Randomized Algorithms

Philip Keen, supervised by Dr Deryk Osthus

April 5, 2007

0.1 Declaration

I warrant that the content of this dissertation is the direct result of my own work and that any use made in it of published or unpublished material is fully and correctly referenced.

Signed:

Date:

0.2 Introduction

A randomized algorithm is an algorithm that, at some point, makes a random decision that affects its future behaviour. There are many situations where such an algorithm is useful in solving a problem. It may be for such a problem that using a determinsitic algorithm to find a solution is extremely time-consuming. It could even be that such an algorithm does not exist. So randomized algorithms can offer fast solutions. Unfortunately, as will be seen, they have limited reliability. It is not always guaranteed that a good, or correct, solution will be found.

This project will consist of two major parts, each considering a counting problem where a randomized algorithm can give an estimated solution. The first part considers the problem of determining the permanent of a (0,1)-matrix, which turns out to be equivalent to counting the number of perfect matchings in a bipartite graph. The second part considers the problem counting the number of k-vertex colourings in a graph. All the algorithms presented will try to take the same basic approach: attempting to give a good estimate by sampling from the input.

Chapter 1

A Problem with (0,1) Matrices

1.1 The Class #P

#P deals with a different sort of problem to the more familiar decision problem. A decision problem is concerned with classifying an object into a class **A** or class not-**A**. So questions such as "Does this graph have a Hamilton cycle?" or "Is this statement self-contradictory?" can be seen as reflecting decision problems. In contrast, **#P** is related to counting. How it is defined though, is dependent upon a particular class of decision problems: **NP**. For the following definition of **#P**, we follow L. Valiant [11].

Let $\mathcal{T}$ be a non-deterministic Turing machine (NTM) associated with a particular decision problem. $\mathcal{T}$ is a *counting machine* if $\mathcal{T}$ has an extra tape (the counting tape) upon which it writes, when $\mathcal{T}$ is given input I, the number of computations that $\mathcal{T}$ could make that would result in $\mathcal{T}$ accepting I. This is equivalent to writing down the number of solutions to I. So if $\mathcal{T}$ was a counting NTM that decided Hamiltonicity, then for any graph G, $\mathcal{T}$ would decide if G contained a Hamilton cycle and on the counting tape write down how many Hamilton cycles are contained in G.

Let $\mathcal{T}$ be a counting machine and let $f_{\mathcal{T}}$ be the function that maps any input I of $\mathcal{T}$ to $\mathcal{T}$'s consequent output on the counting tape. $f_{\mathcal{T}}$ is thus the number of solutions to I. We say the $f_{\mathcal{T}} \in$**#P** if and only if $\mathcal{T}$ is able to decide every input I in time polynomial in the length of I. In other words, $f_{\mathcal{T}} \in$**#P** if and only if the problem that $\mathcal{T}$ decides is in **NP**. Note that the definition of **#P** depends on the time taken to *classify* an input I, not the time it would take to find or write down the number of solutions to I. More informally, **#P** is the class of counting problems that are associated with decision problems in **NP**.

Now, we say that a function f is *#P-complete* if $f \in$**#P** and $\forall g \in$ **#P** there is a polynomial-time reduction from g to f. That is to say, a counting problem is **#P**-complete if it is a **#P** problem and furthermore, any other counting problem in **#P** can be easily turned into a counting problem of the type concerned. So given a **#P**-complete problem y, every other problem in **#P** can be seen as a type of y problem "in disguise".

1.2 A Problem Presented: The Permanent of a (0,1) Matrix

If $\mathbf{A}$ is an $n \times n$ matrix and S_n is symmetric group on {1,. . . ,n}, then the *permanent* of $\mathbf{A}$ is defined to be

$$per(\mathbf{A}) = \sum_{\pi \in S_n} \prod_{i=1}^{n} a_{i,\pi(i)}$$

Thus the permanent of $\mathbf{A}$ has a definition broadly similar to that of the determinant of $\mathbf{A}$:

$$det(\mathbf{A}) = \sum_{\pi \in S_n} (sgn(\pi)) \prod_{i=1}^{n} a_{i,\pi(i)}$$

Allowing for a slight abuse of language, this suggests that the permanent is simply the determinant with addition substituted for subtraction.

The above definition of the permanent of $\mathbf{A}$ suggests a method for calculating this. We take all sequences $\alpha = a_{1,j_1}, a_{2,j_2}, \ldots, a_{n,j_n}$ of entries of $\mathbf{A}$, such that $j_k \neq j_l$ for $k \neq l$. Then for each such α, work out the product of its elements. Adding the results together will yield $per(\mathbf{A})$. The problem is that this method is hopelessly inefficient: we have to consider $|S_n| = n!$ sets drawn from $\mathbf{A}$.

Maybe all is not lost however. After all, except for $sgn(\pi)$, the definitions of $per(\mathbf{A})$ and $det(\mathbf{A})$ are identical. It is well-known that there is a deterministic, polynomial-time algorithm for calculating $det(\mathbf{A})$, so maybe the same is true for the permanent. Would it also be easier if we restrict $\mathbf{A}$ to being a member of a 'simple' class of matrices: the $n \times n$ (0,1) matrices? An $n \times n$ (0,1) matrix is simply a square matrix with entries either 0 or 1. Unfortunately, this is not the case. Valiant [11] showed in 1979 that computing the permanent of an $n \times n$ (0,1) matrix is **#P**-complete. For reasons that will become clear later on, it will be instructive to glance at why this is so. So we must observe two things: namely that the problem under consideration is a **#P** problem, and that every other **#P** problem can be reduced to this problem in polynomial time.

1.2.1 Determining the Permanent of a (0,1) Matrix is #P

Initially, it might not be obvious that the calculation of $per(\mathbf{A})$ for all $n \times n$ (0,1) matrices $\mathbf{A}$ should lie within **#P**. After all, what is $per(\mathbf{A})$ *counting*?

But if the problem is phrased correctly, the reason for locating the problem in **#P** becomes clear.

The idea is to regard $\mathbf{A}$ as a specialized adjacency matrix for a bipartite graph G. We let G be a graph of order $2n$ with vertex classes X and Y such that $|X| = |Y| = n$. For all $x_i \in X$ and for all $y_j \in Y$ we let $x_i y_j \in E(G)$ if and only if $a_{i,j} = 1$. In this way we regard $\mathbf{A}$ as an $X \times Y$ adjacency matrix for G. The rows of $\mathbf{A}$ will indicate the neighbours of vertices in X, the columns will indicate the neighbours of vertices in Y.

