximate counting), and use this to compare different models. Note that $\Pr(X)$ is the probability that the given model generated the observed data, so a model with a large value of $\Pr(X)$ should be preferred to a model with a small value.

References

[DFK89] M. Dyer, A. Frieze and R. Kannan, "A random polynomial time algorithm for estimating volumes of convex bodies," *Proceedings of the 21st ACM STOC*, 1989, pp. 375–381.

[L02] J. Liu, *Monte Carlo Strategies in Scientific Computing*, Springer, 2002.

[LV03] L. Lovász and S. Vempala. "Simulated annealing in convex bodies and an $\tilde{O}(n^4)$ volume algorithm." *Proceedings of the 44th IEEE FOCS*, 2003, pp. 650–659.

[MO94] F. Martinelli and E. Olivieri. "Approach to equilibrium of Glauber dynamics in the one phase region I: The attractive case." *Communications in Mathematical Physics* **161** (1994), pp. 447–486.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 2: September 8

Lecturer: Prof. Alistair Sinclair *Scribes: Anand Bhaskar and Anindya De*

2.1 Markov Chains

We begin by reviewing the basic goal in the Markov Chain Monte Carlo paradigm. Assume a finite state space Ω and a weight function $w : \Omega \to \Re^+$. Our goal is to design a sampling process which samples every element $x \in \Omega$ with the probability $\frac{w(x)}{Z}$ where $Z = \sum_{x \in \Omega} w(x)$ is the normalization factor. Often times, we don't know the normalization factor Z apriori, and in some problems, the real goal is to estimate Z.

With the aforementioned motivation, we now define a Markov chain.

Definition 2.1 *A Markov chain on Ω is a stochastic process $\{X_0, X_1, \ldots, X_t, \ldots\}$ with each $X_i \in \Omega$ such that*

$$\Pr(X_{t+1} = y \mid X_t = x, X_{t-1} = x_{t-1}, \ldots, X_0 = x_0) = \Pr(X_{t+1} = y \mid X_t = x) =: P(x, y)$$

Clearly, the Markov chain in the above matrix can be described by a $|\Omega| \times |\Omega|$ matrix P whose $(x, y)^{th}$ entry is $P(x, y)$. Hence, we sometimes blur the distinction between the Markov chain and its transtion matrix P. We now observe the following two important properties of the matrix P:

- P is non-negative, i.e., $\forall\ x, y \in \Omega,\ \ P(x, y) \geq 0$;
- $\forall x \in \Omega,\ \ \sum_{y \in \Omega} P(x, y) = 1$.

A matrix with the above two properties is called a *stochastic* matrix. We now describe some more notation which will be helpful in the course of our discussion. Let $p_x^{(t)} \in \Re^{1 \times |\Omega|}$ be the row vector corresponding to the distribution of X_t when the Markov chain starts at x, i.e. $X_0 = x$. With this notation, the evolution of the Markov chain can be defined in terms of matrix-vector equations. In particular:

- $p_x^{(t+1)} = p_x^{(t)} P$;
- $p_x^{(t)} = p_x^{(0)} P^t$.

[Note that while we assumed that the starting distribution $p_x^{(0)}$ is a point distribution—i.e., the starting vertex is x with probability 1—the same equations hold even if we start with a general distribution. However, it is usually sufficient to consider point distributions since any distribution is a convex combination of them. Moreover, for quantities such as the mixing time it is clear that the worst case is to start with a point distribution.]

2.1.1 Graphical representation

A Markov chain can also be represented graphically. Consider a Markov chain with state space Ω and transition matrix P. Then a corresponding graphical representation is the weighted graph $G = (V, E)$, where $V = \Omega$ and $E = \{(x, y) \in \Omega \times \Omega \mid P(x, y) > 0\}$. Also, edge (x, y) has weight $P(x, y) > 0$. Note that self-loops are allowed since we can have $P(x, x) > 0$.

Note that an edge is present between x and y if and only if the transition probability between x and y is non-zero. Much of the theory of Markov chains is not critically dependent upon the exact entries $P(x, y)$ but rather on whether a particular entry is zero or not. In terms of the graph, the critical thing is thus the structure of the graph G rather than the values of its edge weights.

Many natural Markov chains have the property that $P(x, y) > 0$ if and only if $P(y, x) > 0$. In this case the graph G is essentially undirected (except for the values of the edge weights). A very important special case is when the Markov chain is *reversible*:

Definition 2.2 *Let $\pi > 0$ be a probability distribution over Ω. A Markov chain P is said to be* reversible *with respect to π if $\forall\ x, y \in \Omega$, $\pi(x)P(x, y) = \pi(y)P(y, x)$.*

Note that any symmetric matrix P is trivially reversible (w.r.t. the uniform distribution π).

A reversible Markov chain can be completely represented by an undirected graph with weight $Q(x, y) := \pi(x)P(x, y) = \pi(y)P(y, x)$ on edge $\{x, y\}$ (without specifying P or π explicitly; and any fixed multiple of Q will do as well). To see this, note that the transition probability $P(x, y)$ can be computed from $P(x, y) = \frac{Q(x,y)}{\sum_z Q(x,z)}$. In fact, as we shall see below, for a reversible Markov chain π must in fact be its stationary distribution, and this can be computed from the $Q(x, y)$ also (using the fact that $\frac{\pi(x)}{\pi(y)} = \frac{P(y,x)}{P(x,y)}$). This is one reason why reversible chains are particularly nice to deal with. For any Markov chain, the quantity $\pi(x)P(x, y)$ is called the *ergodic flow* from x to y, i.e., the amount of probability mass flowing from x to y in stationarity. Reversibility says that the ergodic flows from x to y and from y to x are equal; for this reason, the condition in Definition 2.2 is known as the "detailed balance" condition. Of course, by conservation of mass we always have $\pi(S)P(S, \overline{S}) = \pi(\overline{S})P(\overline{S}, S)$ for any subset of states $S \subseteq \Omega$ (where $\overline{S} = \Omega \setminus S$). Detailed balance says that this also holds *locally*, for every pair of states.

2.1.2 Mixing of Markov chains

We now discuss some definitions and theorems which are important in the context of mixing of Markov chains.

Definition 2.3 *A probability distribution π over Ω is a* stationary distribution *for P if $\pi = \pi P$.*

Definition 2.4 *A Markov chain P is* irreducible *if for all x, y, there exists some t such that $P^t(x, y) > 0$. Equivalently, the graph corresponding to P (denoted by $G(P)$) is strongly connected. In case the graphical representation is an undirected graph, then it is equivalent to $G(P)$ being connected.*

Definition 2.5 *A Markov chain P is* aperiodic *if for all x, y we have* $\gcd\{t : P^t(x, y) > 0\} = 1$.

The following simple exercises are recommended to understand the content of these definitions:

Exercise 2.6 *For a Markov chain P, if $G(P)$ is undirected then aperiodicity is equivalent to $G(P)$ being non-bipartite.*

Exercise 2.7 *Define the period of x as $\gcd\{t : P^t(x,x) > 0\}$. Prove that for an irreducible Markov chain, the period of every $x \in \Omega$ is the same. [Hence, if $G(P)$ is undirected, the period is either 1 or 2.]*

Exercise 2.8 *Suppose P is irreducible. Then P is aperiodic if and only if there exists t such that $P^t(x,y) > 0$ for all $x, y \in \Omega$.*

Exercise 2.9 *Suppose P is irreducible and contains at least one self-loop (i.e., $P(x,x) > 0$ for some x). Then P is aperiodic.*

We now state a theorem which gives a necessary and sufficient condition for convergence of a Markov chain to its stationary distribution regardless of the initial state.

Theorem 2.10 (Fundamental Theorem of Markov Chains) *If a Markov chain P is irreducible and aperiodic then it has a unique stationary distribution π. This is the unique (normalized such that the entries sum to 1) left eigenvector of P with eigenvalue 1. Moreover, $P^t(x,y) \to \pi(y)$ as $t \to \infty$ for all $x, y \in \Omega$.*

In light of this theorem, we shall sometimes refer to an irreducible, aperiodic Markov chain as *ergodic*.

We shall give an elementary probabilistic proof of the above theorem in the next lecture. However, today we sketch an algebraic proof in the special case where the Markov chain is reversible.

In preparation for this, let us verify that, if P is reversible w.r.t. π, then π is a stationary distribution.

Claim 2.11 *If a Markov chain P is reversible w.r.t. π, then π is a stationary distribution for P.*

Proof:

$$(\pi P)(y) = \sum_x \pi(x)P(x,y) = \sum_x \pi(y)P(y,x) = \pi(y)$$

■

Exercise 2.12 *Let P be reversible w.r.t. π. Show that P is similar to a symmetric stochastic matrix, under transformation by the diagonal matrix $\mathrm{diag}(\sqrt{\pi(x)})$.*

Our proof of Theorem 2.10 for reversible chains will make use of the following classical theorem:

Theorem 2.13 (Perron-Frobenius) *Any irreducible, aperiodic stochastic matrix P has an eigenvalue $\lambda_0 = 1$ with unique associated left eigenvector $e_0 > 0$. Moreover, all other eigenvalues λ_i of P satisfy $|\lambda_i| < 1$.*

Proof:(of Theorem 2.10, sketch for reversible case) The fact that P is reversible w.r.t. π means that it is similar to a symmetric matrix (see exercise above). Hence the eigenvalues of P are real, and we can choose a basis of $\Re^{|\Omega|}$ among its eigenvectors. Let the eigenvalues be $\lambda_0 \geq \lambda_1 \geq \ldots \geq \lambda_{|\Omega|-1}$, and the corresponding eigenvectors be $e_0, e_1, \ldots, e_{|\Omega|-1}$. By Theorem 2.13, $\lambda_0 = 1$, and $|\lambda_i| < 1$ for all $i > 0$; also, $e_0 = \pi$ is the unique stationary distribution. To see the convergence property, note that we can write the initial distribution $p^{(0)}$ as a linear combination of eigenvectors, i.e., $p^{(0)} = \sum_i \alpha_i e_i$. But then $p^{(t)} = p^{(0)}P^t = \sum_i \alpha_i \lambda_i^t e_i$. Since $|\lambda_i| < 1$ for all $i > 0$, this implies that $p^{(t)} \to \alpha_0 e_0$, which is the same as e_0 up to a scalar factor. However, by conservation of mass we must have $\alpha_0 = 1$, so the distribution converges to π. ■

If P is not reversible then the Perron-Frobenius theorem still applies but the proof of Theorem 2.10 is a bit more complicated; see, e.g., [Se06] for details.

If P is irreducible (but not necessarily aperiodic), then π still exists and is unique, but the Markov chain does not necessarily converge to π from every starting state. For example, consider the two-state Markov chain with $P = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$. This has the unique stationary distribution $\pi = (1/2, 1/2)$, but does not converge from either of the two initial states. Notice that in this example $\lambda_0 = 1$ and $\lambda_1 = -1$, so there is another eigenvalue of magnitude 1, contradicting the Perron-Frobenius theorem. However, the Perron-Frobenius theorem does generalize to the periodic setting, with the weaker conclusion that the remaining eigenvalues satisfy $|\lambda_i| \leq 1$.

In this course we will not spend much time worrying about periodicity, because of the following simple observation (proof: **exercise!**).

Observation 2.14 *Let P be an irreducible (but not necessarily aperiodic) stochastic matrix. For any $0 < \alpha < 1$, the matrix $P' = \alpha P + (1-\alpha)I$ is stochastic, irreducible and aperiodic, and has the same stationary distribution as P.*

This operation corresponds to introducing a self-loop at all vertices of $G(P)$ with probability $1 - \alpha$. The value of α is usually set to $1/2$. P' is usually called a "lazy" version of P.

In the design of MCMC algorithms, we usually do not worry about periodicity since, instead of simulating the Markov chain P, the algorithm can simulate the lazy version P'. This just has the effect of slowing down the dynamics by a factor of 2.

2.2 Examples of Markov Chains

2.2.1 Random Walks on Undirected Graphs

Definition 2.15 *Random walk on an undirected graph $G(V, E)$ is given by the transition matrix*

$$P(x,y) = \begin{cases} 1/deg(x) & \text{if } (x,y) \in E; \\ 0 & \text{otherwise.} \end{cases}$$

Proposition 2.16 *For random walk P on an undirected graph, we have:*

- *P is irreducible iff G is connected;*
- *P is aperiodic iff G is non-bipartite;*
- *P is reversible with respect to $\pi(x) = deg(x)/(2|E|)$.*

Proof: The first part is straightforward. The second part is a previous exercise. For the third part, we check directly that $\pi(x)P(x,y) = \frac{1}{2|E|} = \pi(y)P(y,x)$. ■

2.2.2 Ehrenfest Urn

In the Ehrenfest Urn, we have 2 urns and n unlabelled balls, where there are j balls in the first urn and $n-j$ balls in the other. Define the state space $\Omega = \{0, 1, ..., n\}$, denoting the number of balls in the first urn. At each step of the Markov chain, we pick a ball u.a.r. and move it to the other urn.

The non-negative entries of the transition matrix are given by

$$\begin{aligned} P(j\ , j+1) &= (n-j)/n, \\ P(j\ , j-1) &= j/n. \end{aligned}$$

The Markov chain is clearly irreducible, and we can check that $\pi(j) = \binom{n}{j}/2^n$ is a stationary distribution (**exercise**). However, P is not aperiodic since the time to return to any given state is even.

2.2.3 Card Shuffling

In card shuffling, we have a deck of n cards, and we consider the space Ω of all permutations of the cards. Thus $|\Omega| = n!$. The aim is to sample from the distribution given by the uniform weight $w(x) = 1\ \forall x \in \Omega$, i.e., to sample a permutation of the cards u.a.r. Thus, in the Markov chain setting, we want the stationary distribution π be uniform.

We look at three different shuffling schemes:

Random Transpositions

Pick two cards i and j uniformly at random with replacement, and switch cards i and j.

This is a pretty slow way of shuffling. The chain is irreducible (any permutation can be expressed as a product of transpositions), and also aperiodic (since we may choose $i = j$, so the chain has self-loops). Since the random transpositions are invertible, we have $P(x, y) = P(y, x)$ for every pair of permutations x, y, so P is symmetric. This implies immediately that its stationary distribution is uniform (since it is reversible w.r.t. the uniform distribution).

Top-to-random

Take the top card and insert it at one of the n positions in the deck chosen uniformly at random.

This shuffle is again irreducible (**exercise**) and aperiodic (due to self-loops). However, note that it is not symmetric (or even reversible): If we insert the top card into (say) the middle of the deck, we cannot bring the card back to the top in one step.

However, notice that every permutation y can be obtained, in one step, from exactly n different permutations (corresponding to the n possible choices for the identity of the previous top card). Since every non-zero transition probability is $\frac{1}{n}$, this implies that $\sum_x P(x, y) = 1$; thus the matrix P is *doubly stochastic* (i.e., its column sums, as well as its row sums, are 1). It is easy to show that the uniform distribution is stationary for doubly stochastic matrices; in fact (**exercise**), π is uniform *if and only if* P is doubly stochastic.

Riffle Shuffle (Gilbert-Shannon-Reeds [Gi55,Re81])

- *Split the deck into two parts according to the binomial distribution $Bin(n, 1/2)$.*
- *Drop cards in sequence, where the next card comes from the left hand L (resp. right hand R) with probability $\frac{|L|}{|L|+|R|}$ (resp. $\frac{|R|}{|L|+|R|}$).*

Note that the second step of the shuffle is equivalent to choosing an interleaving of the two parts uniformly at random (**exercise**).

The chain is irreducible (**exercise**), aperiodic (due to self-loops), and doubly stochastic, and hence its stationary distribution is uniform.

Note: This shuffle is quite different from the "perfect shuffle" performed by professional magicians, who split the deck exactly in half and then perfectly interleave the two halves. The perfect shuffle has no randomness, so the configuration of the deck after any given number of shuffles is known exactly.

References

[Se06] E. Seneta, *Non-negative matrices and Markov chains*, 2nd ed. (revised printing), Springer-Verlag, New York, 2006.

[Gi55] E. Gilbert, "Theory of shuffling," Technical Memorandum, Bell Laboratories, 1955.

[Re81] J. Reeds, Unpublished manuscript, 1981.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 3: September 10

Lecturer: Prof. Alistair Sinclair *Scribes: Andrew H. Chan, Piyush Srivastava*

In the previous lecture we looked at some algorithmic applications of Markov Chains. In this lecture, we will study the Metropolis framework, which gives a recipe for designing Markov Chains that converge to a given stationary distribution, and look at its application to the problem of sampling from the Gibbs distribution on the Ising model. We will also look at another solution to the Ising model called the *Heat Bath Dynamics*, which is less general and works only for spin systems. We end today's lecture with an introduction to mixing times.

3.1 Examples of Markov Chains (continued)

3.1.1 The Metropolis Process

The Metropolis framework addresses the most general version of our sampling problem as follows: Given a set of configurations Ω and a strictly positive weight function $w : \Omega \mapsto \mathbb{R}^+$, we want to sample from Ω with respect to the distribution $\pi(x) = \frac{w(x)}{Z}$, where Z is a normalization constant given by $Z = \sum_{x\in\Omega} w(x)$. Specifically, we want to create a Markov chain whose stationary distribution is the distribution π. In a typical application, Ω may be a set of combinatorial structures, such as the set of satisfying assignments of a boolean formula.

In order to construct a Metropolis process in this setting, we require two ingredients:

Neighborhood Structure Our first requirement is a "neighborhood structure," which is a connected undirected graph with the elements of Ω as its vertices. Typically, two elements are connected by an edge iff they differ by some local change. (For example, in the case of a spin system, we may consider two configurations to be adjacent iff they differ in the spin at exactly one site. Note that we assume the graph is *undirected* and *connected* because the Markov chain must be both irreducible and reversible. Also, we allow self-loops, which may be needed to ensure aperiodicity.

We use the notation $x \sim y$ to denote that x and y are neighbors.

Proposal Distribution Let $\Gamma(\Omega, \mathcal{E})$ be the underlying graph. For each $x \in \Omega$ we define a "proposal distribution," which is a function $\kappa(x, \cdot)$ with the following properties:

- $\kappa(x, y) = \kappa(y, x) > 0$ for all x, y such that $x \sim y, x \neq y$.
- For all $x \in \Omega$, $\sum_{y\neq x} \kappa(x, y) \leq 1$ and $\kappa(x, x) = 1 - \sum_y \kappa(x, y)$.
- For notational convenience, we define $\kappa(x, y) = 0$ if x and y are not neighbors.

The transitions of the Markov chain are now specified as follows. From any state $x \in \Omega$:

- Pick a neighbor $y \sim x$ with probability $\kappa(x, y)$.
- "Accept" the move to y with probability $\min\left\{1, \frac{w(y)}{w(x)}\right\} = \min\left\{1, \frac{\pi(y)}{\pi(x)}\right\}$, else stay at x.

The reason for the term "proposal distribution" for κ is now clear. We "propose" a new vertex according to the distribution induced by κ on the current vertex, and then move to this vertex with a probability that depends on the stationary distribution to which we want the process to converge. Notice that the actual transition probabilities are $P(x,y) = \kappa(x,y) \cdot \min\left\{1, \frac{\pi(y)}{\pi(x)}\right\}$. Crucially, though, we can implement this knowing only the weights w (i.e., we don't need to know the normalizing factor Z, which cancels out in the ratio $\pi(x)/\pi(y)$).

Note that the Metropolis process is always irreducible, as the neighborhood structure is connected and can be made aperiodic by the usual trick of introducing self-loops.

We now show the reversibility of the Metropolis process with respect to π. From the Fundamental Theorem, this implies that π is its unique stationary distribution.

Claim 3.1 *The Metropolis Markov chain defined above is reversible with respect to* $\pi(x) = \frac{w(x)}{Z}$.

Proof: We need to check that $\pi(x)P(x,y) = \pi(y)P(y,x)$ for each x and y. Assume without loss of generality that $w(y) \leq w(x)$. Then

$$\begin{aligned}\pi(x)P(x,y) &= \frac{w(x)}{Z}\kappa(x,y)\frac{w(y)}{w(x)} \\ &= \frac{w(y)}{Z}\kappa(y,x) \\ &= \pi(y)P(y,x),\end{aligned}$$

where the second equality follows from the symmetry of κ. ■

As a technical point, we note that the condition $\kappa(x,y) = \kappa(y,x)$ is not necessary. If κ is not symmetric, we can simply change the acceptance probability to $\min\left\{1, \frac{w(y)\kappa(y,x)}{w(x)\kappa(x,y)}\right\}$. It is easy to verify (**exercise!**) that with this definition reversibility w.r.t. π still holds.

3.1.2 The Ising Model

As an example of the above formalism, we will now consider the Ising model. We recall that the model consists of a graph $G = (V, E)$, as shown in Figure 3.1, where each of the vertices corresponds to a *site* and can be in one of the states $\{+, -\}$. Denoting the set of sites by V, the set of configurations Ω is therefore $\{+, -\}^V$. For a configuration $x \in \Omega$, we denote by $a(x)$ the number of edges in E whose endpoints have agreeing spins, and by $d(x)$ the number of edges whose endpoints have disagreeing spins. Also, for a site $s \in V$, we denote its spin in configuration $x \in \Omega$ by $x(s)$. As discussed in Lecture 1, the Gibbs distribution for the Ising model (with ferromagnetic interaction) is given by

$$\begin{aligned}\pi(x) &\propto \exp\left\{\beta \sum_{\{s,t\}\in E} x(s)x(t)\right\} \\ &= \exp\{\beta(a(x) - d(x))\} \\ &\propto \exp\{2\beta a(x)\} \\ &= \lambda^{a(x)},\end{aligned}$$

where $\lambda = \exp(2\beta)$. The second proportionality here follows from the facts that for any $x \in \Omega$, $a(x) + d(x) = 2|E|$, and that the number of edges $|E|$ is fixed. The parameter β is the inverse temperature. In the case of

ferromagnetic interaction, the lower energy configurations are those in which the neighboring spins are fully aligned, and hence the distributions with more spins aligned have higher weight in the Gibbs distribution. We now show how the Metropolis process can be used to sample from the Gibbs distribution.

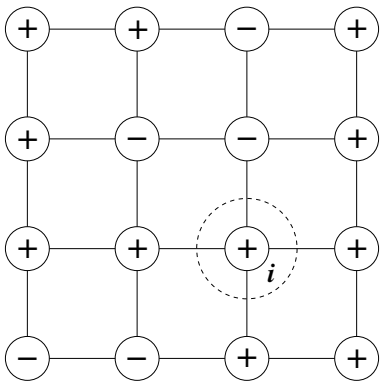


Figure 3.1: Example configuration of a 2-dimensional ferromagnetic Ising model. The dotted circle indicates the vertex i.

We must first define the neighborhood structure on $\Omega = \{+,-\}^V$. A natural choice is to let $x \sim y$ if and only if x and y differ in exactly one spin. Obviously this neighborhood structure is connected. For the proposal distribution, a natural choice is $\kappa(x,y) = \frac{1}{n}$ for all $x \sim y$, where $n = |V|$, the number of sites. Thus, at each step of the Markov chain, we pick a site uniformly at random and "flip" its spin with the appropriate probability given by the Metropolis rule.

As an example, the circled vertex i in Figure 3.1 has three agreeing $+$ neighbors and one disagreeing $-$ neighbor. Let the initial configuration be denoted by x and the configuration resulting from flipping i to $-$ be denoted by y. Then we flip i with probability $\min\left\{1, \frac{w(y)}{w(x)}\right\} = \min\left\{1, \lambda^{a(y)-a(x)}\right\} = \min\left\{1, \frac{1}{\lambda^2}\right\} = \frac{1}{\lambda^2}$, as $\lambda > 1$ for the ferromagnetic model.

3.1.3 The Heat-Bath

Above we saw how the Metropolis process can be used to construct a reversible Markov chain for the ferromagnetic Ising model, or indeed for any spin system. We now mention an alternative approach for spin systems, known as the *heat-bath dynamics*. However, whereas the Metropolis process can be used for an arbitrary probability distribution π, the heat-bath dynamics is suitable only for spin systems.

As an example, let us again consider the ferromagnetic Ising model. The transitions of the heat-bath Markov chain from a state $x \in \Omega$ are as follows:

- Pick a site i uniformly at random.
- Replace the spin $x(i)$ at i by a spin chosen according to the distribution π, conditioned on the spins at the neighbors of i.

So, if d^+ neighbors of i have spin $+$ and d^- have spin $-$, then the conditional probabilities are given by $\Pr_\pi(x(i) = +|d^+, d^-) = \frac{\lambda^{d^+}}{\lambda^{d^+}+\lambda^{d^-}}$ and $\Pr_\pi(x(i) = -|d^+, d^-) = \frac{\lambda^{d^-}}{\lambda^{d^+}+\lambda^{d^-}}$. Returning to Figure 3.1, if vertex i is chosen then its new spin is set to $-$ with probability $\frac{\lambda}{\lambda^3+\lambda}$ and to $+$ with probability $\frac{\lambda^3}{\lambda^3+\lambda}$.

It is an easy **exercise** to check that the heat-bath Markov Chain is aperiodic (because of the presence of self-loops), irreducible (all possible configurations are connected by single spin-flips), and reversible w.r.t. π. This ensures convergence to the Gibbs distribution, as required.

Both the heat-bath dynamics and the Metropolis process for spin systems are special cases of the general framework known as *Glauber Dynamics*. In this framework, the graph G can be arbitrary and a configuration is an assignment of spins from some set $\{1, 2, \ldots, q\}$ to the vertices. In each transition, a vertex is chosen uniformly at random, and its spin is updated according to a local distribution, which depends only on the spins at the vertex and its neighbors. (We will discuss spin systems in more detail later in the course.)

3.2 The Fundamental Theorem and Mixing Time

In this section we present a partial proof of the Fundamental Theorem of Markov Chains. The proof will proceed via estimates of mixing times. We first restate the Fundamental Theorem.

Theorem 3.2 *Let P be the transition matrix of an irreducible and aperiodic Markov chain on a finite set Ω. Then there exists a unique probability distribution π over Ω such that $\pi(x) > 0$ for all $x \in \Omega$, $\pi P = \pi$, and for any initial state $x \in \Omega$*

$$\lim_{t\to\infty} p_x^{(t)} = \pi.$$

Strictly speaking, we will *not* be proving the above theorem completely. We assume that there exists a distribution π such that $\pi P = \pi$ (that is, we assume a stationary distribution exists), and then show that the rest of the theorem holds. We first note that irreducibility of the Markov chain implies that $\pi(x) > 0$ for every x in Ω. We also note that *uniqueness* of π follows from the fact that π satisfies $\pi P = \pi$ and $\lim_{t\to\infty} p_x^{(t)} = \pi$ for every x.

We begin by defining a notion of distance between probability distributions.

Definition 3.3 *For two probability distributions μ and ν on Ω, the* total variation distance *is*

$$\|\mu - \eta\| \equiv \frac{1}{2}\sum_{x\in\Omega} |\mu(x) - \eta(x)| = \max_{A\subseteq\Omega} |\mu(A) - \eta(A)|\,.$$

We note that this is just the ℓ_1 metric, scaled by a factor of $\frac{1}{2}$. This scaling ensures that $\|\mu - \eta\|$ lies in $[0, 1]$.

Exercise: Verify that $\frac{1}{2}\sum_{x\in\Omega} |\mu(x) - \eta(x)| = \max_{A\subseteq\Omega} |\mu(A) - \eta(A)|$.

As an aside, we note that the total variation distance considers the difference between probabilities of all possible events, and hence can be affected severely even by apparently minor changes in the distribution. Let us consider the following example:

Example: Consider n cards. Let μ be the uniform distribution over all permutations of the n cards, and η be the same distribution except that the bottom card is fixed. Then $\|\mu - \eta\| = 1 - \frac{1}{n}$ because the probability of the bottom card of the first deck being the same as the fixed card of the second deck is $\frac{1}{n}$, but for the second deck it is 1.

We will show that the total variation distance between π and $p_x^{(t)}$ decreases exponentially with t/τ_{mix}, where τ_{mix} is a parameter depending on the Markov chain. This gives us an estimate on the time required to get within a given distance of π.

The proof uses the method of *coupled distributions*. We first define the coupling of two distributions and then state the "Coupling Lemma."

Definition 3.4 (Coupling). *Let μ and η be any two probability distributions over Ω. A probability distribution ω over $\Omega \times \Omega$ is said to be a coupling of μ and η if its marginals are μ and η; that is,*

$$\begin{aligned} \mu(x) &= \sum_{y \in \Omega} \omega(x, y), \\ \eta(x) &= \sum_{y \in \Omega} \omega(y, x). \end{aligned}$$

Lemma 3.5 (Coupling Lemma). *Let μ and η be probability distributions on Ω, and let X and Y be random variables with distributions μ and η, respectively. Then*

1. $\Pr[X \neq Y] \geq \|\mu - \eta\|$.
2. *There exists a coupling of (μ, η) such that* $\Pr[X \neq Y] = \|\mu - \eta\|$.

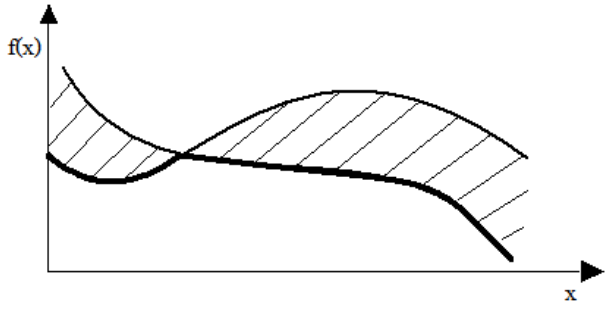


Figure 3.2: Two probability distributions. The bold line indicates the "lower envelope" of the two distributions.

Proof: [Informal sketch] Consider the two probability distributions shown in Figure 3.2. Suppose we try to construct a joint distribution for the two of them that maximizes the probability that they are equal. Clearly, the best we can do is to make $X = Y = z$ with probability $\min\{\Pr(X = z), \Pr(Y = z)\}$ for each value $z \in \Omega$. This is indicated by the bold line in the figure (the "lower envelope" of the two distributions). In this case, the probability that $X \neq Y$ is given by half the area of the shaded region. We then have

$$\frac{1}{2}(\text{Area of Shaded Region}) = \Pr[X \neq Y] = \frac{1}{2} \sum_{x \in \Omega} |\mu(x) - \eta(x)| = \|\mu - \eta\|.$$

■

In order to discuss the convergence of $p_x^{(t)}$ to π, we will need the following definitions:

Definition 3.6

1. *For any $x \in \Omega$, we define $\Delta_x(t) = \|p_x^{(t)} - \pi\|$.*
2. *$\Delta(t) = \max_{x \in \Omega} \Delta_x(t)$ is the maximum possible distance from π after t time steps.*
3. *$\tau_x(\epsilon) = \min\{t : \Delta_x(t) \leq \epsilon\}$ is the first time step t at which the distance $\|p_x^{(t)} - \pi\|$ drops to ϵ.*
4. *$\tau(\epsilon) = \max_{x \in \Omega} \tau_x(\epsilon)$.*

We can now give the key definition of the *mixing time*:

Definition 3.7 (Mixing Time). *The mixing time τ_{mix} of a Markov chain is $\tau(1/2e)$.*

In other words, the mixing time is the time until the variation distance, starting from the worst possible initial state $x \in \Omega$, reaches $1/2e$. This value is chosen for algebraic convenience only, as we shall see below. We now prove some basic facts about the time-dependent behavior of the Markov chain, which will also justify our definition of mixing time. The first fact says that $\Delta(t)$ is non-increasing, so that once the variation distance reaches $1/2e$, it never exceeds it again.

Claim 3.8 *$\Delta_x(t)$ is non-increasing in t.*

Proof: Let $X_0 = x$ and Y_0 have the stationary distribution π. We fix t and couple the distributions of the random variables X_t and Y_t such that $\Pr[X_t \neq Y_t] = \|p_x^{(t)} - \pi\| = \Delta_x(t)$, which is possible because of the Coupling Lemma. We now use this coupling to define a coupling of the distributions of X_{t+1} and Y_{t+1} as follows:

- If $X_t = Y_t$, then set $X_{t+1} = Y_{t+1}$,
- Otherwise, let $X_t \to X_{t+1}$ and $Y_t \to Y_{t+1}$ independently.

Then we have

$$\Delta_x(t+1) \equiv \|p_x^{(t+1)} - \pi\| \leq \Pr[X_{t+1} \neq Y_{t+1}] \leq \Pr[X_t \neq Y_t] = \Delta_x(t).$$

The first inequality holds because of the Coupling Lemma, and the second inequality is true by the construction of the coupling. ■

We now define more general quantities which capture the evolution of distance between corresponding distributions for arbitrary initial configurations.

Definition 3.9

1. $D_{xy}(t) = \|p_x^{(t)} - p_y^{(t)}\|$.
2. $D(t) = \max_{x,y \in \Omega} D_{xy}(t)$.

The following simple relationship between $D(t)$ and $\Delta(t)$ is left as an **exercise**:

Claim 3.10 $\Delta(t) \leq D(t) \leq 2\Delta(t)$.

We now prove that the maximum variation distance, $\Delta(t)$, decays exponentially with time constant τ_{mix}. This will follow from the fact that $D(t)$ is submultiplicative.

Claim 3.11 $\Delta(t) \leq \exp\left(-\left\lfloor \frac{t}{\tau_{\text{mix}}} \right\rfloor\right)$.

Proof: Let $X_0 = x$ and $Y_0 = y$. We use the Coupling Lemma to couple the distributions of X_t and Y_t so that

$$D_{xy}(t) \equiv \|p_x^{(t)} - p_y^{(t)}\| = \Pr[X_t \neq Y_t].$$

We then construct a coupling of X_{t+s} and Y_{t+s} as follows:

- If $X_t = Y_t$ then set $X_{t+i} = Y_{t+i}$ for $i = 1, 2, \ldots s$,
- Otherwise, let $X_t = x'$ and $Y_t = y' \neq x'$. Use the Coupling Lemma to couple the distributions of X_{t+s} and Y_{t+s}, conditioned on $X_t = x'$ and $Y_t = y'$, such that

$$\Pr[X_{t+s} \neq Y_{t+s} | X_t = x', Y_t = y'] = \|p_{x'}^{(s)} - p_{y'}^{(s)}\| = D_{x'y'}(s) \leq D(s). \tag{3.1}$$

The last inequality holds by the definition of $D(t)$. We now have

$$\begin{aligned} D_{xy}(t+s) &= \|p_x^{(t+s)} - p_y^{(t+s)}\| \\ &\leq \Pr[X_{t+s} \neq Y_{t+s}], \text{ by the Coupling Lemma} \\ &\leq D(s) D_{xy}(t), \text{ by the construction of the coupling} \\ &\leq D(s) D(t). \end{aligned}$$

Since this holds for all x, y, we get that $D(t+s) \leq D(s)D(t)$. It follows that $D(kt) \leq D(t)^k$ for all positive integers k. Consequently,

$$\Delta(k\tau_{\text{mix}}) \leq D(k\tau_{\text{mix}}) \leq D(\tau_{\text{mix}})^k \leq (2\Delta(\tau_{\text{mix}}))^k \leq e^{-k}.$$

The last inequality follows from the definition of τ_{mix}, and proves the claim. (It is in the last step that we need $\Delta(\tau_{\text{mix}})$ to be *strictly* less than $\frac{1}{2}$; our choice of $\Delta(\tau_{\text{mix}}) = \frac{1}{2e}$ satisfies this and leads to a particularly simple expression for $\tau(\epsilon)$.) ■

Corollary 3.12 $\tau(\epsilon) \leq \tau_{\text{mix}} \left\lceil \log\left(\epsilon^{-1}\right) \right\rceil$.

The corollary follows immediately from Claim 3.11 and justifies our definition of mixing time: the cost of obtaining any desired variation distance ϵ is only a modest factor times τ_{mix}.

Exercise: The above arguments almost constitute an elementary probabilistic proof of the fundamental theorem. The only hole is that we assumed that the stationary distribution π exists. Show that π exists when P is irreducible and aperiodic. [Hint: Let $x \in \Omega$ be arbitrary. Define $q_x(x) = 1$ and $q_x(y)$ to be the expected number of times that the Markov chain, started in state x, visits y before returning to x. Show that $\pi(y) \propto q_x(y)$ is stationary.] Then assemble all the pieces into a proof of the fundamental theorem.

References

[MRRTT53] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller and E. Teller, "Equations of state calculations by fast computing machines," *J. Chem. Phys*, **21** (1953), pp. 1087–1092.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 4: September 15

Lecturer: Prof. Alistair Sinclair *Scribes: Alistair Sinclair*

4.1 Recapitulation

Recall the following definitions from last time.

$$\begin{aligned}\Delta_x(t) &= \|p_x^{(t)} - \pi\| \\ \Delta(t) &= \max_x \Delta_x(t) \\ \tau(\epsilon) &= \min_t\{\Delta(t) \le \epsilon\} \\ \tau_{\text{mix}} &= \min_t\{\Delta(t) \le \frac{1}{2e}\}\end{aligned}$$

We also proved the following:

$$\begin{aligned}\tau(\epsilon) &\le \tau_{\text{mix}}\lceil \ln \epsilon^{-1}\rceil \\ \Delta(t) &\le e^{-\lfloor t/\tau_{\text{mix}}\rfloor}\end{aligned}$$

4.2 Strong stationary times

Definition 4.1 *A* **stopping time** *is a random variable $T \in \mathbb{N}$ such that the event $\{T = t\}$ depends only on $X_0, X_1, \ldots, X_t$. A stopping time T is a* **strong stationary time** *(SST) if*

$$\forall x,\ \Pr[X_t = x \,|\, T = t] = \pi(x)\ .$$

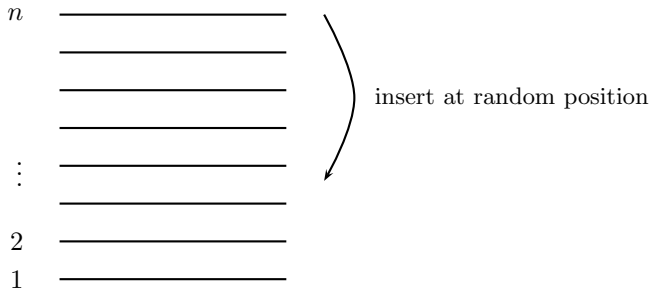
Given the definition of SST, the following claim should be intuitively reasonable:

Claim 4.2 *If T is a SST, then for any starting state x,*

$$\Delta_x(t) \le \Pr[T > t \,|\, X_0 = x]\ .$$

We will prove the claim in a moment. First, however, let us use the claim to bound the mixing time for the top-in-at-random shuffle introduced in a previous lecture.

4.2.1 Example: Top-in-at-random shuffle



Recall that this shuffle is ergodic and its stationary distribution is uniform.

Define

$$T = 1 + \text{time until the original bottom card first reaches the top.}$$

Clearly T is a stopping time. We claim that it is in fact an SST. To see this note that, at any time, all cards below the bottom card were once at the top. Hence each ordering of the cards below the original bottom card is equally likely. Once the bottom card reaches the top, we have a uniformly random permutation on all the other cards; in the next move this card will be inserted in a random position, and we get a random permutation. Hence T is indeed a SST.

We now analyze the tail probability of T. Write T as

$$T = T_1 + T_2 + \ldots + T_{n-1} + 1 \ ,$$

where T_i is the number of steps required for the bottom card to rise from position i to $i+1$ (counting from the bottom of the deck). When the bottom card is at position i, the probability that the top card is inserted below it is $\frac{i}{n}$. So, T_i has a geometric distribution with parameter $\frac{i}{n}$.

