Math 8213A: Rapidly Mixing Markov Chains
Class Notes, 28, 30 October 1996
Professor Dana Randall
Notes TEXed by Christopher Heckman

We want to use Markov Chains to approximate the number of perfect matchings in a bipartite graph $G = (V_1, V_2, E)$. Let $\mathcal{M}_i$ be the set of matchings of size i and $\mathcal{M}$ the set of all matchings. We can use the telescoping product

$$z_G(\lambda) = \frac{z_G(\lambda)}{z_G(\lambda_{r-1})} \cdot \frac{z_G(\lambda_{r-1})}{z_G(\lambda_{r-2})} \cdot \ \cdots \ \cdot \frac{z_G(\lambda_2)}{z_G(\lambda_1)} \cdot \frac{z_G(\lambda_1)}{1}$$

and calculate each factor in turn.

We have two techniques available to show that this is a good method. Recall from before that we defined for an arbitrary graph G

$$z_G(\lambda) = \sum_{M \in \mathcal{M}} w(M) \lambda^{|M|} = \sum_{k=0}^{n} m_k \lambda^k,$$

where m_k is the number of matchings of size k. Our Markov Chain had the additional Metropolis rule that we change location with probability $\min(1, \frac{w(M')}{w(M)})$. We define the *Gibbs Distribution* to be $\Pi_\lambda(M) = \frac{\lambda^{|M|}}{z_G(\lambda)}$, for $M \in \mathcal{M}$. We will show that the following sampling program is a good algorithm:

1. Set M_0 to be the empty matching;
2. Simulate the Markov Chain (with parameter λ) for $\max_x \tau_x(\epsilon)$ steps;
3. Output the current state.

Definition. *A fully polynomial approximate uniform generator (designated FPAUG) is an algorithm which runs in time* $p(n, \lambda, \log \epsilon^{-1})$ *such that* $\Delta_x(t) = \max_{s \in \Gamma} \left| P^t(x, s) - \Pi_\lambda(s) \right| \leq \epsilon$, *for all applicable* x *and* t, *where* $P^t(x, s)$ *is the distribution after* t *steps as run by the algorithm.*

Definition. *A randomized approximation scheme (for our purposes) is an algorithm which takes as input* (G, λ, ϵ) *and outputs a distribution* y *such that* $\Pr[(1-\epsilon)z \leq y \leq (1+\epsilon)z] \geq 3/4$,[1] *where* $z = z_G(\lambda)$. *We say the algorithm is fully polynomial (designated as FPRAS) if the running time is* $p(n, \max(1, \lambda), \epsilon^{-1})$, *for some polynomial* p.

Now, to calculate $z_G(\lambda)$, we write

$$z_G(\lambda) = \frac{z_G(\lambda_r)}{z_G(\lambda_{r-1})} \cdot \frac{z_G(\lambda_{r-1})}{z_G(\lambda_{r-2})} \cdot \ \cdots \ \cdot \frac{z_G(\lambda_1)}{z_G(\lambda_0)},$$

where $z_G(\lambda_0) = 1$ (which we can get by choosing $\lambda_0 = 0$) and $\lambda_r = \lambda$. The selection of the λ's, as well as approximations by the telescoping product, is done by the following algorithm:

1. Set $\lambda_0 = 0$, $\lambda_1 = \frac{1}{2|E|}$, and $\lambda_i = \lambda_1 (1 + \frac{1}{n})^{i-1}$ for $2 \leq i \leq r$.
2. For $i = 1, 2, \ldots, r$,

   Sample $S = S(\epsilon)$ independent matchings $M_1, M_2, \ldots, M_S$ using the distribution Π_λ;
   Define $X_i = \frac{1}{S} \sum_{j=1}^{S} \left(\frac{\lambda_{i-1}}{\lambda_i} \right)^{|M_j|}$;
3. Output $Y = \prod_{i=1}^{r} (X_i)^{-1}$.