With regard to the permanent of $\mathbf{A}$, a sequence α drawn from $\mathbf{A}$ will contribute a product of 1 to the overall sum if and only if each element in α is 1. So calculating $per(\mathbf{A})$ is exactly the same as counting the number of α-sequences from $\mathbf{A}$ that do not contain a 0. But non-zero entries from $\mathbf{A}$ correspond exactly to the edge set of G. So if an α-sequence does not contain a 0, then, in a sense, it has picked out n edges from G. No two of these edges can share an endpoint, otherwise that means the α-sequence has picked two entries from the same row or column, and this cannot happen. Thus, any such α-sequence picks out a perfect matching in G. The converse holds as well. If G has a perfect matching, then there must be n 1s in $\mathbf{A}$ corresponding to these edges. Since no two share an endpoint, no two of the 1s can lie in the same row or column. But then there must be an α-sequence that picks these out. So it should now be clearer that the problem properly belongs to **#P**: $per(\mathbf{A})$ counts the number of perfect matchings in G. Since the question of whether G has a perfect matching is **NP**, the permanent counting problem is in **#P**. To show that the problem is **#P**-complete then, it need only be shown that every problem in **#P** can be turned, in polynomial time, into a problem of calculating the permanent of a (0,1) matrix.

1.2.2 Valiant's Reduction

Valiant [11] managed to show that any **#P** problem can be so transformed by using statements in conjunctive normal form (CNF). He showed that counting the number of satisfying truth assignments of any such statement S could be reduced in polynomial time to calculating the permanents of a set of (0,1) matrices. The technical details need not concern us. But counting the number of satisfying truth assignments for S is **#P**-complete. By implication then, so is the problem of finding the permanent of a (0,1) matrix.

Why should this dampen hopes for a polynomial time algorithm for calculating $per(\mathbf{A})$? Suppose that counting machine $\mathcal{T}$ was faulty and that the output tape for the decision problem was jammed. If $\mathcal{T}$'s counting tape is still working then this will still give us enough information to classify an input I. I will have no solutions if and only if the number on the counting tape is 0. Now, from the previous section, if there is a polynomial time algorithm for calculating the permanent of a matrix, then there is a polynomial time algorithm for counting the number of satisfying truth assignments for any CNF statement S. But this will tell us if S *has* any satisfying truth assignments, and this problem is **NP**-complete. So if such an algorithm exists for the permanent, it follows that $\mathbf{P} = \mathbf{NP}$.

1.3 Lowering Expectations

Suppose we relax the requirement of *determining* the permanent of a matrix $\mathbf{A}$, and instead are happy with an *approximation* of the value. Then the situation is not so hopeless. To achieve this task, our aim will be to find a *fully polynomial randomized approximation scheme* that works.

Definition 1.1. A *randomized algorithm* is an algorithm that, at some step, makes a random choice that affects its future behaviour.

The following definition follows [9].

Definition 1.2. Let I be an instance of a $\#\mathbf{P}$ counting problem and let $\#(I)$ be the number of solutions to I. A *fully polynomial randomized approximation scheme* (FPRAS) is a randomized algorithm that takes I, $\epsilon > 0$ and $\delta < 1$ as inputs, and in time polynomial in $|I|, \frac{1}{\epsilon}, ln\frac{1}{\delta}$ produces an output $F(I)$ such that

$$\mathbf{Pr}[(1-\epsilon)\#I \leq F(I) \leq (1+\epsilon)\#I)] \geq 1-\delta$$

So such an algorithm takes a problem, the number of solutions to which is uncertain, and with ϵ and δ placing bounds on the likely error, outputs a final value.

In what follows, a randomized algorithm will be presented for approximating the permanent of a particular type of $n \times n$ (0,1) matrix. This matrix must be dense with 1's - every row and column must have at least $\frac{1}{2}$ its entries equal to 1. It will be shown that for any such matrix $\mathbf{A}$ a good approximation for the permanent can be found efficiently, and that indeed this algorithm is a fully polynomial randomized approximation scheme for the matrices concerned. This algorithm is described by Rajeev Motwani and Prabhakar Raghavan[10], and is drawn from work by Mark Jerrum and Alistair Sinclair [6].

1.3.1 A Randomized Algorithm for Approximating the Permanent of an $n \times n$ (0,1) Matrix

This randomized algorithm is based on the idea used in showing that calculating $per(\mathbf{A})$ was a counting problem. Instead of approaching the problem in an algebraic manner, the algorithm treats $\mathbf{A}$ as the type of adjacency matrix described above. It attempts to estimate the number of perfect matchings in the associated graph.

Let $\mathbf{A}$ be the matrix under consideration. So each $a_{i,j} = 1$ or 0. Let τ,ω be positive integers. Define a bipartite graph G as follows:
Let $V(G) = \{x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n\}$. So $|G| = 2n$.
Let $(x_i, y_j) \in E(G)$ if and only if $a_{i,j} = 1$. Note that G will then be bipartite with vertex classes $X := \{x_1, x_2, \ldots, x_n\}$ and $Y := \{y_1, y_2, \ldots, y_n\}$.
Define M_k to be the set of all matchings in G of size k. The idea will be to try and sample randomly from $M_n \cup M_{n-1}, M_{n-1} \cup M_{n-2}, \ldots, M_2 \cup M_1$. Using the relative frequencies of each size of matching in each $M_k \cup M_{k-1}$, we can estimate $\frac{|M_n|}{|M_{n-1}|}, \frac{|M_{n-1}|}{|M_{n-2}|}, \ldots, \frac{|M_2|}{|M_1|}$. But $|M_1| = e(G)$. So the product of the ratios when multiplied by $e(G)$ will give an approximation of $per(\mathbf{A})$.

Proceed as follows:

1. Choose any edge $e \in E(G)$. Let E be a "seed matching" from M_1. Let $k = 2$

2. Produce a new matching by using the seed matching as a starting matching m. This will be done in τ steps using a series of random operations described below, Now for each step:

   - Decide with probability $\frac{1}{2}$ whether do do anything this step. If so, continue with one of the operations below. If not, remain idle for this step
   - If we have chosen to do something, define a new matching m' by choosing an edge $e' \in E(G)$ uniformly at random and then doing one of the following four things:
     (a) **Reduce** If $m \in M_k$ and $e' \in m$ then let $m' := m - e' \in M_{k-1}$
     (b) **Increase** If $m \in M_{k-1}$ and $e' \notin m$ then let $m' := m + e' \in M_k$
     (c) **Rotate** If $e' \notin m$ but e' shares an endpoint with exactly one edge $e'' \in m$ then let $m' := (m - e'') + e'$
     (d) **Idle** If none of the operations **Reduce**,**Increase**,**Rotate** can be performed using e', do nothing for this step.
   - Stop after τ steps and record on a "running tally" whether the final $m' \in M_k$ or $m' \in M_{k-1}$.