Now we can relate this to the classical *coupon collector problem*. Recall that, at each time step, the collector gets one out of n coupons with equal probability. His aim is to continue till he has seen every coupon at least once. The time it takes to see the first coupon is 1. For the second, it is geometrically distributed with parameter $\frac{n-1}{n}$. In general, the number of time steps spent waiting for the kth new coupon is geometrically distributed with parameter $\frac{n+1-k}{n}$. The total time taken is the sum of all these times. It is now easy to see that our random variable T above has precisely this "coupon collector" distribution. The expectation of T is nH_n where H_n is the nth harmonic number. Moreover, it is well known that T is tightly concentrated around its expectation. The following inequality is left to the reader as a standard **exercise**:

$$\Pr\left[T > n \ln n + cn\right] \leq e^{-c} \ .$$

(Actually, for any fixed c, the probability tends asymptotically to $1 - e^{-e^{-c}}$.) Using Claim 4.2, we can therefore deduce that

$$\tau(\epsilon) \leq n \ln n + \lceil n \ln \epsilon^{-1} \rceil \ . \tag{4.1}$$

Thus in particular (setting $\epsilon = 1/2e$) we have the mixing time

$$\tau_{\text{mix}} \leq n \ln n + \lceil (1 + \ln 2) n \rceil \ .$$

Note that (4.1) is a better decay with ϵ than that implied by the general bound $\tau(\epsilon) \leq \tau_{\text{mix}} \lceil \ln \epsilon^{-1} \rceil$. In fact, it turns out that this Markov chain exhibits a so-called "sharp cutoff," in the sense that the variation

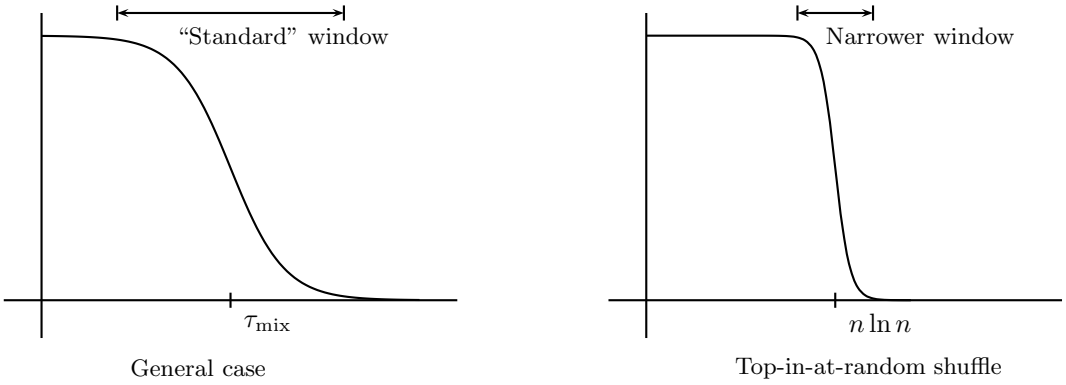


Figure 4.1: The sharp cutoff phenomenon

distance $\Delta(t)$ switches suddenly from close to 1 to close to 0 around the value $t = n \ln n$. I.e., the window over which $\Delta(t)$ decays is of smaller order than τ_{mix} itself (see Figure 4.1).

To say this precisely, write

$$t = n \ln n + n\alpha(n).$$

Then if the function $\alpha(n)$ tends to infinity with n (arbitrarily slowly), we have $\Delta(t) \to 0$ as $n \to \infty$. And if the function $\alpha(n)$ tends to minus infinity (arbitrarily slowly), then $\Delta(t) \to 1$.

The first of these facts is already immediate from (4.1). To see the second fact, we need to prove a *lower* bound on the mixing time, i.e., to prove that the total variation distance from the stationary distribution is large when t is smaller than $n \ln n$. For this, we need to find an event that has very different probabilities under the distribution of the Markov chain at time t and the stationary distribution. Denote the positions of the original bottom k cards by $C_1, \ldots, C_k$. Define the event

$$A_k = \text{“}C_1 < C_2 < \ldots < C_k\text{”} \ ;$$

that is, the cards are in the same relative order as they were initially (though other cards may have been inserted inbetween them). This event certainly holds if the card originally at the kth position from the bottom has not yet reached the top and been reinserted. Using our previous notation, the time it takes for this card to reach the top has the same distribution as

$$T_k + T_{k+1} + \ldots + T_{n-1} \ .$$

So, after t steps of the Markov chain,

$$\begin{aligned} \Pr\left[A_k \text{ holds at time } t\right] &\geq \Pr\left[\text{card at position } k \text{ hasn't been reinserted by time } t\right] \\ &= \Pr\left[T_k + T_{k+1} + \ldots + T_{n-1} + 1 > t\right] \\ &= \Pr\left[\text{coupon collector has } k \text{ or more coupons still to collect after } t \text{ steps}\right]. \end{aligned} \tag{4.2}$$

Now let $t = n \ln n - n\alpha(n)$, where $\alpha(n) \to \infty$ as $n \to \infty$ (arbitrarily slowly). A simple second moment argument (**exercise!**) then shows that, for any fixed k, the probability on the rhs of (4.2) tends to 1 as $n \to \infty$. (I.e., for any fixed k, after time only a little bit less than $n \ln n$ the coupon collector will almost surely still have k coupons left to collect.)

On the other hand, under the uniform distribution π we clearly have $\pi(A_k) = \frac{1}{k!}$. Putting these together gives, for $t = n \ln n - n\alpha(n)$,

$$\|p_x^{(t)} - \pi\| \geq 1 - \frac{1}{k!} - o(1) \quad \text{as } n \to \infty.$$

Since this holds for any fixed k, by taking k large enough and then n large enough (depending on k), we see that $\Delta(t) \to 1$ as $n \to \infty$.

Let us now go back and supply the proof of Claim 4.2.

Proof: [of Claim 4.2]. We want to show that

$$\Delta_x(t) \leq \Pr[T > t \,|\, X_0 = x] \ .$$

Recall the equivalent definition of total variation distance from Lecture 3:

$$\Delta_x(t) \equiv \| \, p_x^{(t)} - \pi \, \| = \max_{A \subseteq \Omega} | \, p_x^{(t)}(A) - \pi(A) \, | \ .$$

In what follows, we denote by T_x the SST for the chain started at state x. Now

$$\begin{aligned}
p_x^{(t)}(A) &\equiv \Pr[X_t \in A] \\
&= \Pr[X_t \in A, T_x > t] + \sum_{t' \leq t} \Pr[X_t \in A, T_x = t'] \\
&= \Pr[X_t \in A \,|\, T_x > t] \Pr[T_x > t] + \pi(A) \sum_{t' \leq t} \Pr[T_x = t'] \\
&= \Pr[X_t \in A \,|\, T_x > t] \Pr[T_x > t] + \pi(A) \left(1 - \Pr[T_x > t]\right) \\
&= \pi(A) + \Pr[T_x > t] \left(\Pr[X_t \in A \,|\, T_x > t] - \pi(A)\right)
\end{aligned}$$

The third line here follows from the definition of a SST. Since the last term in parentheses is the difference of two probabilities, it is bounded in absolute value by 1. Hence $| \, p_x^{(t)} - \pi(A) \, | \leq \Pr[T > t \,|\, X_0 = x]$, as required. ■

4.2.2 Example: Riffle shuffle

Recall from Lecture 2 the Gilbert-Shannon-Reeds model of the riffle shuffle. We now show how a SST can be used to establish an upper bound on τ_{mix} for this shuffle.

Recall that the shuffle proceeds as follows: Cut a pack of n cards into two stacks according to the binomial distribution with parameter $\frac{1}{2}$. Thus, the probability that the cut occurs exactly after k cards is given by $\frac{\binom{n}{k}}{2^n}$. Now choose an interleaving of left and right stacks uniformly at random (u.a.r.). This can be done by dropping cards one at random from the stacks with probability proportional to size of stack at that time (that is, card from left stack is dropped with probability $\frac{|L|}{|L|+|R|}$). Note that the cards of each stack always maintain their relative order. A simple calculation shows that any particular interleaving has probability of $\frac{1}{\binom{n}{k}}$ of occurring. Hence, the probability of any cut followed by any possible interleaving is $\frac{1}{2^n}$.

It turns out that to analyze riffle shuffle, it is easier to work with the *inverse shuffle*:

- Label all cards with 0's and 1's independently and uniformly at random.
- Pull out all the cards with label 0 while maintaining the relative order of the resulting stacks.
- Place the stack with label 0 on top of the stack with label 1.

As an **exercise**, you should check that this permutation is exactly the inverse of the original one, and that both have the same probability $\frac{1}{2^n}$.

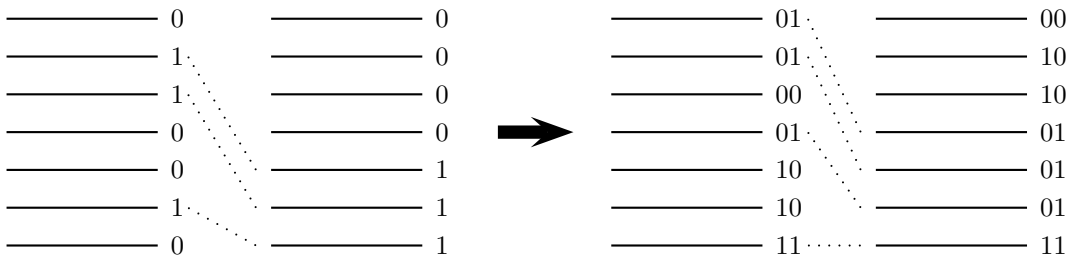


Figure 4.2: Two sequential inverse riffle shuffles

We now claim that it is enough to analyze the inverse shuffle. To justify this, we note that the riffle shuffle (and indeed any card shuffling scheme) is a random walk on a group (in this case, the symmetric group S_n on n elements). We now argue that for a general random walk on a group, $\Delta(t) = \Delta^{inv}(t)$, where Δ^{inv} denotes the variation distance for the inverse walk. More precisely, suppose the original random walk is specified by a set of *generators* $\{g_1, \ldots, g_k\}$ for the group; at each step, a generator is chosen from some fixed probability distribution (so that each generator has non-zero probability of being chosen) and applied to the current state. The inverse random walk is specified in exactly the same way, but using instead the generators $\{g_1^{-1}, \ldots, g_k^{-1}\}$ with the same probability distribution. It is easy to check (**exercise**!) that both the original walk and the inverse walk are doubly stochastic, so both have the uniform stationary distribution.

Now for any given state x, there exists a bijective mapping $\tilde{f}$ between the set of paths of t steps starting at x in the original walk, and the set of paths of t steps starting at x in the inverse walk:

$$\tilde{f}(x \circ \sigma_1 \circ \ldots \circ \sigma_t) = x \circ \sigma_t^{-1} \circ \ldots \circ \sigma_1^{-1} \ .$$

Moreover, the bijection preserves the probabilities of the paths. And if two paths reach the same state, i.e., $x \circ \sigma = x \circ \tau$, then by the group property we must have $x \circ \sigma^{-1} = x \circ \tau^{-1}$, so the paths $\tilde{f}(x \circ \sigma)$ and $\tilde{f}(x \circ \tau)$ also reach the same state. This implies that $\tilde{f}$ induces another bijective mapping f between the set of states reachable from x in t steps of the original walk and the states reachable from x in t steps of the inverse walk (namely, $f(x\sigma) = x\sigma^{-1}$) such that $p_x^{(t)}(y) = p_x^{inv(t)}(f(y))$ for all y. This means that the distributions $p_x^{(t)}$ and $p_x^{inv(t)}$ are identical up to relabeling of the points. And, since the stationary distribution π of both the original walk and the inverse walk is uniform, we conclude that $\| \ p_x^{(t)} - \pi \ \| = \| \ p_x^{inv(t)} - \pi \ \|$. Hence $\Delta(t) = \Delta^{inv}(t)$, as claimed.

Let us return now to the riffle shuffle. Suppose that we perform repeated inverse shuffles starting from an original configuration as shown in Figure 4.2 and we retain the 0–1 labels on the back of cards. After t steps, each card will be labeled by a t-digit binary number. Since during each inverse shuffle the labels are assigned u.a.r., at any time the sets of cards with *distinct* labels are in uniform random relative order (but those with the same label are in the same order as they started in). This implies that the stopping time defined by

$$T = \min\{t : \text{all cards have distinct labels}\}$$

is a SST. After t steps of inverse shuffle, the label of each card is an independent random t-bit binary number. We now make use of the well-known "birthday problem": if n people have their birthdays independent of each other and randomly distributed over cn^2 days, then Pr[some pair has the same birthday] is asymptotically $1 - \exp(-1/2c) \approx 1/2c$. In our case, the number of "birthdays" is the number of t-digit binary numbers, which is 2^t, and we want the probability that some pair of cards has a common label to be small (in particular, less than $1/2e$ to get the mixing time). So we choose c to be a (modest) constant such that $1 - \exp(-1/2c) \leq 1/2e$ and then t so that $2^t \geq cn^2$, i.e., $t \geq 2\log_2 n + \Theta(1)$. Thus the mixing time is $\tau_{\text{mix}} \leq 2\log_2 n + \Theta(1)$.

Aldous [Al83] has shown that $\tau_{\text{mix}} \sim (3/2)\log_2 n$. Thus, we have obtained an almost tight bound. For further information on SST and other stopping rules, the reader is referred to Aldous and Diaconis [AD86] and Lovász and Winkler [LW95]. Finally, Bayer and Diaconis [BD92] have found a way to explicitly calculate $\Delta(t)$ for the riffle shuffle for any value of t and for any number of cards n. (For $n = 52$, the number of possible deck arrangements is $52! \approx 8.07 \times 10^{67}$ and hence a brute force approach is not feasible, so some ingenuity is required to do this.) In particular, for $n = 52$, the exact variation distances are shown in the following table:

t	≤ 4	5	6	7	8	9
$\Delta(t)$	1.00	0.92	0.61	0.33	0.17	0.09

Thus, they argue that for practical purposes like card games in a casino, seven riffle shuffles are sufficient to prevent even the best card player from exploiting any structure remaining in the deck.

For most Markov chains used in practice, there is no obvious choice of a strong stationary time. Hence, more sophisticated methods are needed to bound the mixing time. We will begin to discuss these in the next lecture.

4.3 References

[AD86] D. Aldous and P. Diaconis, "Shuffling cards and stopping times," *American Mathematical Monthly*, **93** (1986), pp. 333–348.

[Al83] D. Aldous, "Random walks on finite groups and rapidly mixing Markov chains," *Séminaire de Probabilites XVII*, Springer Verlag, Lecture Notes in Mathematics **986** (1983), pp. 243–297.

[BD92] D. Bayer and P. Diaconis, "Trailing the dovetail shuffle to its lair," *Annals of Applied Probability*, **2** (1992), pp. 294–313.

[LW95] L. Lovász and P. Winkler, "Mixing of random walks and other diffusions on a graph," In *Surveys in Combinatorics* (P. Rowlinson, ed.), London Mathematical Society Lecture Notes Series 218, Cambridge University Press (1986), pp. 333–348.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 5: September 17

Lecturer: Prof. Alistair Sinclair *Scribes: Suman Ravuri*

5.1 Coupling for bounding the mixing time

Consider as usual an ergodic (i.e., irreducible, aperiodic) Markov chain on some state space Ω. Consider two particles started at positions x and y in Ω, each individually moving through the state space according to the Markov transition matrix P, but whose evolutions may be coupled in some way. We will show below that the time until the two particles meet gives a bound on the mixing time; more precisely, we will show that

$$\Delta(t) \leq \max_{x,y} \Pr[\text{two particles started at positions } x,\, y \text{ have not met by time } t],$$

where we recall that

$$\Delta(t) = \max_x \|p_x^{(t)} - \pi\|.$$

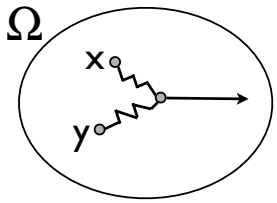


Figure 5.1: Coupling

More formally:

Definition 5.1 *A* coupling *of a Markov chain P is a pair process (X_t, Y_t) such that:*

1. *each of $(X_t, \cdot)$ and $(\cdot, Y_t)$, viewed in isolation, is a faithful copy of the Markov chain; that is,*

$$\Pr[X_{t+1} = b \,|\, X_t = a] = P(a,b) = \Pr[Y_{t+1} = b \,|\, Y_t = a];$$

   and

2. *if $X_t = Y_t$ then $X_{t+1} = Y_{t+1}$.*

Figure 5.1 gives a pictorial illustration of this definition.

Now define the random variable $T_{xy} = \min\{t : X_t = Y_t \,|\, X_0 = x, Y_0 = y\}$ to be the (stopping) time until the two processes meet. The following claim gives the desired upper bound on the mixing time:

Claim 5.2

$$\Delta(t) \leq \max_{x,y} Pr[T_{xy} > t]$$

.

Proof: Recall that we defined in Lecture 3 that:

$$D(t) = \max_{x,y} \|p_x^{(t)} - p_y^{(t)}\|$$

and that $\Delta(t) \leq D(t)$. Now,

$$\begin{aligned} \Delta(t) &\leq D(t) \\ &= \max_{x,y} \|p_x^{(t)} - p_y^{(t)}\| \\ &\leq \max_{x,y} \Pr[X_t \neq Y_t \,|\, X_0 = x, Y_0 = y] \\ &= \max_{x,y} \Pr[T_{xy} > t \,|\, X_0 = x, Y_0 = y]. \end{aligned}$$

The only real content in this proof is the third line, where we use the coupling lemma from Lecture 3. This completes our proof. ■

Coupling ideas for analyzing the time-dependent behavior of Markov chains can be traced back to Doeblin in the 1930s. However, the modern development of the topic was initiated by David Aldous [Al83].

5.2 Examples

5.2.1 Simple random walk on the hypercube $\{0,1\}^n$

The n-dimensional cube is a graph with 2^n vertices, each of which can be encoded as an n-bit binary string $b_1 b_2 \cdots b_n$, whose neighbours are the strings which differ from it by Hamming distance exactly 1. We define a random walk on the cube by the following:

1. With probability 1/2, do nothing.
2. Else, pick a coordinate $i \in \{1, \ldots, n\}$ uniformly at random and flip coordinate x_i (i.e. $x_i \to 1 - x_i$).

This setup is clearly equivalent to the following:

1. Pick a coordinate $i \in \{1, \ldots, n\}$ uniformly at random *and* a bit $b \in \{0, 1\}$ uniformly at random.
2. Set $x_i = b$.

This second description of the random walk dynamics suggests the following coupling: make X_t and Y_t choose the *same* i and b at every step. Clearly this is a valid coupling: obviously each of X_t and Y_t is performing exactly the above random walk.

To analyze the time T_{xy}, notice that once every $i \in \{1, \ldots, n\}$ has been chosen at least once, X_t must equal Y_t. (This is because, once a coordinate i has been chosen, X_t and Y_t agree on that coordinate at all future times.) Thus for any x and y, T_{xy} is stochastically dominated by the time for a coupon collector to collect

all n coupons. Thus $\Pr[T_{xy} > n \ln n + cn] < e^{-c}$, and hence by Claim 5.2 we have $\Delta(n \ln n + cn) \leq e^{-c}$; therefore in particular

$$\tau_{\text{mix}} \leq n \ln n + O(n),$$

and more generally

$$\tau(\varepsilon) \leq n \ln n + \lceil n \ln(\varepsilon^{-1}) \rceil.$$

An exact analysis of this very simple random walk reveals that in fact $\tau_{\text{mix}} \sim (1/2) n \ln n$, so our analysis is tight up to a factor of 2.

5.2.2 Another random walk on the hypercube

The above coupling was a bit trite because of the self-loop probability of 1/2. In this example we consider the same random walk but with a different self-loop probability. This means we will have to be a bit more careful in defining the coupling.

1. With probability $\frac{1}{n+1}$, do nothing.
2. Otherwise, with probability $\frac{1}{n+1}$ for each neighbor, go to one of the neighbors.

This can be reformulated as:

1. Pick $i \in \{0, 1, \ldots, n\}$.
2. If $i = 0$ do nothing; otherwise, flip x_i.

Note that the self-loop probability here is $\frac{1}{n+1}$.

To define a coupling here, we'll write $d(X_t, Y_t)$ for the number of coordinates in which X_t and Y_t differ.

If $d(X_t, Y_t) > 1$, then:

1. if X_t picks $i = 0$ then Y_t picks $i = 0$.
2. if X_t picks coordinate i where they agree, Y_t also picks i.
3. if X_t picks coordinate i where they *disagree*, then Y_t picks coordinate $f(i)$, where f is a cyclic permutation on the disagreeing coordinates.

   (For example, if $X_t = 110100$ and $Y_t = 000010$, then the permutation could be the one that sends $1, 2, 4, 5$ to $2, 4, 5, 1$ respectively.)

If $d(X_t, Y_t) = 1$, then, letting i_0 be the disagreeing coordinate:

1. if X_t picks $i = 0$, then Y_t picks i_0.
2. if X_t picks $i = i_0$, then Y_t picks 0.
3. else, both X_t and Y_t pick the same i.

Note that the distance between X_t and Y_t never increases under this coupling.

In this scenario, the time for the distance between X_t and Y_t to decrease to at most 1 is dominated by the coupon collector for $n/2$ coupons, and so with high probability this is $\sim (n/2)\ln n + O(n)$. The time for the last single disagreement to disappear (that is, the regime in which $d(X_t, Y_t) = 1$, if this happens) has a geometric distribution with mean n, so contributes $O(n)$ to the time for the chains to meet. Summing these two contributions and using Claim 5.2 gives

$$\tau_{\text{mix}} \leq \frac{1}{2} n \ln n + O(n).$$

For this example, Diaconis and Shahshahani [DS81] have shown that the true value of the mixing time is $\tau_{\text{mix}} \sim (1/4)n \ln n + O(n)$, so again our analysis is tight up to a factor of 2.

Exercise: For every self-loop probability δ with $\delta > \text{const}/n$ and $\delta < 1 - \text{const}/n$, show that $\tau_{\text{mix}} \leq c_\delta n \ln n + O(n)$, where c_δ is a constant. (The exact value of c_δ is not important.)

Exercise: Give a strong stationary time argument for the random walk on $\{0,1\}^n$ with self-loop probability $1/2$ (that is, the walk discussed in section 5.2.1).

5.2.3 Top-in-at-Random Shuffle

As previously discussed, the top-in-at-random shuffle involves repeatedly taking the top card from a deck of n cards and inserting it at a position chosen uniformly at random in the deck. Analyzing this shuffle by coupling is best done through its inverse, defined as follows:

- Pick a card c from the deck uniformly at random.
- Move card c to the top of the deck.

Recall from Lecture 4 that the mixing times of the original shuffle and the inverse shuffle are identical.

To construct a coupling, we will envision the two decks of n cards X_t and Y_t at time t. We define the coupling by making both X_t and Y_t choose the *same* card c (which of course is not necessarily in the same position in both decks) and move it to the top. Now the key observation is the following: once a card has been chosen in the coupling, this card will be in the same position in both decks for the rest of time. [**Exercise:** Check this.] T_{xy} is therefore once again dominated by the coupon collector random variable for n coupons. This leads to

$$\tau_{\text{mix}} \leq n \ln n + O(n)$$

and

$$\tau(\varepsilon) \leq n \ln n + \lceil n \ln \varepsilon^{-1} \rceil.$$

As we saw in the previous lecture, this value is tight.

5.2.4 Random Transposition Shuffle

This shuffle is defined as follows:

- Pick cards c and c' uniformly at random.
- Switch c and c'.

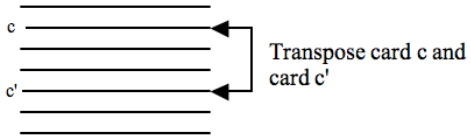


Figure 5.2: Random Transposition Shuffle with cards c and c'

An equivalent, more convenient description is the following:

- Pick card c and position p uniformly at random.
- Exchange card c with the card at position p in the deck.

It is easy to define a coupling using this second definition: namely, make X_t and Y_t choose the same c and p at each step. This coupling ensures that the distance between X and Y is non-increasing. More explicitly, writing $d_t = d(X_t, Y_t)$ for the number of positions at which the two decks differ, we have the following case analysis:

1. If card c is in the same position in both decks, then $d_{t+1} = d_t$.
2. If card c is in different positions in the two decks, there are two possible subcases:
   (a) If the card at position p in both decks is the same, then $d_{t+1} = d_t$.
   (b) Otherwise, $d_{t+1} \leq d_t - 1$.

Thus we get a decrease in distance only in case 2(b), and this occurs with probability

$$\Pr[d_{t+1} < d_t] = \left(\frac{d_t}{n}\right)^2.$$

Therefore, the time for d_t to decrease from value d is stochastically dominated by a geometric random variable with mean $\left(\frac{n}{d}\right)^2$. This implies that $E[T_{xy}] \leq \sum_{d=1}^{n} \left(\frac{n}{d}\right)^2$, which is $O(n^2)$.

Invoking Markov's inequality, we get that $\Pr[T_{xy} > cn^2] < c' = \frac{1}{2\varepsilon}$ for a suitable constant c, which leads to the bound

$$\tau_{\text{mix}} \leq cn^2.$$

Actually, for this shuffle it is known that

$$\tau_{\text{mix}} \sim \frac{1}{2} n \ln n,$$

so our analysis in this case is off by quite a bit.

Exercise: Design a better coupling that gives $\tau_{\text{mix}} \leq O(n \ln n)$.

Remark: It turns out that for any ergodic Markov chain there is *always* a coupling that is optimal, in the sense that the coupling time satisfies

$$Pr[T_{xy} > t] = D_{xy}(t).$$

This is a very general theorem of Griffeath [Gr78]. However, the couplings that achieve the mixing time may involve looking arbitrarily far into the future in the two Markov chains, and thus are not useful in practice. The couplings we have used so far—and almost all couplings used in algorithmic applications—are *Markovian* couplings: the evolution of X_t and Y_t depends only on the current values of X_t and Y_t. For some Markov chains, there is (provably) no Markovian coupling that achieves the mixing time [KR99]. Recently some exciting progress has been made in the use of non-Markovian couplings; see, for example [HV03].

5.2.5 Graph Colorings

This section introduces the problem of sampling colorings of a graph using a Markov chain. This is of interest, for example, when estimating the number of legal colorings for a given graph and color set, a fundamental combinatorial problem. It also has applications to the antiferromagnetic Potts model in statistical physics. Throughout, our input will be an undirected graph $G = (V, E)$, where V is a set of n vertices, and E is a set of edges connecting V. The maximum degree of any vertex in V is denoted Δ. We will also be given a set of colors $Q = \{1, \ldots, q\}$.

Our goal is to sample proper colorings of G uniformly at random. A coloring labels each vertex with a color $c \in Q$. A proper coloring is defined as one where no two vertices connected by an edge share the same color.

The decision problem for graph coloring involves deciding whether it is possible to construct a proper coloring for a given graph. Some important results about the graph coloring decision problem:

- If $q \geq \Delta + 1$, the answer to the decision problem is trivially true.
- if $q = \Delta$, the graph G has a proper coloring unless it contains a $(\Delta + 1)$-clique or is an odd cycle (in which case $\Delta = 2$); this is a classical result known as Brooks' Theorem.
- if $q < \Delta$, the decision problem is NP complete.

Obviously constructing a random coloring is at least as hard as solving the decision problem, so we shouldn't expect to be able to do so in polynomial time unless $q \geq \Delta$. In fact, the decision problem for $q = \Delta$ is also somewhat non-trivial, so we will generally be interested only in the case when $q \geq \Delta + 1$.

We are concerned with sampling proper colorings using a Markov chain. One natural Markov chain is as follows, where the current state is any proper coloring of G:

- Pick a vertex $v \in V$ uniformly at random and a color $c \in \{1, \ldots, q\}$ uniformly at random.
- Recolor v with c if this yields a proper coloring, else do nothing.

This Markov chain is symmetric and aperiodic, but is not always irreducible. In particular, if $q \leq \Delta + 1$, we can have "frozen" colorings in which no move is possible (even though other proper colorings do exist). One very simple such example is given in figure 5.3, in which $\Delta = 2$ and $q = 3$.

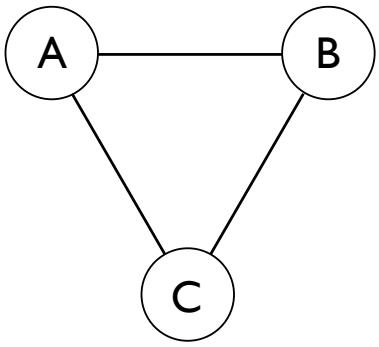


Figure 5.3: Proper coloring, $\Delta = 2$, $q = 3$.

However, it turns out that this problem cannot occur when the number of colors is at least $\Delta + 2$:

Important Exercise: show that the above Markov chain is irreducible if $q \geq \Delta + 2$.

Two interesting conjectures regarding proper graph coloring:

1. Random sampling of proper colorings can be done in polynomial time whenever $q \geq \Delta + 1$.
2. The Markov chain outlined above has mixing time $O(n \ln n)$ whenever $q \geq \Delta + 2$.

Unfortunately, we cannot yet prove either of these conjectures. However, we will now begin to approach the second conjecture by showing that, if $q \geq 4\Delta + 1$, then the mixing time is $O(n \log n)$. Note that intuitively we would expect that, the more colors we have, the easier it should be to prove that the the Markov chain is rapidly mixing. This is because with more colors the vertices are "more independent."

Claim 5.3 *Provided that $q \geq 4\Delta + 1$, the mixing time is $O(n \log n)$.*

This theorem (in the stronger version $q \geq 2\Delta + 1$) is due to Jerrum [Je95], and independently to Salas and Sokal [SS97]. Note that $O(n \log n)$ mixing time is the best we could hope for, since by coupon collecting it takes that long before all the vertices have a chance to be recolored. (Actually the coupon-collecting analogy is not strictly accurate; however, the $\Omega(n \log n)$ lower bound can be proved by a more careful argument [HS05].)

Proof: We will apply coupling to two copies X_t and Y_t of the chain. Our coupling procedure is very simple: Let X_t and Y_t choose the same vertex v and the same color c at each step, and recolor if possible. Let us now analyze this coupling.

Define $d_t := d(X_t, Y_t)$ to be the number of vertices where the colors disagree. For every step in our chain a vertex v and a color c are chosen, which could result in:

- "Good moves" (d_t decreases by 1): the chosen vertex v is a disagreeing vertex, and the chosen color c is not present at any neighbor of v in either X_t or Y_t. In this case, the coupling will recolor v to the same color in both X_t and Y_t, thus eliminating one disagreeing vertex. Since the neighbors of v have at most 2Δ distinct colors in X_t and Y_t, there are at least $d_t(q - 2\Delta)$ good moves available.
- "Bad moves" (d_t increases by 1): the chosen vertex v is not a disagreeing vertex but is a neighbor of some disagreeing vertex v', and the chosen color c is one of the colors of v' (in either X_t or Y_t). In this case, v will be recolored in one of the chains but not the other, resulting in a new disagreement at v. Since each disagreeing vertex v' has at most Δ neighbors, and there are only two corresponding "bad" colors c, the number of bad moves available is at most $2d_t\Delta$.
- "Neutral moves" (d_t is unchanged): all choices of (v, c) that do not fall into one of the above two categories lead to no change in d_t.

Note that the difference between the numbers of good and bad moves is (at least) $d_t(q - 4\Delta)$, so we expect the distance to decrease when $q \geq 4\Delta + 1$. We now make this argument precise.

Since each move has the same probability, namely $\frac{1}{qn}$, we can compute the expected change in d_t under one step of the coupling:

$$E[d_{t+1}|X_t, Y_t] \leq d_t - \frac{d_t(q - 2\Delta)}{qn} + \frac{2d_t\Delta}{qn} = d_t\left(1 - \frac{q - 4\Delta}{qn}\right).$$

Iterating this we get that

$$E[d_t|X_0, Y_0] \leq \left(1 - \frac{q - 4\Delta}{qn}\right)^t d_0 \leq \left(1 - \frac{q - 4\Delta}{qn}\right)^t n \leq \left(1 - \frac{1}{qn}\right)^t n. \tag{5.1}$$

This implies that

$$\begin{aligned}\Pr[d_t > 0 | X_t, Y_t] &= \Pr[d_t \geq 1 | X_t, Y_t] \\ &\leq \mathrm{E}[d_t | X_t, Y_t] \qquad \text{(Markov's Ineq.)} \\ &\leq \left(1 - \frac{1}{qn}\right)^t n.\end{aligned}$$

To ensure that $\Delta(t) \leq \epsilon$, it is sufficient to let $t = qn(\ln n + \ln \epsilon^{-1})$. Thus in particular the mixing time is $\tau_{\text{mix}} = O(qn \log n)$. ■

Remark: The fact that the above mixing time bound increases with q is an artefact of our crude approximation $q - 4\Delta \geq 1$ in the last step of line (5.1). Clearly the value q in the final bound can be replaced by $\frac{q}{q-4\Delta}$.

Exercise (harder): Devise an improved coupling that achieves a similar $O(n \log n)$ mixing time for $q \geq 2\Delta + 1$.

References

[Al83] D. Aldous, "Random walks on finite groups and rapidly mixing Markov chains," *Séminaire de Probabilites XVII*, pages 243- 297. Springer Verlag, 1983. Lecture Notes in Mathematics 986.

[DS81] P. Diaconis and M. Shahshahani, "Generating a random permutation with random transpositions," *Zeitschrift für Wahrscheinlichketistheorie und verwandte Gebiete* **57** (1981), pp. 159–179.

[Gr78] D. Griffeath. "Coupling methods for Markov processes," in *Studies in Probability and Ergodic Theory*, G.-C. Rota ed., Academic Press, 1978, pp. 1–43.

[HS05] T. Hayes and A. Sinclair, "A general lower bound for mixing of single-site dynamics on graphs," *Proceedings of the 46th IEEE Symposium on Foundations of Computer Science (FOCS)*, 2005, pp. 511–520.

[Je95] M.R. Jerrum, "A very simple algorithm for estimating the number of k-colourings of a low-degree graph," *Random Structures and Algorithms* **7** (1995), pp. 157–165.

[HV03] T.P. Hayes and E. Vigoda, "A non-Markovian coupling for randomly sampling colorings," *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science*, 2003, pp. 618–627.

[KR99] V.S. Anil Kumar and H. Ramesh, "Markovian coupling vs. conductance for the Jerrum-Sinclair chain," *Proceedings of the 40th FOCS*, 1999, pp. 241–250.

[SS97] J. Salas and A.D. Sokal, "Absence of Phase Transition for Antiferromagnetic Potts Models via the Dobrushin Uniqueness Theorem," *Journal of Statistical Physics* **98** (2000), p. 551.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 6: September 22

Lecturer: Prof. Alistair Sinclair *Scribes: Alistair Sinclair*

In today's lecture we discuss *path coupling*, a variation on the coupling technique that makes it much easier to apply in practice. We will use path coupling to show that the mixing time for the graph coloring Markov chain discussed last time remains $O(n \log n)$ under a much weaker condition on the number of colors.

6.1 Path coupling

"Path coupling," an idea introduced by Bubley and Dyer [BD97], is a powerful engineering tool that makes it much easier to design couplings in complex examples.

Definition 6.1 *A* pre-metric *on Ω is a connected undirected graph with positive edge weights with the following property: every edge is a shortest path. We call two elements $x, y \in \Omega$* adjacent *if (x, y) is an edge in the pre-metric.*

Notice that a pre-metric extends to a metric in the obvious way (just take shortest path distances in the graph of the pre-metric).

Path coupling says that, when defining a coupling for a Markov chain on Ω, it is enough to specify the coupling *only for pairs of states that are adjacent in the pre-metric*. This will usually be much easier than specifying the coupling for arbitrary pairs of states.

This fact is expressed in the following theorem:

Theorem 6.2 *Suppose there exists a coupling $(X, Y) \to (X', Y')$ defined only on pairs (X, Y) that are adjacent in the pre-metric such that*

$$\mathrm{E}[d(X', Y')|X, Y] \leq (1 - \alpha)d(X, Y) \text{ for some } \alpha \in [0, 1], \tag{6.1}$$

where d is the metric extending the pre-metric. Then this coupling can be extended to a coupling which satisfies (6.1) on all *pairs (X, Y).*

Note that (6.1) says that the distance between X and Y (as measured in the metric d) decreases in expectation by a factor $(1 - \alpha)$. Just as in our analysis of the graph colorings Markov chain in the previous lecture, assuming d takes non-negative integer values this immediately leads to a bound on the mixing time of $\tau_{\mathrm{mix}} = O(\frac{1}{\alpha} \log D)$, where D is the maximum distance between any two states. (In that application, we had $\alpha = \frac{1}{qn}$ and $D = n$.)

Proof: Let (X, Y) be arbitrary, not necessarily adjacent. Consider any shortest path in the pre-metric

$$X = Z_0 \to Z_1 \to Z_2 \to \ldots Z_{k-1} \to Z_k = Y.$$

We construct a coupling of one move of (X, Y) by composing couplings for each adjacent pair (Z_i, Z_{i+1}) along this path, as follows:

- Map (Z_0, Z_1) to (Z'_0, Z'_1) according to the coupling.
- For each $i \geq 1$ in sequence, map (Z_i, Z_{i+1}) to (Z'_i, Z'_{i+1}) according to the coupling, but conditional on Z'_i already being chosen.

This process constructs a coupling $(X, Y) \to (X', Y') = (Z'_0, Z'_k)$. [**Exercise:** convince yourself that this is indeed a valid coupling!]

Now the expected change in distance under the coupling can be analyzed as follows:

$$\begin{aligned}
\mathrm{E}[d(X', Y')] &\leq \mathrm{E}\left[\sum_{i=0}^{k-1} d(Z'_i, Z'_{i+1}) \,\middle|\, X, Y\right] \\
&\leq (1-\alpha)\sum_{i=0}^{k-1} d(Z_i, Z_{i+1}) \\
&\leq (1-\alpha) d(X, Y),
\end{aligned}$$

which establishes (6.1) as required. ■

6.2 Application to graph coloring

We now apply path coupling to the graph coloring Markov chain from the previous lecture, and see how it leads to a simpler, more elegant and tighter analysis.

As before, let $G = (V, E)$ be a graph of maximum degree Δ, and assume that we have q colors at our disposal. A (proper) coloring is an assignment of a color to each node such that no two vertices linked by and edge have the same color.

We proved last time that for $q \geq 4\Delta + 1$, the mixing time is $O(n \log n)$. We will now refine this result using path coupling:

Theorem 6.3 *Provided $q \geq 2\Delta + 1$, $\tau_{\mathrm{mix}} = O(n \log n)$.*

Proof: We will use the following pre-metric: two colorings X, Y are adjacent iff they differ at exactly one vertex. In this case, we set $d(X, Y) = 1$. We will extend this pre-metric to the *hamming metric*: $d(X, Y)$ is the number of vertices at which X, Y differ.

Note, however, that in order to have the hamming distance indeed be the extension of the pre-metric described above, we need to let our state space include non-valid colorings as well. This is because the shortest path from one coloring to another might involve temporarily assigning illegal colors to vertices. We keep the transitions as before, i.e., we do not allow the MC to make a transition to an invalid coloring. Thus the state space is not irreducible, but rather it consists of a single irreducible component (namely, all proper colorings) plus some transient components consisting of invalid colorings. It is easy to see that this Markov chain converges to the uniform distribution on the proper colorings, as before; and, moreover, that a bound on the mixing time derived using coupling extends to a bound on the mixing time of the original Markov chain (without the invalid colorings). [**Exercise:** Verify this by going back to the basic coupling bound in Lecture 5.]

Now let X, Y be any two (not necessarily proper) colorings that are adjacent; this means that they differ at only one vertex, say v_0. We define our coupling for (X, Y) as follows:

- Pick the same vertex v in both chains.
- If v is not in the neighborhood of the unique disagreeing vertex v_0, then pick the same color c in both chains.
- If $v \in N(v_0)$, match up the choice of colors as follows:

$$\begin{aligned} c_X &\longleftrightarrow c_Y \\ c_Y &\longleftrightarrow c_X \\ c &\longleftrightarrow c, \end{aligned}$$

  where c_X, c_Y are the colors of v_0 in X, Y respectively, and $c \notin \{c_X, c_Y\}$.
- Having chosen a vertex v and a color c in both chains, recolor v with c in each chain if possible.

This is clearly a valid coupling because each of X, Y, viewed in isolation, makes a move according to the Markov chain. Note that we don't need to explicitly define the coupling on other pairs (X, Y) as this is taken care of by path coupling.