[1] We can change the value of $3/4$ to $1-\delta$ for small δ by using the (familiar) trick of *amplification*, wherein we run the algorithm $2 \log \delta^{-1}$ times and take the median of the y's. We choose the median as the output instead of the mean because we get junk $1/4$ of the time; the mean would be affected by this junk if it contains some extreme values, whereas the median pays no attention to the extreme values, which are simply truncated.

To see that this algorithm is a FPRAS, we define random variables f_i by

$$f_i(M) = \left(\frac{\lambda_{i-1}}{\lambda_i}\right)^{|M|}, \text{where } M \in_R \mathcal{M}.$$

Note that the expected value $E(f_i)$ is

$$E(f_i) = \sum_{M\in\mathcal{M}} \left(\frac{\lambda_{i-1}}{\lambda_i}\right)^{|M|} \cdot \frac{\lambda_i^{|M|}}{z_G(\lambda_i)} = \frac{1}{z_G(\lambda_i)} \sum_{M\in\mathcal{M}} \lambda_{i-1}^{|M|} = \frac{z_G(\lambda_{i-1})}{z_G(\lambda_i)}.$$

Now we want to show that $\Pr[(1-\epsilon)z \le y \le (1+\epsilon)z] \le 3/4$ (or $\Pr[|y/z - 1| > \epsilon] \le 1/4$), where $z = z_G(\lambda)$. By Chebyshev's Theorem, $\Pr[|y/z - 1| > \epsilon] \le \frac{\text{Var }(y/z)}{\epsilon^2}$. So if we can show the variance is bounded by $\epsilon^2/4$, we're done. Thusly,

$$\begin{aligned}
\text{Var }(y/z) &= \frac{\text{Var }(X_1 \cdots X_r)}{E^2(X_1)\cdots E^2(X_r)}, && \text{by definition;}\\
&= \prod_{i=1}^{r}\left(1 + \frac{\text{Var }(X_i)}{E^2(X_i)}\right) - 1, && \text{by some algebra;}\\
&= \prod_{i=1}^{r}\left(1 + \frac{\text{Var }(f_i)}{SE^2(f_i)}\right) - 1, && \text{since Var }(X_i) = \tfrac{1}{S}\text{Var }(f_i);\\
&\le \prod_{i=1}^{r}\left(1 + \frac{1}{SE(f_i)}\right) - 1, && \text{because } f_i \in [0,1], \text{ so Var }(f_i) \le E(f_i);\\
&\le (1 + \tfrac{e}{S})^r - 1 \le \epsilon^2/4,
\end{aligned}$$

if S is big enough, say $S = 5er\epsilon^2$. The last line follows because

$$\frac{1}{E(f_i)} = \frac{z(\lambda_i)}{z(\lambda_{i-1})} = \frac{\sum m_k \lambda_i^k}{\sum m_k \lambda_{i-1}^k} \le \left(\frac{\lambda_i}{\lambda_{i-1}}\right)^n = (1+\frac{1}{n})^n \le e$$

and

$$z_G(\lambda_1) = z_G(\frac{1}{2|E|}) \le \sum_k m_k \cdot \left(\frac{1}{2|E|}\right)^k \le \sum_k \binom{|E|}{k} \cdot \left(\frac{1}{2|E|}\right)^k \le \sum_k 2^{-k} = 2.$$

Our other method defines new Markov Chains MC_i for $i = 1, 2, \ldots, n$ by:

1. Let the state space be $\mathcal{M}_{i-1} \cup \mathcal{M}_i$;
2. Start with matching $M_0 \in \mathcal{M}_{i-1} \cup \mathcal{M}_i$ (via a perfect matching algorithm and edge deletion);
3. For every step, given matching M, let $M' = M$ with probability $1/2$;
   otherwise, choose an edge $e \in_U E$ (i.e., uniformly) and define

$$M' = \begin{cases} M \setminus e, & \text{if } e \in M;\\ M \cup \{e\}, & \text{if } e \text{ can be added (properly) to } M;\\ M \cup \{e\} \setminus \{e'\}, & \text{if } e \text{ is adjacent to exactly one edge, namely } e';\\ M, & \text{otherwise,} \end{cases}$$

   and set $M \leftarrow M'$ iff $M' \in \mathcal{M}_{i-1} \cup \mathcal{M}_i$.