3. Repeat the 'generating' procedure ω times. If any running of the procedure resulted in a $m' \in M_k$, consider the last such matching produced as a new seed matching

4. Use the relative frequencies in the running tallies to estimate $\frac{|M_k|}{|M_{k-1}|}$. If $|M_k|$ is estimated to be 0, finish the entire algorithm and output the estimated value $per(\mathbf{A}) = 0$

5. If $k < n$, increase k by 1 and repeat from step
6. By this step, estimations for $\frac{|M_n|}{|M_{n-1}|}, \frac{|M_{n-1}|}{|M_{n-2}|}, \ldots, \frac{|M_2|}{|M_1|}$ should have been obtained. Note that no $|M_k|$ should be estimated to be 0 at this stage, else the algorithm would have finished beforehand
7. Calculate $\frac{|M_n|}{|M_{n-1}|}\frac{|M_{n-1}|}{|M_{n-2}|}\cdots\frac{|M_2|}{|M_1|}e(G)$ and output the result as a final approximation for $per(\mathbf{A})$

Having described the algorithm, there are several questions that suggest themselves:

1. How good is the algorithm at estimating $\frac{|M_r|}{|M_{r-1}|}$ for any given $2 \leq r \leq n$?
2. If the aim is to produce a fast algorithm, why does it include a step that deliberately slows it down (ie. the decision with probability $\frac{1}{2}$ to do nothing)?
3. Can the algorithm generate a matching efficiently (ie. in polynomial time)?
4. What value must ω take?

Clearly, the speed of the algorithm ultimately depends on τ and ω. It is safe to assume that the final manipulation of the ratios takes negligible time. In what follows we will attempt to answer the questions given. We will show that the algorithm can produce acceptable measures for each $\frac{|M_r|}{|M_{r-1}|}$. What is more, it will be shown that such values can be obtained when the algorithm runs in polynomial time. Furthermore, the total number of samples needed will be shown to have a polynomial bound. Much of the analysis of the algorithm be drawn from Motwani and Raghavan [10].

Chapter 2

Modelling Matching Generation

To get bounds for τ, we will have to find some way to analyze how the matching-generator behaves. The generator, in effect, starts with one matching and then attempts to find, as randomly as possible, another matching to swap in its place. It does this successively for τ steps. As an aside, note that the search is blind; the algorithm never actually generates $M_r \cup M_{r-1}$ to sample from. How are we to model such a series of swappings, then? Following Motwani and Raghavan [10], we use the most obvious tool available: the Markov chain.

2.1 Markov Chains

Definition 2.1. Let discrete set $T := \{t_0, t_1, t_2, \ldots\}$ be finite or countably infinite. Let S be a finite set and $S \times S$ be a probability space with associated probability function $P : S \times S \to [0, 1]$. A *Markov chain* on S with respect to T is a sequence of random variables $X_{t_0}, X_{t_1}, X_{t_2}, \ldots$ indexed by T such that each $X_i \in S$ and $\mathbf{Pr}(X_{t_{i+1}} = s_k | X_{t_i} = s_j) = P((s_j, s_i))$.

This rather abstract definition does actually capture an idea that is common-sense in operation. We will see this if we show that it applies to our algorithm. Suppose we are attempting to sample from $M_r \cup M_{r-1}$ and start with $m \in M_{r-1}$. Notice that as the algorithm progresses, it will only work through matchings from $M_r \cup M_{r-1}$. So we may take S in this case to be $M_r \cup M_{r-1}$. That means that each X_{t_i} will be a matching. T we we may take, as usual with Markov chains, to be time, measured in discrete time steps. No doubt parameters other than time are possible for T, but a time-dependent process is the most natural setting for a Markov chain. Note that to model something as a Markov chain, it is necessary that the value of X_{t_i} depend only on $X_{t_{i-1}}$, not on any states X_{t_j} before then.

Our algorithm does behave in this manner. What matching m' is swapped for m depends only on an edge randomly selected without any reference to previous matchings.

Rather than a sequence of variables $X_{t_0}, X_{t_1}, X_{t_2}, \ldots$, we can regard a Markov chain as a single random variable X which acts as a discrete-step function from T to S. In this sense, X moves between elements of S as T changes. Viewed in this way, we can see the Markov chain as taking place on a directed multigraph $\overrightarrow{D}$ with $V(\overrightarrow{D}) = S$. The transition probabilities between states will be weights of the edges. The chain will then be a random walk on this multigraph. X will be able to go from s_i to s_j for $s_i, s_j \in S$ if and only if $\overrightarrow{s_i s_j}$ is an edge of $\overrightarrow{D}$.

Let us define an appropriate directed multigraph $\mathcal{Q}$ in the current context. For a running of the matching generator attempting to sample from $M_r \cup M_{r-1}$, let $V(\mathcal{Q}) := M_r \cup M_{r-1}$. Let each edge in $\mathcal{Q}$ have weight $\frac{1}{2|E|}$, where $|E| = e(G)$. When moving from X_{t_i} to $X_{t_{i+1}}$, we know that the Markov chain has to do something. Thus, if the associated walk is currently at matching m_i, it must move to another matching m_j or stay put with probability 1. Hence each vertex in $\mathcal{Q}$ must have outdegree $2|E|$ to reflect this. These $2|E|$ edges must be positioned so as to reflect the possible movements of the chain. If we look back at the algorithm we note that at each step the algorithm decided with probability $\frac{1}{2}$ whether to do anything that step. If, for each vertex, we make $\frac{1}{2}(2|E|) = |E|$ edges self-loops, then this will reflect that decision. The other $|E|$ edges can be positioned to reflect all the $|E|$ possible actions resulting from choosing an edge of G at random. In this way, if the choice of an edge results in a **Reduce**,**Increase** or **Rotate** operation being performed on the current matching m, then there will be an edge from m to the resulting matching m'. If the choice of edge results in an **Idle** step, then the associated edge is a self-loop.

2.1.1 The Structure of $\mathcal{Q}$

We may begin here by making a few observations. $\mathcal{Q}$ has regular outdegree, namely $2|E|$. Also note that each operation **Reduce**,**Increase** and **Rotate** is reversible. If **Reduce** produced matching $m - e$ from m, then there is an **Increase** operation that produces $(m - e) + e = m$ from $m - e$. The converse holds as well. Also note that if **Rotate** produces matching $(m - e) + e'$ from m then there is another **Rotate** operation that gets $(((m - e) + e') - e') + e = m$ from $(m - e) + e'$. So for all distinct m_1, m_2 in $V(\mathcal{Q})$, if $\overrightarrow{m_1 m_2}$ is a directed edge in $\mathcal{Q}$ then $\overleftarrow{m_1 m_2}$ is as well. Now, suppose we take a self-loop to contribute 1 to vertex's outdegree and 1 to its indegree as well. Then each vertex in $V(\mathcal{Q})$ has regular indegree, since for every edge it sends out it receives an edge.

In this section, we will also show that $\mathcal{Q}$ is connected. This will require more work. To aid us, we will turn our attention to the graph G. It is worthwhile noting here that G and $\mathcal{Q}$ are separate structures. In fact, they do not have many features in common. This section uses arguments taken from [10].

Proposition 2.1. Let m be a matching in G of size $n' < n$. Then m has an augmenting path of length at most 3.