To analyze the above coupling, we count the numbers of "good" moves and "bad" moves. A "good" move corresponds to choosing the disagreeing vertex v_0 and some color not present among its neighbors; hence there are at least $q - \Delta$ good moves. A "bad" move corresponds to picking a neighbor $v \in N(v_0)$ together with one choice of color (namely, the combination c_Y in X and c_X in Y, for then v_0 is recolored to different colors in the two chains and becomes a new disagreeing vertex; note that the complementary choice c_X in X and c_Y in Y is not a bad move because neither chain will recolor v in this case). Hence the number of bad moves is at most Δ (i.e., the number of neighbors of v_0). All other moves are neutral (do not change the distance).

Since each move occurs with the same probability $\frac{1}{qn}$, and since $d(X_t, Y_t) = 1$ for all adjacent pairs X_t, Y_t, we can conclude that

$$E[d(X_{t+1}, Y_{t+1})|X_t, Y_y] \leq \Big(1 - \frac{q - 2\Delta}{qn}\Big) d(X_t, Y_t).$$

Thus, assuming $q \geq 2\Delta + 1$, (6.1) holds with $\alpha \geq \frac{q-2\Delta}{qn}$ and $D = n$, so we get $\tau_{\text{mix}} = O(n \log n)$ as claimed. ■

Theorem 6.3 was first proved by Jerrum [Je95] (see also Salas and Sokal [SS97]) using a direct coupling rather than path coupling. The resulting analysis, though still elementary, is quite a bit messier than that above.

6.3 Going beyond the 2Δ bound

The ultimate goal of research on MCMC for graph coloring is to resolve the following:

Conjecture 6.4 *τ_{mix} is $O(n \log n)$ for all $q \geq \Delta + 2$.*

Note that this is precisely the range of q for which the Markov chain is guaranteed to be connected. The Conjecture has been recently proved for the case of Δ-regular trees [MSW06].

There is a further interesting related conjectured connection between colorings on trees and on general graphs. The range $q \geq \Delta + 1$ is precisely the region in which the Gibbs measure on the infinite Δ-regular

tree is *unique*, i.e., if we set any boundary condition (fixed coloring) on the leaves of the tree, the asymptotic influence of this boundary condition on the distribution of the color at the root of the tree goes to zero with distance if and only if $q \geq \Delta + 1$. It is believed that the same holds for general graphs (i.e., among all graphs of maximum degree Δ, the decay of influence with distance is slowest for the tree). Resolving either this conjecture, or the related one above for Markov chain mixing, would be of great interest in combinatorics, computer science and statistical physics.

We now briefly examine some recent work that approaches Conjecture 6.4.

6.3.1 Ideas for improving $2\Delta \to 1.76\Delta$

Recall our path coupling analysis in the proof of Theorem 6.3 above. Our bound of $q - \Delta$ on the number of good moves there was pessimistic, because in a typical coloring we would expect significantly fewer than Δ colors (the maximum possible number) to be represented in the neighborhood $N(v_0)$.

Let $A(X, v)$ denote the number of *available* colors at v in coloring X, i.e., the number of colors not represented in $N(v)$. Since the number of bad moves is (at most) Δ, the previous analysis will still go through provided we have $A(X, v) > \Delta$.

To get a handle on $A(X, v)$, suppose each vertex is colored independently and u.a.r. in X. Then by linearity of expectation we have

$$\mathrm{E}[A(X, v)] = q\Big(1 - \frac{1}{q}\Big)^{\Delta} \approx qe^{-\frac{\Delta}{q}}$$

is the expected value. Thus we will have $\mathrm{E}[A(X, v)] > \Delta$ provided $qe^{-\frac{\Delta}{q}} > \Delta$, which is true whenever $q > \alpha\Delta$ where α is the solution to $x = e^{\frac{1}{x}}$. In particular, $\alpha \approx 1.76$. So we might hope that we get $O(n \log n)$ mixing time for $q > 1.76\Delta$.

This is the crux of the proof presented in [DF03]. However, quite a bit of work remains to be done: we need to justify why it is OK to work with a random coloring; in a random coloring, the neighbors of v are not colored independently; and we cannot work just with the *expected* value $\mathrm{E}[A(X, v)]$.

To sketch how to turn the above intuition into a real proof, we follow the development of [HV05].

Definition 6.5 *Say that a coloring X of G satisfies the* local uniformity *property if, for all v, $A(X, v)$ is at least $q(e^{-\Delta/q} - \delta)$.*

Here δ is an arbitrary small constant. Thus local uniformity says that the number of available colors at all vertices is not much less than the expected value we calculated above.

The following Fact captures formally the intuition from our informal calculation above:

Fact 6.6 *Let G be triangle-free and have maximum degree Δ. Assume that the number of colors satisfies $q \geq \max\{\Delta + 2/\delta, C \log n\}$, where C is a constant that depends on δ. Then a random q-coloring of G satisfies the local uniformity property (w.r.t. δ) with probability at least $1 - O(n^{-4})$.*

The Fact is really only interesting in the case where $\Delta \geq C \log n$. In that case the condition on q is only $q \geq \Delta + \text{const}$, which will certainly hold in our application.

We will not prove this Fact here; the proof follows from a straightforward application of large deviation bounds on the colors of the neighbors of a given vertex v, conditional on an arbitrary fixed coloring of the

rest of the graph. (Note that, since G is triangle-free, these colors are conditionally independent. And since the number of colors is $\geq C \log n$, large deviation bounds hold.) For the details, see [HV05].

To use this fact, we assume $\Delta \geq C \log n$ and we return to our original coupling idea (without path coupling). (As mentioned earlier, [Je95] shows how to obtain $q \geq 2\Delta + 1$ using a direct coupling.) Now we note that, rather than coupling two arbitrary initial states (X_0, Y_0), it is enough to couple an arbitrary state X_0 with a *stationary* Y_0, i.e., in the coupling we may assume that Y_0 is a uniformly random coloring. [**Exercise:** Check this by going back to our original justification of coupling at the beginning of Lecture 5.] So at all times t, Y_t is a uniformly random coloring, and thus by Fact 6.6 it is locally uniform with high probability. In fact, if we let $T = cn \log n$ (for some constant c), then applying a union bound over times t we see that Y_t satisfies local uniformity for $t = 0 \ldots T$ with probability $\geq 1 - O(n^{-2})$.

Now, exploiting local uniformity to bound the number of good moves as sketched earlier, we get an expected contraction $1 - O(\frac{1}{n})$ in distance at each step, from which we can conclude that if $q \geq 1.76\Delta$ then $\Pr[X_T \neq Y_T] \leq O(n^{-2})$. (Here we are choosing the constant c large enough that the probability of not having coupled is this small. Note that some work still needs to be done here; in particular, we need to check that it is enough that just one of the two colorings, Y_t, satisfies local uniformity.)

Hence, we obtain that

$$E[d(X_T, Y_T)|X_0] \leq n(O(n^{-2}) + O(n^{-2})) = O(n^{-1}),$$

where the first $O(n^{-2})$ term bounds the probability that local uniformity fails, and the second bounds the probability that $\Pr[X_T \neq Y_T]$, and the factor n comes from the fact that we can always bound $d(X_T, Y_T)$ by its maximum value n. Thus, using Markov's inequality we get

$$\Pr[d(X_T, Y_T) > 0|X_0] = \Pr[d(X_T, Y_T) \geq 1|X_0] \leq E[d(X_T, Y_T)|X_0] \leq O(n^{-1}).$$

This establishes the result with $q \geq 1.76\Delta$ for triangle-free graphs of degree $\Delta \geq C \log n$.

See [HV05] for the missing details of the above argument.

6.3.2 Further Reading

Dyer and Frieze [DF03] were the first to obtain $\tau_{\text{mix}} = O(n \log n)$ for $q \geq \alpha\Delta$, with $\alpha \approx 1.76$, but using a more complicated argument than the one sketched above. (Rather than assuming Y_0 is stationary, they instead argued that, after sufficiently many steps T, Y_T satisfies a similar local uniformity condition.) In [M04], Molloy improves the $q \geq \alpha\Delta$ bound from [DF03,HV05] to $q \geq \beta\Delta$, where $\beta \approx 1.49$, for graphs meeting the above requirements (i.e., triangle-free and maximum degree $\Delta = \Omega(\log n)$). In [HV03], Hayes and Vigoda use a very delicate *non-Markovian* coupling to demonstrate $O(n \log n)$ mixing time for $q \geq (1 + \epsilon)\Delta$, but the $\Delta = \Omega(\log n)$ requirement remains, and moreover the associated constant in the O-expression depends on ϵ. [DFHV04] demonstrates that the $q \geq \alpha\Delta$ result of [DF03] holds for constant-degree graphs, provided $\Delta = \Omega(1)$ is sufficiently large. It is still open whether the stronger results also apply for constant-degree graphs. Random graphs with constant (average) degree are discussed in [DFFV06], where it is shown that, with high probability over the choice of the graph, $O(n \log n)$ mixing time holds for q down to about $o(\log \log n)$, which is much smaller than the maximum degree (which is of order $\frac{\log n}{\log \log n}$). Finally, while a graduate student at Berkeley, Eric Vigoda [Vi99] was the first to show that a (slightly more complicated) local Markov chain on colorings has mixing time $O(n \log n)$ for q below 2Δ; specifically, his result holds for $q \geq \frac{11}{6}\Delta$. This remains the best known result for a local Markov chain without any additional restrictions on the graph.

References

[BD97] R. Bubley and M.E. Dyer, "Path coupling: a technique for proving rapid mixing in Markov chains," *Proceedings of the 38th IEEE Symposium on Foundations of Computer Science (FOCS)*, 1997, pp. 223–231.

[DFFV06] M. Dyer, A. Flaxman, A. Frieze and E. Vigoda, "Randomly coloring sparse random graphs with fewer colors than the maximum degree," *Random Structures and Algorithms* **29** (2006), pp. 450–465.

[DF03] M. Dyer and A. Frieze, "Randomly coloring graphs with lower bounds on girth and maximum degree," *Random Structures and Algorithms* 23, 2003, pp. 167–179.

[DFHV04] M. Dyer, A. Frieze, T.P. Hayes, E. Vigoda, "Randomly Coloring Constant Degree Graphs," *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, 2004 pp. 582–589

[HV03] T.P. Hayes and E. Vigoda, "A Non-Markovian Coupling for Randomly Sampling Colorings," *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science*, 2003, pp. 618–627

[HV05] T.P. Hayes and E. Vigoda, "Coupling with the stationary distribution and improved sampling for colorings and independent sets," *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, 2005, pp. 971–979

[Je95] M.R. Jerrum, "A very simple algorithm for estimating the number of k-colourings of a low-degree graph," *Random Structures and Algorithms* **7** (1995), pp. 157–165.

[MSW06] F. Martinelli, A. Sinclair and D. Weitz, "Fast mixing for independent sets, colorings and other models on trees," *Random Structures & Algorithms*, 2006, to appear imminently.

[M04] M. Molloy, "The Glauber Dynamics on Colorings of a Graph with High Girth and Maximum Degree," *SIAM Journal on Computing* 33, 2004, pp. 721–737

[SS97] J. Salas and A.D. Sokal, "Absence of Phase Transition for Antiferromagnetic Potts Models via the Dobrushin Uniqueness Theorem," *Journal of Statistical Physics* **98** (2000), p. 551.

[Vi99] E. Vigoda, "Improved bounds for sampling colorings," *Proceedings of the 40th IEEE Symposium on Foundations of Computer Science*, 1999, pp. 51–59.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 7: September 29

Lecturer: Alistair Sinclair *Scribes: Alistair Sinclair*

In today's lecture we will continue the discussion on path coupling and see how it can be applied to bound the mixing time of a Markov chain defined on the space of all linear extensions of a partial order. Towards the end of the lecture we will introduce yet another type of coupling called monotone coupling.

7.1 Mixing time using path coupling

Suppose we have a pre-metric d on the state space Ω of an ergodic Markov chain. If we can define a coupling $(X, Y) \mapsto (X', Y')$ for pairs of adjacent (in the pre-metric) states (X, Y) such that

$$\mathbf{E}[d(X', Y')|X, Y] \leq (1 - \alpha)d(X, Y) \tag{7.1}$$

for some $0 \leq \alpha \leq 1$, then this coupling can be extended to all pairs (X, Y) which also satisfy (7.1). Moreover, if $\alpha > 0$ and d is integer-valued then we have $\tau_{\text{mix}} = \mathrm{O}(\alpha^{-1} \log D)$, where $D = \max_{x,y} d(x, y)$ is the maximum distance between any pair of states under the metric extension of d. The proof of this follows along the same lines as the final step in our analysis of path coupling for the colorings Markov chain in the previous lecture.

Now consider the case in which (7.1) holds with $\alpha = 0$. Then one cannot expect the above result to be true since the distance does not decay geometrically as before. Still one can prove by standard martingale arguments (**exercise!**) that, when d is integral,

$$\tau_{\text{mix}} = \mathrm{O}(\beta^{-1} D^2),$$

where $\beta = \min_{X,Y \in \Omega} \mathbf{E}\Big[\big(d(X', Y') - d(X, Y)\big)^2\Big]$ is the variance of the change in d. (Note that a crude bound on the above would be $\beta = \min_{X,Y \in \Omega} \mathbf{P}\big(|d(X', Y') - d(X, Y)| \geq 1\big)$.)

Caution: In the definition of β, it is **not** sufficient to take the minimum over *adjacent* pairs (X, Y) (**exercise**).

7.2 Linear Extensions of a Partial Order

In this section we illustrate path coupling in the context of sampling a linear extension of a partial order uniformly at random.

Input: A partial order[1] $\preceq$ on $V = \{1, 2, \ldots, n\}$.

[1] A *partial order* $(V, \preceq)$ is a binary relation $\preceq$ over a set V which is reflexive, antisymmetric, and transitive, *i.e.*, for all a, b, and $c \in V$, we have, (i) $a \preceq a$, (ii) $a \preceq b$ and $b \preceq a \implies a = b$, (iii) $a \preceq b$ and $b \preceq c \implies a \preceq c$.

A *linear extension* of $\preceq$ is a total order $\sqsubseteq$ on V which respects $\preceq$, *i.e.* for all $x, y \in V$ $x \preceq y$ implies $x \sqsubseteq y$. Hence a linear extension of $\preceq$ can be written as a permutation $\sigma = (\sigma(1), \sigma(2), \ldots, \sigma(n))$ of $\{1, 2, \ldots, n\}$ such that $\sigma(i) \leq \sigma(j)$ if $v_i \preceq v_j$.

Note that, given a partial order $\preceq$ on $V = \{1, 2, \ldots, n\}$, one can easily construct a linear extension of $\preceq$ (**exercise**). For example, in figure 7.1, the right hand picture represents a linear extension of the partial order shown on the left.

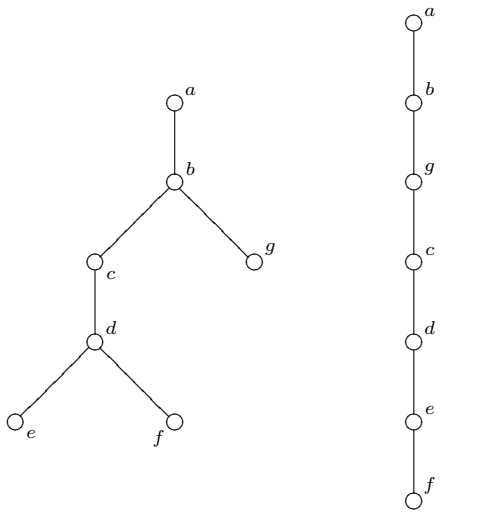


Figure 7.1: A partial order and one of its linear extensions

Goal: Sample a linear extension of $\preceq$ uniformly at random.

Let $\Omega = \Omega(\preceq)$ denote the set of all linear extensions of $\preceq$. Being able to sample from Ω has a variety of applications in combinatorics, near-optimal sorting and decision theory. Brightwell and Winkler [BW91] showed that counting linear extensions is $\#P$-complete. But if we have an efficient algorithm for sampling (almost) uniformly from Ω, then that can be used recursively, as described in lecture note 1, to get an efficient approximation algorithm for $|\Omega|$.

As usual, we propose to sample from Ω by constructing an ergodic Markov Chain on state space Ω, whose stationary distribution is uniform. Define a Markov Chain on Ω as follows.

Markov Chain:

1. With probability 1/2 do nothing.
2. Else pick a position $p \in \{1, 2, \ldots, n-1\}$ uniformly at random.
3. Exchange elements at position p and $p+1$ if "legal", *i.e.,* if the resulting total order is a linear extension of $\preceq$.

It is easy to verify that this Markov Chain is symmetric and aperiodic. Irreducibility of the chain follows from the following exercise.

Exercise: Prove that it is always possible to reach one linear extension from another linear extension using at most $\binom{n}{2}$ "legal" exchanges of consecutive elements.

Hence the above chain is ergodic and converges to the uniform distribution π on Ω. To apply path coupling, we need first to define a pre-metric on the state space Ω.

Pre-metric:
Two states X and Y in Ω are adjacent iff they differ at exactly two positions, say, i and j, $1 \leq i < j \leq n$. We denote this by $X = Y \circ (i, j)$. In this case, the distance $d(X, Y)$ between X and Y is defined to be $j - i$ (see figure 7.2).

Exercise: Check that the above is indeed a pre-metric.

This pre-metric extends to a metric on Ω, denoted again by d. Note that this metric may be rather complicated to describe, but the power of path coupling lies in the fact that we never need to do so. Next we define the coupling for adjacent states.

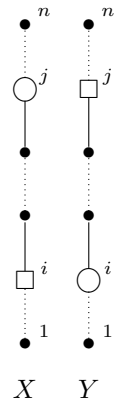


Figure 7.2: Two adjacent states

7.2.1 Coupling for adjacent pairs

Let (X, Y) be a pair of adjacent states. Let i and j be the positions where they differ, $1 \leq i < j \leq n$. Then the transition $(X, Y) \mapsto (X', Y')$ is defined as follows.

Case 1: If $j \neq i + 1$,

- With probability $\frac{1}{2}$ do nothing in both.
- Else choose the same p from $\{1, 2, \ldots, n-1\}$ uniformly at random in both and exchange elements at position p and $p + 1$ if "legal".

Case 2: If $j = i + 1$,

- With probability $\frac{1}{2(n-1)}$ do nothing in X and pick $p = i$ in Y and exchange the elements at i and j in Y. (Note that the elements at i and j are incomparable in $\preceq$ and hence this exchange is always legal.)
- With probability $\frac{1}{2(n-1)}$ pick $p = i$ in X and make a similar exchange and do nothing in Y .
- With probability $\frac{n-2}{2(n-1)}$ do nothing in both.
- Else choose same p from $\{1, 2, \ldots, n-1\} \setminus \{i\}$ uniformly at random in both and exchange elements at position p and $p + 1$ if "legal".

(Note that X' and Y' may no longer be adjacent after a move under this coupling.)

7.2.2 Analysis

To analyze the Markov Chain we need to consider three cases. Note that in any coupled move there is at most one position p that is chosen by either or both of X, Y; we give a case analysis according to the value of p.

Case I: $p \notin \{i - 1, i, j - 1, j\}$.
In this case the exchanges in X and Y are either both legal or both not legal. Thus X' and Y' differ exactly at i and j and $d(X', Y') = j - i = d(X, Y)$.

Case II: $p = i - 1$ or $p = j$.

These two cases are symmetric, so we consider only the case $p = i - 1$. If the exchanges are both legal in X and Y, then we have $d(X', Y') = j - (i - 1) = d(X, Y) + 1$. If exactly one of them is legal (say in X) then $Y' = Y = X \circ (i, j) = X' \circ (i - 1, i) \circ (i, j)$ and $d(X', Y') = (j - i) + 1 = d(X, Y) + 1$. If the exchanges are not legal in both X and Y then of course $d(X', Y') = d(X, Y)$. Hence d increases by at most 1 in this case. The probability of this case is $2 \times \frac{1}{2(n-1)} = \frac{1}{n-1}$.

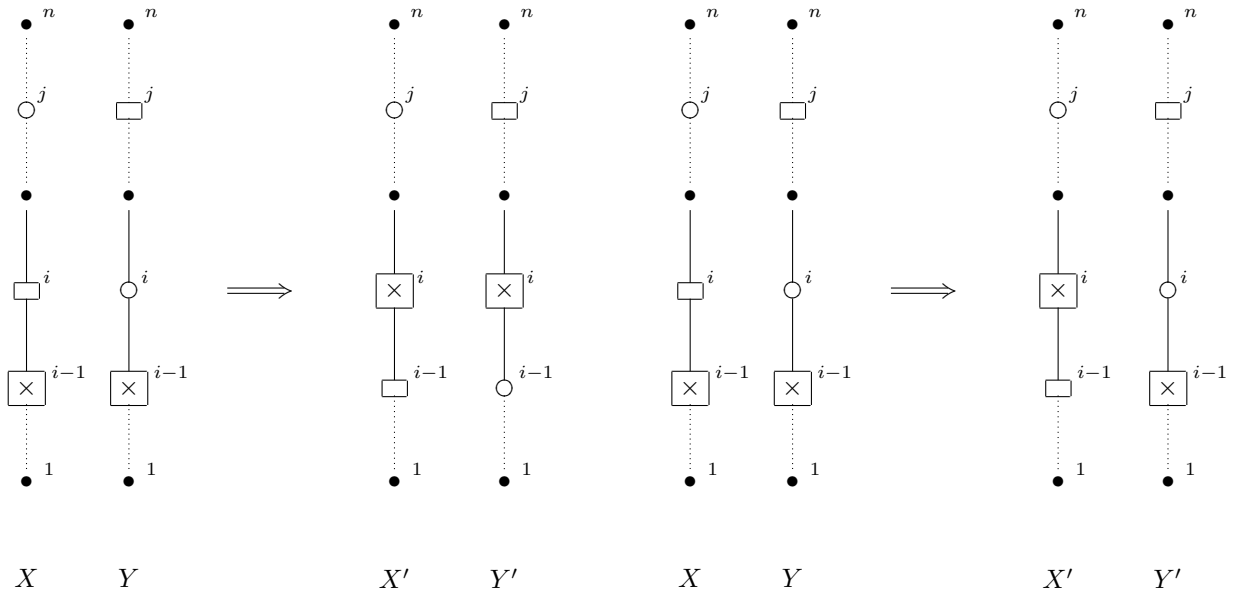


Figure 7.3: Graphical illustration of case II. *Left:* both exchanges are legal. *Right:* only exchange in X is legal.

Case III: $p = i$ or $p = j - 1$.

There are two subcases, depending on the value of $j - i$.

- $j - i = 1$.
  Notice that in this case we make an exchange between the elements at position i and $j = i + 1$ in exactly one of X and Y keeping the other undisturbed and the exchange is always legal. So, $d(X', Y') = 0 = d(X, Y) - 1$. The probability of this case is $\frac{2}{2(n-1)} = \frac{1}{n-1}$.
- $j - i > 1$.
  Again, by symmetry we need only consider the case $p = i$. But the crucial observation is that the elements at position i, $i + 1$ and j are incomparable in $\preceq$. Hence the exchanges in X and Y are both legal and $d(X', Y') = j - (i + 1) = d(X, Y) - 1$. The probability of this case is $2 \times \frac{1}{2(n-1)} = \frac{1}{n-1}$.

Hence $d(X', Y') = d(X, Y) + 1$ with probability *at most* $1/(n - 1)$, and $d(X', Y') = d(X, Y) - 1$ with probability exactly $1/(n - 1)$. Otherwise $d(X', Y') = d(X, Y)$. Therefore we have

$$\mathbf{E}\left[d(X', Y') | X, Y\right] \leq d(X, Y).$$

Applying the path coupling lemma with $\alpha = 0$, we have $\tau_{\text{mix}} = \mathrm{O}(\beta^{-1} D^2)$ where $D = \max_{X,Y \in \Omega} d(X, Y) \leq \binom{n}{2}$ and $\beta = \min_{X,Y \in \Omega} \mathbf{P}(|d(X', Y') - d(X, Y)| \geq 1) \geq \frac{c}{n}$ for some constant $c > 0$ (**exercise**). Therefore,

$$\tau_{\text{mix}} = \mathrm{O}(n^5).$$

Bubley and Dyer [BD99] showed that the convergence can be accelerated by picking position p according to some other distribution instead of choosing it uniformly at random. We will discuss this now.

Modified Markov Chain

1. With probability 1/2 do nothing.
2. Else pick a position $p \in \{1, 2, \ldots, n-1\}$ with probability $\frac{Q(p)}{Z}$ where $Z = \sum_{i=1}^{n-1} Q(i)$.
3. Exchange elements at position p and $p+1$ if "legal".

Note that this Markov chain is still symmetric and aperiodic. If $Q(p) > 0$ for all $1 \leq p \leq n-1$ then it is also irreducible and hence ergodic with uniform stationary distribution. Note that the same coupling stated before can be defined for this Markov chain except only that the position p is now chosen with probability $Q(p)/Z$. The same analysis holds here too and we have that $d(X', Y') = d(X, Y) + 1$ with probability *at most* $[Q(i-1) + Q(j)]/2Z$ and $d(X', Y') = d(X, Y) - 1$ with probability $[Q(i) + Q(j-1)]/2Z$. Otherwise $d(X', Y') = d(X, Y)$. Therefore,

$$\mathbf{E}\left[d(X', Y') - d(X, Y) | X, Y\right] \leq \frac{1}{2Z}\left[Q(i-1) + Q(j) - Q(i) - Q(j-1)\right].$$

We want the R.H.S. to be $\leq -\alpha \cdot d(X, Y) = -\alpha(j-i)$. Observe that the R.H.S. is the difference between the discrete derivatives of the function Q at j and at i, which will be linear if we take $Q(p)$ to be a quadratic polynomial, say $Q(p) = ap^2 + bp + c$. We need to choose the parameters a, b and c in such a way that the value of α is maximized. The optimal choice of Q is given by $Q(p) = p(n-p)$ (**Exercise:** check this!). This gives $Z = \sum_p Q(p) = (n^3 - n)/6$ and $\alpha = 1/Z \geq n^3/6$.

Plugging in, we get the expected change in distance

$$\mathbf{E}\left[d(X', Y') - d(X, Y) | X, Y\right] \leq -\frac{6}{n^3} d(X, Y).$$

Hence, $\tau_{\text{mix}} = \mathrm{O}(\alpha^{-1} \log D) = \mathrm{O}(n^3 \log n)$.

7.2.3 Remarks

1. Wilson [W04] improved the upper bound on the mixing time for the original Markov chain to $\mathrm{O}(n^3 \log n)$ and showed that this is tight. Interestingly, this is the correct order for the mixing time even in the case when $\preceq$ is empty (so that Ω includes all permutations of the elements); in this case the Markov chain is equivalent to shuffling cards by random adjacent transpositions.
2. It remains open whether the mixing time of the "improved" Markov chain can be strictly better than the original.

7.3 Monotone Coupling

Let P be the transition matrix of an ergodic Markov chain on state space Ω, and $\mathcal{F}$ be a probability distribution on functions $f : \Omega \to \Omega$ such that $\mathbf{P}[f(x) = y] = P(x, y)$ for all x, y. We say that the random function f (or, more correctly, the distribution $\mathcal{F}$) is *consistent* with the Markov chain. Note that $\mathcal{F}$ defines a *coupling* for the Markov chain, namely $(X, Y) \mapsto (f(X), f(Y))$. We call such a coupling a *complete coupling*,

because it defines a move not only for each pair (X, Y) but for all states simultaneously. Note also that $\mathcal{F}$ actually specifies the Markov chain, via the relation

$$\mathbf{P}_{\mathcal{F}}(f(x) = y) = P(x, y) \quad \forall x, y \in \Omega.$$

It is easy to check that the couplings we defined for random walk on the hypercube and for graph coloring are actually complete couplings. [**Exercise:** Which other couplings in the course so far have been complete couplings?]

Definition 7.1 *Suppose the states in Ω have a partial order $\preceq$. A complete coupling is said to be* monotone *(w.r.t. $\preceq$) if $x \preceq y \implies f(x) \preceq f(y)$ with $\mathcal{F}$-probability* 1.

Let us now describe why monotone (complete) couplings are useful. Assume that $(\Omega, \preceq)$ has unique minimal and maximal elements, denoted by $\perp$ and $\top$ respectively. Then the following claim ensures that it is enough to consider these extremal states for analyzing the coupling time.

Claim: The coupling time $T_{X,Y}$ for any any pair (X, Y) is stochastically dominated by $T_{\perp,\top}$.

Proof: Let $F_t = f_t \circ f_{t-1} \circ \cdots \circ f_1$, where the f_i are independent samples from $\mathcal{F}$. After time t, (X, Y) moves to $(F_t(X), F_t(Y))$ and by monotonicity, $F_t(\perp) \preceq F_t(X), F_t(Y) \preceq F_t(\top)$. So, $F_t(\perp) = F_t(\top) \Rightarrow F_t(X) = F_t(Y)$.

This means that, to analyze the coupling time for the chain, it is enough to bound the coupling time for the two states $(\perp, \top)$. Moreover, even in cases where we cannot get a good analytical bound on this coupling time, we can perform numerical experiments to get at least a good statistical estimate of the coupling time just by observing how long it takes for $(\perp, \top)$ to meet.

We now give an important example of a monotone coupling.

7.3.1 Example: Ising model

Consider the "heat-bath" Markov chain for the Ising model. Let V be the set of vertices and let $\sigma \in \Omega = \{-, +\}^V$ be an Ising configuration. Recall that the Gibbs distribution for the Ising model is given by

$$\begin{aligned} \pi(\sigma) &\propto \exp\{\beta(a(\sigma) - d(\sigma))\} \\ &\propto \exp\{2\beta a(\sigma)\} = \lambda^{a(\sigma)}, \end{aligned}$$

where $a(\sigma)$ = number of pairs of neighboring vertices whose spins agree in σ , $d(\sigma)$ = number of pairs of neighboring vertices whose spins disagree in σ, $\lambda = \exp(2\beta) \geq 1$ and β is inverse temperature. The heat-bath Markov chain makes transitions from any state $\sigma \in \Omega$ as follows:

- Pick a vertex v uniformly at random.
- Replace the spin σ_v at v by a random spin chosen from the distribution π conditioned on the spins σ_u of the vertices $u \in \text{Nbd}(v)$. Specifically, set σ_v to '+' with probability $p_v^+(\sigma) = \lambda^{n_v^+}/(\lambda^{n_v^+} + \lambda^{n_v^-})$, where n_v^+ and n_v^- are the numbers of neighbors of v with spin '+' and '-' respectively; set σ_v to '$-$' with probability $p_v^-(\sigma) = 1 - p_v^+(\sigma)$.

Let us now define a partial order $\leq$ on Ω as follows:

For $\sigma, \tau \in \Omega$, $\sigma \preceq \tau$ iff $\sigma_v \leq \tau_v$ for all vertices $v \in V$. (Convention: $- \leq +$). Obviously, $\perp$ and $\top$ will be the configurations with all '$-$'s and all '+'s respectively.

We describe a complete coupling as follows, where σ is the current state:

- Pick vertex v uniformly at random.
- Pick $r \in [0, 1]$ uniformly at random.
- If $r \leq p_v^+(\sigma)$ then set the spin σ_v to '+', else set it to '$-$'.

Exercise: Check that this coupling preserves the partial order, and hence is monotone.

References

[BW91] G. Brightwell and P. Winkler, "Counting linear extensions." *Order*, 8(3):225242, 1991.

[BD99] R. Bubley and M E. Dyer, "Faster random generation of linear extensions", *Discrete Mathematics*, 1999, vol. 201, pp. 81–88.

[W04] D.B. Wilson, "Mixing times of lozenge tiling and card shuffling Markov chains," *The Annals of Applied Probability*, 14(1):274–325, 2004.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 8: October 1

Lecturer: Prof. Alistair Sinclair *Scribes: Alistair Sinclair*

8.1 Lattice Tilings

Consider a simply connected region S of the two-dimensional Cartesian lattice (e.g. a $2n \times 2n$ chessboard). A *tiling* of S is a covering of all its squares by non-overlapping dominoes, each of which occupies two adjacent squares of the lattice. We want to generate u.a.r. a tiling of S. Such problems arise in statistical physics, where the tilings correspond to configurations of a so-called *dimer system* on S. Various physical properties of the system are related to the expected value, over the uniform distribution, of some function defined over configurations, such as the number of horizontal dominoes or the correlation between the orientation of dominoes at two given squares. The Markov chain most commonly used to generate u.a.r. a tiling of S picks a 2×2 square in S u.a.r. and rotates it if it contains exactly two dominoes. (See, e.g., Figure 8.1.) Thurston [Th90] proved that the set of all tilings is connected by such moves (and we shall see this below).

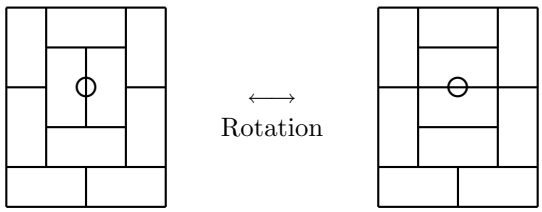


Figure 8.1: Domino Tilings

8.2 Lozenge Tilings

For the purpose of this lecture, we will analyze the mixing time of an analogous Markov chain for *lozenge* tilings on the triangular lattice as shown in Figure 8.2. (A *lozenge* is the analogue of a domino in the Cartesian lattice). This analysis is due to Luby, Randall and Sinclair [LRS01], and extends with a little more work to the Cartesian lattice. We use the following Markov chain to sample lozenge tilings of a simply connected region S of the triangular lattice.

- Pick a point in the interior of region S u.a.r.
- If the hexagon surrounding the chosen point contains exactly three lozenges, rotate it by 60° (as shown in Figure 8.2), else do nothing.

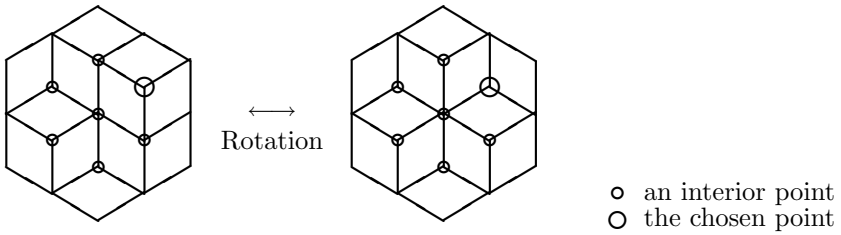


Figure 8.2: Lozenge Tilings

It turns out that it is easier to analyze the above process when viewed over so-called *routings*, which are defined below.

8.2.1 Correspondence With Routings

The lozenge tilings of a region of the triangular lattice are in 1-1 correspondence with *routings* on an associated Cartesian lattice. Given a simply connected region S of the triangular lattice, we can define a corresponding region S' of the Cartesian lattice as follows. There is a vertex in region S' corresponding to the midpoint of a vertical edge in region S. Two vertices in region S' are connected iff the corresponding points in region S lie on adjacent triangles. The vertices in S' that correspond to the vertical edges on the left (or right) sided boundary of S are called sources (or sinks). If a lozenge tiling exists for S, then the number of sources in S' is equal to the number of sinks. For example, in Figure 8.3, there are two sources s_1, s_2 and two sinks t_1, t_2. A *routing* in S' is a set of non-intersecting shortest paths from each source to the corresponding sink. Note that every routing contains the same number of points of S'; indeed, the path for each source-sink pair contains the same number of points in every routing. We call the interior points (i.e., those that are not sources or sinks) the *points along* the routing.

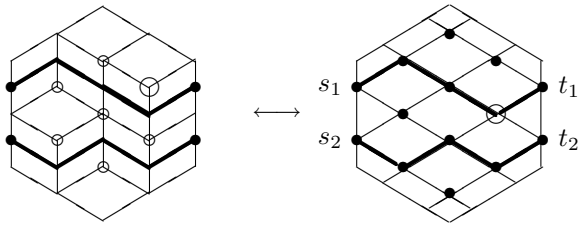


Figure 8.3: Lozenge Tiling $\equiv$ Routing

What does the above Markov chain look like in the routings world? Well, rotation of a hexagon containing three lozenges in a tiling corresponds to flipping a "valley" to a "peak" or vice-versa in the corresponding routing (see Figure 8.3). This leads us to the following description of a Markov chain on routings:

- Pick a point $p \in \{1, \ldots, n\}$ u.a.r. along the routing, and a direction $d \in \{\uparrow, \downarrow\}$ u.a.r.
- Move point p in direction d if possible.

Note that if the chosen point p is not a peak or a valley, it will not move in either direction d. Moreover, not all peaks and valleys can move: some are "blocked" because of the constraint that the paths be non-intersecting.

The above Markov chain is aperiodic since there is at most one direction in which a point along a routing can be moved, and hence choosing direction d u.a.r. implies a self-loop probability of at least $\frac{1}{2}$ at every state. The Markov chain is irreducible since it is possible to get from any routing to the maximum routing (i.e., the routing in which every path is "as high as possible") by iteratively converting valleys into peaks by rotation, processing the paths top-to-bottom. Similarly, it is possible to go from the maximum routing to any other routing as the Markov chain is symmetric. The Markov chain thus converges to the uniform distribution over the set of all routings (or equivalently tilings) Ω.

We will use a coupling argument to bound the mixing time of this Markov chain. We will exploit the fact that the chain admits a *complete* coupling that is *monotone* w.r.t. a natural partial order on Ω, as defined in the previous lecture.

The partial order is the following.

Definition 8.1 *For two routings X and Y, we say that $X \preceq Y$ iff X does not lie above Y at any corresponding point along their paths.*

Thus the unique maximum routing $\top$ is the one in which all paths are "as high as possible", and the minimum routing $\bot$ is the one in which all paths are "as low as possible."

The complete coupling is defined by the following distribution over functions $f : \Omega \to \Omega$. The function f is specified by choosing a point p and a direction d, both u.a.r. Then, for any routing $X \in \Omega$, the image $f(X)$ is obtained by moving point p in direction d if possible. It is clear that this distribution is consistent with the Markov chain, so we have a valid complete coupling.

It is easy to verify [**exercise!**] that this coupling is monotone w.r.t. $\preceq$. As discussed in the previous lecture, this implies that the coupling time T_{xy} for any pair of initial states x, y is bounded by $T_{\top\bot}$, the coupling time for the two extremal states $\top$ and $\bot$. This gives us an *algorithmic* method for estimating the mixing time, even if we have no analytical bound: namely, simulate the coupled evolution starting from the states $\top$ and $\bot$ and observe the time for them to meet.

However, we now go further and show how to analyze the coupling time rigorously. Note that monotonicity also simplifies this analysis: first, we need only consider initial states $X_0 = \top$ and $Y_0 = \bot$; second, we can assume that at all future times the two copies satisfy $X_t \succeq Y_t$.

8.2.2 Coupling Analysis of Single Path Routings

As described earlier, the coupling between two routings X and Y is the following. We choose the same point p and direction d in both X and Y. Then we make the corresponding move in both (if possible). For the purpose of simplicity, we will initially restrict attention to routings in which there is only one path: this is simpler since no moves are ever blocked by the constraint that the paths be non-intersecting. By monotonicity, we may assume that one path always lies above the other; see Figure 8.4. We define the *distance* between the two paths as the area enclosed between them.

Now consider the possible moves under the coupling. We can mark some points on the two paths as either good (G) or bad (B). A point is marked G if by selecting that point and making the corresponding move, we decrease the area between the two paths (by one unit). It is marked B if the corresponding move increases the area (by one unit). The following claim is not hard to prove and is left as an exercise.

Claim 8.2 $\#G \geq \#B$.

This claim implies that the expected change in distance is non-positive. Also, it is easy to see that when the

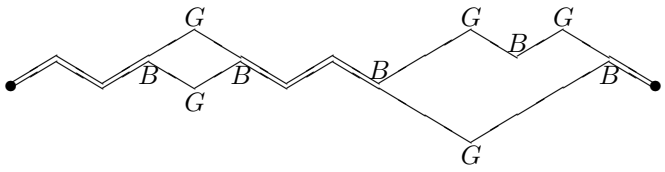


Figure 8.4: Distance = area between the two paths = 6

paths differ there is always at least one good move. Hence, the probability of a change in the distance is at least $\frac{1}{2N}$, where N is the number of points along a routing. Taking as a natural measure of problem size the area n of S, we have clearly that $N \leq n$, and hence the coupling time is stochastically dominated by the hitting time to 0 of a random walk on the integers $[0, n]$ which has zero drift and probability at least $\frac{1}{2n}$ of making a move. Thus the mixing time of our process is $O(n^3)$.