Now we use the algorithm:

1. For each $i = 1, 2, \ldots, n$, do:
   Run the Markov Chain MC_i and produce $2t$ outputs $a_1, \ldots, a_{2t}$;
   Set $f_i \leftarrow \frac{1}{t}|\{j \le t : a_j \in M_i\}|$ and $g_i \leftarrow \frac{1}{t}|\{j > t : a_j \in M_{i-1}\}|$;
   Set $X_i \leftarrow \frac{f_i}{g_i}$;
2. Output $Y = \prod X_i$.

Note that

$$f_i \approx \frac{|\mathcal{M}_i|}{|\mathcal{M}_{i-1} \cup \mathcal{M}_i|}, \quad g_i \approx \frac{|\mathcal{M}_{i-1}|}{|\mathcal{M}_{i-1} \cup \mathcal{M}_i|}, \quad X_i \approx \frac{f_i}{g_i} \approx \frac{|\mathcal{M}_i|}{|\mathcal{M}_{i-1}|}, \text{ and } \quad Y \approx \frac{|\mathcal{M}_n|}{|\mathcal{M}_0|} = |\mathcal{M}_n|.$$

if we choose enough samples. To make things easier for us to prove, we will assume that D is a distribution over the set U such that $\frac{1}{2}\sum_{x\in U} |D(x) - \Pi(x)| < \xi$, and $|D(x) - \Pi(x)| < \xi$, for all $x \in U$, where $0 < \xi < 1$. To see how many samples we need, we prove the claim that:

Claim. *Let $S \subset U$ and $p = \sum_{i\in S} \Pi(i)$. Let p' be the expected value of X and ξ and δ be positive numbers. Then* $\Pr[(1-9\xi)\,X \le p \le (1+9\xi)\,X] \ge 1-\delta$, *where*

$$p = \frac{|\mathcal{M}_i|}{|S|} \text{ and } t \ge \frac{3}{\xi^2 p} \ln\left(\frac{2}{\delta}\right).$$

Proof: We claim that $(1-4\xi)\,p' \le p \le (1+4\xi)\,p'$. (This follows from the worst-case result, namely:

$$D(x) = \begin{cases} (1+\xi)\Pi(x), & \text{if } x \in S_i; \\ (1-\xi)\Pi(x), & \text{otherwise.} \end{cases}$$

Then we find that

$$\frac{1-\xi}{1+\xi}\,p' \le p \le \frac{1+\xi}{1-\xi}\,p'$$

and since $\frac{1+\xi}{1-\xi} = 1 - 2\xi + 4\xi^2 - \ldots \ge 1 - 4\xi$, the smaller claim holds.) Now we note that

$$\begin{aligned} \Pr[(1-9\xi)\,X \le p \le (1+\xi)\,X] &\ge \Pr[(1-\xi)\,X \le p' \le (1+\xi)\,X] \\ &= \Pr[(1-\xi)\,p' \le X \le (1+\xi)\,p'] \\ &= \Pr[|X - p'| \le p'\xi] \\ &\ge 1 - 2e^{-\frac{\xi^2 p^t}{3}} \ge 1 - \delta, \end{aligned}$$

the last line following from the Chernoff Bound and the definition of t. ∎

Given ϵ and n from the input, we will use this result, where $\xi = \frac{\epsilon}{36n}$ and $\delta = \frac{1}{8n}$. Now suppose we have estimates for each f_i (as defined in our algorithm) to within $1 + \frac{\epsilon}{4n}$ with confidence $1 - \frac{1}{8n}$.