Proof. Let $a \in X$, $b \in Y$ be unmatched vertices in G. If one of a, b has an unmatched vertex v as a neighbour then either av or bv will be an augmenting path for m of length 1. So suppose both $N_G(a)$ and $N_G(b)$ are covered by m. Now $d_G(a), d_G(b) \geq \frac{n}{2}$ by assumption. So $|N_G(a)| \geq \frac{|Y|}{2}$. Let $X' \subseteq X$ be all the vertices matched to vertices in $N_G(a) \subseteq Y$ by m. So $|X'| = |N_G(a)| \geq \frac{|Y|}{2} = \frac{|X|}{2}$. Clearly, $N_G(b) \subseteq X$. Since a is not covered by m, $a \notin N_G(b)$ and $a \notin X'$. So $N_G(b) \cup X' \subseteq X - a$ and $|N_G(b)| + |X'| \geq \frac{|X|}{2} + \frac{|X|}{2} = |X| > |X| - 1 = |X - a|$. Then by the pigeonhole principle, there must be an $a' \in N_G(b) \cap X'$. Hence, a' is matched to some $b' \in Y$ by m. Therefore, $ab'a'b$ is an augmenting path of length 3 for m. $\square$

Given the cautionary note before the statement of the Proposition, we may well wonder how talking about G will help in showing that $\mathcal{Q}$ is strongly connected. This is because augmenting paths enable us to produce new matchings.

Proposition 2.2. $\mathcal{Q}$ is strongly connected.

Proof. $\mathcal{Q}$ is strongly connected if and only if for any distinct x, y in $V(\mathcal{Q})$ there is a directed path from x to y. It was observed before that for every edge $\overrightarrow{v_1v_2}$ there is an opposing edge $\overleftarrow{v_1v_2}$. As a consequence we need only prove that for any two vertices, a directed path exists between them in one direction.

Suppose $V(\mathcal{Q}) = M_r \cup M_{r-1}$. Let $m \in M_{r-1}$. We know from the Proposition above that m has an augmenting path of length at most 3. We saw in the proof that this path must have length 1 or 3. If m has an augmenting path ab of length 1 then there is an **Increase** operation from m to matching $m + ab \in M_r$. Suppose m has an augmenting path $ab'a'b$ of length 3. Then there is a **Rotate** operation that takes m to $(m - b'a') + a'b =: m' \in M_{r-1}$. There is a further **Increase** operation that produces $m' + ab' \in M_r$. Therefore, if $m \in M_{r-1}$ then there is an $m'' \in M_r$ such that there is a directed path from m to m''. If we manage to show that for each distinct $m_1, m_2 \in M_r$ there is a directed path from m_1 to m_2 we are done. M_r can then act as a "hub". For any two vertices in $V(\mathcal{Q})$ there will be a directed path between them that goes through M_r.

Let m_1, m_2 be matchings in M_r, then. Consider $m_1 \triangle m_2$. Since $|m_1| = |m_2|$, $|m_1 \triangle m_2|$ must be even. This is because for every edge of m_1 not in

m_2 there must be an edge of m_2 not in m_1, and *vice versa*. Also, note that a vertex in $m_1 \triangle m_2$ must have degree at most 2, otherwise m_1 or m_2 have edges that meet, and this cannot happen. This forces the constituent parts of $m_1 \triangle m_2$ to be cycles or paths. Consider a cycle C in $m_1 \triangle m_2$. Since both m_1 and m_2 cannot have edges that meet, the edges of C must alternate between m_1 and m_2. So C has even length. For any path P in $m_1 \triangle m_2$ its edges must alternate between m_1 and m_2, as for C.

Hence $m_1 \triangle m_2$ consists of alternating paths and cycles. We can use these to move from m_1 to m_2 in $\mathcal{Q}$.

Let C be a cycle in $m_1 \triangle m_2$. Then we can produce a new matching $m_\alpha \in M_r$ from m_1 by using a process called "unwinding the cycle". Let $v_1v_2 \in m_1$ lie on C. Regard C as being $v_1v_2v_3 \ldots v_1$. **Reduce** m_1 to get $m_1 - v_1v_2 \in M_{r-1}$ (Throughout all the manipulations of the matchings, we must always be careful to remain in $M_r \cup M_{r-1}$ else we go "off the map" as far as the Markov chain is concerned). Now, $v_2v_3 \in m_2$ and $v_3v_4 \in m_1$. As $m - 1$ has lost v_1v_2, there is nothing to prevent from using the **Rotate** transition to swap v_2v_3 for v_3v_4 in the current matching. We can then go around the cycle C like this, swapping an edge in m_1 for the one proceeding it in m_2. As **Rotate** does not alter the size of a matching, any matching produced in this way will still lie in M_{r-1}. However, this process will finish with an isolated edge $v_kv_1 \in m - 2$, because C alternates. But then an **Increase** operation can be used to add v_kv_1 to the current matching, and so gain a matching in M_r (see Figure 2.1).

Suppose P is a path in $m_1 \triangle m_2$. If P is of even length, then $P = v_1v_2v_3v_4 \ldots v_{2k+1}$ where, without loss of generality,

$$v_1v_2 \in m_2, v_2v_3 \in m_1, v_3v_4 \in m_2, \ldots, v_{2k}v_{2k+1} \in m_1.$$

In this case, the **Rotate** operation may be used k times to produce a new matching in M_r. For the first such operation, we swap v_2v_3 for v_1v_2. This is allowable since v_1 must be unmatched in m_1. Were it not, then the edge e involved must also appear in m_2, since it does not appear in $m_1 \triangle m_2$. But this means that m_2 has two adjacent edges e and v_1v_2, which is impossible. For the next operation, we may swap $v_3v_4 \in m_2$ for the next edge $v_4v_5 \in m_1$, and so on until P is traversed. Note that any matching generated here will lie in M_r as **Rotate** does not alter the size of a matching.

If P is of odd length then there must be another path P' in $m_1 \triangle m_2$ of odd length. This is because $|m_1 \triangle m_2|$ is even and any cycle C in $m_1 \triangle m_2$ is of even length. Therefore, any paths left over must account for an even number of edges. What is more, since cycles use the same number of edges from m_1 and m_2, the paths in $m_1 \triangle m_2$ must altogether use the same number of edges from m_1 and m_2. Therefore, if P starts and ends with edges from m_1, we may take P' to start and end with edges from m_2. In this case, we **Reduce** m_1 by the starting edge of P to get a matching in M_{r-1}. The