8.2.3 Coupling Analysis of Multi-Path Routings

Let us now extend the above analysis to routings with multiple paths. The difficulty here lies in taking care of blocked moves: because some moves are blocked, we can no longer argue that $\#G \geq \#B$. We shall get around this with a trick: we shall modify the Markov chain so that no moves are blocked! I.e., whenever the chosen point p is a peak or a valley, we will always make a move. To describe this, note that any peak (valley) defines a *tower* of some height $h \geq 1$, which is the maximal contiguous sequence of peaks (valleys) above (below) it. The tower of an unblocked peak/valley has height 1; if it is blocked the height will be greater than 1. Note that a tower of any height can be rotated in an analogous fashion to the simple rotations (of towers of height 1) considered earlier; see Figure 8.5.

Here is the modified Markov chain, with tower moves:

- Pick a point p and direction d as before.
- Rotate the tower at p in direction d if possible (i.e., if p is a peak and $d =\downarrow$, or p is a valley and $d =\uparrow$) with probability $1/h$, where h is the height of the tower.

Note that we rotate a tower only with probability inversely proportional to its height. This is because rotating a tower of height h changes the distance by $\pm h$.

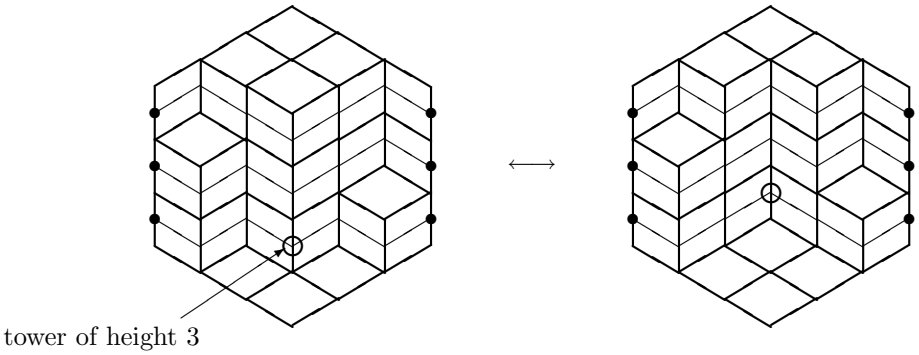


Figure 8.5: Tower Rotations

It is easy to check that the same complete coupling f as before (modified to allow tower moves) is still monotone w.r.t. $\preceq$. Hence, we can work with the coupling time for the min ($\perp$) and max ($\top$) routings, and can assume that the two routings at any future time lie one above the other. We define the *distance* between any two routings as the sum of the areas between their corresponding paths; it is not hard to check [**exercise**] that the maximum possible distance (i.e., that between $\perp$ and $\top$) is $O(n^{3/2})$, where again n is the area of S.

As before, identify a peak or valley as good (G) or bad (B) according to whether it decreases or increases the distance when it moves. Let us mark all bad nodes with a tower of height h above/below them as B_h, and all good nodes with a tower of height h above/below them as G_h. We can now analyze the expected change in distance as follows:

$$\begin{aligned} E[\Delta(Distance)] &= \sum_h \frac{1}{2Nh}(\#B_h - \#G_h)h \\ &= (\frac{1}{2N})\sum_h (\#B_h - \#G_h) \\ &\leq 0, \end{aligned}$$

where $N \leq n$ is the total number of points along a routing. In the first line here, we are using the facts that a tower of height h is rotated with probability $1/2Nh$, and that the resulting change in area is $\pm h$. In the last line we are using the combinatorial fact seen earlier that the number of good moves is at least as large as the number of bad moves; this property is inherited from the single-path case as no moves are blocked.

Thus, this process is stochastically dominated by the hitting time to 0 of a random walk on the integers $[0, n^{1.5}]$ which has zero drift, and whose variance in one step is at least $\frac{1}{2n}$.[1] Hence the mixing time is $O(n^4)$.

Note: Wilson [W04] subsequently obtained a tight bound of $O(n^3 \log n)$ for the mixing time on a hexagonal region, using a more technical argument.

References

[LRS01] M. Luby and D. Randall and A. Sinclair, "Markov chain algorithms for planar lattice structures," *SIAM Journal on Computing* **31** (2001), pp. 167–192. Preliminary version in *ACM STOC*, 1995.

[Th90] W. Thurston, "Conway's tiling groups," *American Mathematical Monthly* **97** (1990), pp. 757–773.

[W04] D.B. Wilson, "Mixing times of lozenge tiling and card shuffling Markov chains," *The Annals of Applied Probability*, 14(1):274–325, 2004.

[1] Note that the process can take jumps of arbitrary size h, not just 1 as in simple random walk. However, standard martingale techniques still imply that the expected hitting time to 0 for such a process on $[0, D]$ with zero drift is D^2/V, where V is a lower bound on the variance of a move (i.e., the expected squared distance moved in one step). In our case, we can take $V \geq \frac{1}{2Nh} \times h^2 = \frac{h}{2N} \geq \frac{1}{2n}$.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 9: October 6

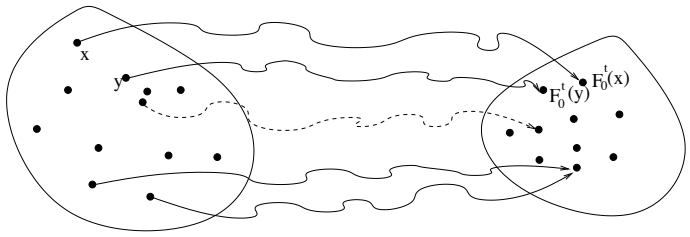
Lecturer: Prof. Alistair Sinclair *Scribes: Alistair Sinclair*

9.1 Random function representation

Let P be an ergodic Markov chain on state space Ω. Recall the "random function" representation of P, namely as a probability distribution over functions $f : \Omega \to \Omega$ such that

$$\Pr[f(x) = y] = P(x, y) \qquad \forall x, y, \tag{9.1}$$

where the probability is over the random choice of f.



Let us fix a labeling of steps as "points in time" $(-\infty, \ldots, -1, 0, 1, \ldots, \infty)$ with arbitrary "current time" 0. Define F_i^j as the $(j-i)$-step evolution of the Markov chain M from time i to j, and decompose it into $(j-i)$ independent applications of a random function f consistent with M; that is

$$F_i^j = f_{j-1} \circ f_{j-2} \circ \cdots \circ f_{i+1} \circ f_i \tag{9.2}$$

where f_t is the random function chosen at time t. Two cases will play a special role for us in this lecture: firstly, F_0^t is the standard "forwards" simulation of M for t steps (starting at time 0). Secondly, F_{-t}^0 is the t-step evolution of M from time $-t$ to time 0 (i.e., "from the past")

9.2 Coupling from the past

Define the "coalescence time" T as

$$T = \min\{t : F_0^t \text{ is a constant function}\}. \tag{9.3}$$

Note that F_0^t being constant means that, after t steps, all paths in the Markov chain have reached the same state (for all possible initial states).

In view of what we know about coupling, we might be tempted to conjecture that the distribution of $F_0^T(x)$ (which, by definition, is the same for all x) is the stationary distribution π. However, as we shall see in a

moment, this is spectacularly false. But remarkably, if we try the same trick "from the past", we *do* get the correct distribution.

To state this formally, define

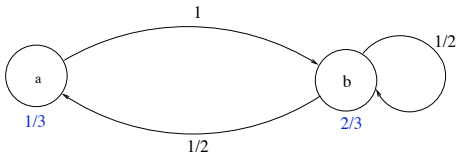
$$T = \min\{t : F^0_{-t} \text{ is a constant function}\}. \tag{9.4}$$

The following theorem, due to Propp and Wilson [PW96], shows that if we take T as a stopping time for the simulation from the past, then the resulting constant value of the function F^0_{-T} will have distribution exactly π:

Theorem 9.1 *Assuming T is finite with probability 1, then the constant value $Z^0_{-\infty} = F^0_{-T}(x)$ has distribution exactly π.*

Before proving the theorem, we will give an example to illustrate the difference between the forward and "from the past" simulations.

Consider the following very simple Markov chain M, whose stationary distribution π is $(1/3, 2/3)$:



In this case, there is a *unique* random function f consistent with M, as follows:

$$f_1(a) = b \qquad f_2(a) = b$$
$$f_1(b) = a \qquad f_2(b) = b$$

$$f = \begin{cases} f_1 \text{ with probability } 1/2 \\ f_2 \text{ with probability } 1/2 \end{cases}$$

Exercise: Verify that f is unique.

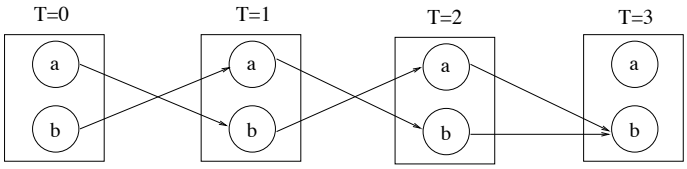


Figure 9.1: One particular forward simulation where $T = 3$. Note that $F^T_0(x) = b$ always.

Figure 9.1 shows a sample forwards simulation with stopping time $T = 3$. You should convince yourself that such a simulation must always stop in state b, i.e., in distribution $(0, 1)$, which is very far from the stationary distribution $(1/3, 2/3)$. Figures 9.2 and 9.3 show two possible simulations from the past, one of which stops in state b and the other in state a.

Exercise: Verify that in the above example, the distribution of F^0_{-T} is indeed π.

Exercise: Show that, although the distributions of F^0_{-T} and F^T_0 (where in each case T is the appropriate stopping time) are quite different, the distributions of the two *stopping times* are the same.

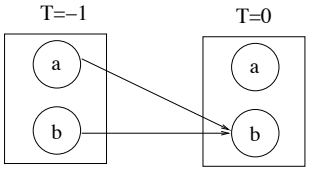


Figure 9.2: Simulation from the past with $T = 1$.

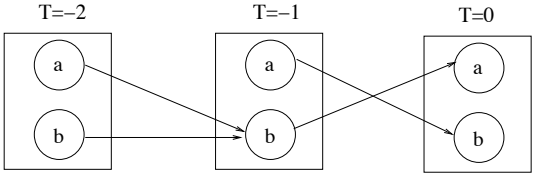


Figure 9.3: Simulation from the past with $T = 2$.

Now we will go back and prove Theorem 9.1.

Proof: Since T is finite with probability 1, $Z^0_{-\infty}$ is well defined with probability 1. Define

$$T' = \min\{t : F^1_{-t} \text{ is constant}\}.$$

Couple the processes for F^0_{-t} and F^1_{-t}, so that they use the same $f_{t'}$ at time t' (hence $T' \leq T$). Let $Z^1_{-\infty} = F^1_{-T'}(x)$ be the constant value of the function $F^1_{-T'}$, and π_0, π_1 be the distributions of $Z^0_{-\infty}$ and $Z^1_{-\infty}$ respectively. Since $Z^0_{-\infty}$, $Z^1_{-\infty}$ have the same distribution (both are just the value of the constant function obtained by coupling from the past up to some fixed time, 0 or 1 respectively), we have $\pi_0 = \pi_1$. Also, it should be clear that $Z^1_{-\infty} = f(Z^0_{-\infty})$, since the former is obtained by extending the simulation for one further step. Hence the common distribution $\pi_0 = \pi_1$ is a fixed point of f, so it must be the (unique) stationary distribution of M. ■

9.3 An exact sampling algorithm

Theorem 9.1 immediately suggests an algorithm for sampling *exactly* from the stationary distribution π. (Note that our standard Markov chain simulation does not quite achieve this: there is always some non-zero variation distance.) Here is the algorithm:

1. Compute F^0_{-1}, F^0_{-2}, ..., F^0_{-t}, ... until F^0_{-t} is constant.
2. Output the constant value of F^0_{-t}.

Unfortunately this algorithm is in general not very efficiently implementable: in order to check if F^0_{-t} is constant, we need to compute $F^0_{-t}(x)$ for every state x. This is hopeless in most of our examples, where the number of states $|\Omega|$ is huge.

However, there is a situation in which we can turn the above into an efficient algorithm: namely, if the random function f is a *monotone* coupling then to test if F^0_{-t} is constant it is enough just to test if $F^0_{-t}(\top) = F^0_{-t}(\bot)$, where $\top$ and $\bot$ are the unique maximum and minimum elements respectively (see Figure 9.4). [**Exercise:** Check carefully that this is equivalent to testing if F^0_{-t} is constant.]

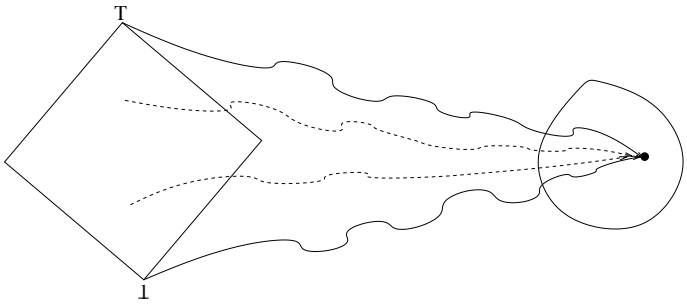


Figure 9.4: For a monotone coupling, top and bottom states converging to the same state after time T implies that all other states converge to the same state by that time.

We can gain a further improvement in running time by noting that, instead of trying $F^0_{-1}, F^0_{-2}, F^0_{-3}, F^0_{-4}, \ldots$ in sequence, it is enough to try the times $F^0_{-1}, F^0_{-2}, F^0_{-4}, F^0_{-8}, \ldots$. This is OK because "overshooting" the stopping time T does not affect the distribution of the final state. (I.e., if F^0_{-t} is constant and has distribution π, then the same is true of $F^0_{-t'}$ for all $t' \geq t$.) And if the true stopping time is T, then the above doubling scheme will stop after at most $2T$ steps. Moreover, the number of steps it will simulate is at most $1 + 2 + 4 + 8 + \cdots + 2T \leq 4T$. Contrast this with the naive single-step scheme which will simulate $1 + 2 + 3 + 4 + \cdots + T = \Omega(T^2)$ steps.

Here, then, is the final algorithm for the monotone case:

$T \leftarrow 1$
repeat
  bottom $\leftarrow \perp$
  top $\leftarrow \top$
  for $t \leftarrow -T$ **to** -1 **do**
    bottom $\leftarrow f_t$(bottom)
    top $\leftarrow f_t$(top)
    $T \leftarrow 2T$
  end
until bottom=top ;
Output: top

Algorithm 1: A "Coupling from the Past" algorithm for exact sampling from π

Note that, in this algorithm, it is important to store and re-use the random choices f_t as you go, rather than to recompute them at each step. [**Why?**]

It is interesting to observe that, in the monotone setting, Coupling from the Past never adds a huge overhead compared to standard forwards simulation of the Markov chain: specifically, one can show that the expected value of the coalescence time T from the past is bounded above by $O(h\tau_{\text{mix}})$, where h is the *height* of the partial order (i.e., the length of a longest chain of comparable elements between $\perp$ and $\top$). [**Exercise**: Prove this!] Thus in order to bound *a priori* the amount of time required to produce a sample, it is enough to bound the mixing time. In many practical situations, however, the actual time until Coupling from the Past produces a sample is much less than the (best known estimate of) the mixing time. Note also that we can use CFTP even when no analytical estimate of the running time is available (but then we have no bound on how long we may have to wait for a result).

Finally, we should inject a caveat here. Coupling from the Past only guarantees a sample from distribution π if we are willing to wait as long as it takes for the procedure to terminate. If, e.g., we truncate (and restart)

the algorithm whenever it runs for too long, we can introduce an arbitrary bias into the distribution. There are so-called "interruptible" versions of CFTP (see, e.g., [Fi98,FMMR00]) which do allow one to terminate without introducing bias, but these are much less elegant and less useful in practice than the basic version above.

9.4 Extensions

The condition of monotonicity is in fact not necessary for an efficient implementation of CFTP; in certain non-monotone settings the algorithm can be made to work efficiently also. Recall that the essential issue is testing efficiently whether the random function F^0_{-t} is constant. Here is a trick, due to Häggstrom and Nelander [HN98] and Huber [Hu98], for doing this in the case of the Hard Core Model (or Independent Sets), one of the most important models in Statistical Physics.

Here we are given a graph $G = (V, E)$ and a parameter $\lambda > 0$, and the set of configurations Ω is the set of all independent sets (of any size) in G. We refer to vertices in the independent set as "occupied", and the remainder as "unoccupied". Thus the constraint is that no two adjacent vertices may be occupied. Each independent set $x \in \Omega$ has weight $w(x) = \lambda^{|x|}$, and as usual our goal is to sample from the distribution $\pi(x) \propto w(x)$.

The heat-bath dynamics for this model makes transitions from configuration x as follows:

- pick a vertex $v \in V$ u.a.r. (and ignore the current state of v)
- with probability $\frac{1}{1+\lambda}$ make v unoccupied; with probability $\frac{\lambda}{1+\lambda}$ make v occupied if possible (i.e., if none of its neighbors is occupied), else make it unoccupied.

Exercise: Verify that this Markov chain is ergodic and reversible w.r.t. the distribution π.

The natural random function (complete coupling) representation of this dynamics is the following. We pick a pair (v, r) u.a.r., where $v \in V$ and $r \in [0, 1]$; then we make the update at vertex v, using r to decide whether to try to occupy v or not (i.e., if $r \leq \frac{1}{1+\lambda}$ make v unoccupied, else make it occupied if possible).

There is no monotone coupling for this dynamics, so the standard approach to CFTP of the previous section does not work. (Of course, we can always apply CFTP to any Markov chain; the problem is that we have no efficient way of telling when F^0_{-t} is constant.) The trick is to consider a modified Markov chain in which each vertex may be either occupied, unoccupied, or in state "?". The idea is that any configuration in this enlarged model represents a *set* of configurations in the original model: a vertex that is (un)occupied in the enlarged model means that that vertex is (un)occupied in every configuration in the set; a vertex that is "?" in the enlarged model means that that vertex may be both occupied and unoccupied in configurations in the set.

Transitions in the enlarged model are made as follows:

- pick a vertex $v \in V$ u.a.r. (and ignore the current state of v)
- with probability $\frac{1}{1+\lambda}$ make v unoccupied; with probability $\frac{\lambda}{1+\lambda}$, make v occupied if all of its neighbors are unoccupied, make it unoccupied if all of its neighbors are occupied, else make it "?".

Thus after each such update the 0, 1, ? configuration describes the set of possible independent sets that the original Markov chain could be in after one of the original heat-bath updates. This dynamics has essentially the same random function representation as above.

Now it is not hard to check **[exercise!]** that if the enlarged chain, starting from the all-? configuration, ever reaches a configuration y in which there are no ?'s, then the original heat-bath dynamics on independent sets, using the same realizations of the random function at each step, maps every initial configuration to the same destination, namely this same y. Hence we can use this as an efficient test for coalescence.

We face the same challenge as with standard CFTP if we want to estimate the running time in advance (and in fact we can't appeal to the previously mentioned result relating the coalescence time to the mixing time, since that required monotonicity). It has been shown [HN99] that the number of "?"'s decays exponentially fast provided $\lambda < \frac{1}{\Delta}$, where Δ is the maximum degree of G. Thus CFTP can be implemented efficiently in this range. In practice, however, for many graphs of interest (such as regular grids) the method works efficiently for much larger values of λ than this theoretical guarantee.

We will have more to say about the Hard Core Model, and the significance of the parameter λ, later in the course.

References

[Fi98] J.A. Fill, "An interruptible algorithm for perfect sampling via Markov chains", *Annals of Applied Probability* **8** (1998), pp. 131–162.

[FMMR00] J.A. Fill, M. Machida, D.J. Murdoch and J.S. Rosenthal, "Extension of Fill's perfect rejection sampling algorithm to general chains", *Random Structures & Algorithms***17** (2000), pp. 290–316.

[HN98] O. Häggstrom and K. Nelander, "Exact sampling from anti-monotone systems", *Statistica Neerlandica* **52** (1998), pp. 360–380.

[HN99] O. Häggstrom and K. Nelander, "On exact simulation of Markov random fields using coupling from the past", *Scandinavian Journal of Statistics* **26** (1999), pp. 395–411.

[Hu98] M. Huber, "Exact sampling and approximate counting techniques", *Proceedings of the 30th STOC*, 1998, pp. 31–40.

[PW96] J. Propp and D. Wilson, "Exact Sampling with Coupled Markov Chains and Applications to Statistical Mechanics", *Random Structure and Algorithms*, 1996, pp. 9:223-252.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 10: October 08

Lecturer: Prof. Alistair Sinclair *Scribes: Jörg Lässig and Dirk Sudholt*

10.1 Multicommodity Flow

In this lecture we consider multicommodity network flows. As usual, we assume an ergodic Markov chain P whose stationary distribution is π. We say that an edge $e = (z, z')$ has *capacity* $C(e) := \pi(z)P(z, z')$. Note that $\pi(z)P(z, z')$ is also called the *ergodic flow* from z to z'. It represents the flow of probability mass along the edge when the Markov chain is at stationarity. For all pairs $(x, y) \in \Omega \times \Omega$ we have a *demand* $D(x, y) := \pi(x)\pi(y)$. A *flow* f is now any scheme that routes $D(x, y)$ units of flow from x to y simultaneously for all pairs x, y. The commodities are disjoint; we can think (say) of routing different types of oil between any pair of nodes. More formally, a flow is a function $f\colon P \to \mathbb{R}^+ \cup \{0\}$ where $P = \bigcup_{xy} P_{xy}$ and P_{xy} denotes the set of all simple paths from x to y s. t.

$$\sum_{p \in P_{xy}} f(p) = D(x, y) \ .$$

Let $f(e) = \sum_{p \ni e} f(p)$ be the total flow along e. The *cost* $\rho(f)$ of a flow f is given by $\rho(f) = \max_e f(e)/C(e)$. The length $\ell(f)$ of f is the length of a longest flow-carrying path, i.e., $\ell(f) := \max_{p\colon f(p)>0} |p|$.

Note that the demands must always be satisfied, and the flow through an edge may exceed its capacity. Equivalently we could have asked for the largest F such that $FD(x, y)$ units are routed between each x, y, with no edge capacity being exceeded. The cost would then be $1/F$.

Our goal in this lecture is to prove the following theorem relating mixing times to flows. Recall from Lecture 2 that a *lazy* version of a Markov chain is obtained by introducing additional self-loop probabilities of $1/2$ at each state.

Theorem 10.1 *For any lazy, ergodic Markov chain P and any flow f, we have*

$$\tau_{\text{mix}} \leq O\left(\rho(f)\ell(f)\ln \pi_{\min}^{-1}\right) \ ,$$

where $\pi_{\min} = \min_{x\in\Omega} \pi(x)$.

The main implication of this theorem is that the mixing time is roughly proportional to the cost $\rho(f)$ of a flow; the factors $\ell(f)$ and $\ln \pi_{\min}^{-1}$ are usually easily shown to be small (i.e., low-degree polynomials in the problem size n). In particular, in most applications the flows that we use will route all flow along *shortest* paths; in this case $\ell(f)$ is bounded by the *diameter* of the Markov chain, which is typically polynomial in n. Also, the size of the state space $|\Omega|$ is typically singly exponential in the measure n of problem size (e.g., for colorings n is the size of the underlying graph); so if π is uniform then $\ln \pi_{\min}^{-1}$ is $O(n)$. In fact, when π is not uniform we can replace $\pi_{\min}$ in the above theorem by $\pi(x)$, where x is the initial state, so provided we start off in a state of (near-)maximum probability this factor will still be small.

Remark: Note that *any* flow f gives an upper bound on the mixing time. There is a similar lower bound on the mixing time in terms of flows; however, to use that we would need to consider all possible flows and show that none of them can be good. For lower bounds it makes more sense to consider the "dual" problem of *sparsest cut*; then *any* bad cut will give us a lower bound on mixing time. We will consider the sparsest cut problem, and its relation to multicommodity flows, later in the course.

We will spend the rest of the lecture proving Theorem 10.1. First, we need some definitions.

Definition 10.2 *Let $\varphi\colon \Omega \to \mathbb{R}$ be any real-valued function. Define the expectation $E_\pi\varphi = \sum_x \pi(x)\varphi(x)$. The* (global) variance *of ϕ is given by*

$$\begin{aligned}
\mathrm{Var}_\pi\varphi &:= \sum_x \pi(x)\cdot(\varphi(x) - E_\pi\varphi)^2 \\
&= \sum_x \pi(x)\varphi(x)^2 - (E_\pi\varphi)^2 \\
&= \sum_x \pi(x)\varphi(x)^2 \sum_y \pi(y) - \sum_x \pi(x)\varphi(x)\sum_y \pi(y)\varphi(y) \\
&= \sum_{xy} \big(\pi(x)\pi(y)\varphi(x)^2 - \pi(x)\pi(y)\varphi(x)\varphi(y)\big) \\
&= \frac{1}{2}\sum_{xy} \pi(x)\pi(y)\cdot(\varphi(x)-\varphi(y))^2 \ .
\end{aligned}$$

By analogy with the last line above, it is natural to define the "local variance" by considering only adjacent pairs x, y as follows.

Definition 10.3 *The* local variance *is defined as*

$$\mathcal{E}_\pi(\varphi,\varphi) = \frac{1}{2}\sum_{xy} \pi(x)P(x,y)\cdot(\varphi(x)-\varphi(y))^2 \ .$$

$\mathcal{E}_\pi(\cdot,\cdot)$ is also known as the "Dirichlet form."

A *Poincaré inequality* bounds the ratio of the local to the global variance for any function φ. This leads to the following definition.

Definition 10.4 *The* Poincaré constant *is defined by*

$$\alpha := \inf_{\varphi \text{ non-constant}} \frac{\mathcal{E}_\pi(\varphi,\varphi)}{\mathrm{Var}_\pi\varphi} \ .$$

It should be intuitively reasonable that the Poincaré constant provides an upper bound on the mixing time, essentially because the local variance $E_\pi(\varphi,\varphi)$ determines the rate at which φ is "averaged" per step of the Markov chain. This intuition is captured in the following general theorem.

Theorem 10.5 *For any lazy ergodic P and any initial state $x \in \Omega$,*

$$\tau_x(\varepsilon) \le \frac{1}{\alpha}\left(2\ln\varepsilon^{-1} + \ln\pi(x)^{-1}\right) \ .$$

The second ingredient in the proof of Theorem 10.1 is a bound on the Poincaré constant in terms of multi-commodity flows.

Theorem 10.6 *For any ergodic P and any flow f for P*

$$\alpha \geq \frac{1}{\rho(f)\ell(f)} .$$

It is immediate that these theorems together imply Theorem 10.1. Plugging the lower bound for α from Theorem 10.6 into Theorem 10.5 and letting $\varepsilon := 1/(2e)$ yields an upper bound $O(\rho(f)\ell(f)\ln\pi(x)^{-1})$ on $\tau_x(1/(2e))$. This is actually a stronger result than Theorem 10.1, which is obtained by taking a worst-case starting point x.

We now proceed to prove the above theorems. The first, Theorem 10.5, is a version of a classical result. We follow the proofs of Jerrum [Je03] and Mihail [Mi89].

Proof of Theorem 10.5: Because P is lazy we have $P = \frac{1}{2}(I + \widehat{P})$, where $\widehat{P}$ is stochastic and has the same stationary distribution π as P. Define a function $[P_\varphi]\colon \Omega \to \mathbb{R}$ by $[P_\varphi](x) := \sum_y P(x,y)\varphi(y)$. This function is called the "one-step averaging of φ" as we take averages over φ-values after one step. Similarly, $[P^t\varphi](x) = \sum_y P^t(x,y)\varphi(y)$ is the t-step averaging of φ. Since P is ergodic, $[P^t\varphi]$ converges to the constant function $E_\pi\varphi$, so

$$\mathrm{Var}[P^t\varphi] \xrightarrow{t\to\infty} 0 .$$

Note also that $E_\pi[P^t\varphi] = E_\pi\varphi$ for all t.

The following lemma is the main content of the proof.

Lemma 10.7 (Main Lemma) *For any $\varphi : \Omega \to \mathbb{R}$*

$$\mathrm{Var}_\pi[P_\varphi] \leq \mathrm{Var}_\pi\varphi - \mathcal{E}_\pi(\varphi,\varphi) .$$

We defer the proof of the Main Lemma for a moment. For now, we state an immediate corollary that establishes a contraction $1-\alpha$ for the variance in each step of the chain.

Corollary 10.8 $\mathrm{Var}_\pi[P^t_\varphi] \leq (1-\alpha)^t \cdot \mathrm{Var}_\pi\varphi$.

Proceeding with the proof of Theorem 10.5, let $A \subseteq \Omega$ be arbitrary. Define

$$\varphi_A(x) := \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{otherwise} . \end{cases}$$

Then $\mathrm{Var}_\pi\varphi_A \leq 1$, so $\mathrm{Var}_\pi\left[P^t_{\varphi_A}\right] \leq (1-\alpha)^t \leq e^{-\alpha t}$. Setting $t = \frac{1}{\alpha}(\ln\pi(x)^{-1} + 2\ln\varepsilon^{-1})$, we have $\mathrm{Var}_\pi[P^t_{\varphi_A}] \leq \varepsilon^2 \cdot \pi(x)$. But on the other hand

$$\begin{aligned} \mathrm{Var}_\pi[P^t_{\varphi_A}] &\geq \pi(x)\left(P^t_{\varphi_A}(x) - E_\pi[P^t_{\varphi_A}]\right)^2 \\ &= \pi(x)\left(P^t_x(A) - \pi(A)\right)^2 , \end{aligned}$$

where the last equality follows from the definition of φ_A and the fact that $E_\pi[Py] = E_\pi y$ for all y. Together, $\left(P^t_x(A) - \pi(A)\right)^2 \leq \varepsilon^2$. Since A is arbitrary, we get $||P^t_x - \pi|| \leq \varepsilon$ which proves the theorem. ∎

Proof of Main Lemma:

$$\begin{aligned}
[P_\varphi](x) &= \sum_y P(x,y)\varphi(y) \\
&= \frac{1}{2}\varphi(x) + \frac{1}{2}\sum_y \widehat{P}(x,y)\varphi(y) \\
&= \frac{1}{2}\sum_y \widehat{P}(x,y)\left(\varphi(x)+\varphi(y)\right) \ .
\end{aligned}$$

Assume w. l. o. g. that $E_\pi \varphi = 0$ (shifting φ by a constant value does not affect any of the quantities we are interested in, all of which are variances). Then

$$\begin{aligned}
\mathrm{Var}_\pi[P\varphi] &= \sum_x \pi(x)\left([P\varphi](x)\right)^2 \\
&= \frac{1}{4}\sum_x \pi(x)\left(\sum_y \widehat{P}(x,y)\left(\varphi(x)+\varphi(y)\right)\right)^2 \\
&\le \frac{1}{4}\sum_{xy} \pi(x)\widehat{P}(x,y)\left(\varphi(x)+\varphi(y)\right)^2 \ ,
\end{aligned}$$

where the last inequality follows from the Cauchy-Schwarz inequality (or the fact that the square of an expectation is bounded by the expectation of the square). Moreover, we can rewrite the variance yet again as

$$\begin{aligned}
\mathrm{Var}_\pi\varphi &= \frac{1}{2}\sum_x \pi(x)\varphi(x)^2 + \frac{1}{2}\sum_y \pi(y)\varphi(y)^2 \\
&= \frac{1}{2}\sum_{xy} \pi(x)\varphi(x)^2\widehat{P}(x,y) + \frac{1}{2}\sum_{xy} \pi(x)\widehat{P}(x,y)\varphi(y)^2 \\
&= \frac{1}{2}\sum_{xy} \pi(x)\widehat{P}(x,y)\left(\varphi(x)^2+\varphi(y)^2\right) \ .
\end{aligned}$$

Taking differences,

$$\mathrm{Var}_\pi\varphi - \mathrm{Var}_\pi[P_\varphi] \ \ge \ \frac{1}{4}\sum_{xy} \pi(x)\widehat{P}(x,y)\left(\varphi(x)-\varphi(y)\right)^2 \ .$$

Observe that all entries in $\widehat{P}$ are twice as large as the entries in its lazy version P, except for the diagonal elements. Diagonal elements can be ignored, however, as they contribute a value of 0 to the above sum. Hence, the right-hand side is equal to

$$\frac{1}{2}\sum_{xy} \pi(x)P(x,y)\left(\varphi(x)-\varphi(y)\right)^2 = \mathcal{E}_\pi(\varphi,\varphi) \ .$$

This completes the proof of the Main Lemma and the proof of Theorem 10.5. ■

Remark 1: The above proof assumes that the Markov chain is lazy. This device ensures that there are no periodicity issues. It can be avoided by passing to a *continuous time* version of the chain, in which periodicity can never arise. We omit the details. ■

Remark 2: If the Markov chain P is reversible, then we can take an alternative approach to proving Theorem 10.5, based on the following.

Fact 10.9 *If P is* reversible *then its eigenvalues are*

$$1 = \lambda_1 > \lambda_2 > \cdots > \lambda_N > -1$$

and its spectral gap $1 - \lambda_2$ *satisfies*

$$1 - \lambda_2 = \inf_{\varphi \text{ non-constant}} \frac{\mathcal{E}_\pi(\varphi, \varphi)}{\mathrm{Var}_\pi \varphi} \,. \tag{10.1}$$

Thus in the reversible case the Poincaré constant has an alternative interpretation as the spectral gap of the transition matrix. (The formula (10.1) follows from the standard variational characterization of eigenvalues of symmetric matrices; since P is not necessarily symmetric, but is reversible—and hence similar to a symmetric matrix—the standard formula has to be suitably weighted by the principal eigenvector π.)

This observation yields an entirely different way of proving Theorem 10.5 in the reversible case by expressing the distribution $x^{(t)}$ at time t as a linear combination of eigenvectors, and noting that the slowest rate of decay to 0 of the eigenvectors is $\max_{i \geq 2} |1 - \lambda_i|$. If in addition P is lazy, then all eigenvectors are non-negative (i.e., $\lambda_N \geq 0$), so $\max_{i \geq 2} |1 - \lambda_i| = 1 - \lambda_2$. Thus the rate at which $x^{(t)}$ approaches π can be bounded in terms of the spectral gap. The overhead in approximating the rate of decay of all the other eigenvalues by just the second is captured by the factor $\ln \pi(x)^{-1}$. For an extension of this approach to non-reversible chains (from which you can also deduce the details of the simpler reversible case) see [Fi91]. ■

We now turn to the second main ingredient of our analysis, which is a bound on the Poincaré constant in terms of flows. This theorem is due to [Si92,DS91].

Proof of Theorem 10.6: We rewrite $\mathrm{Var}_\pi \varphi$ using $\pi(x)\pi(y) = D(x, y) = \sum_{p \in P_{x,y}} f(p)$ as

$$\begin{aligned} 2\mathrm{Var}_\pi \varphi &= \sum_{xy} \pi(x)\pi(y) \left(\varphi(x) - \varphi(y)\right)^2 \\ &= \sum_{xy} \sum_{p \in P_{xy}} f(p) \left(\varphi(x) - \varphi(y)\right)^2 \,. \end{aligned}$$

We are aiming at an expression made up of local variances. Therefore we use the following telescoping sum: for any path $p \in P_{xy}$, $\varphi(x) - \varphi(y) = \sum_{(u,v) \in p} (\varphi(v) - \varphi(u))$. Thus we may continue the above derivation as follows:

$$\begin{aligned} &\sum_{xy} \sum_{p \in P_{xy}} f(p) \left(\sum_{(u,v) \in p} (\varphi(v) - \varphi(u)) \right)^2 \\ &\leq \sum_{xy} \sum_{p \in P_{xy}} f(p)|p| \sum_{(u,v) \in p} \left(\varphi(v) - \varphi(u)\right)^2 \end{aligned}$$

by the Cauchy-Schwarz inequality. Switching the order of summation, this equals

$$\begin{aligned} &\sum_{e=(u,v)} \left(\varphi(v) - \varphi(u)\right)^2 \sum_{p \ni e} f(p)|p| \\ &\leq \ell(f) \sum_{e \in (u,v)} \left(\varphi(v) - \varphi(u)\right)^2 \sum_{p \ni e} f(p) \,. \end{aligned}$$

Recalling $\sum_{p \ni e} f(p) = f(e)$ and $\rho(f) = \max_e f(e)/C(e)$, the above term is at most

$$\ell(f)\rho(f) \sum_{e \in (u,v)} \left(\varphi(v) - \varphi(u)\right)^2 C(e)$$

and substituting $C(e) = \pi(u)P(u,v)$ results in the bound

$$2\ell(f)\rho(f)\mathcal{E}_\pi(\varphi,\varphi)\ .$$

This completes the proof. ■

In the next lecture we shall see some applications of Theorem 10.1, by constructing suitable flows in various Markov chains.

References

[DS91] P. Diaconis and D. Stroock, "Geometric bounds for eigenvalues of Markov chains," *Annals of Applied Probability* **1**, 1991, pp. 36–61.

[Fi91] J.A. Fill "Eigenvalue Bounds on Convergence to Stationarity for Nonreversible Markov Chains, with an Application to the Exclusion Process," *Annals of Applied Probability* **1** (1991), pp. 62–87.

[Je03] M. Jerrum, *Counting, sampling and integrating: Algorithms and complexity*, Birkhäuser Lectures in Mathematics, 2003.

[Mi89] M. Mihail, "Conductance and convergence of Markov chains: A combinatorial treatment of expanders," *Proceedings of the 30th STOC*, 1989, pp. 526–531.

[Si92] A. Sinclair, "Improved bounds for mixing rates of Markov chains and multicommodity flow," *Combinatorics, Probability and Computing* **1**, 1992, pp. 351–370.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 11: October 13

Lecturer: Alistair Sinclair *Scribes: Alistair Sinclair*

11.1 Review

Recall from the previous lecture the definition of the *Poincaré constant*

$$\alpha := \inf_{\varphi \text{ non-constant}} \frac{\mathcal{E}_\pi(\varphi,\varphi)}{\mathrm{Var}_\pi \varphi} \ ,$$

where the infimum is over all non-constant functions $\varphi : \Omega \to \mathbb{R}$. In the above expression, the numerator $\mathcal{E}_\pi(\varphi,\varphi)$ is the "local variance" and the denominator is the global variance of φ w.r.t. the stationary distribution π.

In the previous lecture we proved the following theorem relating the mixing time to α.

Theorem 11.1 *For any lazy ergodic P and any initial state $x \in \Omega$,*

$$\tau_x(\varepsilon) \le \frac{1}{\alpha}\left(2\ln \varepsilon^{-1} + \ln \pi(x)^{-1}\right) \ .$$

We then proved the following combinatorial bound on α in terms of multicommodity flows on the Markov chain (viewed as a network).

Theorem 11.2 *For any ergodic P and any flow f for P*

$$\alpha \ge \frac{1}{\rho(f)\ell(f)} \ ,$$

where $\rho(f)$ is the cost of f and $\ell(f)$ is the length of f.

The above two results lead to the following immediate corollary:

Corollary 11.3 *For any lazy, ergodic Markov chain P and any flow f, the mixing time from any initial state $x \in \Omega$ is bounded by*

$$\tau_x(\varepsilon) \le \rho(f)\ell(f)\left(2\ln \varepsilon^{-1} + \ln \pi(x)^{-1}\right) \ .$$

We briefly mention here, without proof, some (partial) converses to the above bounds.

- Converse for Theorem 11.1: $\tau_{mix} \ge \text{constant} \cdot \frac{1-\alpha}{\alpha}$.
- Converse for Theorem 11.2: There exists a flow f with $\alpha \le \text{constant} \cdot \frac{\log|\Omega|}{\rho(f)}$. (This is a consequence of the $\log n$ approximation to sparsest cut devised by Leighton-Rao using multicommodity flows; see [Si92].)

- Converse for Corollary 11.3: There exists a flow f with $\rho(f) \leq \text{constant} \cdot \tau_{\text{mix}}$. (This flow is exactly that generated by the probability mass flow in the Markov chain itself; see [Si92].)

In this lecture we will apply the above flow technology to obtain bounds on the mixing time of various Markov chains. We begin with a few very simple warm-up examples.