Claim. *Y estimates $|\mathcal{M}_n|$ to within $1+\epsilon$ with probability 3/4.*

Proof: For each i,

$$\begin{aligned} \Pr[(1-\tfrac{\epsilon}{2n})X_i \le \tfrac{|\mathcal{M}_i|}{|\mathcal{M}_{i-1}|} \le (1+\tfrac{\epsilon}{2n})X_i] &\ge \Pr[\tfrac{1-\frac{\epsilon}{4n}}{1+\frac{\epsilon}{4n}}X_i \le \tfrac{|\mathcal{M}_i|}{|\mathcal{M}_{i-1}|} \le \tfrac{1+\frac{\epsilon}{4n}}{1-\frac{\epsilon}{4n}}] \\ &\ge \Pr[\left(1-\tfrac{\epsilon}{4n}\right) f_i \le \tfrac{|\mathcal{M}_i|}{|\mathcal{M}_i \cup \mathcal{M}_{i-1}|} \le \left(1+\tfrac{\epsilon}{4n}\right) f_i \\ &\qquad \text{and } \left(1-\tfrac{\epsilon}{4n}\right) g_i \le \tfrac{|\mathcal{M}_{i-1}|}{|\mathcal{M}_i \cup \mathcal{M}_{i-1}|} \le \left(1+\tfrac{\epsilon}{4n}\right) g_i] \\ &\ge \left(1-\tfrac{1}{8n}\right)^2 \ge 1 - \tfrac{1}{4n}, \text{ so} \end{aligned}$$

$$\begin{aligned} \Pr[(1-\epsilon)Y \le |\mathcal{M}_n| \le (1+\epsilon)Y] &\ge \Pr[(1-\tfrac{\epsilon}{2n})^n \textstyle\prod_{i=1}^n X_i \le Y \le (1+\tfrac{\epsilon}{2n})^n \prod_{i=1}^n X_i] \\ &\ge \Pr[(1-\tfrac{\epsilon}{2n})X_i \le \tfrac{|\mathcal{M}_i|}{|\mathcal{M}_{i-1}|} \le (1+\tfrac{\epsilon}{2n})X_i, \text{ for all } i = 1, \ldots, n] \\ &\ge \left(1-\tfrac{1}{4n}\right)^n \ge 3/4. \blacksquare \end{aligned}$$

Note that our first algorithm is polynomial in n and $\log \xi^{-1}$ and the second is polynomial in n, ϵ^{-1}, $\log \delta^{-1}$, and $\frac{|\mathcal{M}_{i-1}|}{|\mathcal{M}_i|}$. To get a fully polynomial algorithm, we will need to bound the last term. This can be seen as equivalent to bounding $\frac{|\mathcal{M}_{n-1}|}{|\mathcal{M}_n|}$, by concavity. Unfortunately, there is no universal polynomial bound, but there are several classes of graphs which have such bounds. It is interesting to note that a random graph (with each edge having probability $p = {}^c/_n$, for some $c > 0$) is likely to be so bounded. (This is due to Jerrum and Sinclair.)

Definition. *A bipartite graph $G = (V_1, V_2, E)$ with $|V_1| = |V_2| = n$ is said to be dense if for all $v \in V_1 \cup V_2$, $\deg v \geq cn$, for some constant $c > 0$.*

Theorem (Broder). *If G is a dense graph (with $c = 1/2$), then*

$$\frac{|\mathcal{M}_{n-1}|}{|\mathcal{M}_n|} \leq n^2.$$

Proof: Define a map $\phi : \mathcal{M}_{n-1} \longrightarrow \mathcal{M}_n \times E$ in the following way: Let u, v be the two (unique) unmatched vertices in $M \in \mathcal{M}_{n-1}$. Then let

$$\phi(M) = \begin{cases} \Big(M \cup \{(u,v)\}, (u,v)\Big), & \text{if } (u,v) \in E; \\ \Big(M \setminus \{(u',v')\} \cup \{(u,v'),(u',v)\}, (u,v)\Big), & \text{otherwise,} \end{cases}$$

where the vertices u', v' are such that $(u', v') \in M, (u, v') \in E$, and $(u', v) \in E$ (note this is an augmenting path of length 3). Such vertices exist because u, v have degree $\geq n/2$, and so some neighbor of u must be connected to a neighbor of v in M, by the Pigeonhole Principle. Since ϕ is invertible (with respect to its image, as it may fail to be surjective), it is injective, which is equivalent to the result. ∎

Remark: Dagum and Luby showed that this result holds for any $c > 0$.