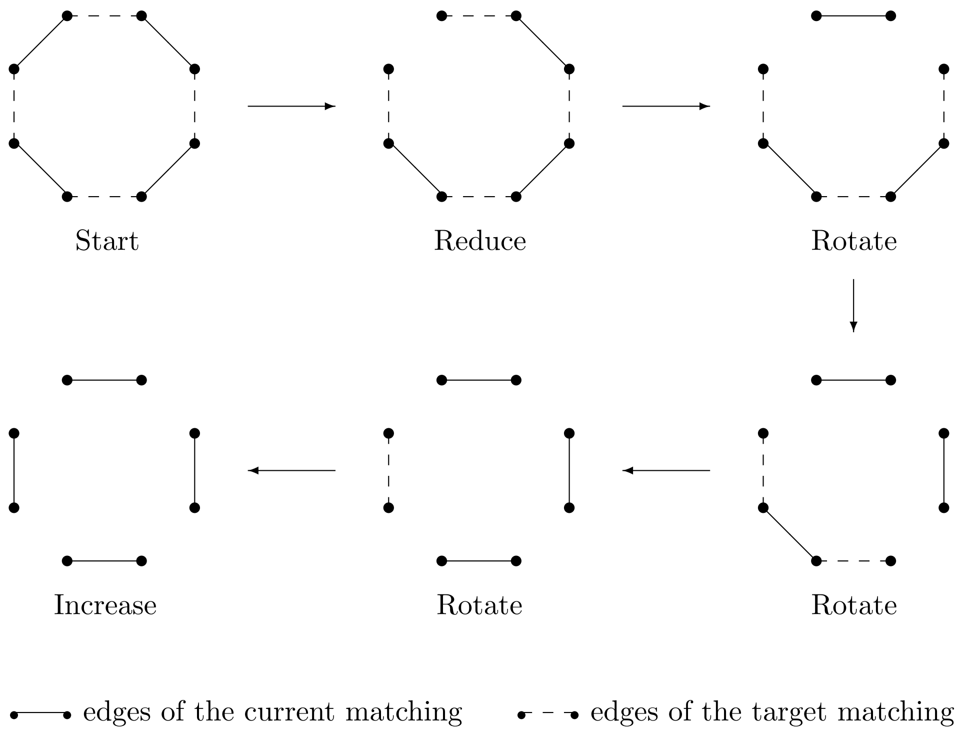

Figure 2.1: Unwinding a cycle

rest of P is of even length so now we may use a sequence of **Rotations** as above to produce a new matching in M_{r-1} using edges of P. Now, ignore the ending edge of P'. The rest of P' is of even length, so once again we may use successive **Rotate** operations to produce a new matching $m_3 \in M_{r-1}$. Note that edges from m_1 in P' will have been swapped for *preceding* edges in m_2. So now no edges in m_3 are adjacent to the ending edge of P'. Therefore, we may **Increase** m_3 with this edge to get a matching in M_r.

By working through $m_1 \triangle m_2$ in this way, it is obvious that we may use a sequence of the defined operations to produce m_2 from m_1. This establishes the Proposition. $\square$

2.2 The Canonical Path Argument

The results in this final section of the chapter are drawn from [10]. We will seek to establish a result for a particular instance of $\mathcal{Q}$ - that of when $V(\mathcal{Q}) = M_n \cup M_{n-1}$. Indeed, for reasons that will be made clear later, the major results in the analysis of the algorithm will be proved directly only for the case $M_n \cup M_{n-1}$. Suppose we have an arbitrary partition of $V(\mathcal{Q})$ into two sets, S and $\overline{S}$. Roughly speaking, we seek to establish in this section how densely packed with edges the connections between these sets will be. Since we are concerned with moving between vertices in this section, we can ignore self-loops in $\mathcal{Q}$. Also, as has been noted before, directed edges between vertices come in opposing pairs. Given these two observations, we can restrict ourselves to considering a simple, undirected version $\mathcal{Q}'$ of $\mathcal{Q}$, where $V(\mathcal{Q}') = V(\mathcal{Q}) = M_n \cup M_{n-1}$ and $xy \in E(\mathcal{Q}')$ if and only if $\overrightarrow{xy} \in E(\mathcal{Q})$ for $x \neq y$.

2.2.1 A Specification of Canonical Paths in $\mathcal{Q}'$

The idea here is to find a "skeleton graph" in $\mathcal{Q}'$ made of canonical paths such that if there is a path from vertex x to vertex y in $\mathcal{Q}'$ then there is a canonical path from x to y in the skeleton graph. The ultimate aim will be to get a lower bound for the number of canonical paths between two regions in $\mathcal{Q}'$. Obviously the lower bound for the skeleton graph will provide a lower bound for the number of paths between the two regions in $\mathcal{Q}'$.

It is worthwhile mentioning here that "canonical" is sometimes taken to mean "standard" in mathematical settings. However, there will nothing "standard" about the canonical paths chosen below. A random walk between x and y in $\mathcal{Q}'$ will not choose to take the canonical path with greater probability. Here, canonical paths will simply be used as a means of measurement.

The General Form of a Canonical Path in $\mathcal{Q}'$
Let x, y be vertices in $V(\mathcal{Q}')$ and let the canonical path between them contain three sections of two types of path:

- A first section that starts at x of type 1
- A second section of type 2
- A third section that ends at y of type 1

For each $z \in M_n \cup M_{n-1}$ we associate a single $z' \in M_n$ with it. A type 1 section joins z to its associated z'. If $x \in M_n$, take $x' := x$, and so the first section is an empty path. Similarly, if $y \in M_n$, take $y' := y$.

Suppose $x \in M_{n-1}$, then. As we saw in the arguments for the connectivity of $\mathcal{Q}$, there must be a path of length 1 or 3 from x to some $x' \in M_n$. If there is a path of length 1 from x to some perfect matching, then take x' to be this perfect matching and the first type 1 path to be the edge between them. Note that there will only be one such x' in this case: $x \in M_{n-1}$ and so G contains exactly 2 vertices unmatched by x. A path of length 1 from x to x' can only be an **Increase** operation, and there is only one edge of G that can be added: the one between the unmatched vertices (if it exists).

Otherwise, take x' to be one of the perfect matchings of distance 2 from x and choose a path of distance 2 between them to be the first segment. Note that the intermediate vertex $v \in M_{n-1}$ on this path will choose x' as its associate as well, since there is a path of length 1 between v and x'.

The situation for $y \in M_{n-1}$ will be symmetric to $x \in M_{n-1}$.

A type 2 section joins $x' \in M_n$ to $y' \in M_n$. As we saw in the argument establishing the connectivity of $\mathcal{Q}$ (see Proposition 2.2), we can always find such a path. However, we will ensure that these paths are not arbitrary. To make later analysis easier, we give type 2 paths a more definite specification.

First, we claim that $x' \triangle y'$ is composed only of even-length, disjoint cycles. To establish this we recall from a proof above that $x' \triangle y'$ must be composed of paths or even-length, disjoint cycles. If we show that $x' \triangle y'$ cannot contain any paths then we have the result we want. So suppose P is a path in $x' \triangle y'$ and let ab be its final edge with b its endpoint. Suppose, without loss of generality, that $ab \in x'$. Now $ab \notin y'$ else ab would not appear in $x' \triangle y'$. But y' is a perfect matching, so b must be matched by an edge ba' in y'. Now, ba' cannot appear in x' as ab and ba' are adjacent. So that means that $ba' \in x' \triangle y'$ and P includes ba' after ab. But this is a contradiction as ab was the final edge of P. So $x' \triangle y'$ contains no paths.