11.2 Example: Random Walk on the Hypercube

Consider the usual lazy random walk on the hypercube $\{0,1\}^n$. Let $N = 2^n$ be the number of vertices. Then $\pi(x) = \frac{1}{N}$. The edge capacity is $C(u,v) = \pi(u)P(u,v) = \frac{1}{N}\frac{1}{2n} = \frac{1}{2Nn}$ for all edges (u,v). The demand between vertices x, y is $D(x,y) = \pi(x)\pi(y) = \frac{1}{N^2}$.

Now we need to construct a flow f satisfying the demands above. Moreover, to obtain the best bound possible we want to do this maximizing the ratio $\frac{1}{\rho(f)\ell(f)}$. Intuitively, this is achieved by spreading the flow between x and y evenly among all shortest paths from x to y. So we define the flow f in this way.

To compute $\rho(f)$ we now need to get a handle on $f(e)$ for all e. This is not difficult if we use the symmetry of the hypercube to notice that $f(e) = f(e')$ for all edges e, e', so that:

$$f(e) = \frac{\sum_{e \in E} f(e)}{|E|} = \frac{\frac{1}{N^2}\sum_{x,y \in V}\{\text{length of shortest path between } x,y\}}{Nn}$$

Notice $|E| = Nn$ as, in the context of flow, we consider edges to be directed. Observe that the average distance between two vertices in the hypercube is $n/2$ so that:

$$\sum_{x,y \in V} \{\text{length of shortest path between } x,y\} = N^2 \frac{n}{2}$$

Thus we have:

$$f(e) = \frac{\frac{1}{N^2}\frac{n}{2}N^2}{Nn} = \frac{1}{2N} \ .$$

Hence, $\rho(f) = \max_{e \in E} \frac{f(e)}{C(e)} = \frac{1}{2N} 2Nn = n$. Moreover, as all the flow goes along shortest paths we have also that $\ell(f) = n$, so from Corollary 11.3 we get (for some constant c)

$$\begin{aligned} \tau_x(\varepsilon) &\leq \rho(f)\ell(f)\left(c + \ln \pi(x)^{-1}\right) \\ &= n \cdot n \cdot O(n) = O(n^3). \end{aligned}$$

This is not tight, as we saw in an earlier lecture that $\tau_{\text{mix}} \sim \frac{n}{2}\ln n$ (and indeed obtained a similar bound, up to a constant factor, by coupling). However, our flow argument does yield a polynomial upper bound on the mixing time despite the fact that the size of the cube is exponential in n, so this is a non-trivial result. The slackness in our bound is typical of this method, which is heavier-duty than coupling.

Let's look in a bit more detail at where we lose here. In the case of the hypercube, it is actually possible to compute exactly the Poincaré constant α (or, equivalently since this is a reversible chain) the spectral gap $1 - \lambda_2$) to be $\sim \frac{1}{n}$. This shows that both Theorems 11.1 and 11.2 are not tight, as:

- Using the above exact value of α, Theorem 11.1 gives $\tau_{mix} = O(n^2)$, which is off by a factor of almost n from the true value. This error results from the factor $\ln \pi(x)^{-1}$, which arises essentially because we approximate the mixing time using only the second eigenvalue, treating the other eigenvalues pessimistically.
- Theorem 11.2 bounds α by $\frac{1}{\rho(f)\ell(f)} = 1 - \frac{1}{n^2}$, which is also loose as the true value is $\alpha \sim \frac{1}{n}$. This demonstrates the potential slackness in our bound relating α to flows.

11.3 Example: Random Walk on a Line

We consider the Markov chain for lazy random walk on the line $\{1, 2, \ldots, N\}$, with self-loop probability $1/2$ at every state (except at the endpoints where it is $3/4$). (Thus the chain moves left or right from each position with probability $1/4$ each.) The stationary distribution is uniform: $\pi(x) = 1/N$. The edge capacities and demands are

$$C(e) \;=\; \pi(x)P(x,y) \;=\; \frac{1}{4N} \quad \text{for all non-self loop edges ,}$$

$$D(x,y) \;=\; \pi(x)\pi(y) \;=\; \frac{1}{N^2}\ \forall x, y\ .$$

There is only one simple path between each pair of vertices, so there is a unique flow f here. The amount of flow on any edge $(i, i+1)$ is given by

$$f\big((i,i+1)\big) \;=\; i(N-i)\frac{1}{N^2} \;\le\; \frac{1}{4}\ .$$

(The maximum is achieved on the middle edge.) The cost of the flow f is

$$\rho(f) \;=\; \max_e \frac{f(e)}{C(e)} \;\le\; \frac{1/4}{1/(4N)} \;=\; N\ ,$$

and the length is

$$\ell(f) \;=\; N\ .$$

Thus from Theorem 11.2 the Poincar'e constant is bounded by

$$\alpha \;\ge\; \frac{1}{\rho(f)\ell(f)} \;\ge\; \frac{1}{N^2}\ ,$$

which happens to be asymptotically tight for this Markov chain. From Corollary 11.3 the bound on the mixing time is

$$\tau_{mix} = O(N^2 \log N).$$

This is off from the true answer only by the $O(\log N)$ factor (which again arises from Theorem 11.1).

11.4 Example: Random Walk on $K_{2,N}$

Consider lazy random walk on the complete bipartite graph $K_{2,N}$, with a self-loop of $1/2$ at every vertex. Label the two vertices on the small side of the graph s, t respectively.

The stationary distribution is

$$\pi(s) \;=\; \pi(t) \;=\; \frac{1}{4}\ , \qquad \pi(x) \;=\; \frac{1}{2N}\ \forall x \neq s, t.$$

The capacity $C(e) = 1/(8N)$ for all edges and the demands are given by:

$$\begin{aligned} &D(s,t) \;=\; D(t,s) \;=\; 1/16, \\ &D(s,x) \;=\; D(t,x) \;=\; D(x,s) \;=\; D(x,t) \;=\; 1/(8N), \\ &D(x,y) \;=\; 1/(4N^2). \end{aligned}$$

Consider two strategies to route flow between any pair of states:

(1) Send flow along a single shortest path for each pair of states. Then exactly four edges will carry 1/16 units of flow for the $s \to t$ and $t \to s$ paths. Construct the flow f such that the edges carrying the maximum flow only carry flows from/to s and t. Then,

$$\max_e f(e) \;\le\; \frac{1}{8N} + \frac{1}{16} \;,$$

which gives,

$$\rho(f) \;\le\; \frac{1/(8N) + 1/16}{1/(8N)} \;=\; constant \times N \;.$$

Also $\ell(f) = 2$. Hence (assuming we start at either s or t), we get a bound on the mixing time of $O(N)$. This is a severe over-estimate as the true mixing time (starting from any state) is $\Theta(1)$ (why?)

(2) Distribute $s \to t$ and $t \to s$ flows along all shortest paths evenly. Then,

$$\max_e f(e) \;\le\; \frac{1}{8N} + \frac{1}{16N} + \frac{1}{4N^2}\frac{N-1}{2} \;\le\; constant \times \frac{1}{N} \;,$$

which gives,

$$\rho(f) \;\le\; \frac{constant \times 1/N}{1/(8N)} \;=\; constant \;.$$

Hence, the mixing time starting from s or t is $O(1)$, which is correct.

This simple example shows that it is sometimes necessary to spread the flow over many paths in order to get a good value for $\rho(f)$.

11.5 Generic Flow Calculation

Our examples so far have been on symmetric graphs where computing the cost of a flow was easy. But in general, it is difficult to determine how much flow is carried on an edge. To solve this, we will develop technology to count paths and calculate flows in a generic setup.

Let $|\Omega| = N$ be the size of the state space of an ergodic, lazy Markov chain, where N is exponential in n, the natural measure of problem size. Assume the stationary distribution is uniform ($\pi(x) = 1/N$). Also assume $P(u, v) \;\ge\; 1/poly(n)$ for all non-zero transition probabilities. (This implies that the degree of the underlying graph is not huge, and holds in most of our examples.) Then the capacity of any edge is

$$C(u, v) \;=\; \pi(u)P(u, v) \;\approx\; \frac{1}{Npoly(n)} \;.$$

Our goal is to construct a flow f such that

$$\frac{f(e)}{C(e)} \;\le\; poly(n), \qquad \ell(f) \;\le\; poly(n) \;,$$

in order to provide a polynomial upper bound on mixing time. Hence we must have

$$f(e) \;\le\; \frac{poly(n)}{N} \;. \tag{11.1}$$

On the other hand, the number of edges in the graph is less than $N \times poly(n)$ and the total flow along all paths is $\sum_{x,y} \frac{1}{N^2} \approx 1$. Hence *some* edge must carry at least $1/(N \times poly(n))$ flow, i.e.,

$$f(e) \;\ge\; \frac{1}{N \times poly(n)} \;.$$

Comparing this with Eq. (11.1) we see that any good flow has to be optimal up to a polynomial factor.

Finally, suppose the flow $x \to y$ goes along a *single* path $\gamma_{x,y}$. (Again, in many of our examples this will be the case.) Let $paths(e)$ denote the set of paths through edge e under flow f. Then

$$f(e) \ = | \ paths(e) \ | \ \times \frac{1}{N^2} \ .$$

Since we wanted $f(e) \leq poly(n)/N$, this implies that the flow must satisfy

$$| \ paths(e) \ | \leq \ N \times poly(n) \ .$$

The goal, then, is to set up a flow such that the number of paths along any edge is $N \times poly(n)$. This will give us a polynomial bound on the mixing time using Theorem **??**. However, we typically don't know N, the size of the state space; in fact, this is often what we are trying to compute! So we can't hope to explicitly count the paths through an edge and compare this value with N. The way we get around this is to construct an *injective map*,

$$\eta_e : \ paths(e) \hookrightarrow \Omega \ .$$

Such a map (if we can construct it) will show implicitly that $|paths(e)| \leq |\Omega|$, which is what we want, without any counting. We continue this discussion in the next lecture.

References

[Si92] A. Sinclair, "Improved bounds for mixing rates of Markov chains and multicommodity flow," *Combinatorics, Probability and Computing* **1**, 1992, pp. 351–370.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 12: October 15

Lecturer: Alistair Sinclair *Scribes: Alistair Sinclair*

12.1 Flow encodings

Recall the generic calculation from the end of last lecture: in a typical Markov chain of the type that occurs in our applications, if the stationary distribution is uniform and if we send the $x \to y$ flow along a single path γ_{xy}, then in order for this flow to be good it must satisfy

$$|\text{paths}(e)| \leq \text{poly}(n)|\Omega| \qquad \forall e, \tag{12.1}$$

where $\text{paths}(e)$ is the set of paths that go through edge e. If we allow multiple paths from x to y then the same calculation applies but in an average sense, and if the stationary distribution is non-uniform then again the same applies but with everything suitably weighted.

Equation (12.1) indicates that we must compare $|\text{paths}(e)|$ with the size of the state space $N = |\Omega|$. However, since computing N is often our goal, this seems difficult. The following machinery is designed to get around this problem. The key idea is to set up an *injective mapping* from $\text{paths}(e)$ to Ω which allows us to compare their sizes implicitly. This technology is employed in almost all non-trivial applications of multicommodity flows in the analysis of mixing times.

Definition 12.1 *An* encoding *for a flow f (that uses only single paths γ_{xy} for each x, y) is a set of functions $\eta_e : \text{paths}(e) \to \Omega$ (one for each edge e) such that*

1. *η_e is injective*
2. *$\pi(x)\pi(y) \leq \beta\pi(z)\pi(\eta_e(x, y))$ $\forall (x, y) \in \text{paths}(e)$, where $e = (z, z')$.*

Remark: Property 1 says that η_e is an injection, as motivated earlier. Property 2 says that η_e is in addition weight-preserving up to a factor β (which is assumed to be not too large: constant or polynomially bounded). Note that property 2 is automatically satisfied with $\beta = 1$ when π is uniform. In some applications, we may usefully weaken property 1 slightly as follows: we may not have a perfect injection, but may require a small amount of "extra information" to invert η_e. As should be clear from the proof of the following claim, this will just insert a modest additional factor into the bound on the mixing time.

Claim 12.2 *If there exists an encoding for f as above, then $\rho(f) \leq \beta \max_{P(z,z')>0} \frac{1}{P(z,z')}$.*

Proof: Let $e = (z, z')$ be an arbitrary edge. Then

$$f(e) = \sum_{(x,y)\in\text{paths}(e)} \pi(x)\pi(y) \leq \beta \sum_{(x,y)\in\text{paths}(e)} \pi(z)\pi(\eta_e(x, y)) \leq \beta\pi(z).$$

In the first inequality here we have used property 2, and in the second we have used property 1.

Finally, $C(e) = \pi(z)P(z,z')$, so $f(e)/C(e) \le \beta/P(z,z')$. ■

We now give a simple example of an analysis using flow encodings, followed by a more serious example.

12.2 Example: Cube $\{0,1\}^n$

Previously, in analyzing the random walk on the cube by flows, we spread flow evenly and used the symmetry of the cube to bound $\rho(f)$. In more sophisticated applications, we will not have this symmetry and not know $N = 2^n$. Flow encodings let us proceed without appealing to these properties of the cube.

Recall that the capacity of an edge e is $C(e) = 1/(2nN)$, and the demand between two vertices x, y is $D(x,y) = 1/N^2$. Now consider a flow that sends all the $x \to y$ flow along the "left-right bit-fixing path" γ_{xy}. That is, correct each bit of x sequentially, left-to-right, until arriving at y. Then $\ell(f) = n$, clearly.

We now bound the cost of this flow using the encoding technique. Consider an arbitrary edge $e = (z, z')$, where z and z' differ in bit position i. Consider any pair $(x,y) \in \text{paths}(e)$. What do we know about x and y? Notice that y agrees with z in the first $i-1$ bits (which have already been corrected), and x agrees with z in the last $n-i$ bits. We therefore define $\eta_e : \text{paths}(e) \to \Omega$ by $\eta_e(x,y) = x_1 x_2 \dots x_i y_{i+1} y_{i+2} \dots y_n$. I.e., $\eta_e(x,y)$ is the $0,1$-string that agrees with x on the first i bits and with y on the rest.

Now it is easy to see that we can uniquely recover (x,y) from $\eta_e(x,y)$ and e (this is why we constructed η_e this way!), so η_e is an injection. Since the stationary distribution is uniform, it is trivially weight-preserving. Hence η_e is a valid encoding. By the above Claim therefore,

$$\rho(f) \le \max_{z,z'} \frac{1}{P(z,z')} = 2n \ .$$

Up to a constant, this is the same bound we obtained for the cube by spreading flow uniformly and appealing to symmetry. In particular, we again get $\tau_{\text{mix}} \le O(n^3)$. However, the key point about this alternative flow and its analysis is that we never used the fact that $N = 2^n$, nor any of the cube's symmetry. In the next section, we'll see a much harder example in which this feature is crucial.

12.3 Matchings in a graph

We now use encodings to analyze a more complicated Markov chain that samples matchings in a graph, due to [JS89].

Recall that a *matching* in G is a set of edges of G such that no two edges share a vertex. Given an undirected graph $G = (V, E)$ and a parameter $\lambda \ge 1$,[1] we wish to sample from the set Ω of all matchings in G according to the *Gibbs distribution*,

$$\pi(M) = \frac{1}{Z}\lambda^{|M|},$$

where $|M|$ is the number of edges in the matching M and $Z = Z(\lambda)$ is the normalizing factor (partition function). If m_k is the number of k-matchings of G (matchings with k edges), then $Z(\lambda) = \sum_k m_k \lambda^k$, the matching polynomial of G. This problem is motivated both by its combinatorial significance and because it corresponds to the so-called *monomer-dimer* model of statistical physics: in this model, vertices connected by

[1] Actually the algorithm and its analysis are essentially the same for $\lambda < 1$, but we consider only the more important case $\lambda \ge 1$ for definiteness.

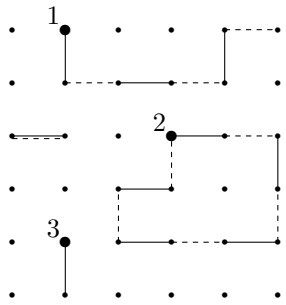


Figure 12.1: The matchings x (solid lines) and y (dotted lines).

an edge in the matching correspond to diatomic molecules (dimers), and unmatched vertices to monatomic molecules (monomers). We note also that computing the partition function $Z(\lambda)$ is #P-complete for any fixed $\lambda > 0$.

Markov chain: We define a Markov chain on the space of matchings using three kinds of transitions: edge addition, edge deletion, and edge exchange (deleting an edge and adding an edge sharing one vertex with the deleted edge). We make the Markov chain lazy and use the Metropolis rule to make the stationary distribution match the Gibbs distribution, as follows. At a matching $M \in \Omega$,

- (Laziness) With probability 1/2, stay at M.
- Otherwise, choose an edge $e = (u, v) \in E$ u.a.r.
- (Edge addition) If both u and v are unmatched in M, then go to $M + e$.
- (Edge deletion) If $e \in M$, then go to $M - e$ with probability $1/\lambda$ (else stay at M).
- (Edge exchange) If exactly one of u and v is matched in M, let e' be the unique edge in M containing u or v, and go to $M + e - e'$.
- If both u and v are matched, then do nothing.

This Markov Chain follows the Metropolis rule. Edge addition and edge exchange either increase or do not change the weight of the matching, so we always perform these moves. However, an edge deletion decreases the weight by a factor of λ, so we only accept an edge deletion with probability $1/\lambda$.

Flow: To define a flow f, we pick a path γ_{xy} for every pair of matchings x and y and send all of the flow along it. Imagine that we color x red and y blue, then superimpose the two matchings to form $x + y$. Because x and y are matchings, the connected components of $x + y$ consist of (closed) cycles of alternating red and blue edges, (open) paths of alternating red and blue edges, and edges that are both red and blue (or equivalently, trivial alternating cycles of length two).

Now fix (for the sake of analysis only) an arbitrary total ordering on the set of *all* open paths and even-length cycles (of length at least four) in G. Designate one vertex of each such cycle and path as its "start vertex"; the start vertex must be an endpoint in the case of a path. This ordering induces an ordering on those paths and cycles that actually appear in $x+y$. Figure 12.1 shows an example; the start vertices in each component are marked with a large circle and the induced ordering is shown.

To define the flow from x to y, we process the paths and cycles in the given order. To process a path, let its consecutive edges be $e_1, e_2, \ldots, e_r$, where e_1 is the edge containing the start vertex, and apply the following transitions to x:

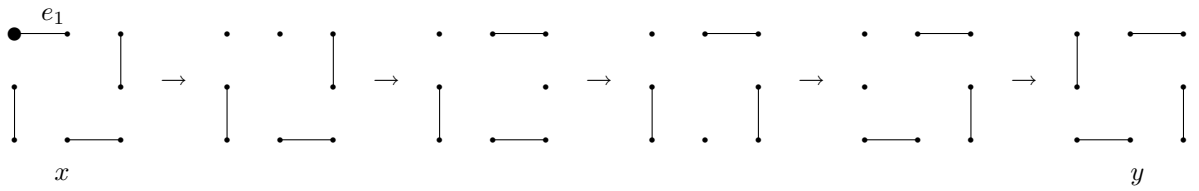


Figure 12.2: Unwinding a single cycle along the path from x to y.

- If e_1 is red, then remove e_1, exchange e_3 for e_2, exchange e_5 for e_4, and so on. If e_r is blue, we have the additional move of adding e_r at the end.
- If e_1 is blue, then exchange e_2 for e_1, exchange e_4 for e_3, and so on. If e_r is blue, we have the additional move of adding e_r at the end.

For cycles, the processing is similar. Suppose that the edges of the cycle are $e_1, \ldots, e_{2r}$, where e_1 is the red edge adjacent to the start vertex. Process the cycle by first removing e_1, then exchanging e_3 for e_2, exchanging e_5 for e_4, ..., exchanging e_{2r-1} for e_{2r-2}, and finally adding e_{2r}.

By processing the components of $x + y$ in succession, we have constructed a path from x to y; let γ_{xy} be this path.

Encoding: We will define the encoding η_t for a transition $t = (z, z')$ corresponding to an edge exchange. For the other types of transition t, η_t is defined similarly.

Suppose transition t involves exchanging e' for e. Now let x and y be matchings for which the path γ_{xy} traverses t, i.e., $(x, y) \in \text{paths}(t)$. Consider the multiset $x + y$. If z and z' are part of a cycle of $x + y$, then let e_1 be the first edge removed in processing $x + y$, and define

$$\eta_t(x, y) = (x + y) \setminus (z \cup z' \cup \{e_1\}).$$

Otherwise, define

$$\eta_t(x, y) = (x + y) \setminus (z \cup z').$$

It is not hard to check [**exercise!**] that $\eta_t(x, y)$ is always a matching, so η_t is well-defined. Now this definition may seem mysterious, but $\eta_t(x, y)$ is really the disjoint union of $x \cap y$ (i.e., all edges of $x + y$ that are both red and blue) and $(x \oplus y) \setminus (z \cup z')$ (with e_1 also deleted if we are processing a cycle). The latter term is just all the edges in the paths and cycles of $x + y$ that are missing from $z \cup z'$. It will be convienient for us to write this as

$$\eta_t(x, y) = x \oplus y \oplus (z \cup z') \setminus \{e_1\}. \tag{12.2}$$

As an example, let x and y be as in Figure 12.2 and let t be the middle (third) transition in the example. Then the encoding $\eta_t(x, y)$ is as shown in Figure 12.3.

We need to check that η_t is indeed an encoding. To verify the injectivity property, we need to be able to recover (x, y) uniquely from $\eta_t(x, y)$. Using (12.2), we see that $x \oplus y = (\eta_t(x, y) \cup \{e_1\}) \oplus (z \cup z')$, so we can recover $x \oplus y$. (There is a detail here: because of the absence of edge e_1 from both $z \cup z'$ and $\eta_t(x, y)$, we can't tell the difference between a cycle and a path that differ up to addition of e_1. But we can overcome this with a trick: we can use the sense of unwinding of the path/cycle to distinguish the two possibilities.) Now we can partition $x \oplus y$ into x and y by using the ordering of the paths. That is, from the transition t we can identify which path/cycle is currently being processed. Then for every path that preceeds the current one, we know that the path agrees with y in z and with x in $\eta_t(x, y)$. For the paths following the current one, the parity is reversed. Finally, for the current path we know that z agrees with y up to the point of the

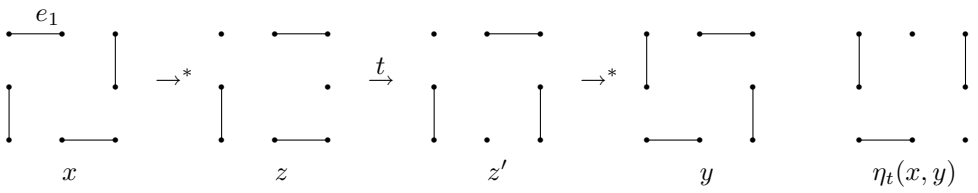


Figure 12.3: The transition t and its encoding $\eta_t(x, y)$.

current transition, and with x beyond it. Finally, we can easily identify the edges that belong to both x and y as they are just $z \cap z' \cap \eta_t(x, y)$. Thus we see that η_t is indeed an injection.

Notice that for a transition $t = (z, z')$ on a path from x to y, we have that the multiset $z \cup \eta_t(x, y)$ has at most two fewer edges than the multiset $x \cup y$ (arising from the possible edge deletion that starts the processing of a path, and from the edge missing from the current transition). Therefore, $\pi(x)\pi(y) \leq \lambda^2 \pi(z)\pi(\eta_t(x, y))$, so η satisfies property (ii) in the definition of encoding, with $\beta = \lambda^2$.

Analysis of τ_{mix}: For any transition $t = (z, z')$, $P(z, z') \geq \frac{1}{2\lambda|E|}$ (attained for edge deletions t). Thus by the above encoding and Claim 12.2,

$$\rho(f) \leq \lambda^2 \cdot \left(\frac{1}{2\lambda|E|}\right)^{-1} = O(\lambda^3|E|).$$

Now the length of γ_{xy} is at most $|x| + |y|$ because every edge of $x \cup y$ is processed at most once. Since x and y are matchings, $|x|$ and $|y|$ are at most $|V|/2$. Hence the length of γ_{xy} is at most $|V|$, and $\ell(f) \leq |V|$. By Theorem 11.2 of Lecture 11, the Poincaré constant α of P satisfies

$$\alpha \geq \frac{1}{\rho(f)\ell(f)} \geq \Omega\left(\frac{1}{\lambda^3|V||E|}\right).$$

Then Corollary 11.3 of Lecture 11 implies that

$$\tau_x(\epsilon) \leq O\left((\log \pi(x)^{-1} + \log \epsilon^{-1}) \cdot \frac{1}{\alpha}\right).$$

From this we obtain

$$\tau_x(\epsilon) \leq O\left(\lambda^3|E||V|(\log \pi(x)^{-1} + \log \epsilon^{-1})\right).$$

Note that Ω is a subset of the $2^{|E|}$ subgraphs of G, so $|\Omega| \leq 2^{|E|}$. Also, there exists a polynomial time algorithm to find a matching x of G with the maximum number of edges. This x has maximal weight in the Gibbs distribution, so

$$\pi(x) \geq \frac{1}{|\Omega|} \geq \frac{1}{2^{|E|}}.$$

Therefore, the mixing time starting from a maximum matching x is bounded by

$$O\left(\lambda^3|E|^2|V|\right),$$

which is polynomial in λ and in the size of the graph. [As a side note, we can reduce the upper bound on the mixing time to $O\left(\lambda^2|E|^2|V|\right)$ by taking a little more care in the above encoding analysis.]

In the next lecture, we will look at some applications and extensions of this Markov chain on matchings.

References

[JS89] M. Jerrum and A. Sinclair, "Approximating the Permanent," *SIAM Journal on Computing* **18** (1989), pp. 1149–1178.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 13: Oct 20

Lecturer: Alistair Sinclair *Scribes: Alistair Sinclair*

Last time we saw an MCMC algorithm that samples matchings M in an undirected graph $G = (V, E)$ from the Gibbs distribution $\pi_\lambda(M) = \frac{1}{Z(\lambda)}\lambda^{|M|}$ with running time poly(n, λ), where $n = |V|$. Here the normalizing factor $Z(\lambda) = \sum_M \lambda^{|M|} = \sum_k m_k \lambda^k$, where m_k is the number of k-matchings in G, is known in the statistical physics context as the *partition function*, and in the combinatorial context as the *matching polynomial*. In this lecture we look at some applications and extensions of this Markov chain on matchings, taken from [JS89].

13.1 Extensions and Applications

13.1.1 Estimating the Partition Function Z(λ)

Computing $Z(\lambda)$ exactly for any fixed value of $\lambda > 0$ is #P-complete. So the best we can hope for is an approximation algorithm.

First, we express $Z(\lambda)$ as a telescoping product as follows:

$$Z(\lambda) = \frac{Z(\lambda_r)}{Z(\lambda_{r-1})} \times \frac{Z(\lambda_{r-1})}{Z(\lambda_{r-2})} \ldots \times \frac{Z(\lambda_1)}{Z(\lambda_0)} \times Z(\lambda_0), \text{ where } 0 = \lambda_0 < \lambda_1 < \lambda_2 < \ldots < \lambda_r = \lambda.$$

Notice that $Z(\lambda_0) = Z(0) = 1$ because there is exactly one 0-matching (the empty matching) in any G. The sequence λ_i is chosen to increase slowly so that each factor in the product is bounded by a constant, allowing it to be estimated efficiently by random sampling.

Accordingly, define the sequence as: $\lambda_1 = \frac{1}{|E|}, \lambda_i = (1+\frac{1}{n})\lambda_{i-1}$ for $i = 2, \ldots, r-1$, and $\lambda_r = \lambda \leq (1+\frac{1}{n})\lambda_{r-1}$. Notice that this gives $r = O(n \log \lambda)$, so there are not many factors in the product. Also, we can easily verify the following upper bound for each factor:

$$\frac{Z(\lambda_i)}{Z(\lambda_{i-1})} = \frac{\sum_k m_k \lambda_i^k}{\sum_k m_k \lambda_{i-1}^k} \leq \left(1 + \frac{1}{n}\right)^n \leq \text{e}.$$

The final observation is that each such factor can be expressed as the expectation of an appropriate random variable defined over the Gibbs distribution on matchings with parameter λ_{i-1} This implies that we can estimate each factor by random sampling from the Gibbs distribution π_{λ_i}.

Claim 13.1 $\frac{Z(\lambda_i)}{Z(\lambda_{i-1})} = E_{\pi_{\lambda_{i-1}}}\left[\left(\frac{\lambda_i}{\lambda_{i-1}}\right)^{|M|}\right]$

Proof:

$$\begin{aligned} E_{\pi_{\lambda_{i-1}}}\left[\left(\frac{\lambda_i}{\lambda_{i-1}}\right)^{|M|}\right] &= \frac{1}{Z(\lambda_{i-1})}\sum_k m_k \lambda_{i-1}^k \left(\frac{\lambda_i}{\lambda_{i-1}}\right)^k \\ &= \frac{1}{Z(\lambda_{i-1})}\sum_k m_k \lambda_i^k \\ &= \frac{Z(\lambda_i)}{Z(\lambda_{i-1})}. \end{aligned}$$

■

Our algorithm will estimate $Z(\lambda)$ by estimating each $Z(\lambda_i)/Z(\lambda_{i-1})$ in succession and taking the product as above. Each of these factors is estimated by random sampling from π_{λ_i}. Since, as we saw above, the expectation of each such r.v. is bounded by a constant, the number of samples required for each factor in order to ensure an overall estimate within a factor of $(1\pm\epsilon)$ is about $O(n^2\epsilon^{-2})$. (See the analogous calculation in a future lecture for graph colorings.) Also, by our mixing time analysis, the time to obtain each sample is bounded by $\text{poly}(n,\lambda)$.

The bottom line is a *fully-polynomial randomized approximation scheme* for $Z(\lambda)$, that is, we get an estimate of $Z(\lambda)$ within ratio $(1\pm\epsilon)$ with high probability in total time $\text{poly}(n,\lambda,\epsilon^{-1})$.

13.1.2 Estimating the Coefficients m_k

The coefficient m_k represents the number of k-matchings in the graph. When $k = |V|/2$, m_k is the number of perfect matchings. If the graph G is bipartite with $|V|$ vertices on each side, counting the number of perfect matchings is equivalent to computing the *permanent* of the adjacency matrix A_G of G. (This is an easy **exercise**.)

Definition 13.2 *The permanent of an $n\times n$ matrix A is defined as*

$$\text{per}(A) = \sum_\sigma \prod_{i=1}^n A(i,\sigma(i)),$$

where the sum is over all permutations σ of $\{1,\ldots,n\}$.

Note the similarity with the determinant:

$$\det(A) = \sum_\sigma (-1)^{sign(\sigma)} \prod_{i=1}^n A(i,\sigma(i)).$$

However, while we have several polynomial time algorithms (such as Gaussian elimination) for computing $\det(A)$, it is known that evaluating $\text{per}(A)$ is #P-complete. (This is a celebrated result of Valiant [Val79a].)

We return now to estimating m_k. If we could compute the partition function $Z(\lambda)$ *exactly*, we could also get all the m_k exactly by computing $Z(\lambda)$ at $n+1$ points λ and using interpolation. (Note that Z is a polynomial of degree at most n.) However, polynomial interpolation is not robust, so if we only know the value of $Z(\lambda)$ approximately this could result in large errors in its coefficients. Instead we use a different approach. We need the following two claims before describing the algorithm.

Claim 13.3 *The sequence $\{m_k\}$ is* log-concave*: i.e., $m_{k-1}m_{k+1} \le m_k^2$ for all k.*

This is left as an **exercise**. [Hint: set up an injective mapping that takes a pair of matchings, one with $k-1$ edges and the other with $k+1$ edges, to a pair of k-matchings. This uses similar ideas to our encoding analysis for the mixing time in the last lecture.]

Claim 13.4 *If $\lambda = \frac{m_{k-1}}{m_k}$, then the ratio $m_{k'}\lambda^{k'}$ is maximized at $k' = k$ and $k' = k-1$.*

Proof: First notice that $m_k\lambda^k = (m_k\lambda)\lambda^{k-1} = m_{k-1}\lambda^{k-1}$. Next, since the log function is monotonic, it suffices to check that $\log(m_{k'}\lambda^{k'}) = \log m_{k'} + k' \log\lambda$ is maximized at $k' = k$. Since the sequence $\{\log m_{k'}\}$ is concave (by the previous Claim), it suffices to show that $m_k\lambda^k \geq m_{k+1}\lambda^{k+1}$ and that $m_{k-1}\lambda^{k-1} \geq m_{k-2}\lambda^{k-2}$. To see the first of these, note that by log-concavity

$$\frac{m_{k+1}\lambda^{k+1}}{m_k\lambda^k} = \lambda\frac{m_{k+1}}{m_k} = \frac{m_{k-1}m_{k+1}}{m_k^2} \leq 1.$$

The argument for the second inequality is entirely similar. ■

This suggests the following algorithm to successively estimate m_k. Trivially $m_0 = 1$, so we can estimate m_k by successively estimating each of the ratios $m_{k'}/m_{k'-1}$ for $2 \leq k' \leq k$ and multiplying them together. To estimate the ratio $m_{k'}/m_{k'-1}$, the idea is to raise λ until $(k'-1)$- and k'-matchings appear frequently enough in the stationary distribution π_λ to estimate their ratio reliably:

- Gradually increase λ and sample from the Gibbs distribution π_λ until we see "lots of" $(k'-1)$- and k'-matchings in stationary distribution. "Lots of " means that their probability is at least about $\frac{c}{n}$ for some constant c.
- For this λ, use samples from the MC to estimate $m_{k'}/m_{k'-1}$. The estimator is the ratio of the number of k'-matchings to the number of $(k'-1)$-matchings in the sample, multiplied by the factor $1/\lambda$.
- To obtain the estimate of m_k, multiply the estimates of $m_{k'}/m_{k'-1}$ obtained as above for $2 \leq k' \leq k$.

Claim 13.4 confirms that, in each stage of the above algorithm, λ won't need to be larger than $m_{k'-1}/m_{k'}$, which by log-concavity is at most m_k/m_{k-1} for all $k' \leq k$. By our Markov chain analysis from the last lecture, the time to obtain a sample is polynomial in n and λ, and by similar standard statistical arguments as in the previous section, the total number of samples required for a $(1 \pm \epsilon)$ estimate of m_k is polynomial in n and ϵ^{-1}. Hence the entire algorithm takes time $\text{poly}(n, m_{k-1}/m_k, \epsilon^{-1})$ to get a $(1 \pm \epsilon)$ estimate of m_k/m_{k-1} with high probability. Thus the algorithm works well as long as m_{k-1}/m_k is not too large, i.e., smaller than some fixed polynomial $\text{poly}(n)$.

Exercises: (i) In a graph with $2n$ vertices (so that a maximum matching has size n), show how to use the above algorithm to estimate $m_{(1-\epsilon)n}$ in time $n^{O(1/\epsilon)}$.
(ii) Show how to use the above algorithm to find a matching of size at least $(1-\epsilon)k^*$ in any given graph, where k^* is the (unknown) size of a maximum matching in the graph. Your algorithm should run in time $n^{O(1/\epsilon)}$.

Let us now focus on the important problem of counting *perfect* matchings, which as we have noted is equivalent to computing the permanent of a 0-1 matrix. We will assume that our graph has $|V| = 2n$ vertices, so the size of a perfect matching is n. Our goal is thus to compute m_n. The above algorithm will run in polynomial time provided the ratio m_{n-1}/m_n is polynomially bounded. This property happens to hold for many interesting classes of graphs, including, for example, dense graphs (all vertex degrees are at least $n/2$), random graphs (in the $G_{n,1/2}$ model, with high probability), and regular lattices (e.g., finite square regions of the grid Z^2, which are important in physical applications such as the dimer model). However, unfortunately it is not too hard to construct examples of graphs where this ratio grows exponentially with n.

The figure below shows such an example. The graph consists of ℓ squares connected in a line (so the number of vertices is $2n = 4\ell + 2$). There is a single perfect matching, but the number of "near-perfect matchings" (i.e., those of size $n-1$) is at least $2^\ell = 2^{(n-1)/2}$ (corresponding to leaving the two endpoints unmatched and having two choices on each square). Thus $m_{n-1}/m_n \geq 2^{(n-1)/2}$.

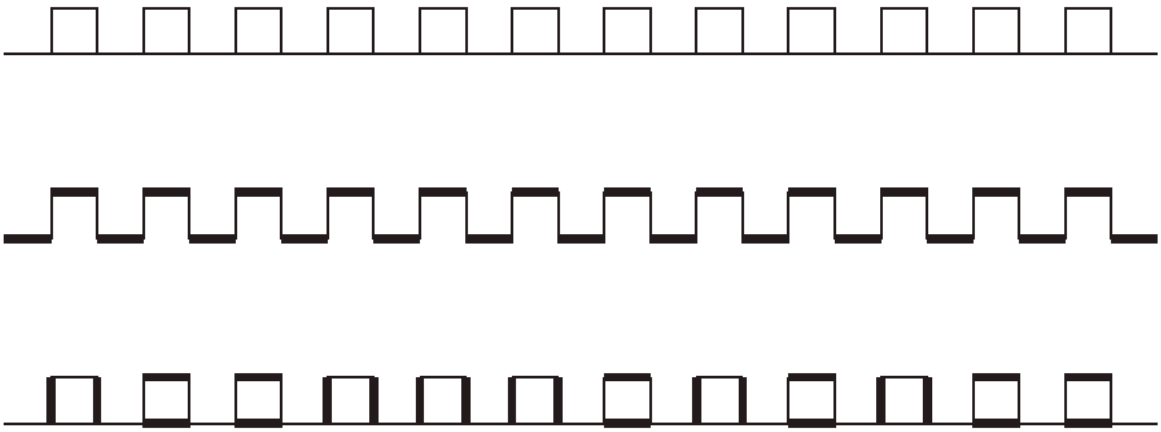


Figure 13.1: The first row shows a graph; the second row shows its unique perfect matching; the third row shows one of many near-perfect matchings

13.2 Perfect Matchings in an Arbitrary Graph / Permanent of Arbitrary Positive Matrix

How can we handle graphs in which the ratio m_{n-1}/m_n of near-perfect matchings to perfect matchings is very large? This was done in [JSV04] by an extension of the above MCMC approach for all matchings. The key new idea is to reweight the matchings in such a way that the perfect matchings have large enough weight in the stationary distribution; the problem is that the required weights are very hard to compute, but we can get the MC itself to succcessively *learn* the correct weights.

Our starting point is a slightly different MC for sampling k-matchings to the one above (whose states were all the matchings, of all sizes). This MC has as states only the k- and $(k-1)$-matchings, and its stationary distribution is uniform. The transitions (edge additions, deletions and exchanges) are the same as for our previous chain. By an analysis very similar to the one we used for the all-matchings MC, it can be shown that the mixing time for this MC is poly$(n, \frac{m_{k-1}}{m_k})$. Since we are interested in perfect matchings, we will focus on the chain whose states are perfect and near-perfect matchings (i.e., $k = n$).

We partition the set of near-perfect matchings into sets $\{N(u,v)\}$, where $N(u,v)$ denotes the set of near-perfect matchings with "holes" (unmatched vertices) at u and v. Let P be the set of perfect matchings. A typical distribution of $\{|N(u,v)|\}$ and $|P|$ (with all matchings equally weighted) is sketched on the left side of Figure 13.2. Note that in a "bad" graph at least one of the $|N(u,v)|$ is exponentially larger than $|P|$.

Now assign to each matching M in the graph a weight $w(M)$ defined as:

$$w(M) = \begin{cases} w(u,v) := \frac{|P|}{|N(u,v)|} & \text{if } M \in N(u,v); \\ 1 & \text{if } M \in P; \\ 0 & \text{otherwise.} \end{cases}$$

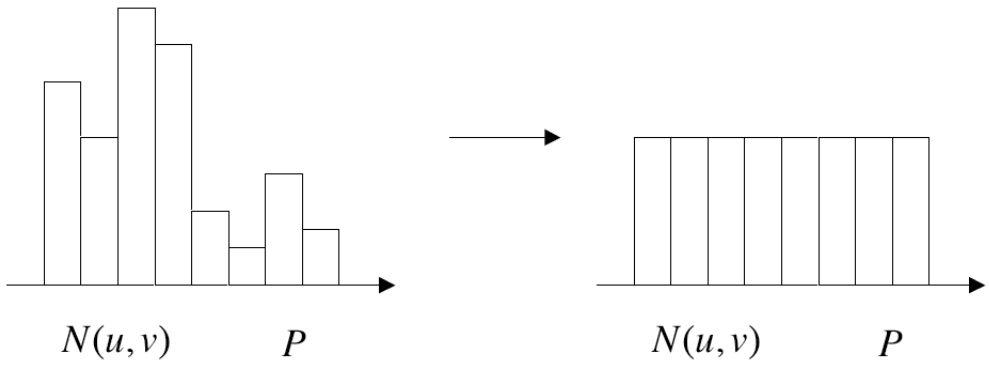


Figure 13.2: The histogram of $\{|N(u,v)|\}$ and $|P|$ before and after reweighting.