Definition. *A periodic lattice graph (with one set of periodic boundary conditions) is a bipartite graph with vertex set $V = Z_k \times [a_1, b_1] \times \ldots \times [a_m, b_m]$ which has an even number of vertices, $k \geq 2$, $m \geq 1$, and $E = \{(u,v) \in V^2 : \sum_{i=0}^{m} (u_i - v_i)^2 = 1\}$. The 0^{th} coordinates are said to be subject to periodic boundary conditions, and the i^{th} coordinates are said to be subject to non-periodic boundary conditions, for $1 \leq i \leq m$.*

Theorem (Kenyon, Randall, Sinclair). *If G is a lattice graph, then*

$$\frac{|\mathcal{M}_{n-1}|}{|\mathcal{M}_n|} \leq n^2.$$

Proof: Let $\mathcal{M}_{n-1}(u,v)$ be the set of near-perfect matchings such that $u, v \in V$ are the only unmatched vertices. Then it is enough to show that $|\mathcal{M}_{n-1}(u,v)| \leq |\mathcal{M}_n|$, because we can add up these inequalities over all u, v. Furthermore, we will show that $|\mathcal{M}_{n-1}(u,v)|^2 \leq |\mathcal{M}_n|^2$, which is clearly equivalent to what we want.

For a vertex u, define the vertex u' to be $u + (1, 0, \ldots, 0)$. We trivially have that $|\mathcal{M}_{n-1}(u,v)| = |\mathcal{M}_{n-1}(u',v')|$, because we can map matchings to matchings in the obvious way. Now, to prove the result, we show that there is a function $\phi : \mathcal{M}_{n-1}(u,v) \times \mathcal{M}_{n-1}(u',v') \longrightarrow \mathcal{M}_n \times \mathcal{M}_n$ which is invertible (with respect to its image).

Consider two matchings, $M \in \mathcal{M}_{n-1}(u,v)$ and $M' \in \mathcal{M}_{n-1}(u',v')$. In the graph induced by edge-set $M \oplus M'$, every vertex has degree 2, except for u, v, u', and v'. If we add the edges

$(u, u'), (v, v')$, we obtain a 2-regular graph; (part of) this setup is shown below, for the case $m = 1$:

M

Now, the cycle C containing the edge (u, u') must also contain the edge (v, v'); otherwise, C would be an odd cycle in a bipartite graph. (For the proof of this, trace C in both directions, edge by edge, from the edge (u, u'); because of the alternation between edges of M and M', there is no edge connecting two previous edges. Thus, after the cycle has been traced, two edges (one each from M and M') are adjacent. But then C is an odd cycle. (It has the "unmatched" edge and a set of edges from M and a set from M'. The number of edges in the last two sets is equal, because every other edge *is* matched.) The picture below shows this.)

So, if we alternate the edges between u and v on one side (from u to u' to v (or v', whichever is encountered first))[2], then we obtain an even cycle with alternating edges. This, with the other even cycles (and the edges $e \in M \cap M'$), we obtain a graph which is the sum of two perfect matchings, which can be read off in a well-defined (canonical) way.

Note that, if we have two perfect matchings which were constructed this way, and the special vertices u, v (subject to the additional constraint that they are in the same "component"), we can recover the original (near-perfect) matchings by reversing this process, so ϕ is invertible (with respect to its image), so is injective, and the result holds.

Remark: The result also holds for two-dimensional lattice graphs with non-periodic boundary conditions.

[2] We choose this rule to be canonical (i.e. in a consistent way) to avoid a 2-1 function.