Now, given previous discussions, it is obvious that we will move from x' to y' by unwinding a succession of cycles in $x' \triangle y'$. In order to completely determine the path between x' and y' though, we must state the order in which cycles are to be unwound, and also give a direction around each cycle which its unwinding will follow. To this end, we suppose that $V(G) = \{1, \ldots, 2n\}$. For each cycle C_i in $x' \triangle y'$ we consider the vertex v_1 with the lowest value in C_i to be its "starting vertex". Let v_2 be its lowest valued neighbour on C_i. We can fix a direction for unwinding the cycle C_i by moving in the direction from v_1 to v_2. So the unwinding of C_i begins by **Reducing** C_i by the edge in x' that matches v_1. Then we continue with the unwinding by performing a series of rotations, moving in the direction of v_2 around C_i.

We can order the cycles of $x' \triangle y'$ by their starting vertices. Start with the cycle with the starting vertex of lowest value, and place the subsequent cycles in order of increasing starting vertices. This suffices to determine a path between x' and y', and so suffices to determine all paths of type 2.

Altogether then, between any x and y we can find a canonical path of the above form for all $x, y \in V(\mathcal{Q}')$.

2.2.2 Counting Canonical Paths

Let $N := |M_n \cup M_{n-1}| = |\mathcal{Q}|$. The following proposition will come in useful for the major proposition in this section.

Proposition 2.3. $\Delta(\mathcal{Q}') \leq 2n$.

Proof. Let $m \in V(\mathcal{Q}')$. Edges in $\mathcal{Q}'$ only join distinct matchings. Thus, determining $d_{\mathcal{Q}'}(m)$ is clearly the same as determining the number of edges of G that can be used to produce a new matching m' from m using a **Reduce**, **Increase** or **Rotate** operation.

If $m \in M_n$ then m can only be altered by a **Reduce** operation. There are n edges available for m to be **Reduced** by, so $d_{\mathcal{Q}'}(m) = n$.

If $m \in M_{n-1}$ then m can only be altered by an **Increase** or **Rotate** operation. There is at most one unmatched pair of vertices a, b in G, so there is at most one **Increase** operation from m. A **Rotate** operation must connect a matched vertex to an unmatched one. Each of the two unmatched vertices can be connected to at most $n-1$ unmatched ones, so there are at most $2(n-1)$ **Rotate** operations from m. So $d_{\mathcal{Q}'}(m) = 2(n-1)+1 \geq n$

So $d_{\mathcal{Q}'}(m) \leq 2n$ □

Proposition 2.4. Each edge in $\mathcal{Q}'$ lies on at most $48n^4N$ canonical paths.

Proof. We will prove this by proving two claims. Let e be an edge in $\mathcal{Q}'$.

Claim 2.4.1. *The total number number of type 1 segments containing e is at most* $12n^2N$.

To prove this, we consider a vertex $z' \in M_n$ and count the number of vertices z distinct from z' that could be joined to z' by a type 1 path. We know that all such z are of distance 1 or 2 from z'. By Proposition 2.3, there must be at most $d(z') \leq 2n$ vertices at distance 1 from z'. By the same Proposition, there must be at most $\Delta(\mathcal{Q}')^2 \leq 4n^2$ vertices of distance 2 from z'. Altogether then, there must be at most $2n + 4n^2 \leq 6n^2$ vertices that lie on a type 1 path joined to z'.

Now suppose $e =: (u, v)$ lies on a type 1 section of a canonical path, with the path moving in the direction of u to v. Suppose, to start with, that e lies on the first section of a canonical path. Let $v' \in M_n$ be the vertex associated with v by the type 1 path between them. If the first section is of length 1 then $v' = v$ is the endpoint of the section. If the first section is of length 2, then e could be the first or second edge. If it is the first then the second edge must be vv' since the second operation is an **Increase** operation, and v will have chosen the perfect matching that results from this operation as its associate. If e is the second edge, then the section ends on $v = v'$. Whatever the case then, if e lies on the first section, this section ends with v'. We have seen above that there can be at most $6n^2$ type 1 paths connected to v'. Now, there are $N-1$ canonical paths from v' to all the other vertices of $\mathcal{Q}'$. So the number of canonical paths with e in their first section must be at most $6n^2N$.

Now suppose that e lies on the third section of a canonical path. But this situation is symmetric to the case of e lying in a first section. Using reasoning similar to the above, we find that the second section must begin with u', the vertex in M_n associated with u. Again, there will be at most $6n^2$ type 1 paths connected to u', and there are $N-1$ canonical paths connecting u to the other vertices. So the number of canonical paths with e in their last section must be at most $6n^2N$.

Altogether then, e must lie in the type 1 sections of at most $12n^2N$ canonical paths. Thus Claim 1 is established.

Now suppose that e lies on a type 2 segment. We prove the following claim.

Claim 2.4.2. *There are at most N type 2 segments that use e.*

We begin the proof this claim by reminding ourselves that for any pair x', y' of perfect matchings in M_n, there is a canonical type 2 path between them, given the way our canonical paths were defined.

To prove the claim, we consider all pairs $(x', y') \in M_n \times M_n$ that use e in the type 2 segment between them. Let A be the set of all such pairs. If we show that there is an injective map from A into $M_n \cup M_{n-1}$ then this suffices to prove the claim.

Let $e = (u, v)$, and suppose, without loss of generality, that the canonical path from x' to y' arrives at u before v. It is worthwhile reminding ourselves precisely what u and v are. u and v are matchings in their own right. As we follow the canonical path from x' to y', we are, in effect, following a transformation of x' into y'. When we look at u, say, we are in a sense looking at a snapshot of x' mid-transformation. Also, u and v will differ by one edge as the move from u to v involves one operation. If moving from u to v involves a **Reduce** operation then $u \cup v = u$ contains all edges in u and v, including the one that is being removed. If the move involves an **Increase** operation, then $u \cup v = v$ contains all edges in both u and v, together with the one that is being added at this step. In the case of a **Rotate** operation, $u \cup v$ contains all the edges common to both u and v, as well as both the new edge and the edge it is being swapped for amongst other edges.

Consider the set

$$S := (x' \triangle y') \triangle (u \cup v).$$

This set will contain all edges of x' not in y', u or v. But these are precisely those edges which have been lost from x' by the time the path reaches u. Neither $u \cup v$ nor y' contain the edges that have been removed from x' to reach this point. Similarly, the set will contain all the edges of y' not in x', u or v. $u \cup v$ contains all the edges of y' that have been added in the transformation of x' so far. So S contains the edges of y' that the path has yet to include to reach y'. Finally, the set will contain all the edges of $u \cup v$ that do not lie in $x' \triangle y'$. In other words, the set will contain all edges that x' and y' agree upon (see Figure 2.2).