These weights are chosen precisely so that the weight of each of the sets $N(u,v)$ is the same, and equal to that of the perfect matchings P. In other words, we have reweighted the distribution so that it looks like the right-hand side of Figure 13.2. We can easily modify our Markov chain so that it has this distribution as its stationary distribution (i.e., $\pi(M) \propto w(M)$) by using the Metropolis rule (i.e., transitions are accepted with probability proportional to the ratio of the weights of the two matchings). This Markov chain has two nice properties: (i) it allocates total weight at least $\frac{\text{const}}{n^2}$ to the set of perfect matchings P (because there are at most n^2 different possible hole patterns (u,v)); and (ii) the mixing time is poly(n) (this follows by a flow argument similar to the one we used for the all-matchings chain but a bit more technical; the weights on the matchings have the effect of cancelling the factor m_{n-1}/m_n from the mixing time).

So we have a rapidly mixing Markov chain that samples perfect matchings u.a.r. with good probability—and it seems that we are done! But there is a catch. In order to implement the above algorithm we need to know the weights $w(u,v)$ (for the Metropolis rule). But these are defined as ratios of the form $\frac{|P|}{|N(u,v)|}$, which involve quantities like $|P|$ which is what we are trying to compute in the first place! So how can we get hold of the weights?

The final trick here is to introduce edge activities λ_e, so that the weights also depend on these. We start off with trivial activities $\lambda_e = 1$ for *all possible* edges (even those that are not in G). For these trivial activities, the weights $w(u,v)$ are easy to compute. Then we gradually reduce the activities of the non-edges until they become negligible. As we do this, we are able to *learn* the weights $w(u,v)$ for the new activities using observations from the MC itself!

Here is a summary of the procedure:

- Introduce edge activities λ_e for all $e \in |V| \times |V|$, and define:

$$\lambda(M) := \prod_{e \in M} \lambda_e; \quad \lambda(P) := \sum_{M \in P} \lambda(M); \quad \lambda(N(u,v)) := \sum_{M \in N(u,v)} \lambda(M).$$

  Here $\lambda(M)$ is the activity of a matching, and $\lambda(P)$, $\lambda(N(u,v))$ are just weighted versions of the cardinalities $|P|$ and $|N(u,v)|$ from before.

- Redefine the hole weights as $w_\lambda(u,v) := \frac{\lambda(P)}{\lambda(N(u,v))}$. This introduction of activities λ has essentially no effect on the mixing time analysis, so the Markov chain still mixes in polynomial time. The stationary distribution becomes $\pi_\lambda(M) \propto \lambda(M) w_\lambda(u,v)$.

- Start with $\lambda_e = 1$ for all edges e (including "non-edges" $e \notin E(G)$). In other words, we start out with the complete bipartite graph $K_{n,n}$. For this graph, computing the hole weights $w_\lambda(u, v)$ is trivial.
- Gradually reduce the activities λ_e of non-edges $e \notin E(G)$ until $\lambda_e \ll 1/n!$; at this point the stationary weight of any matching containing a non-edge is negligible, so we are effectively just sampling perfect and near-perfect matchings in the graph G. Since the activities of all the edges in G are 1, the stationary distribution is essentially the same as in the right-hand side of Figure 13.2. We reduce the activities in stages: at each stage we reduce λ_e for just one of the non-edges e by a factor of 2. The number of stages we need is thus about $O(n^3 \log n)$.
- Assume inductively that we have the correct hole weights for the current set of activities $\{\lambda_e\}$. Now we reduce one of the λ_e, so the hole weights are no longer correct. However, it is not hard to see that reducing a single λ_e by a factor of 2 can affect each of the hole weights by at most a factor of 2 as well. So we can continue to use our old hole weights with the new activities. The constant factor error in hole weights just means that the stationary distribution will be slightly biased (by at most a constant factor), and that the mixing time may increase (again by at most a constant factor). So we still have a rapidly mixing Markov chain for the new activities. Now the crucial point is that, by taking samples from the stationary distribution of this Markov chain, we can easily estimate the *correct* hole weights for the new activities. Hence we can get very accurate estimates of these, and we are back in the situation we started in but with the new activities.

References

[1] M. Jerrum and A. Sinclair, "Approximating the permanent." *SIAM Journal on Computing* **18** (1989), pp. 1149–1178.

[2] M. Jerrum, A. Sinclair and E. Vigoda, "A polynomial-time approximation algorithm for the permanent of a matrix with non-negative entries." *Journal of the ACM* **51** (2004), pp. 671–697.

[V79a] L.G. Valiant, "The complexity of computing the permanent," *Theoretical Computer Science* **8**(1979), pp. 189–201.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 14: October 22

Lecturer: Alistair Sinclair *Scribes: Alistair Sinclair*

14.1 Relating mixing times using flows

14.1.1 Theory

This section describes a flow-based method for bounding the Poincaré constant (or, in the reversible case, the spectral gap) of a Markov chain based on that of another Markov chain, due to Diaconis and Saloff-Coste [DS93]. Assume we are given two ergodic, lazy Markov chains P and $\tilde{P}$ which share the same stationary distribution π over some state space Ω. Furthermore, assume that we have already proved a lower bound on the Poincaré constant $\tilde{\alpha}$ of $\tilde{P}$ by some other method. By constructing a flow on P that "simulates" the transitions of $\tilde{P}$, we can bound the ratio of the Poincaré constant of P to that of $\tilde{P}$ by a constant determined by the characteristics of the flow.

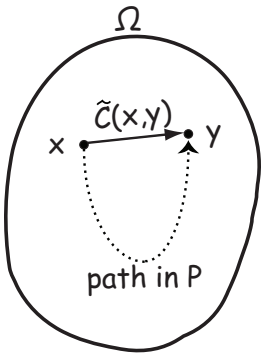


Figure 14.1: A $(P, \tilde{P})$-flow

Definition 14.1 *Let $\mathcal{Q}_{xy}$ denote the set of all simple $x \rightsquigarrow y$ paths in P, and $\mathcal{Q} = \cup_{x,y} \mathcal{Q}_{xy}$. A $(P, \tilde{P})$-*flow *is a function $f : \mathcal{Q} \to \mathbb{R}^+ \cup \{0\}$ subject to demand constraints*

$$\forall x, y : \quad \sum_{p \in \mathcal{Q}_{xy}} f(p) = \tilde{C}(x, y),$$

where $\tilde{C}(x, y) = \pi(x)\tilde{P}(x, y)$ (the capacity of edge (x, y) in $\tilde{P}$).

Recall that we define the cost of a flow f as

$$\rho(f) = \max_e \frac{f(e)}{C(e)}$$

and the length of a flow as

$$\ell(f) = \max_{p:f(p)>0} |p|.$$

Claim 14.2 *For any $(P, \tilde{P})$ flow f and function $\varphi : \Omega \to \mathbb{R}$, we have*

$$\sum_{x,y} (\varphi(x) - \varphi(y))^2 C(x,y) \geq \frac{1}{\rho(f)\ell(f)} \sum_{x,y} (\varphi(x) - \varphi(y))^2 \tilde{C}(x,y).$$

The proof of this fact follows that of Theorem 10.6 in Lecture 10; it involves expanding out $\tilde{C}(x,y)$, applying the Cauchy-Schwartz inequality, and summing over paths.

Exercise 14.3 *Prove Claim 14.2 via pattern matching on the proof of Theorem 10.6.*

Theorem 14.4 *For any $(P, \tilde{P})$-flow f,*

$$\frac{\alpha}{\tilde{\alpha}} \geq \frac{1}{\rho(f)\ell(f)}$$

Proof: Recall from Lecture 10 that

$$\alpha = \inf_{\varphi} \frac{\sum_{x,y} (\varphi(x) - \varphi(y))^2 C(x,y)}{\sum_{x,y} (\varphi(x) - \varphi(y))^2 \pi(x)\pi(y)},$$

where the inf is taken over non-constant functions $\varphi : \Omega \to \mathbb{R}$. Thus, we can conclude

$$\begin{aligned}
\frac{\alpha}{\tilde{\alpha}} &= \frac{\inf_{\varphi} \frac{\sum_{x,y}(\varphi(x)-\varphi(y))^2 C(x,y)}{\sum_{x,y}(\varphi(x)-\varphi(y))^2 \pi(x)\pi(y)}}{\inf_{\varphi} \frac{\sum_{x,y}(\varphi(x)-\varphi(y))^2 \tilde{C}(x,y)}{\sum_{x,y}(\varphi(x)-\varphi(y))^2 \pi(x)\pi(y)}} \\
&\geq \inf_{\varphi} \frac{\sum_{x,y} (\varphi(x) - \varphi(y))^2 C(x,y)}{\sum_{x,y} (\varphi(x) - \varphi(y))^2 \tilde{C}(x,y)} \\
&\geq \frac{1}{\rho(f)\ell(f)}.
\end{aligned}$$

■

Note that the original flow theorem from Lecture 10 can be seen as a special case of Theorem 14.4 in which $\tilde{P}$ is the trivial Markov chain with $\tilde{P}(x,y) = \pi(y)$, which mixes in one step and has a Poincaré constant of 1. Theorem 14.4 can be very useful, because other methods of analysis are often rather sensitive to the details of the Markov chain being analyzed; the theorem allows us to bootstrap the analysis of different, perhaps more complex Markov chains by relating their convergence to that of simpler versions that we can analyze directly.

14.1.2 Example

In this section, the technique just presented will be used to prove the convergence properties of the "vanilla" Markov chain on multi-path routings without tower moves (from Lecture 8).

Recall that there is a one-to-one correspondence between "lozenge tilings" of regular hexagons and "routings" on the rectangular lattice where each point belongs to at most one path (see Figure 14.2). Thus, a Markov

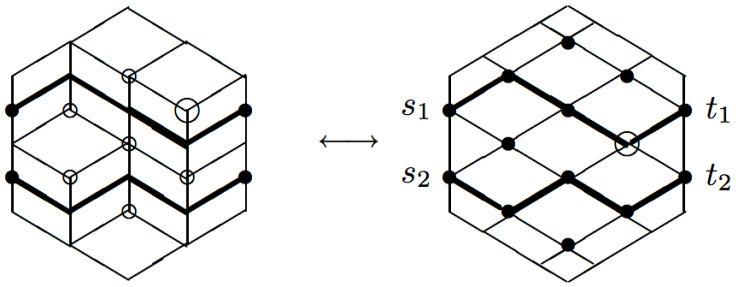


Figure 14.2: Lozenge tiling ≡ routing (reprint of Figure 8.3)

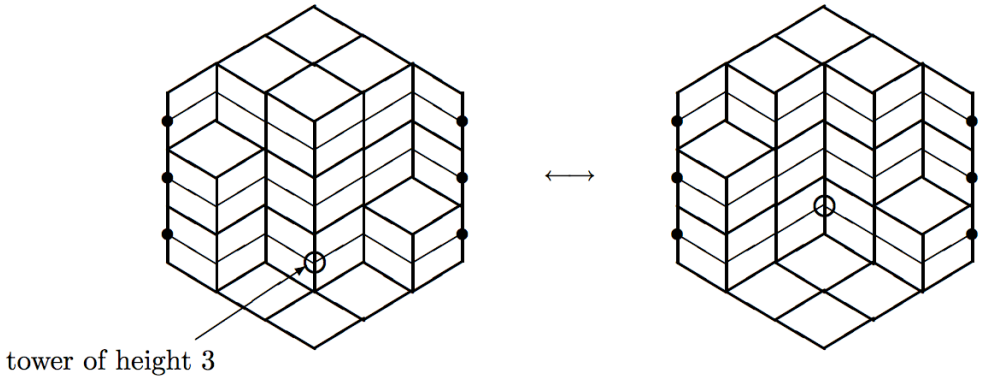


Figure 14.3: Tower rotations (reprint of Figure 8.5)

chain with a stationary distribution that is uniform on such routings can be used to produce uniform samples of lozenge tilings.

In Lecture 8, a "vanilla" Markov chain on multi-path routings was first presented: choose a point p along some path and a direction $d \in \{\uparrow, \downarrow\}$ uniformly at random, and, if possible, move p in direction d. $\uparrow/\downarrow$ moves are possible only for "valleys"/"peaks," and only when the move would not be blocked by another path. This Markov chain was shown to be irreducible, aperiodic and lazy, but its convergence time was difficult to analyze for multi-path routings due to the possibility of blocked moves.

Thus, a related Markov chain with "tower rotations" was introduced: choose a point p and direction d as before, and then rotate the *tower* at p in direction d if possible (i.e., if d is $\downarrow$ and p is a peak or d is $\uparrow$ and p is a valley) with probability $1/2h$, where h is the height of the tower. Because no moves are ever blocked by other paths in this Markov chain, we found it easier to analyze and proved its mixing time to be $O(n^4)$, where n is the total number of points in the lattice. (A more involved analysis by Wilson improves this to $O(n^3)$, which is tight.) However, the analysis of this modified chain told us nothing about the convergence time of the original vanilla Markov chain.

In this section, we will show how to use the technique introduced in Section 14.1.1 to bound the mixing of the vanilla Markov chain in terms of the mixing time of the modified chain with tower moves. The application is fairly straightforward: P is the vanilla chain and $\tilde{P}$ is the chain with tower moves.

To compare P with $\widetilde{P}$, we must specify, for each tower move, a way to "implement" the move by one or more

sequences of simpler non-tower moves. In fact we will route all the flow for a tower move along a *single* path of non-tower moves in an obvious way: simply flip one level of the tower at a time, starting from the top in the case of a valley tower, and from the bottom in the case of a peak tower.

To analyze this flow, consider any edge e in P. This edge may lie on the paths corresponding to several tower moves; however, it should be clear that the number of tower moves of height h whose paths can pass through this edge e is at most h. Therefore we have

$$\begin{aligned} f(e) &\leq \sum_h h \times \frac{1}{2Nhn} && (14.1)\\ &\leq \frac{1}{2nN} \sum_h 1 && (14.2)\\ &\leq \frac{m}{2nN}, && (14.3) \end{aligned}$$

where m is the maximum height of any tower.

On the other hand, the capacity of the edge e is

$$C(e) = \frac{1}{N} \times \frac{1}{2n}. \tag{14.4}$$

Hence we have

$$\rho(f) = \max_e \frac{f(e)}{C(e)} \leq m.$$

Since $\ell(f)$ is also bounded by m, and m is bounded above (very crudely) by n, we may apply Theorem 14.4 to conclude that

$$\alpha \geq (\tilde{\alpha}) \times n^{-2} \tag{14.5}$$

Thus, the eigenvalue gap of the vanilla chain is at least an n^2 fraction of that of the enhanced chain. There is a small hitch: we proved the mixing time of the enhanced chain directly using coupling, and never proved a bound on its eigenvalue gap. Previously we showed that the mixing time of a chain is bounded by its eigenvalue gap (or Poincaré constant) via

$$\tau_{mix} \leq \frac{C}{\alpha} \log(\pi_{min}^{-1}).$$

We can also easily prove a bound in the opposite direction:

Claim 14.5 *$\alpha \geq \frac{C'}{\tau_{mix}}$ where C' is some other constant.*

Combining these facts, we can bound the mixing time of the vanilla chain:

$$\begin{aligned} \tau_{mix} &\leq \frac{C}{\alpha} \log(\pi_{min}^{-1})\\ &\leq \frac{Cn}{\alpha}\\ &\leq \frac{Cn^3}{1-\tilde{\alpha}}\\ &\leq C''\tilde{\tau}_{mix}n^3\\ &= \mathrm{O}(n^3)n^3\\ &= \mathrm{O}(n^6). \end{aligned}$$

Note that this analysis was rather loose, and may have introduced several unnecessary factors of n. (Indeed, it is conjectured that the true mixing time of this chain is $O(n^4)$, or even $\tilde{O}(n^3)$.) Notwithstanding, it provided us with a polynomial bound on the mixing time of the vanilla chain, which we were not able to prove directly using coupling.

14.2 Approximate counting

14.2.1 The class $\#P$

Definition 14.6 *$\#P$ is the class of functions $f : \Sigma^* \to \mathbb{N}$ such that there exists a polynomial-time nondeterministic Turing Machine M which, on input x, has exactly $f(x)$ accepting computation paths.*

This is a natural analog of the class NP of search or decision problems, with typical elements of $\#P$ being counting versions of NP search problems: the canonical example is $\#SAT$, which counts the number of satisfying assignments of a boolean formula in CNF. Most natural such problems are $\#P$-complete. Interestingly, this applies not only to counting versions of NP-complete problems, such as $\#SAT$ or $\#COL$ (counting graph colorings), but also to several important problems whose decision version is in P, such as $\#MATCHINGS$ (counting the number of matchings in a graph), or $\#DNFSAT$ (counting the number of satisfying assignments of a DNF formula).

Given this state of affairs, much attention has focused on the design of efficient *approximation algorithms* for problems in $\#P$. We make the following definition:

Definition 14.7 *Let f be a function in $\#P$. A* fully polynomial randomized approximation scheme (fpras) *for f is a randomized algorithm that on input (x, ε), where $x \in \Sigma^*$ and $\varepsilon \in (0, 1]$, outputs a random variable Z such that $\Pr[f(x)(1-\varepsilon) \le Z \le f(x)(1+\varepsilon)] \ge \frac{3}{4}$, and runs in time polynomial in x and ε^{-1}.*

The $\frac{3}{4}$ constant above is chosen for algebraic convenience only:

Claim 14.8 *If there exists an fpras for f, we can boost the confidence from $\frac{3}{4}$ to $1-\delta$ at the cost of a slowdown by a factor of $O(\log \delta^{-1})$.*

Proof: Take $t = O(\log \delta^{-1})$ independent trials of the the fpras and output the *median* of the results. The median falls outside $[f(x)(1-\varepsilon), f(x)(1+\varepsilon)]$ only if at least $t/2$ trials land outside that range, and the probability of this is less than the probability that a coin with $\frac{3}{4}$ probability of landing "heads" comes up heads less than $t/2$ times in t tosses. By a standard Chernoff bound, this probability is bounded by e^{-ct} for a constant c, and thus by δ when $t = O(\log \delta^{-1})$. ■

We have mentioned several times in this course that random sampling (obtained, e.g., by MCMC) can be used to perform approximate counting. In the next section we spell out this connection in more detail for a representative example, namely counting the colorings of a graph.

14.2.2 From sampling to approximate counting: graph coloring

Let us consider the problem of approximately counting q-colorings of a graph $G = (V, E)$. We will restrict our attention to the case $q \ge 2\Delta + 1$, where Δ is the maximum degree of the graph, for which we know that we have an MCMC algorithm to sample colorings uniformly at random with mixing time $O(n \log n)$.

Consider the sequence of graphs $G_0 = (V, E_0), G_1 = (V, E_1), \ldots, G_m = G$, where $m=|E|$, $E_0 = \emptyset$, and each step in the sequence adds one of E's edges, $E_i = E_{i-1} \cup \{e_i\}$.

Let $\Omega(G)$ be the set of all colorings of a graph G. We can represent $|\Omega(G)|$ by a telescoping product:

$$|\Omega(G)| = |\Omega(G_0)| \prod_{i=1}^{m} \frac{|\Omega(G_i)|}{|\Omega(G_{i-1})|}.$$

Since G_0 has no edges, $|\Omega(G_0)| = q^n$ (where $n = |V|$).

We estimate $|\Omega(G_i)| \,/\, |\Omega(G_{i-1})|$ by sampling colorings of G_{i-1} and outputting the proportion of samples which give different colors to the endpoints of e_i, which correspond precisely to the proper colorings of G_i.

To estimate the required sample size, consider the injective, multivalued map μ from the set $\Omega(G_{i-1})\backslash\Omega(G_i)$ of G_{i-1} colorings which aren't proper G_i colorings to the set of proper G_i colorings $\Omega(G_i)$. Define this map to take the lexicographically first endpoint u_i of e_i and recolor it with any color allowed for it in G_i. Since the degree of u_i in G_i is at most Δ, there are at least $q - \Delta \geq \Delta + 1$ such colorings for each coloring in $\Omega(G_{i-1})\backslash\Omega(G_i)$. The map is injective since the original coloring χ in $\Omega(G_{i-1})\backslash\Omega(G_i)$ can be obtained from a coloring $\chi' \in \mu(\chi)$ by changing u_i's color in χ' to the color of the other endpoint of e_i in χ'. We thus have (for $\Delta > 1$; the problem is trivial for graphs with $\Delta \leq 1$):

$$\begin{aligned} |\Omega(G_i)| &\geq (\Delta+1)\,|\Omega(G_{i-1})\backslash\Omega(G_i)| \\ &\geq (\Delta+1)(|\Omega(G_{i-1})| - |\Omega(G_i)|); \\ \frac{|\Omega(G_i)|}{|\Omega(G_{i-1})|} &\geq \frac{\Delta+1}{\Delta+2} \geq \frac{3}{4}. \end{aligned}$$

A standard variance analysis (via Chebyshev's inequality) shows that taking $t \approx \frac{16m^3}{3\varepsilon^2}$ ensures that each ratio in the telescoping product is within $1 \pm \frac{\varepsilon}{2m}$ of its true value with probability at least $1 - \frac{1}{4m}$. (This assumes that the MCMC yields an exactly uniform distribution over colorings, but the deviation from uniformity is of lower order and can easily be absorbed into the above bound.) Since there are m factors in the product, we see that $O(\frac{m^4}{\varepsilon^2})$ samples are sufficient to ensure that the product estimates $|\Omega(G)|$ within ratio $1 \pm \varepsilon$ with probability at least $\frac{3}{4}$ (the confidence follows by a union bound, and the accuracy from the fact that $(1 \pm \frac{\varepsilon}{2m})^m \in [1-\varepsilon, 1+\varepsilon]$). Since each sample takes $O(n \log n)$ time to generate, the overall running time of the algorithm is $O(m^4 \varepsilon^{-2} n \log n)$, which is polynomial in n and ε^{-1}, as required for an fpras.

Exercise 14.9 *Verify the variance analysis in the previous paragraph.*

Exercise 14.10 *By analyzing the variance of the product estimator itself (rather than of each term of the product separately as we did above), show that the number of samples required can be reduced by a factor of m^2.*

References

[DS93] P. Diaconis and L. Saloff-Coste, "Comparison theorems for reversible Markov chains," *Annals of Applied Probability* 3 (1993), pp. 696–730.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 15: October 27

Lecturer: Alistair Sinclair *Scribes:*

15.1 Introduction

This lecture further develops the connection between random sampling and approximate counting. Informally, the two main results we will prove in this lecture are as follows:

1. Random sampling is "equivalent" to approximate counting.
2. A counting problem either has a very good approximation algorithm or it has none at all (within any polynomial factor).

Recall a counting problem is a function $f : \Sigma^* \mapsto \mathbb{N}$ that maps input (strings) to the number of solutions. For example, in 3-SAT, the input is a CNF formula and the output is the number of satisfying assignments. Almost all interesting counting problems belong to the class #P, which means that there is a polynomial time nondeterministic Turing machine with the property that, for all x, $f(x)$ is the number of accepting computations of the machine on input x.

A *fully polynomial randomized approximation scheme* (f.p.r.a.s.) for f is a randomized algorithm which, on input $(x, \epsilon) \in \Sigma^* \times (0, 1]$, outputs a random variable Z such that

$$\Pr[(1+\epsilon)^{-1} f(x) \leq Z \leq (1+\epsilon) f(x)] \geq 3/4$$

and runs in time $\text{poly}(|x|, \epsilon^{-1})$.

If Z is in the range $[(1+\epsilon)^{-1} f(x), (1+\epsilon) f(x)]$, we say that Z estimates $f(x)$ *within ratio* $(1+\epsilon)$. We choose to work with this definition rather than $(1-\epsilon) f(x) \leq Z \leq (1+\epsilon) f(x)$ (which is almost equivalent for small ϵ)because it makes the algebra a little cleaner, and also still makes sense when ϵ is larger than 1 (as it will be in Section 15.3).

Note that the estimate could be arbitrarily bad with probability 1/4. However, this probability can be reduced to any desired $\delta > 0$ by performing $O(\log \delta^{-1})$ trials and taking the median.

A f.p.r.a.s. is considered a very good algorithm, since most interesting counting problems we care about are #P-complete, so a polynomial time exact algorithm is impossible (unless P = NP). A f.p.r.a.s. is essentially the best approximation one can hope for for such problems. Note also that we cannot even hope for a f.p.r.a.s. for counting versions of NP-complete problems, because a f.p.r.a.s. always allows us to tell the difference between 0 and non-zero w.h.p., which would allow us to solve the decision problem (w.h.p.) in polynomial time. Hence interest focuses on those counting problems that are #P-complete but whose decision version is in P. There are many such examples, including: counting matchings (monomer-dimer systems), counting perfect matchings (dimer systems), computing the partition function of the Ising model, computing the volume of a convex body, counting independent sets (the hard core model), counting bases of a matroid, counting colorings of a graph with $q \geq \Delta + 1$ colors, etc.

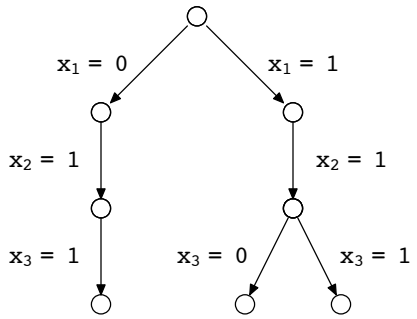


Figure 15.1: An example of a self-reducibility tree for the SAT instance $(\neg x_1 \vee x_2) \wedge (x_1 \vee x_2) \wedge (x_1 \vee x_3)$.

15.2 Equivalence between counting and sampling

Last time, we showed how to count the number of colorings of a graph G by reducing the problem to sampling colorings of G, thus obtaining a f.p.r.a.s. by invoking sampling. Although a similar approach can be used for a number of other problems, this technique is rather problem-specific. Today, we will give a *generic* reduction from counting to sampling and vice-versa, which is captured in the following theorem:

Theorem 15.1 *For all* self-reducible *NP problems, there exists a f.p.r.a.s. for counting iff there exists a polynomial time algorithm for (almost) uniform sampling.*

Rather than giving a formal definition of self-reducibility, we illustrate it with an example:

Example (SAT) Consider a boolean formula, for example,

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

We can represent the solutions to this SAT instance by a *self-reducibility tree*, as show in Figure 15.1. Each root-to-leaf path in this tree is a satisfying assignment. Along the path, the edges leaving the i-th internal node correspond to the assignments of variable i that eventually lead to complete satisfying assignments. Note that one or two outgoing edges are both possible, but not zero, because we would not have reached that node in the first place. Another way to see the tree is to consider it as the set of all accepting paths (witnesses) of a nondeterministic Turing machine that solves the decision version of the SAT problem.

The tree is of depth m, where m is the number of variables in the SAT formula. Each subtree at level i corresponds to a smaller SAT instance, which is the original instance but with the variables $x_1, \cdots, x_i$ already set to certain values. This recursiveness is exactly the self-reducibility property.

More generally, for an NP search problem, self-reducibility means that the set of witnesses/solutions can be partitioned into a polynomial number of sets, each of which is in 1-1 correspondence with the set of solutions of a smaller instance. The number of partitions is the maximum branching factor of the tree, which in our example is two, but more generally could be any polynomial in size of the instance. Note that the tree has polynomial depth because the size of the instance decreases at each level of the tree. We may assume w.l.o.g. that all leaves are at the same depth. The number of leaves is exactly the number of solutions, i.e., the function value $f(x)$.

Proof: We will now prove Theorem 15.1, dealing with each direction separately.

⇐: Our goal is to show how to use an algorithm for random sampling in order to construct a f.p.r.a.s. For simplicity, we will assume that we can sample exactly uniformly although the theorem requires only sampling almost uniformly. As will be evident from the proof, this extra error can easily be absorbed into the error of the f.p.r.a.s. We will also assume for simplicity that the tree has branching factor 2.

Assume that we can draw perfect samples in polynomial time. We will now give a polynomial time algorithm to approximately count. The idea is to work with the self-reducibility tree. Recall that the number of solutions is the number of leaves.

In the first stage of the algorithm, we sample some leaves uniformly at random. Using these samples, we can get an estimate $\hat{r}$ of the following quantity:

$$r = \frac{\text{number of leaves in left subtree}}{\text{total number of leaves}}.$$

Without loss of generality, assume that the left subtree contains at least half of the leaves. Now, recursively estimate number of leaves in the left subtree; call this estimate $\hat{N}_1$. The key observation is that, by virtue of self-reducibility, this is just a smaller instance of the same problem, so counting the number of leaves in this subtree can be handled by invoking the same approximate counting procedure on the smaller instance. Eventually, the recursion bottoms out at a leaf, at which point the algorithm returns 1. Our final estimate of the total number of leaves is $\hat{N} = \frac{\hat{N}_1}{\hat{r}}$.

If m is the depth of the tree, a standard second moment calculation (as sketched in the previous lecture) tells us that the number of samples needed to ensure the final estimate is within ratio $(1+\epsilon)$ with probability $3/4$ is $O(m\epsilon^{-2})$ per level, or $O(m^2\epsilon^{-2})$ total. Since by assumption each sample is produced in polynomial time, we have a f.p.r.a.s.

⇒: Assume that we have a f.p.r.a.s. for counting. We will give an algorithm for almost uniform sampling. This direction is even easier than the other direction. The idea is to simply incrementally construct a path going down the tree.

At each node, we use the f.p.r.a.s. to estimate the number of leaves in the left and right subtrees, $\hat{N}_l$ and $\hat{N}_r$, respectively. Now we branch left with probability $\frac{\hat{N}_l}{\hat{N}_l+\hat{N}_r}$, right otherwise, and repeat.

To ensure a variation distance at most ϵ from uniform in the distribution on the leaves, we need the error in each invocation of the f.p.r.a.s. to be within ratio $(1+\epsilon/2m)$, where m is the number of levels in the tree; for this gives an accumulated bias at most $(1+\epsilon/2m)^{2m} \leq \epsilon$ at any leaf.

We thus invoke the f.p.r.a.s. $O(m)$ times, each time requesting an approximation within ratio $(1+\epsilon/2m)$. Since the running time of the f.p.r.a.s. scales only polynomially with the inverse error, the cost of each invocation is polynomial in m and $\log(m/\epsilon)$. Hence the entire sampling algorithm is polynomial in m and $1/\epsilon$. ■

Rejection sampling Note that, in the second implication proved above, the resulting random sampling algorithm had a running time that depended polynomially on ϵ^{-1}. We can do better than this using the idea of *rejection sampling*. Suppose we have an algorithm that runs in time polynomial in m and counts solutions within ratio $(1+1/m)$. This is certainly the case if we have a f.p.r.a.s., since we can just set its error parameter to $\epsilon = 1/m$.

Now apply the above algorithm, branching down the tree with probabilities proportional to these counting estimates. Suppose we reach leaf ℓ. The probability, p_ℓ, of reaching ℓ will deviate from its ideal value $\frac{1}{N}$ (where N is the total number of leaves) by a factor of at most $(1+1/m)^{2m} \leq e^2$. I.e., the error is bounded

by a constant factor. (We will use this fact below.) Another crucial point is that we can compute p_ℓ, simply by multiplying the branching probabilities along the path we took.

Once we have reached ℓ, we do the following. We output ℓ with probability $\frac{\alpha}{p_\ell}$, and we *fail* otherwise. Here α is a constant chosen so that $\alpha \leq p_\ell$ for all ℓ (so that the above value really is a probability). Note that this will ensure that each leaf is output with *exactly uniform* probability α. For α, we can take the value $\alpha = (1+1/m)^{-(2m+1)}\hat{N}^{-1}$; for then we have

$$p_\ell \geq \frac{1}{N}(1+1/m)^{-2m} \geq \frac{1}{\hat{N}}(1+1/m)^{-(2m+1)} = \alpha.$$

And the failure probability is given by

$$1 - N\alpha \leq 1 - \frac{N}{\hat{N}}(1+1/m)^{-(2m+1)} \leq 1 - (1+1/m)^{-(2m+2)} \leq 1 - c$$

for a constant $c > 0$. Since the failure probability in one trial is at most a constant, if we repeat until we output a leaf we get a perfectly uniform sample in $O(1)$ expected trials, each of which takes time polynomial in m. Moreover, if we repeat for a fixed number $O(\log \epsilon^{-1})$ of trials, and just output the final leaf if all of them fail, then we get a leaf within variation distance ϵ from uniform in time polynomial in m and $\log \epsilon^{-1}$.

Note that this approach breaks down if our counting estimates are within a ratio larger than $(1 + c/m)$ for constant c, since the accumulated error at the leaves is too large to be efficiently corrected by rejection sampling (the failure probability would be too large).

Note: We can also apply rejection sampling in the case where the f.p.r.a.s. may produce estimates that are not within ratio $(1 + \epsilon)$ with small probability δ. In that case the above analysis still holds except on an event of probability at most $m\delta$, so the bias in our sample is at most $m\delta$. Note that this can be made as small as we like by reducing δ by repeated trials of the f.p.r.a.s.; as we saw in the last lecture, the cost of this is $O(\log \delta^{-1})$.

15.3 All or nothing

In the following we show the surprising fact that a counting problem either has a very good approximation (in the strong sense of an f.p.r.a.s.), or cannot be approximated in any reasonable sense in polynomial time. Specifically, we prove the following theorem from [JS89]:

Theorem 15.2 *For a self-reducible problem, if there exists a polynomial time randomized algorithm for counting within a factor of* $(1 + \mathit{poly}(|x|))^{\pm 1}$*, then there exists a f.p.r.a.s.*

Note that this theorem says that, if we can approximately count colorings (say) in polynomial time within a factor of 1000, or even within a factor of n^{1000}, then we can get an f.p.r.a.s. for colorings!

Corollary 15.3 *For a self-reducible counting problem, one of the following two holds:*

1. *there exists a f.p.r.a.s.;*
2. *there does not exist a polynomial time approximation algorithm within any polynomial factor.*

This dichotomy between approximable and non-approximable is very different from the situation with optimization problems, for which many different degrees of approximability exist (e.g., approximation schemes $(1 \pm \epsilon$ for any $\epsilon)$; constant factor; logarithmic factor, polynomial factor etc.)

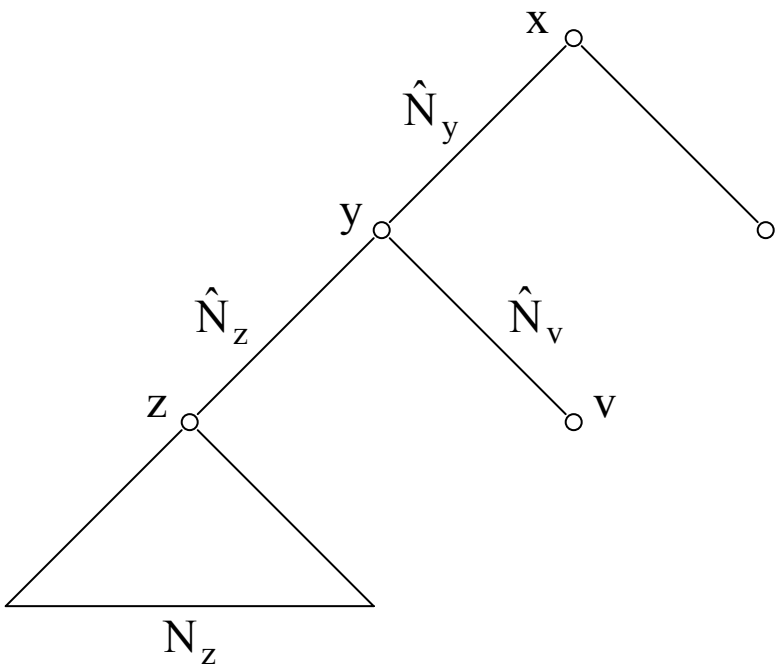


Figure 15.2: Part of a self-reducibility tree. $\hat{N}_z$ denotes the estimated number of leaves in the subtree rooted at z. The edges connected to a node y are weighted according to the sizes of subtrees as shown.

We now give a constructive proof of Theorem 15.2.

Proof: We again work with the self-reducibility tree. For any node z of the tree, let N_z denote the number of leaves in the subtree rooted at z. We assume we have an algorithm (a black box) that provides counting estimates within a factor of $\alpha = \text{poly}(n)$; i.e., for each z the algorithm outputs an estimate $\hat{N}_z$ that satisfies

$$\alpha^{-1} N_z \leq \hat{N}_z \leq \alpha N_z. \tag{15.1}$$

As will be clear from the proof below (**exercise**), we can allow Z to be a random variable, and we need only assume that (15.1) holds with high probability. However, for simplicity we just work with (15.1) deterministically.

Consider Figure 15.2. Assume the tree height is m. Weight each edge in the tree as $w_{zy} = \hat{N}_z$, where y is the parent of z; thus the weight of an edge is an estimate (provided by the counter) of the number of leaves in the subtree rooted at its lower end. Now consider the Markov chain defined by weighted random walk on the tree; i.e., at any given vertex, we choose an incident edge proportional to its weight and move along that edge. Note that both upward moves (to the parent) and downward moves (to a child) are allowed. We may assume w.l.o.g. that the weight on every leaf edge is exactly 1 (because these correspond to trivial instances of the counting problem, for which we can assume the f.p.r.a.s. has no error).

The intuition for this MC is the following. The downward moves correspond exactly to the vanilla process we defined earlier, in which we use counting estimates to guide our choice of a root-leaf path. The problem with that process, however, is that it requires the counting estimates to be within ratio about $(1 \pm 1/m)$; otherwise we can build up a large bias by the time we reach the leaves. (For example, if the counting estimates are off by a constant factor, we can get a bias at the leaves that is exponential in m.) This was not a problem before because we assumed we had an f.p.r.a.s. Now, however, we are allowed to assume only that our counting

estimates are within some (possibly very large) constant or even polynomial. This is handled by the upward moves of the walk: suppose the weight on some edge is too large (i.e., it is an overestimate of the number of leaves in the subtree below it). Then this very same edge will tend to pull the process back upward in the next step! In this sense the process is "self-correcting." Remarkably, this self-correction actually works in a rigorous sense, as we will now prove.

We will prove the following three claims about the above Markov chain:

1. The stationary distribution π is uniform over the leaves;
2. The total weight of the leaves in π is at least $\frac{\text{constant}}{\alpha m}$;
3. The mixing time is $\tau_{\text{mix}} = O(m^2\alpha^2 \log(\alpha m))$.

Note that the above three facts establish the theorem. Claim 1 gives us uniformity over solutions; claim 2 means that we only have to take expected $O(\alpha m)$ samples from the stationary distribution until we see a leaf (solution) (if the final state of the MC is not a leaf, we reject and start a new simulation); and claim 3 ensures that the time to generate a sample is polynomial. The overall expected running time of the random sampling algorithm will be $O(\alpha^3 m^3)$.

Let us start by observing that the stationary distribution of a weighted random walk is always proportional to the (weighted) vertex degrees, i.e., $\pi(x) = D(x)/D$, where $D(x) = \sum_{x\sim y} w_{xy}$ and $D = \sum_x D(x)$. Note that $D(x) = 1$ for every leaf x, which establishes claim 1.

Next, let us compute the total weight of the leaves in the stationary distribution:

$$\sum_{\text{leaves } x} \pi(x) = \frac{\#\ \textit{leaves}}{D} = \frac{N}{D}.$$

The denominator here is

$$D = \sum_x D(x) = 2 \times \text{sum of all edge weights} \le 2\alpha m N, \tag{15.2}$$

where the inequality follows from the fact that the sum of the edge weights on any given level of the tree approximates N (the total number of leaves) within a factor of α. Hence,

$$\sum_{\text{leaves } x} \pi(x) \ge \frac{N}{2\alpha m N} = \frac{1}{2\alpha m},$$

which establishes claim 2.

Finally, we bound the mixing time using flows arguments. Note that since the underlying graph of the MC is a tree, there is a unique simple path $x \rightsquigarrow y$ for any pair of vertices x, y. Hence our choice of flow here is forced. Considering a generic edge e with lower vertex y, the flow along e (in either direction) is given by

$$f(e) = \sum_{z\in T_y, v\notin T_y} \pi(z)\pi(v) = \pi(T_y)\pi(\overline{T}_y) \le \pi(T_y),$$

where T_y denotes the subtree rooted at y. Now we have

$$\pi(T_y) = \sum_{z\in T_y} \frac{D(z)}{D} \le \frac{1}{D} 2\alpha m N_y,$$

where the inequality follows as in equation (15.2) above. Hence

$$f(e) \leq \frac{1}{D} 2\alpha m N_y. \tag{15.3}$$

On the other hand, the capacity of edge e is

$$C(e) = \pi(y)\frac{\hat{N}_y}{D(y)} = \frac{D(y)}{D} \times \frac{\hat{N}_y}{D(y)} = \frac{\hat{N}_y}{D}. \tag{15.4}$$

Combining (15.3) and (15.4) we get

$$\frac{f(e)}{C(e)} \leq \frac{2\alpha m N_y}{\hat{N}_y} \leq 2\alpha^2 m,$$

where we used the fact that $\frac{N_y}{\hat{N}_y} \leq \alpha$. Recalling that $\rho = \max_e \frac{f(e)}{C(e)}$, the mixing time assuming we start at the root is

$$\tau_{\text{mix}} = O(\rho \ell \log \pi(\text{root})^{-1}),$$

where $\ell = 2m$ is the length of a longest flow-carrying path. But $\pi(\text{root}) = \frac{D(\text{root})}{D} \geq \frac{1}{2\alpha m}$, yielding a mixing time of $\tau_{\text{mix}} = O(\alpha^2 m^2 \log(\alpha m))$, which is claim 3. This concludes the proof. ■

Exercise. One might think that, rather than rejecting the sample and restarting if the final state is not a leaf, it is OK to simply wait until the first time after τ_{mix} that a leaf is hit and output that leaf. This would save a factor of $O(\alpha m)$ in the running time because of the waiting time until we get a leaf. Show that this is **not** OK because it may introduce bias into the leaf probabilities.

Exercise. The waiting time factor $O(\alpha m)$ mentioned in the previous exercise can in fact be saved, by modifying the bottom of the tree slightly (adding a suitable self-loop to each leaf). Show how to do this and argue that your method does not increase the bound on the mixing time.

References

[JS89] M.R. Jerrum and A.J. Sinclair. "Approximate Counting, Uniform Generation and Rapidly Mixing Markov Chains," *Information and Computation* **82** (1989), pp. 93–133.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 16: October 29

Lecturer: Alistair Sinclair *Scribes:*

Disclaimer: *These notes have not been subjected to the usual scrutiny reserved for formal publications. They may be distributed outside this class only with the permission of the Instructor.*

16.1 Volume of a Convex Body

We will discuss algorithms for estimating the volume of convex bodies. We'll try not to get bogged down in technical details, but there are some things we'll need to sift through. The problem definition is the following.

Input: A convex body $K \subseteq R^n$.

Goal: Compute the volume of K, $\mathbf{vol}_n(K)$.

We assume throughout that the body K is specified by a *membership oracle*, i.e., a black box which, given a point $x \in R^n$, replies whether or not $x \in K$.

The problem is known to be ♯P-hard [DF88, K93]. Moreover, it has been shown that there is no deterministic algorithm which approximates $\mathbf{vol}_n(K)$ within any "reasonable" factor [BF87]: for any deterministic sequence of oracle queries of length polynomial in n, there exist two different convex bodies consistent with the answers to the queries which have exponentially large volume differences. Nevertheless, as we shall see next, randomized algorithms allow us to approximate the volume of a convex body to arbitrary accuracy in polynomial time.

Claim 16.1 *There exists a f.p.r.a.s. for the volume problem in the oracle model.*

The first such algorithm was devised by Dyer, Frieze and Kannan [DFK91] and its running time was about $\tilde{O}(n^{26})$. (The notation $\tilde{O}$ indicates that factors of polylog(n) are omitted, in addition to constant factors.) A long sequence of subsequent improvements (mainly in the mixing time analysis, often involving major conceptual and technical advances) by various combinations of these three authors together with Lovász, Simonovits [LS93] and others, led to the current state of the art algorithm due to Lovász and Vempala [LV03] with a running time of $\tilde{O}(n^4)$.

The basic idea of the algorithm is to select an increasing sequence of concentric balls $B_0 \subseteq B_1 \subseteq \ldots \subseteq B_r$, such that $B_0 \subseteq K \subseteq B_r$, where, without loss of generality, B_0 can be assumed to be the unit ball and B_r to have radius of $O(\sqrt{n})$. (The fact that K can always be "rounded" in this way is a standard but non-trivial geometric fact that we will not prove here.)

Note that, since $\frac{\mathbf{vol}B_r}{\mathbf{vol}B_0}$ is exponential in n, a naive Monte-Carlo algorithm which samples from B_r and tests whether the sampled point belongs to K does not in general give an efficient algorithm.

Let us, however, express the volume of K as follows:

$$\mathbf{vol}(K) = \mathbf{vol}(K \cap B_r) = \frac{\mathbf{vol}(K \cap B_r)}{\mathbf{vol}(K \cap B_{r-1})} \times \ldots \times \frac{\mathbf{vol}(K \cap B_1)}{\mathbf{vol}(K \cap B_0)} \times \mathbf{vol}(K \cap B_0).$$

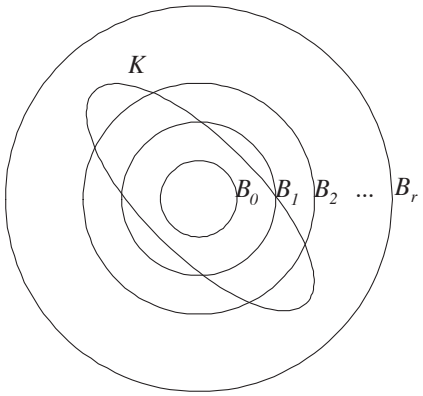


Figure 16.1: Sequence of concentric balls used in volume algorithm

Now note that $\mathbf{vol}(K \cap B_0) = \mathbf{vol}(B_0)$ is trivial, and that each ratio $\frac{\mathbf{vol}(K \cap B_i)}{\mathbf{vol}(K \cap B_{i-1})}$ can be estimated by sampling u.a.r. from $K \cap B_i$.

If we set $\frac{\mathbf{rad}(B_i)}{\mathbf{rad}(B_{i-1})} = 1 + \frac{1}{n}$ for each i, where $\mathbf{rad}(B)$ denotes the radius of B, it follows that $\frac{\mathbf{vol}(K \cap B_i)}{\mathbf{vol}(K \cap B_{i-1})} = (1 + \frac{1}{n})^n \leq \mathrm{e}$ (a constant). Thus each ratio can be estimated accurately (say, within ratio $(1 + \frac{\epsilon}{r})$) using only $O(r^2 \epsilon^{-2})$ random samples. And, since $\mathbf{rad}(B_r) = O(\sqrt{n})$, the number of balls we require is only $r = O(n \log n)$. Thus we deduce the following:

$\exists$ a f.p.r.a.s. for estimating $\mathbf{vol}(K)$ given a black box for sampling (almost) u.a.r. from $K \cap B_i$.

Note that $K \cap B_i$, being the intersection of two convex bodies, is itself convex. Thus we have a polynomial time reduction from computing the volume of a convex body to sampling u.a.r. in a convex body.

In the remainder of this lecture and next, we focus on algorithms for sampling from a convex body. This is the main content of work in this area.

16.1.1 Random Sampling

We consider two variants of a basic random-walk based sampling algorithm. Let $B(x, \delta)$ denote the ball of radius δ centered at x.

Heat-Bath Ball Walk: at point $x \in K$, move to a point in $B(x, \delta) \cap K$ u.a.r.

Metropolis Ball Walk: at point $x \in K$, pick $y \in B(x, \delta)$ u.a.r.;
if $y \in K$ then move to y, else stay at x.

Note that the heat-bath version of the ball walk is not easily implementable, since at every step it requires uniform sampling from the intersection of K and a ball around the current point. Moreover, as we shall soon see, it does not result in a uniform stationary distribution, since there is higher chance of moving away from the boundary. On the other hand, the Metropolis version is easy to implement and yields a uniform stationary distribution. Nevertheless, it turns out that it is easier to analyze the heat-bath version. What we shall do next will be to analyze the heat-bath version of the ball walk and then argue that the mixing time of the Metropolis version is similar.

16.1.2 Notation for Continuous Space Markov Chains

For a point $x \in R^n$ and a (measurable) set $A \subseteq K$, denote by $p(x, A)$ the transition probability from x to A, i.e.,

$$p(x, A) = \Pr[X_{t+1} \in A | X_t = x].$$

Then the t-step transition probability is recursively defined as follows:

$$p^t(x, A) = \int_K p^{t-1}(x, dy) p(y, A).$$

For the heat-bath ball walk we have

$$p(x, A) = \frac{\mathbf{vol}(B(x, \delta) \cap A)}{\mathbf{vol}(B(x, \delta) \cap K)}, \text{ where } A \subseteq K.$$

This walk is specified buy the infinitesimal generator

$$p(x, dy) = \begin{cases} \frac{dy}{\mathbf{vol}(B(x,\delta) \cap K)} & \text{if } y \in B(x, \delta) \cap K; \\ 0 & \text{otherwise.} \end{cases}$$

A probability distribution μ over K is *a stationary measure for P* if

$$\mu(A) = \int_K p(x, A) \mu(dx), \ \forall \text{ measurable } A \subseteq K. \tag{16.1}$$

Definition 16.2 *Define the density $\ell(x) := \frac{\mathbf{vol}(B(x,\delta) \cap K)}{\mathbf{vol}(B(x,\delta))}$, and set $L := \int_K \ell(x) dx$.*

Claim 16.3 *The distribution $\mu(A) = \frac{\int_A \ell(x) dx}{L}$ is stationary for the heat bath ball walk.*

Proof: With this definition of μ, the right-hand side of (16.1) becomes

$$\begin{aligned} \int_K p(x, A) \mu(dx) &= \int_A \int_K p(x, dy) \mu(dx) \\ &= \int_A dy \int_{B(y,\delta) \cap K} \frac{\mu(dx)}{\mathbf{vol}(B(y, \delta) \cap K)} \\ &= \frac{1}{L} \int_A dy \int_{B(y,\delta) \cap K} \frac{\ell(x) dx}{\mathbf{vol}(B(x, \delta) \cap K)} \\ &= \frac{1}{L} \int_A dy \int_{B(y,\delta) \cap K} \frac{dx}{\mathbf{vol}(B(x, \delta))} \\ &= \frac{1}{L} \int_A \ell(y) dy = \mu(A), \end{aligned}$$

which verifies the required condition (16.1). (In the second line here we have used the fact that $y \in B(x, \delta) \cap K \Leftrightarrow x \in B(y, \delta) \cap K$.) ■

Note that the density ℓ puts less weight near the boundary of K, as we had guessed earlier from the definition of the heat-bath walk. Also, at this stage we are not in a position to claim that the heat-bath ball walk always converges to μ as we have not developed a theory of Markov chains with continuous state spaces. However, the fact that it does converge to μ will emerge as a consequence of our analysis of its mixing time in the next lecture.

Exercise 16.4 *Verify that the uniform measure is stationary for the Metropolis version of the ball walk.*

16.1.3 Curvature Assumption

To simplify the analysis, we'll make the following curvature assumption, which is **not** without loss of generality:

$$\forall x \in K, \exists y \in K \text{ s.t. } B(y,1) \subseteq K \text{ and } d(x,y) \leq 1.$$

The assumption, which is illustrated in Figure 16.2, corresponds to a smoothness assumption for the boundary of K. It will eliminate a lot of technical complexity from the analysis of the ball walk while retaining most of the interesting ideas.

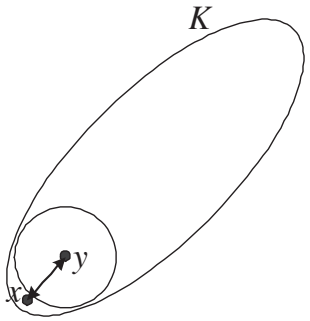


Figure 16.2: The curvature assumption

Claim 16.5 *Under the curvature assumption, if we also assume that $\delta \leq \frac{c}{\sqrt{n}}$, where c is a universal constant, then $\forall x \in K, \frac{2}{5} \leq \ell(x) \leq 1$.*

Observe that the above claim implies that we can efficiently implement the heat-bath ball walk by "rejection sampling" based on the Metropolis version. (The expected time to make one step of the heat-bath walk is at most 5/2.) Also, given a sample produced by the heat-bath ball walk from its stationary distribution μ (with density ℓ), we can make this uniform by accepting with probability $\frac{2/5}{\ell(x)} \leq 1$. (The latter is based of course on the fact that, because $\ell(x)$ is bounded from above and below by constants, it can be estimated within very high accuracy by sampling.)

Proof: Let $x \in K$, $y \in K$ as in the curvature assumption. Then by definition of $\ell(x)$ we have

$$\ell(x) \geq \frac{\mathbf{vol}(B(x,\delta) \cap B(y,1))}{\mathbf{vol}(B(x,\delta))} \geq \frac{1}{2} - \frac{\text{volume of shaded cylinder of Figure 16.3}}{\mathbf{vol}(B(x,\delta))},$$

where we have used the fact that $\mathbf{vol}(B(x,\delta) \cap B(y,1))$ is minimized for x lying on the boundary of $B(y,1)$.

The height of the cylinder is $h = \delta^2/2$ (basic geometry: exercise). The area of the base of the cylinder is $b = \mathbf{vol}_{n-1}(B(x,\delta)))$. Using the fact that $\frac{\mathbf{vol}_{n-1}(B(x,\delta))}{\mathbf{vol}_n(B(x,\delta))} \geq \frac{c'\sqrt{n}}{\delta}$ for a constant c', we get

$$\ell(x) \geq \frac{1}{2} - \frac{\delta^2}{2} \times \frac{c'\sqrt{n}}{\delta} = \frac{1}{2} - \frac{c'\delta\sqrt{n}}{2} \geq 2/5 \text{ by taking } \delta \leq \frac{1}{5c'\sqrt{n}}.$$

■

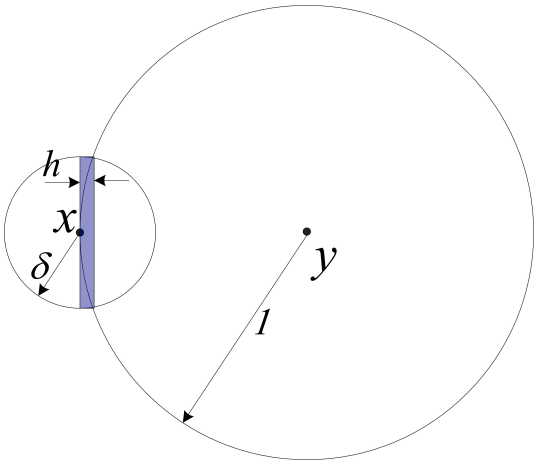


Figure 16.3: Proof of Claim 16.5

References

[1] I. Bárány and Z. Füredi. "Computing the volume is difficult," *Discrete and Computational Geometry*, 2:319–326, 1987.

[2] M.E. Dyer and A.M. Frieze. "The complexity of computing the volume of a polyhedron," *SIAM Journal of Computing*, 17:967–974, 1988.

[3] M. Dyer, A. Frieze and R. Kannan. "A random polynomial-time algorithm for approximating the volume of convex bodies," *J. Assoc. Comput. Mach.,* 38:1–17, 1991.

[4] L.G. Khachiyan. "Complexity of polytope volume computation," *New Trends in Discrete and Computational Geometry*, J. Pach editor, pages 91–101, Springer Verlag, Berlin, 1993.

[5] L. Lovász and M. Simonovits. "Random walks in a convex body and an improved volume algorithm," *Random Structures and Algorithms*, 4:359–412, 1993.

[6] L. Lovász and S. Vempala. "Simulated annealing in convex bodies and an $\tilde{O}(n^4)$ volume algorithm," *IEEE FOCS*, 2003.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 17: November 3rd

Lecturer: Alistair Sinclair *Scribes:*

17.1 Volume of a Convex Body

Our main goal in this lecture is to analyze the mixing time of the (heat bath) Ball Walk on a convex body K, as defined in the previous lecture. In this walk, from any point $x \in K$, we move to a new point chosen uniformly at random in $B(x,\delta) \cap K$, where $B(x,\delta)$ denotes the ball of radius δ centered at x, and $\delta = \Theta(\frac{1}{\sqrt{n}}$. Recall that this walk has stationary measure $\mu(x) = \frac{\ell(x)}{L}$, where $L = \int_K \ell(x)dx$ is the normalizing factor and the density $\ell(x)$ is defined by

$$\ell(x) = \frac{\mathrm{Vol}(B(x,\delta) \cap K)}{\mathrm{Vol}(B(x,\delta))}.$$

We will throughout assume a bound on the curvature of K given by the *Curvature Assumption*, which together with the value of δ (as we saw in the last lecture) ensures that $\ell(x) \geq 0.4$ for all $x \in K$. This in turn means that we can effectively implement the ball walk, and also that its stationary distribution is not too far from uniform. (In particular, given samples from μ we can produce uniform random samples by rejection sampling with expected constant overhead.)

17.2 Mixing Time

We will analyze the mixing time by examining how the variance of a measurable function φ decays as the Ball Walk evolves. Recall that in the case of discrete Markov chains, we defined the Poincaré constant

$$\alpha := \inf_{\varphi} \frac{\mathcal{E}_\pi(\varphi,\varphi)}{\mathrm{Var}_\pi(\varphi)} = \frac{\sum_{x,y}(\varphi(x)-\varphi(y))^2\pi(x)P(x,y)}{\sum_{x,y}(\varphi(x)-\varphi(y))^2\pi(x)\pi(y)},$$

where the infimum is over all non-constant functions $\varphi : \Omega \to \mathbf{R}$. The quantity α has a natural interpretation as the ratio of "local variation" to "global variation" of any function φ.

We then showed that the following holds for any lazy discrete Markov chain:

$$\tau_x(\epsilon) \leq O\big(\frac{1}{\alpha}(\log(\pi(x)^{-1}) + \log(\epsilon^{-1})\big). \tag{17.1}$$

Thus the mixing time is controlled by α, modulo a dependence on the starting state x. It turns out that a version of (17.1) (suitably modified) extends to the continuous setting of our Ball Walk. To state this, define

$$\begin{aligned} h_\varphi(x) &:= \frac{1}{2}\int_K (\varphi(x)-\varphi(y))^2 P(x,dy) \\ \alpha &:= \inf_{\varphi} \frac{\int_K h_\varphi(x)\mu(dx)}{\int_K \varphi(x)^2\mu(dx) - (\int_K \varphi(x)\mu(dx))^2} \equiv \inf_{\varphi} \frac{\int_K h d\mu}{\int_K \varphi^2 d\mu - (\int_K \varphi d\mu)^2}, \end{aligned}$$

where the infimum is over non-constant, measurable, real-valued functions φ.[1] Note that this is exactly the analog of α as defined in the discrete case; the numerator is a "local variation" and the denominator is just the "global variation" (or variance) of φ.

We then have the following claim,

Claim 17.1 *Suppose that the Ball Walk is started in an initial distribution that is uniform in a ball $B(x,\delta) \subseteq K$. Then the mixing time satisfies*

$$\tau_x(\epsilon) \leq O\Big(\frac{1}{\alpha}\left(n\log(D/2\delta) + \log\epsilon^{-1}\right)\Big)$$

where $D = \sup_{x,y\in K}\|x-y\|_2$ is the diameter of the convex body K.

Note that this has essentially the same form as (17.1). The only significant difference is that the dependence on the initial distribution is replaced by the term $n\log(D/2\delta)$, reflecting the fact that we begin at the uniform distribution in a δ-ball.[2]

We shall prove this general Claim in the next lecture. For now we just assume it. As a result, our analysis of the mixing time of the Ball Walk will boil down to the following:

Theorem 17.2 *For the Ball Walk, under the curvature assumption and with $\delta = \frac{c}{\sqrt{n}}$ for a suitable constant c, we have*

$$\alpha \geq \frac{C\delta^2}{D^2 n}$$

for some universal constant C.

An immediate consequence of this gives our main result:

Corollary 17.3 *For the Ball Walk started in an initial distribution that is uniform in a ball $B(x,\delta) \subseteq K$, under the curvature assumption and with $\delta = \frac{c}{\sqrt{n}}$, we have*

$$\tau_{mix} = O\big(\frac{D^2 n}{\delta^2} n\log(D/2\delta)\big) = O(D^2 n^3 \log(D\sqrt{n})) = O(n^4 \log n).$$

Note that we cannot expect to do much better than this, for the following reason. In one step of the Ball Walk, the expected distance traveled parallel to any fixed axis is $O(\delta/\sqrt{n}) = O(1/n)$. So in particular, the expected number of steps needed to explore the entire "longest" axis of the body (whose length is the diameter, D), is about $O(D^2 n^2)$. This is only about a factor of n less than the bound we obtain.

We will first give a "big picture" proof of Theorem 17.2 before proceeding to the technical details. The proof is by contradiction, and is in three parts, following the exposition in [Jer]. Define $\alpha^* = \frac{C\delta^2}{D^2 n}$ and assume for contradiction that

$$\frac{\int_K h d\mu}{\int_K \varphi^2 d\mu} < \alpha^* \tag{17.2}$$

for some measurable function φ with $\int \varphi d\mu = 0$. (Note that we can always assume w.l.o.g. that the expectation $\int \varphi d\mu = 0$ simply by adding an appropriate constant to φ.) We will find successively smaller subsets

[1] From now on, where no confusion arises, we will omit the subscript φ from h and also suppress the variable of integration by abreviating expressions like $\int h(x)\mu(dx)$ to $\int h d\mu$. Thus, for example, $\int \varphi d\mu$ is just the μ-expectation $\int_K \varphi(x)\mu(dx)$ of φ.

[2] The previous value $\log \pi(x)^{-1}$ would clearly have no sensible meaning in the continuous setting.

of K for which (17.2) (or a similar condition) continues to hold. Eventually we will end up with a set that is so small that (17.2) cannot hold. Thus we get a contradiction.

Here are the three stages of the proof:

Proof Outline:

1. There exists a convex subset $K_1 \subseteq K$ that also satisfies inequality (17.2) and is "needle-like", i.e.,

   $$K_1 \subseteq [0, \epsilon]^{n-1} \times [0, D]$$

   for an arbitrarily small $\epsilon > 0$.

2. We subdivide K_1 into sections of length $\eta = c' \frac{\delta}{\sqrt{n}}$ along its long axis. We show that some region K_0 consisting of a pair of adjacent such sections satisfies

   $$\frac{\int_{K_0} h d\mu}{\int_{K_0} (\varphi - \bar{\varphi})^2 d\mu} \leq \frac{1}{10}, \tag{17.3}$$

   where $\bar{\varphi} = \frac{1}{\mu(K_0)} \int_{K_0} \varphi d\mu$ is the expectation of φ on K_0. Thus K_0 also satisfies (17.2) but with a weaker bound of $\frac{1}{10}$ rather than α. Here we can take $c' > 0$ to be as small as we like.

3. We show that (17.3) is a contradiction, since K_0 is so small that its local and global variation cannot differ by much (since the reachable regions for any pair of points in K_0 have large intersection).

Proof: We prove the theorem in the three steps outlined above. Part 1 uses a classical argument due to Payne & Weinberger together with an induction on dimension. Part 2 is mostly technical and will be omitted. Part 3 involves a direct calculation.

Proof of Part 1: We call a dimension "fat" if the size of the projection of K onto it is $> \epsilon$. We proceed by induction on the number of fat dimensions. Given a set K_j with $j \geq 2$ fat dimensions that satisfies (17.2) for some φ with $\int_{K_j} \varphi d\mu = 0$, we construct a convex set K_{j-1} which has only $j-1$ fat dimensions for which φ still satisfies (17.2). By repeated application, we will end up with a set K_1 which has only one fat dimension, as required.

So let $K_j \subseteq [0, D]^j \times [0, \epsilon]^{n-j}$, and let S be the projection of K onto any two of the fat dimensions. We need a basic fact about convex sets S in two dimensions:

Fact: For any convex set $S \subseteq \mathbf{R}^2$, there exists a point $x \in S$ such that every line through x cuts S into two pieces each of size $\geq \frac{1}{3}\text{Area}(S)$.

Consider the hyperplanes through x and orthogonal to S. They partition $\mathbf{R}^n$ into H^+ and H^-. Now because $\int_{K_j} \varphi d\mu = 0$ we have that

$$\int_{K_j \cap H^+} \varphi d\mu + \int_{K_j \cap H^-} \varphi d\mu = 0$$

and so we can find a hyperplane so that

$$\int_{K_j \cap H^+} \varphi d\mu = \int_{K_j \cap H^-} \varphi d\mu = 0$$

(If $\int_{K_j \cap H^+} \varphi d\mu$ is positive then after rotating the hyperplane 180 degrees it will be negative because before the rotation $\int_{K_j \cap H^-} \varphi d\mu$ was negative. Now just use the intermediate value theorem.)

(a) (b)

Figure 17.1:

Since K_j satisfies (17.2), at least one of $K_j \cap H^+$ and $K_j \cap H^-$ must satisfy it also. Moreover, by the above Fact, the projection of both of these bodies onto the plane of S has area at most $\frac{2}{3}\text{Area}(S)$. Now replace K_j by the appropriate half and repeat this procedure until $\text{Area}(S) \leq \frac{1}{2}\epsilon^2$.

Finally, the width of a convex set of area A in $\mathbf{R}^2$ is $\leq \sqrt{2A}$, so $\text{width}(S) \leq \epsilon$ and the number of fat dimensions is reduced by one. Note that the above argument allows us to achieve any desired value $\epsilon > 0$.

Proof of Part 2 This step is rather technical and is omitted. For the details, see [Jer]. In this part we use the value of δ.

Proof of Part 3 Set $\delta' = \delta - \epsilon\sqrt{n}$. Let K_0 be the set that we are left with after Part 2. K_0 is contained in some prism of dimension $[0, \frac{c'\delta}{\sqrt{n}}] \times [0, \epsilon]^{n-1}$ (see figure 17.1(a)), where ϵ and c' are arbitrarily small positive values. Let z_1 and z_2 be the midpoints of the ($(n-1)$-dimensional) end faces of this prism.

We focus on the set $I := B(z_1, \delta') \cap B(z_2, \delta') \cap K$.

Fact 1: Any point in I can be reached in one step of ball walk by every point in the prism (and thus by every point in K_0, see figure 17.1(b)).

This Fact follows easily from our definition of $\delta' = \delta - \epsilon\sqrt{n}$.

Fact 2: $\text{Vol}(I) \geq \frac{1}{5}\text{Vol}(B(0, \delta))$.

This Fact follows from the Curvature Assumption, in similar fashion to our argument in the previous lecture that showed $\text{Vol}(B(x, \delta) \cap K) \geq \frac{2}{5}\text{Vol}(B(x, \delta))$. Here we are working with the intersection of two such balls whose centers, z_1 and z_2, are very close compared to their radii δ' (recall that both ϵ and η can be chosen very small). This causes us to lose at most a factor of 2 in the argument.

We will now show that, for any function φ,

$$\frac{\int_{K_0} h\, d\mu}{\int_{K_0} (\varphi - \bar{\varphi}) d\mu} > \frac{1}{10}$$

which will give us our desired contradiction. From Fact 1 and the definition of h and the ball walk transition density $P(x, dy)$:

$$\int_{K_0} h d\mu \geq \frac{1}{2} \int_{K_0} \frac{\mu(dx)}{\text{Vol}(B(x, \delta) \cap K)} \int_I (\varphi(x) - \varphi(y))^2 dy$$

Now applying Fubini's Theorem to interchange the order of integration, this becomes

$$\int_I dy \int_{K_0} \frac{(\varphi(x) - \varphi(y))^2}{\text{Vol}(B(x, \delta) \cap K)} \mu(dx) \geq \frac{1}{2\text{Vol}(B(0, \delta))} \int_I dy \int_{K_0} (\varphi(x) - \varphi(y))^2 \mu(dx)$$

Now, using the fact that $\varphi(y)$ is constant in the inner integral, and that $\int(\varphi - c)^2 d\mu$ is minimized by taking $c = \frac{1}{\mu(K_0)} \int_{K_0} \varphi d\mu$, i.e., the μ-expectation of φ, the above expression is bounded below by

$$\frac{1}{2\text{Vol}(B(0,\delta))} \int_I dy \int_{K_0} (\varphi - \bar{\varphi}(y))^2 d\mu \geq \frac{1}{10} \int_{K_0} (\varphi - \bar{\varphi}(y))^2 d\mu,$$

where in this final inequality we have used Fact 2.

This completes the proof. ■

References

[Jerrum] M.R. Jerrum, "Volume of a convex body," Draft lecture notes, available at http://homepages.inf.ed.ac.uk/chap6.ps.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 18: November 5

Lecturer: Prof. Alistair Sinclair *Scribes: Sharmodeep Bhattacharyya and Bharath Ramsundar*

18.1 Continuation of Volume Estimation

In the past two lectures, we have been analyzing the construction of an f.p.r.a.s. to estimate the volume of a convex body $K \subseteq \mathbb{R}^n$ in the membership oracle model. We assumed a general upper bound on the mixing time of heat-bath ball walks, and used this assumption to analyze our construction. In this section, we prove the assumed bound. Recall that the Poincaré constant is defined by $\alpha = \inf_{\varphi: K \to \mathbb{R}} \frac{\int_K h_\varphi d\mu}{\mathrm{Var}_\mu \varphi}$, where the local variance is $h_\varphi = \frac{1}{2}\int_K (\varphi(x) - \varphi(y))^2 p(x, dy)$. This is a continuous analog of the discrete version introduced in an earlier lecture.

First we need to generalize our definition of mixing time to the continuous setting:

Definition 18.1 *For $B \subseteq \mathbb{R}^n$, let $\tau_B(\epsilon) := \min\{t \in \mathbb{N}^+ | \forall$ measurable $S \subseteq K,\ |p^t(B, S) - \mu(S)| \leq \epsilon\}$ where μ is the stationary measure for the given Markov chain, and $p^t(B, S)$ denotes the probability that the chain is in S after t steps starting from the uniform distribution on B.*

Claim 18.2 *Let $B \subseteq K$ be a δ-ball contained in K. (This exists since K is assumed to contain the unit ball.) Then*

$$\tau_B(\epsilon) \leq O(\frac{1}{\alpha}(n \ln(\frac{D}{2\delta}) + \ln(\epsilon^{-1}))),$$

where D is the diameter of K.

Proof: Let μ be the stationary measure on K for the given Markov chain. By mimicking the proof in the discrete case, we have that $\mathrm{Var}_\mu P\varphi \leq \mathrm{Var}_\mu \varphi - \alpha \mathrm{Var}_\mu \varphi$, where the operator P is defined in analogous fashion to the discrete case, i.e., $P\varphi(x) = \int_K p(x, dy)\varphi(y))$. Thus, by induction, $\mathrm{Var}_\mu P^t\varphi \leq \exp(-\alpha t)\,\mathrm{Var}_\mu \varphi$.

Now let S be any measurable subset of K, and φ the indicator function of S. Since $\mathrm{Var}_\mu \varphi \leq 1$ we have

$\mathrm{Var}_\mu P^t\varphi \le \exp(-\alpha t)$. Also,

$$\begin{aligned}
\mathrm{Var}_\mu P^t\varphi &= \int_K (P^t\varphi(y) - \mu(S))^2\mu(dy) \\
&= \int_B (P^t\varphi(y) - \mu(S))^2\mu(dy) \\
&\ge \frac{2}{5}\frac{\mathrm{vol}\, B}{\mathrm{vol}\, K}\int_B \frac{1}{\mathrm{vol}\, B}(P^t\varphi(y) - \mu(S))^2 dy \\
&\ge \frac{2}{5}\frac{\mathrm{vol}\, B}{\mathrm{vol}\, K}\left(\int_B \frac{1}{\mathrm{vol}\, B}(P^t\varphi(y) - \mu(S))dy\right)^2
\end{aligned}$$

In the second line we may restrict the integral to B because we start with density zero outside B. In the third line we have used our curvature assumption to write $\mu(dy) \ge \frac{2}{5}\frac{dy}{\mathrm{vol}\, K}$. The last inequality follows from the fact $E[X^2] \ge E[X]^2$.

To bound the mixing time, we want to find t such that $|p^t(B,S) - \mu(S)| \le \epsilon$. Note that $p^t(B,S) - \mu(S) = \int_B \frac{1}{\mathrm{vol}\, B}(P^t\varphi(y) - \mu(S))dy$. Thus, it follows that if $\mathrm{Var}_\mu P^t\varphi \le \epsilon^2\frac{2}{5}\frac{\mathrm{vol}\, B}{\mathrm{vol}\, K}$, then our bound is proven. But $\mathrm{Var}_\mu P^t\varphi \le \exp(-\alpha t)$, so we need only set $t = \frac{1}{\alpha}\ln(\frac{5}{2\epsilon^2}\frac{\mathrm{vol}\, K}{\mathrm{vol}\, B})$. Since the volume of K is bounded above by the volume of the n-dimensional ball of diameter D, we have that $\frac{\mathrm{vol}\, K}{\mathrm{vol}\, B} \le (\frac{D}{2\delta})^n$. Thus, $t \le O(\frac{1}{\alpha}(n\ln(\frac{D}{2\delta}) + \ln(\frac{1}{\epsilon})))$, as required. ■

18.2 Conductance and Sparsest Cut

Once again, we return to the framework of looking at a discrete Markov chain as a multicommodity flow network. In earlier lectures, we saw that any flow gave us an upper bound on mixing time. We will now look at *cuts*, the dual of flows, to develop a method for lower-bounding the mixing time of a Markov chain. Recall that in the single source/single sink model, the size of a maximum flow equals the size of a minimum cut. This statement isn't necessarily true when there are multiple sources and sinks, but we can still get bounds.

Definition 18.3 *For any subset $S \subseteq \Omega$,*

$$\Phi(S) := \frac{\sum_{\substack{x\in S\\ y\in S}} C(x,y)}{\pi(S)} = \frac{C(S,\overline{S})}{\pi(S)}$$

We may think of this quantity as the normalized flow passing out of subset S, or equivalently as the probability of escaping from S in one step of the Markov chain in the stationary distribution, conditional on starting in S.

Definition 18.4 *Define the* conductance Φ *as*

$$\Phi \equiv \min_{S:0<\pi(S)\le 1/2} \Phi(S)$$

The conductance corresponds to the "sparsest cut" in the graph. Combinatorially, a small cut is a bottleneck in the graph. Probabilistically, it is a subset which the Markov chain is unlikely to leave in one move. Consequently, if Φ is very small, mixing time should be very large and vice versa. We note that restricting attention to sets with $\pi(S) \leq 1/2$ is natural, for if S is large then we should not be too concerned if it takes a long time to leave it. However, we can also define the following more symmetrical quantities:

Definition 18.5 *For any subset $S \subseteq \Omega$,*

$$\Phi'(S) = \frac{C(S,\overline{S})}{\pi(S)\pi(\overline{S})} = \frac{C(S,\overline{S})}{D(S,\overline{S})}$$

$$\Phi' = \min_{S:0<\pi(S)\leq 1/2} \Phi'(S)$$

Here $D(S,\overline{S}) = \sum_{x\in S,\, y\in\overline{S}} \pi(x)\pi(y)$ is the total demand between vertices in S and in $\overline{S}$.

For $\pi(S) \leq 1/2$, it is clear that $\Phi(S) \leq \Phi'(S) \leq 2\Phi(S)$. Hence Φ and Φ' differ by at most a factor of 2. It is also obvious that for any flow f, $\rho(f) \geq \frac{1}{\Phi'}$ (the quantity on the r.h.s is the maximum ratio of demand to capacity for sets, so in any flow there must be at least one edge that has this ratio of demand to capacity). Less obviously, we have that $\rho(f) \leq O(\frac{\log|\Omega|}{\Phi'})$. This result follows from the approximation to sparsest cut devised by Leighton and Rao [LR99]; see [Si92] for the application to Markov chains. This bound is tight, as the following exercise asks you to verify.

Exercise 18.6 *Verify that in a constant degree expander, we have $\rho(f) = \Theta(\log|\Omega|)$, but Φ' is constant.*

We now state two theorems relating conductance and mixing time. The first is an upper bound on mixing time:

Theorem 18.7 *For any lazy, reversible Markov chain,*

$$\tau_X(\epsilon) \leq \text{const} \times \left[\frac{1}{\Phi^2}\left(\log \pi(x)^{-1} + \log \epsilon^{-1}\right)\right] \tag{18.1}$$

Proof: **[Sketch]** The proof follows from the following bound on eigenvalue gap (sometimes known as "Cheeger's inequality" because of its more classical continuous analog):

$$1 - \lambda_2 \geq \text{const.}\Phi^2 \tag{18.2}$$

So, by Fact 10.9 in lecture 10, $\alpha = 1 - \lambda_2 \geq \text{const.}\Phi^2$ and then by Thm 10.5 in lecture 10, we get that

$$\tau_x(\epsilon) \leq \text{const}\left[\frac{1}{\Phi^2}\left(\log \pi(x)^{-1} + \log \epsilon^{-1}\right)\right].$$

■

We will not prove (18.2) (which is quite non-trivial) here; for a proof, see [SJ89]. We also will not dwell on the above theorem as it is usually less useful than the analogous upper bound based on flows (Corollary 11.3

from Lecture 11). The reason for this is that, while *any* flow gives us an upper bound on the mixing time, in order to use Theorem 18.7 we need to quantify over *all* cuts, which is inherently harder. Thus, with the exception of a few geometric examples where Φ can be bounded directly using an *isoperimetric inequality* (i.e., a bound on surface area vs volume), Theorem 18.7 is not usually the best path to upper bounds.

On the other hand, by the same token, conductance *is* very useful for proving *lower* bounds on mixing times, because any set S provides such a bound as the following easy result confirms:

Theorem 18.8 *For any Markov chain, and any $S \subseteq \Omega$ with $\pi(S) \leq \frac{1}{2}$,*

$$\tau_X(1/4) \geq \frac{1}{4\Phi(S)}. \tag{18.3}$$

Proof: Consider initial distribution

$$p^0(x) \quad = \quad \begin{cases} \frac{\pi(x)}{\pi(S)}, & \text{if } x \in S \\ 0, & \text{if } x \notin S \end{cases}$$

Then the distribution p^1 after 1 step satisfies

$$\begin{aligned} ||p^1 - p^0||_{TV} &= \frac{1}{2}\sum_y \left|\sum_x p^0(x)P(x,y) - p^0(y)\right| \\ &= \sum_{y\in\bar{S}} \left|\sum_x p^0(x)P(x,y) - p^0(y)\right| \\ &= \sum_{x\in S}\sum_{y\in\bar{S}} p^0(x)P(x,y) \qquad (\because p^0(y) = 0, \text{ for } y \in \bar{S}) \\ &= \Phi(S). \end{aligned}$$

The second equality above follows from the fact that, $||p^1 - p^0||_{TV} = \sum_{y\in A} \left|p^1(y) - p^0(y)\right|$, where $A = \{y \in \Omega : |p^1(y) - p^0(y)| > 0\}$, together with the observation that in this case $A = \bar{S}$. **Exercise:** check this!