Using S, we will define a mapping $f : A \to M_n \cup M_{n-1}$. Before we begin the definition though, we must introduce a piece of notation. For each cycle C_i in $x' \triangle y'$ let s_i be the edge that is removed from C_i by a **Reduce** operation. That is, the edge that is removed at the beginning of

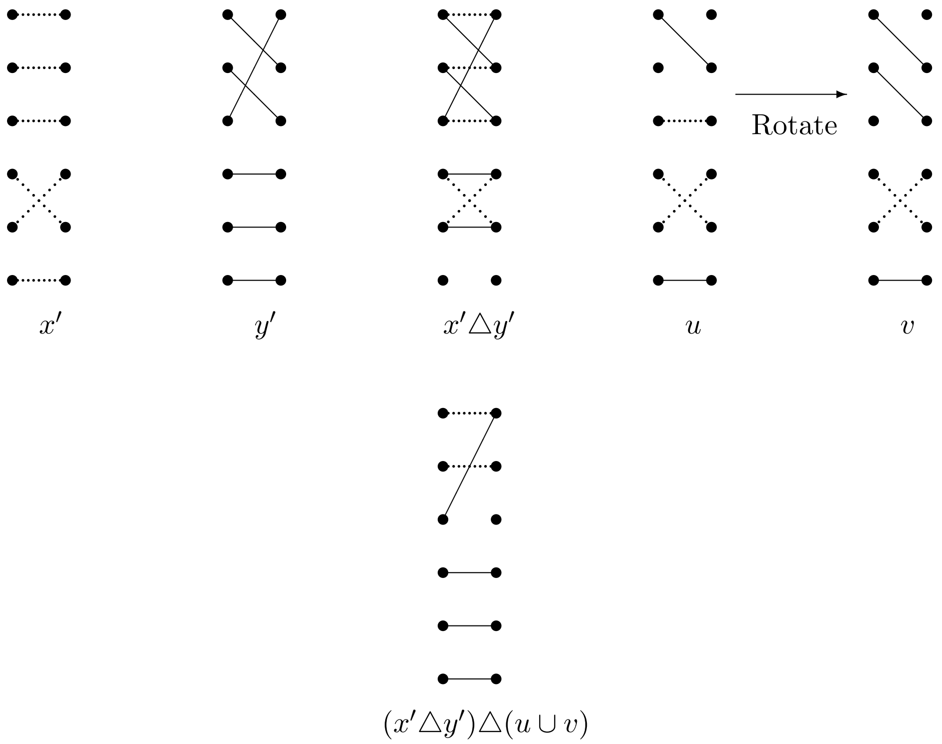

Figure 2.2: Creating S

an unwinding. Define f then, by

$$f(x', y') = \begin{cases} S, & \text{if } e \text{ corresponds to a } \textbf{Reduce/Increase} \text{ pair;} \\ S \triangle s_i, & \text{if } e \text{ corresponds to a } \textbf{Rotate} \text{ pair.} \end{cases}$$

We will show that f has the required properties in three steps. First, we will show that f is well-defined. Second, we will show that the image of f lies in $V(Q)$. Finally, we will then show that f is injective.

Now we must show that f is a well-defined function. That is, $f(x', y')$ unambiguously identifies a matching for all pairs x', y'. We must also show that for any $e \in E(\mathcal{Q}')$, $f(x', y') \in M_n \cup M_{n-1}$.

First, we must show that $f(x', y')$ is a matching in G. None of the edges from $u \cup v$ will be adjacent to an edge in $x' \triangle y'$. This is because $u \cup v$ provides the edges common to both x' and y', and x' and y' are matchings. Hence, no other edge of x' or y' in $x' \triangle y'$ will be adjacent to them. The same fact can be used to conclude that none of the edges that $u \cup v$ provide will be adjacent to each other. If we manage to show that none of the edges from x' in $f(x', y')$ are adjacent to the edges from y' in $f(x', y')$, then we have shown what we wanted.

Suppose the move from u to v involves a **Reduce** operation. Then given the way type 2 paths have been defined, e must begin the unwinding of a cycle. That means that all the edges from x' in $f(x', y')$ must lie on cycles that have already been dealt with. But these are disjoint from the current cycle, and the cycles yet to be unwound. The edges from y' in $f(x', y')$ lie only in these latter cycles, so we have the result we want in this case.

Suppose the move from u to v involves an **Increase** operation. Then e must finish the unwinding of a cycle. Now, the edges from y' in $f(x', y')$ must belong only to cycles yet to be dealt with, and so must be disjoint from any edges that x' contributes. So we have the result we wanted in this case.

Suppose the move from u to v involves a **Rotate** operation, then. Figure 2.2 displays such a transition. So e must lie in the middle of an unwinding of a cycle C_i. Given the above discussion, obviously edges from x' in cycles already dealt with are disjoint from y''s edges in cycles to be unwound. Now, the edge chosen to be added to x' and the edge it will replace are both in $u \cup v$, so will not appear in $f(x', y')$. Recall that s_i is taken to denote the edge of C_i that was first removed by a **Reduce** operation. So, by definition of f, s_i will not appear in $f(x', y')$. Now each move between the removal of s_i and the current position must have been a **Rotate** transition that considered pairs of x', y' edges in turn. From each of these pairs, the edges from x' will appear in $f(x', y')$. These, of necessity, will be disjoint from any other edge in $f(x', y')$, as each such edge is only adjacent to the edges that precede and succeed it on the cycle. So consider the edges on the cycle that follow the current position. The path will go on to consider subsequent pairs, performing a **Rotate** operation with each, until it reaches the final edge of the cycle. Each of these pairs of edges will contribute one edge from y' to $f(x', y')$. These will be disjoint from any other edge in $f(x', y')$. The final edge will be added from y' to $f(x', y')$. The only danger here is that it will be adjacent to $s_i \in x'$, as s_i has been removed previously. However, f is so defined that $s_i \notin f(x', y')$. So the final edge will disjoint from every other edge in $f(x', y')$. Hence $f(x', y')$ is a matching G in this case too. Altogether then, $f(x', y')$ must be a matching in G.

Next, we must argue that $|f(x', y')|$ is $n - 1$ or n. Suppose the move from u to v is a **Reduce** operation. This operation only occurs at the start of unwinding a cycle. But this means that any previous cycle has been dealt with, and so the matching has gained as many edges from y' as it has lost from x'. So if x' and y' agree on exactly a edges, and x' has so far lost b edges, we have yet to gain $n - a - b$ edges from y' after e. $u \cup v$ provide the a edges the two endpoints agree on. So altogether, $|f(x', y')| = |S| = b + n - a - b + a = n$.

If the shift from u to v involves an **Increase** operation, then this transition must lie at the end of an unwinding of a cycle. In this case, by the time the path reaches u, it will have lost one more edge from x' than we have gained from y'. So e will "even things up". So if, as before, x' has lost b edges, there will be $n - a - b$ edges to be gained from y' with and after e.

So once again, $|f(x', y')| = |S| = n$.