We leave the following simple observation as an exercise:

Exercise 18.9 $||p^{t+1} - p^t||_{TV} \leq ||p^t - p^{t-1}||_{TV}$

Iterating the result of this Exercise gives $||p^t - p^0|| \leq t\Phi(S)$. Thus, using the triangle inequality and the fact that $||p^0 - \pi|| = \pi(\overline{S}) \geq \frac{1}{2}$, we get

$$\begin{aligned} ||p^t - \pi|| &\geq ||p^o - \pi|| - ||p^t - p^0|| \\ &\geq \frac{1}{2} - t\Phi(S) \\ &\geq \frac{1}{4} \qquad (\text{assuming } t \leq \frac{1}{4\Phi(S)}). \end{aligned}$$

So, $\tau_x\left(\frac{1}{4}\right) \geq \frac{1}{4\Phi(S)}$, as claimed. ■

Corollary 18.10 *If we can find $S \subseteq \Omega$ with $\pi(S) \leq \frac{1}{2}$ and $\frac{\pi(\partial S)}{\pi(S)} \leq \delta$, where ∂S denotes the interior boundary of S (i.e., points in S that are connected to a point in $\overline{S}$), then $\tau_{mix} \geq \frac{1}{4\delta}$.*

The proof of the Corollary follows immediately from the easy observation that $\frac{\pi(\partial S)}{\pi(S)} \geq \frac{C(S,\bar{S})}{\pi(S)}$.

18.3 Independent Sets

Input: Graph, $G = (V, E)$, Δ =maximum degree of the graph.
Goal: Sample an independent set (of any size) of vertices in G u.a.r. (or, equivalently, give a f.p.r.a.s. for counting independent sets).

Facts 18.11 *1. In general there can be no polynomial time sampling algorithm (or equivalently, no f.p.r.a.s.) for the problem unless $NP = RP$.*

2. The same holds even for graphs with maximum degree $\Delta \geq 25$.

3. A simple MCMC algorithm (with local moves) solves the problem in polynomial time for graphs with $\Delta \leq 5$.

4. No MCMC algorithm (from a very wide class) can work in polynomial time for $\Delta = 6$.

We will prove Fact 1 (which is very easy) below. Fact 2, due to Dyer, Frieze and Jerrum [DFJ02], is a more refined version of Fact 1, which replaces the simple NP-hardness of independent set by a hardness of approximation derived from the PCP theorem; we will not prove this. Fact 3 is a recent, and rather substantial result of Weitz [We06]; we may prove this later if there is time. Finally, we will prove the negative Fact 4 (also due to [DFJ02]) in the next lecture, using the lower bound on mixing times given in Theorem 18.8 above.

Before proceeding, note that the above Facts naturally suggest the following intriguing conjecture:

Conjecture 18.12 *There can be no polynomial time algorithm for the problem even for graphs with maximum degree $\Delta \geq 6$.*

This would extend the negative result of Fact 2 all the way down to $\Delta = 6$, and provide a sharp dichotomy since by Fact 3 the problem is tractable for $\Delta \leq 5$. In the next lecture we will prove Fact 4, which can be viewed as strong evidence for the Conjecture (since MCMC is the only algorithmic tool we know here, and the proof of Facts 3 and 4 also suggest a qualitative change in the problem as Δ moves from 5 to 6).

We end today's lecture with the proof of the easy Fact 1.

Proof: (of (1)) Let $G = (V, E)$ be arbitrary, $|V| = n$. Let $I \subseteq V$ be an independent set of size k in G. Construct graph G' by replacing each vertex of G by a group of r vertices with no edges between them, and

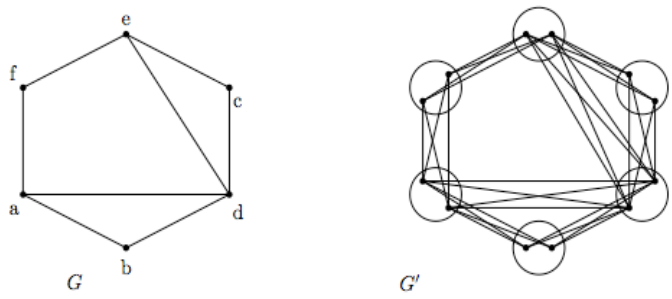


Figure 18.1: G and G' for $r = 2$.

connecting all pairs of vertices in two such groups iff there is an edge between the two corresponding vertices of G. The construction of G' is explained in Fig. 1 for $r = 2$.

Thus each independent set I of size k in G gives rise to $(2^r - 1)^k$ independent sets in G' corresponding to I (since we can choose any non-empty subset of each of the k associated groups). So, in G', the number of independent sets of size $< k$ is $\leq 2^n(2^r - 1)^{k-1}$.

Now choose r such that $(2^r - 1)^k \gg 2^n(2^r - 1)^{k-1}$. (Taking $r = cn$ for a suitable constant c suffices; note that G' can be constructed in polynomial time from G.) In other words, when translated to G', even a single independent set of size k in G completely swamps all the smaller independent sets.

Thus we see that, if we could approximately count (even very crudely) the number of independent sets in G', then we could determine the maximum size of an independent set in G. Since this latter problem is NP-hard, this would imply that $NP = P$ (or, if the approximate counting algorithm is randomized, that $NP = RP$). ■

References

[DFJ02] M. Dyer, A. Frieze and M. Jerrum. On counting independent sets in sparse graphs. *SIAM Journal on Computing* **31** (2002), pp. 1527–1541.

[LR99] F.T. Leighton and S.B. Rao. Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms. *Journal of the ACM* **46** (1999), pp. 787–832.

[Si92] A. Sinclair. Improved bounds for mixing rates of Markov chains and multicommodity flow. *Combinatorics, Probability and Computing* **1**, 1991, pp. 351–370.

[SJ89] A. Sinclair and M. Jerrum. Approximate counting, uniform generation and rapidly mixing Markov chains. *Information & Computation* **82** (1989), pp. 93–133.

[We06] D. Weitz, Counting Independence Sets up to the tree threshold. *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, 2006, Session 4A, pp. 140–149.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 19: November 10

Lecturer: Prof. Alistair Sinclair *Scribes: Kevin Dick and Tanya Gordeeva*

19.1 Independent Sets (Continuation)

In the previous lecture, we discussed the problem of sampling independent sets uniformly at random.

Input: Graph $G = (V, E)$ on $|V| = n$ vertices.

Goal: Sample an independent set of vertices of G uniformly at random.

Definition 19.1 *Call a Markov Chain with state space Ω equal to the independent sets of G and uniform stationary distribution γ-*cautious *if, in any step, it changes the disposition of at most γn vertices of G.*

We'll prove the following theorem, due to Dyer, Frieze, and Jerrum [DFJ]:

Theorem 19.2 *There exist $\gamma > 0$ and graph G with maximum degree $\Delta = 6$ such that any γ-cautious Markov chain on independent sets of G has mixing time* $\exp(\Omega(n))$.

Note that this result is best possible: a theorem of Weitz [Wei] says that there exists a local Markov chain (which changes one vertex per step) with mixing time $O(n \log n)$ for all G with $\Delta \leq 5$.

To prove the theorem, we'll use a result from the previous lecture:

Theorem 19.3 *For any Markov chain, and for all $S \subseteq \Omega$ with $\pi(S) \leq 1/2$, $\tau_{mix} \geq \frac{1}{4\Phi(S)}$, where*

$$\Phi(S) = \frac{C(S, \overline{S})}{\pi(S)} = \sum_{x \in S, y \in \overline{S}} \frac{\pi(x)P(x, y)}{\pi(S)}.$$

Corollary 19.4 $\tau_{mix} \geq \frac{|S|}{4|\partial S|}$, *where ∂S is the* inner boundary *of S, i.e., the set of states in S that are connected to some state outside S.*

Proof: (of Theorem 19.2) Let $G = (L, R, E)$ be a *random* bipartite graph, with $|L| = |R| = n$, constructed by throwing down Δ independent random perfect matchings (and erasing duplicate edges). Clearly the maximum degree is at most Δ.

Definition 19.5 *An* (α, β)-independent set *is one with αn vertices in L and βn vertices in R.*

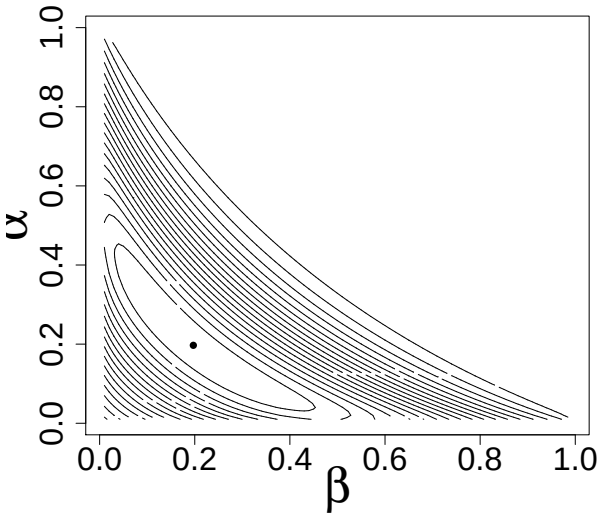


Figure 19.1: $f(\alpha, \beta)$ with $\Delta = 5$.

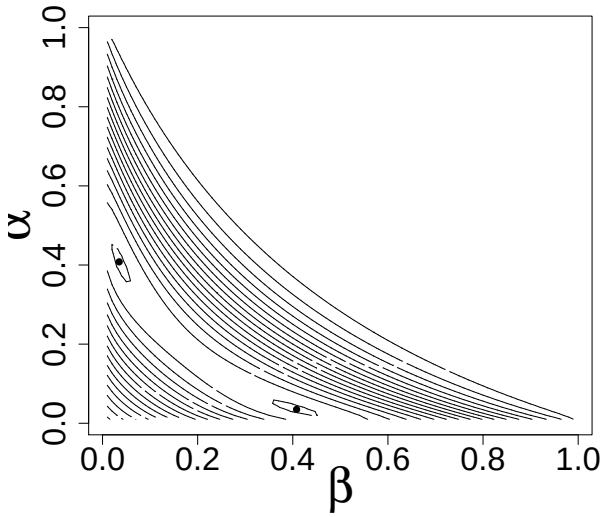


Figure 19.2: $f(\alpha, \beta)$ with $\Delta = 6$.

Let $\mathcal{E}(\alpha, \beta)$ be the expected number of (α, β)-independent sets. We have $\binom{n}{\alpha n}$ ways to choose αn vertices in L and $\binom{n}{\beta n}$ ways to choose βn vertices in R. Moreover, the probability that a given set of αn vertices in L won't hit some given set of βn vertices in R under a single random perfect matching is the probability that the αn outgoing edges from L only hit the other $(1-\beta)n$ vertices in R, which is $\binom{(1-\beta)n}{\alpha n} / \binom{n}{\alpha n}$. Since we have Δ independent random perfect matchings, we see that

$$\mathcal{E}(\alpha, \beta) = \binom{n}{\alpha n}\binom{n}{\beta n}\left[\binom{(1-\beta)n}{\alpha n} \Big/ \binom{n}{\alpha n}\right]^{\Delta}.$$

By Stirling's approximation, $m! \sim \sqrt{2\pi m}(m/e)^m$, we have

$$\binom{n}{\alpha n} = \left(\frac{1}{\alpha^{\alpha}(1-\alpha)^{1-\alpha}}\right)^{n} \theta\left(\frac{1}{\sqrt{n}}\right),$$

which yields

$$\mathcal{E}(\alpha, \beta) = \left[\frac{(1-\beta)^{(\Delta-1)(1-\beta)}(1-\alpha)^{(\Delta-1)(1-\alpha)}}{\alpha^{\alpha}\beta^{\beta}(1-\alpha-\beta)^{\Delta(1-\alpha-\beta)}}\right]^{n+o(n)}.$$

Thus we can write $\mathcal{E}(\alpha, \beta) = \exp\{f(\alpha, \beta)(n + o(n))\}$, where

$$f(\alpha, \beta) := -\alpha \ln \alpha - \beta \ln \beta - \Delta(1-\alpha-\beta)\ln(1-\alpha-\beta) + (\Delta-1)((1-\alpha)\ln(1-\alpha) + (1-\beta)\ln(1-\beta)).$$

The behavior of f is qualitatively very different for the cases of $\Delta \leq 5$ and $\Delta \geq 6$. In particular, note the following properties:

- f is symmetric in α, β, and has no local maxima (other than its global maxima).
- When $\Delta \leq 5$, there is a unique maximum, which falls on the line $\alpha = \beta$ (Figure 19.1).
- When $\Delta \geq 6$, there are two (symmetric) maxima, neither of which fall on the line $\alpha = \beta$ (Figure 19.2).

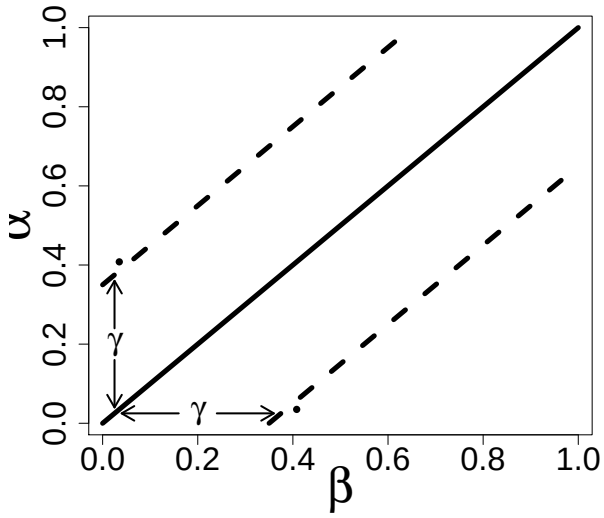


Figure 19.3: Strip of width 2γ between the two maxima.

We'll study the case of $\Delta \geq 6$. Let (α^*, β^*) be the location of (one of the two) maxima. (For $\Delta = 6$, $\alpha^* \approx 0.035$ and $\beta^* \approx 0.408$.) Numerically, $f(\alpha^*, \beta^*) > c = 0.71$. So $\mathcal{E}(\alpha^*, \beta^*) \geq e^{cn}$ for sufficiently large n.

Choose $\gamma = .35$. Then, again numerically, the maximum value of $f(\alpha, \beta)$ on the strip of width 2γ in Figure 19.3 is less than $c - \delta$, for some $\delta \geq .0001$. Since there are clearly at most n^2 values of (α, β) in this strip, the expected number of independent sets with (α, β) in this strip is at most $n^2 e^{(c-\delta)n} < e^{c'n}$, for some $c' < c$ and sufficiently large n.

Let I_{left} be the set of (α, β)-independent sets with $\alpha > \beta$, I_{right} the set of (α, β)-independent sets with $\alpha < \beta$, and I_{mid} the set of (α, β)-independent sets with (α, β) in the strip. Ignoring the possible case of $\alpha = \beta$, we see that $I_{\text{mid}} \subseteq I_{\text{left}} \cup I_{\text{right}}$.

By Markov's inequality, $|I_{\text{mid}}| \leq e^{c'n}$ with high probability (with $c' = 0.706$). Deterministically, the number of (α, β)-independent sets I with $|I \cap L| = \alpha n$ is at least $\binom{n}{\alpha n} 2^{(1-\Delta\alpha)n}$, since for every choice of αn vertices in L we can select some subset of the (at least) $(1 - \Delta\alpha)n$ vertices in R that aren't adjacent to any of the chosen vertices in L. By a Stirling-like argument as above, this quantity is $\exp(g(\alpha)(n - o(n)))$, where $g(\alpha) = -\alpha \ln \alpha - (1 - \alpha) \ln(1 - \alpha) + (\ln 2)(1 - \Delta\alpha)$.

Setting $\Delta = 6$ and optimizing over α gives a lower bound on $\exp(g(\alpha)(n - o(n)))$ of $e^{c''n}$, where $c'' > c'$ (in fact, $c'' = .708$).

Clearly this analysis holds for both $I \cap L$ and $I \cap R$. Choose S to be the smaller of I_{left} and I_{right}, so that $\pi(S) \leq 1/2$. Then

$$\frac{|\partial S|}{|S|} \leq \frac{e^{c'n}}{e^{c''n}} = \frac{1}{e^{(c''-c')n}} \leq e^{-0.002n}.$$

By Theorem 19.3, this completes the proof. ■

19.2 The Ising model in two dimensions

Recall that the Ising model assigns + or − to vertices in a $\sqrt{n} \times \sqrt{n}$ box in the two dimensional square lattice with the probability of a configuration σ given by the *Gibbs distribution*:

$$\pi(\sigma) = Z^{-1} \exp\left(\sum_{i \sim j} \beta \sigma_i \sigma_j\right) = \exp\left(\beta(\#\text{ of agreeing neighbors} - \#\text{ of disagreeing neighbors})\right).$$

(The factor Z is just a normalizing factor, or partition function.) The sum is over adjacent pairs i, j. The parameter β is the inverse of the temperature (in physics, the model simulates ferromagnetism). When β is low, the Gibbs distribution approaches the uniform distribution over configurations, corresponding to the fact that at high temperatures σ will not exhibit macroscopic order. Conversely, when β is high (low temperatures), the distribution assigns much higher weight to configurations that are highly organized. There is a critical (inverse) temperature β_{crit} at which the model undergoes a *phase transition*, that is, it switches between being organized and being disorganized. A more formal approach to β_{crit} examines the correlation between the value of the spin at the origin and the value of spins at the boundary as n tends to infinity. For $\beta < \beta_{\text{crit}}$, the spin at the origin will be uncorrelated with the boundary as $n \to \infty$; for $\beta > \beta_{\text{crit}}$, there may be positive correlation with the boundary (e.g., for the all-plus or all-minus boundary) even as $n \to \infty$.

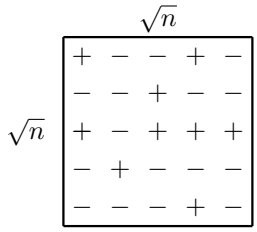


Figure 19.4: An example of a configuration on the 5x5 lattice

The heat bath (Glauber) dynamics is a Markov chain on spin configurations starting from any initial configuration. At each step, a site i is chosen u.a.r. from the lattice, and σ_i is resampled from the conditional distribution on σ_i given the neighbors. Let m_i^+ be the number of neighbors of i with value +, and let m_i^- be the number of neighbors with value −. Given that i is chosen, the probability that the new spin at i will be + is given by $\frac{\exp(\beta(m_i^+ - m_i^-))}{\exp(\beta(m_i^+ - m_i^-)) + \exp(\beta(m_i^- - m_i^+))}$. This gives a reversible Markov chain that converges to the Gibbs distribution.

The following remarkable fact about the Glauber dynamics has been proved relatively recently for such a classical model (see, e.g., [M98]):

Theorem 19.6 *The mixing time of the Glauber dynamics for the Ising model on a $\sqrt{n} \times \sqrt{n}$ box in the 2-dimensional square lattice is:*

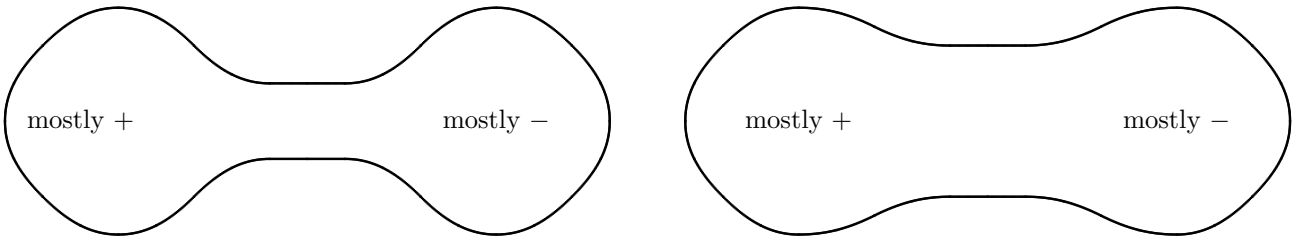
$$\begin{cases} O(n \log n) & \text{if } \beta < \beta_c; \\ e^{\Omega(\sqrt{n})} & \text{if } \beta > \beta_c, \end{cases}$$

where β_{crit} is the critical point (i.e., phase transition).

The full proof of this result requires more time than we have available in this class. However, we'll give elementary arguments that show that the mixing time is $O(n \log n)$ for sufficiently small β, and that it is

$\exp(\Omega(\sqrt{n}))$ for sufficiently large β. Pushing both these results to the threshold value β_{crit} is technically rather involved.

For the rest of this lecture, we concentrate on showing a lower bound $\exp(\Omega(\sqrt{n}))$ on the mixing time for $\beta > \beta_0$ for some constant β_0 (sufficiently low temperature). We can think of the configurations at low temperatures as being divided into two "phases" where the spins are mostly + or mostly − respectively, with a low probability of transitioning between the phases. (At high temperatures such a bottleneck does not exist because the "intermediate" configurations, which are the most numerous, have sufficiently large weight.)



Low-temperature regime High-temperature regime

Figure 19.5: The low-temperature regime has a bottleneck between states of mostly + and mostly −.

Claim 19.7 *There exists a large, but finite β such that the mixing time is* $\exp(\Omega(\sqrt{n}))$.

The key idea in the proof is the concept of a *fault line*, due to [Ran] and [Per].

Definition 19.8 *A* fault line *is a line that crosses the $\sqrt{n} \times \sqrt{n}$ box either left-right or top-bottom and separates + from −.*

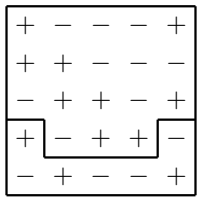


Figure 19.6: An example of a fault line

Fact 1: Let F be the set of configurations that contain a fault line. Then $\pi(F) \leq e^{-C\sqrt{n}}$ for some $C > 0$, provided β is large enough.

Proof: Fix a fault line L with length ℓ. Let $F(L)$ be the set of configurations containing ℓ. Take any configuration $\sigma \in F(L)$, and flip the spins of σ on one side of the fault line. The weight of σ will go up by a factor of $e^{2\beta\ell}$ since $\sigma_i\sigma_j$ only changes for i, j neighbors on opposite sides of the fault line. For a fixed fault line L, this operation is one-to-one on $F(L)$. Therefore $\pi(F(L)) < e^{-2\beta\ell}$. Now sum over L to get

$$\pi(F) \leq 2\sqrt{n} \sum_{\ell \geq \sqrt{n}} \gamma^\ell e^{-2\beta\ell} \leq e^{-c\sqrt{n}} \quad \text{when } \beta > \frac{1}{2} \ln \gamma$$

The factor of $2\sqrt{n}$ accounts for the starting points of the fault lines and whether the fault line is left-right or top-bottom. The quantity γ is the connective constant, $\gamma = \lim_{\ell\to\infty} |\Gamma_\ell|^{1/\ell}$ where $|\Gamma_\ell|$ is the number of self-avoiding walks of length ℓ. For our purposes, it's enough that $\gamma \leq 3$ since there are at most 3 directions to choose from at each step in the fault line. ■

Fact 2: If there does not exist a + crossing or a − crossing from left to right, then there must be a fault line going top-bottom.

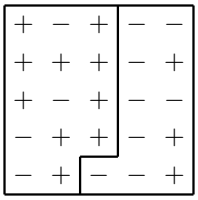


Figure 19.7: An example of a top-bottom fault line when there are no monochromatic left-right crossings

Proof: (Sketch) There must a leftmost path L of + vertices from top to bottom and a rightmost path R of − vertices from top to bottom, otherwise there would be either a − crossing or a + crossing from left to right. Suppose L is always to the right of R (otherwise take the rightmost path of +s and the leftmost path of −s). Then it is possible to construct a fault line between L and R with + to the left of the fault line and − to the right of the fault line. ■

Fact 3: If there exists a vertex v that has a + path and a − path to the top, then there exists a "fault line" from v to the top (ie, a line starting from v to the top that separates + and −).

We leave the proof of Fact 3 as an **exercise** (see Figure 19.8 for inspiration).

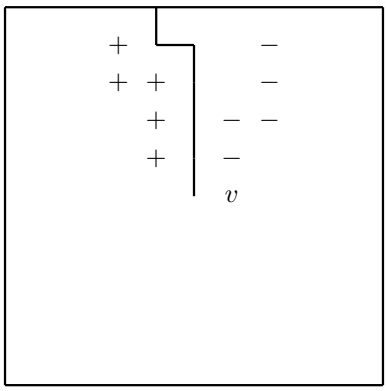


Figure 19.8: The vertex v has paths of +s and −s to the top, so it has a fault line to the top.

Now define S_+ to be the set of configurations with a left-right + crossing and a top-bottom + crossing. Define S_- similarly, except with − crossings. If $\sigma \in \overline{(S_t \cup S_i)}$, then σ either has no monochromatic left-right crossing or no monochromatic top-bottom crossing; hence it has a fault line by Fact 2. Therefore $\pi(S_-) = \pi(S_+) \to 1/2$ as $n \to \infty$ by Fact 1.

Let $\tilde{\delta}S_+$ be the *exterior boundary* of S_+ (i.e., configurations adjacent to S_+ that are not in S_+).

Claim 19.9 $\pi(\tilde{\delta}S_+) \leq e^{-c'n}$.

Proof: If $\sigma \in \overline{(S_+ \cup S_-)}$, then σ has a fault line and therefore has exponentially small probability by Fact 1. So consider $\sigma \in \tilde{\delta} S_+ \cap S_-$. For any such σ there must be a vertex v with spin $-$ having $+$ crossings from v to the top, bottom, left, and right. (This follows because flipping the spin at v must put the configuration in S_+.) There also must be a $-$ crossing from left to right, or from top to bottom, that passes through v (because flipping v takes the configuration out of S_-); w.l.o.g. suppose there exists a left-right such crossing. Therefore there are fault lines from the left to v and from v to the right. Concatenating these two fault lines gives us a true top-bottom fault line, except that the two lines may not match up exactly at v (e.g., one may end at the top of v and the other at the bottom of v). Hence we get a fault line with at most a small (constant) number of imperfections. But it should be clear that the proof of Fact 1 is robust enough to handle a constant number of imperfections. So once again, σ has exponentially small probability. ■

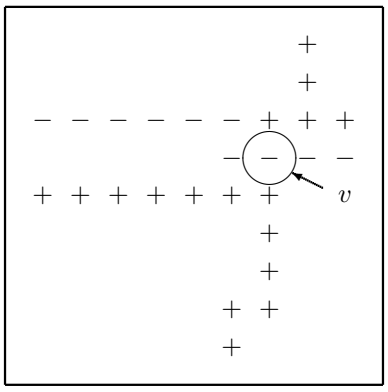


Figure 19.9: There is one vertex which blocks top-bottom and left-right + crossings.

References

[DFJ] M. Dyer, A. Frieze and M. Jerrum. "On counting independent sets in sparse graphs." *SIAM Journal on Computing* **31** (2002), pp. 1527–1541.

[M98] F. Martinelli. Lectures on Glauber dynamics for discrete spin models. *Lectures on Probability Theory and Statistics*, Springer Lecture Notes in Mathematics **1717**, 1998, pp. 93–191.

[Per] Y. Peres. "On the lower bound for mixing time of Glauber dynamics for the Ising model." Manuscript, 2006.

[Ran] D. Randall. "Slow mixing of Glauber dynamics via topological obstructions." *Proceedings of SODA* 2006, pp. 870–879.

[Wei] D. Weitz. "Counting independent sets up to the tree threshold." *Proceedings of the 38th Annual ACM Symposium on Theory of Computing (STOC)*, 2006, pp. 140–149.

CS294 Markov Chain Monte Carlo: Foundations & Applications **Fall 2009**

Lecture 20: November 12

Lecturer: Prof. Alistair Sinclair *Scribes: S. Negahban and D. Omidiran*

20.1 Glauber Dynamics for the 2-d Ising model

Recall the fundamental result quoted last time (see, e.g., [M98]):

Theorem 20.1 *The mixing time of the Glauber dynamics for the Ising model on a $\sqrt{n} \times \sqrt{n}$ box in the 2-dimensional square lattice is:*

$$\begin{cases} O(n \log n) & \text{if } \beta < \beta_c; \\ e^{\Omega(\sqrt{n})} & \text{if } \beta > \beta_c, \end{cases}$$

where β_c is the critical (inverse) temperature, $\beta_c = \frac{1}{2} \ln (1 + \sqrt{2})$.

Last time we showed that the mixing time is $e^{\Omega(\sqrt{n})}$ for sufficiently large (but finite) β (though not all the way down to β_c). Now we will show that the mixing time is $O(n \log n)$ for sufficiently small (but finite) β (again, not all the way up to β_c). Getting both of these results to go all the way to β_c is rather challenging and beyond the scope of this course.

We also mention the following interesting conjecture:

Conjecture: For the Ising model with the + (or −) boundary condition, the mixing time is poly(n) for *all* $\beta > 0$. [Intuition: The obstacle to rapid mixing for $\beta > \beta_c$ is the bottleneck between the plus-phase and the minus-phase; but one of the phases disappears with such a boundary condition.]

We now turn to the result claimed above, that the mixing time is $O(n \log n)$ for sufficiently small β.

Let $d(X, Y)$ denote the number of disagreements between the configurations X, Y. We use path coupling, meaning that we need consider only pairs X, Y that have one disagreement.

Consider X_t, Y_t with disagreement only at i_0. Define the coupling in which X_t and Y_t always pick the same site i and update it "optimally" (i.e., so as to maximize the probability of agreement).

- Good moves: $i = i_0 \to \Delta d = -1$ with probability 1.
- Bad moves: $i \sim i_0 \to \Delta d = +1$ with probability at most $\frac{\exp(2\beta) - \exp(-2\beta)}{2 + \exp(2\beta) - \exp(-2\beta)}$, where $i \sim i_0$ means that i is a neighbor of i_0.

To see this probability of a bad move, let $\alpha = \alpha^+ - \alpha^-$ where α^+ denotes the number of + neighbors of i excluding i_0, and α^- denotes the number of − neighbors excluding i_0.

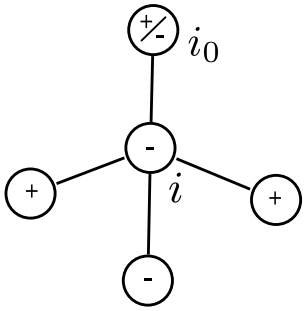


Figure 20.1: Here, X_t, Y_t differ only at node i_0 and are equal everywhere else. Node i is an arbitrary neighbor of i_0. In this example, $\alpha^+ = 2$ and $\alpha^- = 1$.

Letting $\Pr(X_t(i) = +)$ denote the probability that node i is set to $+$ conditioned on its neighbors, then the probability of creating an additional disagreement is $|\Pr(X_t(i) = +) - \Pr(Y_t(i) = +)|$. A simple calculation shows that

$$|\Pr(X_t(i) = +) - \Pr(Y_t(i) = +)| \;=\; \frac{\exp(2\beta) \;- \exp(-2\beta)}{\exp(2\beta\alpha) + \exp(-2\beta) + \exp(2\beta) + \exp(-2\beta\alpha)},$$

which is maximized by letting $\alpha = 0$. This gives the expression claimed above.

Since all other moves leave $d(X_t, Y_t)$ unchanged, the expected change in distance in one move (bearing in mind that each node i_0 has four neighbors), is at most

$$\mathrm{E}[\Delta d] \le \frac{1}{n}\Big[-1 + 4\,\frac{\exp(2\,\beta) - \exp(-2\,\beta)}{2 + \exp(2\,\beta) - \exp(-2\,\beta)}\Big].$$

Therefore, by our usual path coupling analysis, since the maximum distance is at most n, we will get $O(n \log n)$ mixing time if

$$4\,\frac{\exp(2\,\beta) - \exp(-2\,\beta)}{2 + \exp(2\,\beta) - \exp(-2\,\beta)} \;<\; 1.$$

Setting $z = \exp(2\,\beta)$ this becomes $4z - 4z^{-1} \;<\; 2 + z + z^{-1}$. Multiplying out by z and factoring yields $(3z - 5)(z + 1) < 0$, so that we have $O(n \log n)$ mixing time for $\beta < \frac{1}{2}\ln(5/3)$. (Note that this is rather smaller than the critical value $\beta_c = \frac{1}{2}\ln(1 + \sqrt{2})$.)

20.2 Spatial and Temporal Mixing

In this section we develop a relationship between "Spatial Mixing" and "Temporal Mixing." That is, roughly speaking, we will prove that

"Spatial Mixing" (decay of correlations) $\iff$ "Temporal Mixing" (fast mixing time)

For simplicity we will focus on the 2-d Ising model in a $\sqrt{n} \times \sqrt{n}$ box, but our results actually hold for much more general spin systems on more general graphs. We will demonstrate each of the above two implications separately, beginning with the $\Leftarrow$ direction. Let π^+, π^- denote the Gibbs measures (stationary distributions) for the all-plus and all-minus boundary conditions, respectively. Let $\pi^+(\sigma_0 = +)$ denote the probability under π^+ that the origin has spin $+$.

Theorem 20.2 *Suppose, $\tau_{mix} = O(n \log n)$ for all boundary conditions. Then*

$$|\pi^+[\sigma_0 = +] - \pi^-[\sigma_0 = +]| \to 0 \text{ as } n \to \infty.$$

Exercise: With a bit more care in the argument, one can show exponential decay of correlations, i.e., $|\pi^+[\sigma_0 = +] - \pi^-[\sigma_0 = +]| \leq \exp(-c\sqrt{n})$.

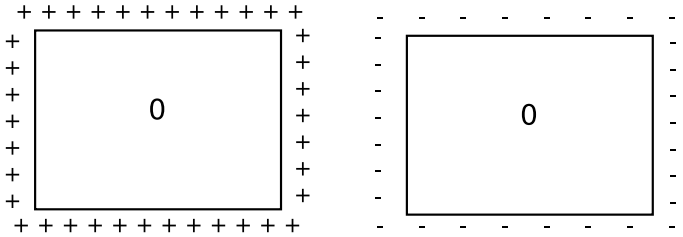


Figure 20.2: (a) Markov chain (X_t) with boundary conditions set to all plus. (b) Markov chain (Y_t) with boundary conditions set to all minus. The initial configurations X_0, Y_0 are identical except on the boundary. The stationary distributions of $(X_t), (Y_t)$ are π^+, π^- respectively.

Proof: Set the initial condition $X_0 = Y_0$ (except on the boundary). Couple (X_t) and (Y_t) such that:

- They choose the same site i to be updated at all times t.
- If the neighborhoods of the i are equal, then they both perform the same update, otherwise they update independently conditional on their neighbors.

We then have

$$\left|\, \pi^+[\sigma_0 = +] - \pi^-[\sigma_0 = +] \,\right| \leq \underbrace{\left|\, \pi^+[\sigma_0 = +] - X_t[\sigma_0 = +] \,\right|}_{a} + \underbrace{\left|\, X_t[\sigma_0 = +] - Y_t[\sigma_0 = +] \,\right|}_{b} + \underbrace{\left|\, Y_t[\sigma_0 = +] - \pi^-[\sigma_0 = +] \,\right|}_{c}$$

Choose $t = Cn \log^2 n > \tau_{mix} \log n$. Then, $a \leq \frac{1}{n} \to 0$ and $c \leq \frac{1}{n} \to 0$, by standard properties of the mixing time. To analyze the remaining term b, note that

$$b \leq \Pr[\,\exists \text{ disagreement at 0 within } t \text{ steps }] = \Pr[\exists \text{ path of disagreement from boundary to 0}].$$

Here, a "path of disagreement" means a contiguous sequence of sites, starting at the boundary and leading to the origin 0, that are chosen to be updated in sequence. Note that only if this happens is it possible for the spins at the origin in X_t, Y_t to differ, since they start out identical except on the boundary.

Now, we may bound the latter term by

$$\Pr[\exists \text{ path of disagreement from boundary to 0}] \leq \sum_{k \geq \frac{1}{2}\sqrt{n}} (4\sqrt{n}\, 4^k) \binom{t}{k} \left(\frac{1}{n}\right)^k,$$

where $(4\sqrt{n}\, 4^k)$ is the number of paths, $\binom{t}{k}$ counts the number of ways of choosing the sequence of update times along the path, and $(\frac{1}{n})^k$ is the probability that this sequence of updates is actually chosen. The above

quantity is then bounded by

$$4\sqrt{n}\sum_{k\geq\frac{1}{2}\sqrt{n}}\Big(\frac{4et}{kn}\Big)^k \;\leq\; \underbrace{4\sqrt{n}}_{\text{“noise”}}\sum_{k\geq\frac{1}{2}\sqrt{n}}\Big(\frac{4ec\log^2 n}{k}\Big)^k \to 0 \text{ as } n\to\infty,$$

■

We turn now to the $\Rightarrow$ direction of the above equivalence: i.e., we will show that exponential decay of correlations implies $O(n\log n)$ mixing time for all boundary conditions. We will actually prove the mixing time bound not for the standard Glauber (heat-bath) dynamics, but for the *block* version of this dynamics, which is the same except that, instead of choosing a random site and updating it, we choose a random $L\times L$ block of sites and update the entire block (again, conditional on the spins on the neighbors of the block). By a result of Peres and Winkler [PW05], this implies a similar mixing time for the single-site version of the dynamics, at least for monotone systems such as the Ising model.

Theorem 20.3 *Suppose we have exponential decay of correlations, i.e., for all pairs of sites i,j,*

$$\big|\pi^{i=+}[\sigma_j=+]-\pi^{i=-}[\sigma_j=+]\big| \;\leq\; \exp\big(-\alpha d(i,j)\big),$$

where $d(i,j)$ is the distance between i and j and $\alpha>0$ is a constant. Then the mixing time of the $L\times L$-block dynamics (for sufficiently large L) is $O(n\log n)$ for all boundary conditions.

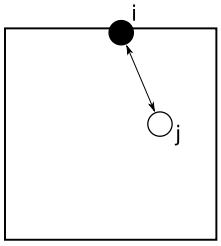


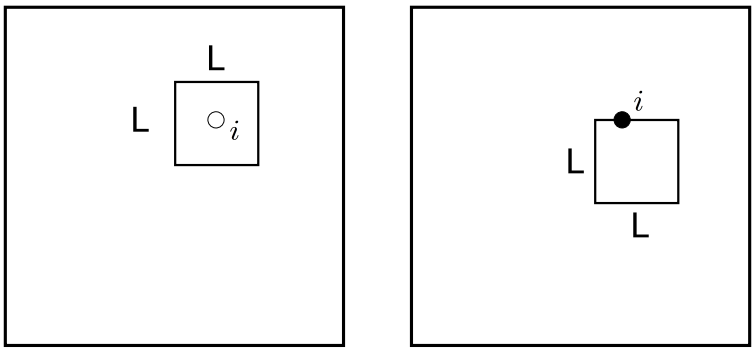
Figure 20.3: The effect of changing the spin at i on the distribution of the spin at j decays exponentially with the distance $d(i,j)$.

Proof: We use path coupling. Consider two copies $(X_t),(Y_t)$ of the Markov Chain with same boundary conditions and arbitrary initial states X_0 and Y_0. Consider the case where X_t and Y_t differ at one site i.

- Good moves: We select a point in the $L\times L$ block centered around i. This implies that $\Delta d=-1$ with probability 1. There are L^2 of these choices.
- Bad moves: We select a point j such the node i is on the edge of the $L\times L$ box centered around the point j. In such a case, $\mathrm{E}[\Delta d]=c'L+L^2\exp\left(-\alpha c\sqrt{L}\right)\leq c''L$ for arbitrarily small $c''>0$ by taking L large enough. (The first term here comes from the sites in the box that are at distance at most $c\sqrt{L}$ from i; clearly there are at most cL of these, for some constant c. The second term comes from the remaining sites in the box, for which we apply the decay of correlations in the hypothesis. By choosing c appropriately small, and then L appropriately large, we can make c'' as small as we like.) There are $4L$ of these choices, corresponding to the size of the boundary of an $L\times L$ block.

  Since all other moves do not change the distance d, we have

$$E[\Delta d]\leq\frac{1}{n}[-L^2+4c''L^2]\leq-\frac{C}{n},$$



(a) (b)

Figure 20.4: (a) Good moves. (b) Bad moves

provided we make c'' small enough. Hence, since the maximum distance is n, by our usual path coupling analysis we get that $\tau_{mix} = O(n \log n)$. ■

References

[M98] F. Martinelli. Lectures on Glauber dynamics for discrete spin models. *Lectures on Probability Theory and Statistics*, Springer Lecture Notes in Mathematics **1717**, 1998, pp. 93–191.

[PW05] Y. Peres, *Mixing for Markov chains and spin systems.* Lecture Notes for PIMS Summer School at UBC, August 2005. www.stat.berkeley.edu/∼peres/ubc.pdf