If the shift from u to v is made using a **Rotate** operation then this transition must lie in the middle of an unwinding of a cycle C_i. Suppose C_i has length $2c$. Divide C_i into c pairs of edges as follows. Pair s_i and the final edge on C_i together. For the edges in between, pair these off into successive pairs. So each **Rotate** operation in the unwinding deals with one of these pairs at a time. Now, $s_i \notin f(x', y')$ but the final edge is in $f(x', y')$ as it is yet to be added to obtain y'. Hence the first pair contributes one edge to $f(x', y')$. No edge from the current pair will appear in $f(x', y')$ as both edges appear in $u \cup v$. For every other pair, though, one edge will appear in $f(x', y')$, either as a discarded edge or an edge yet to be added. So each pair except one contributes an edge to $f(x', y')$. Hence C_i contributes $c - 1$ edges to $f(x', y')$. Now, for cycles already dealt with, as many edges are lost from y' in $f(x', y')$ as gained from x', as only the edges from x' appear from these. Also, for cycles yet to come, as many edges are lost from x' in $f(x', y')$ as gained from y'. If x' and y' agree on a edges then $|x' \triangle y'| = 2(n - a)$. Furthermore, by the above all cycles but C_i contribute half of their edges towards $f(x', y')$. C_i itself contributes half less one of its edges to $f(x', y')$. So $|f(x', y')| = \frac{1}{2}[2(n - a)] - 1 + a = n - 1$. As in the previous case, the summand a arises from the edges contributed by $u \cup v$.

This shows that $|f(x', y')|$ will always be a matching of an acceptable size. Also, given $(x', y') \in A$, the construction of $f(x', y')$ is clearly unambiguous. Hence, f is well-defined.

Now all that remains to be proved is that f is an injective map. So we need to show that if $f(x', y') = f(x'', y'')$ then $(x', y') = (x'', y'')$. This can be done if we can recover x' and y' from $f(x', y')$ and $e = (u, v)$. After all, if both $f(x', y')$ and e can be used to uniquely determine x' and y' then it can never be the case that two distinct pairs in A are mapped to the same matching by f.

Suppose the move from u to v along e is a **Reduce** or **Increase** operation. Let $B := u \cup v \cup f(x', y')$. Note that there are four sorts of edges in $u \cup v$:

1. Edges common to both x' and y'. These are unaltered throughout the path
2. Edges that have been added from y'
3. Edges of x' that have yet to be removed or replaced
4. Edges of x' or y' that are being used in the current operation

$f(x', y')$ contains all the edges of x' that have been lost; while $u \cup v$ contains all the edges of x' that are to be lost, are in the process of being removed, or will be unchanged. But then both sets must account for all edges of x'. Hence $x' \subseteq u \cup v \cup f(x', y') = B$. Likewise, $f(x', y')$ contains the edges of

y' yet to be added, while $u \cup v$ must contain all the other edges of y'. So $y' \subseteq B$. But no edges from $G - (x' \cup y')$ are introduced in the transversal of the path, so it must be that $B = x' \cup y'$. It immediately follows that $x' \triangle y' \subseteq B$.

Now, let e' be an edge common to both x' and y'. e' must be disjoint from every other edge in $x' \cup y'$ as x' and y' are matchings. So this means that $x' \cup y' = B$ must consist of disjoint edges and disjoint, even-length cycles from $x' \triangle y'$. Clearly then, $x' \triangle y'$ is easily recoverable from B. It will be the set of all even length cycles in B.

But once we have $x' \triangle y'$, the numbering of the vertices will give a clear indication of the order in which these cycles are unwound in the path. u and v must tell us precisely which cycle is being dealt with at the present time and at what stage of the unwinding the path has reached. But then we know at precisely what point in the path we are at. For any cycle already dealt with, the edges it has in common with u must lie in y'. The other edges must lie in x'. If the move from u to v is a **Reduce** operation then $u = v \cup \{g\}$ where g is an edge of G. Then $g \in x'$, given the way type 2 paths are specified. For any edge g' following g on the current cycle, if $g' \in v$ then $g' \in x'$ as it will eventually be removed. Any other edge on the cycle will be in y'. If the move is an **Increase** operation then $v = u \cup \{g\}$ where g is an edge of G. Then $g \in y'$, given the way type 2 paths are specified. For any edge g' before g on the current cycle, if $g' \in u$ then $g' \in y'$ as this portion of the cycle has been dealt with. Any other edge of the cycle will be in x'. Now, for any subsequent cycles, any edges they have in v will be edges in x', yet to be removed. All other edges on the cycles will be edges in y'. In this way, we can completely recover x' and y' from B in this case.

Now suppose that the move from u to v is a **Rotate** operation. Define B as above. Using similar reasoning to the above we get that $B = x' \cup y' - \{s_i\}$, since s_i was explicitly removed from $f(x', y')$ and does not appear in $u \cup v$ (as s_i was removed from the matching x' before the path reached u). But recovering $x' \triangle y'$ in this case is an easy matter. Since the cycles involved are of even length, they must have length at least 4 (a cycle of length 2 is impossible in $\mathcal{Q}'$). That means the path P left by removing s_i must have an odd length at least 3. This is easy to pick out in a set otherwise made up of disjoint, even-length cycles and disjoint edges. Also, it will be obvious precisely what the edge s_i is: It will be the edge joining the endpoints of P together to form the current cycle C_i. What is more, $s_i \in x'$ and C_i is an alternating cycle, so this enables us to determine which edges of C_i are in x' and which are in y'. For the cycles already dealt with and yet to be dealt with, the situation is the same as for the case above. It is easy to see, then, that x' and y' can be recovered in this situation as well.

So the map f must be injective, and this proves the Claim.

We must now use the two Claims to prove the Proposition. Let e be an edge of $\mathcal{Q}'$. If e lies on a canonical path between two vertices then e must either appear in one of its type 1 segments or in its type 2 segment (but not both).

Claim 1 establishes that e can lie in the type 1 segments of at most $12n^2N$ canonical paths. Suppose that e lies on a type 2 segment then. Recall that set A was defined to be the set of all pairs of perfect matchings x', y' such that e lay on the type 2 segment between them. But each of x', y' has at most $6n^2$ type 1 paths connected to them. So the number of canonical paths with e lying in their type 2 segment must be at most $6n^2 \times |A| \times 6n^2 = 36n^4|A|$. But Claim 2 established that $|A| \leq N$, so there are at most $36n^4N$ canonical paths that have e in their middle section.

So e must lie on at most $12n^2N + 36n^4N \leq 12n^4N + 36n^4N = 48n^4N$ canonical paths. Therefore the Proposition holds true. □

Proposition 2.5. Let $S \subseteq V(\mathcal{Q})$. Then the number of edges going from S to $\overline{S}$ in $\mathcal{Q}$ is at least $\frac{|S|}{96n^4}$.

Proof. We may assume that $|S| \leq \frac{N}{2}$, otherwise swap the labels of S and $\overline{S}$. For each edge sent out from S one is returned, so the Proposition will still be established if labels are swapped. There are canonical paths from each vertex in S to each vertex in $\overline{S}$. $|\overline{S}| = N - |S|$, so the number of such paths must be $|S|(N - |S|)$. However, $N - |S| \geq \frac{N}{2}$, so $|S|(N - |S|) \geq |S|\frac{N}{2}$. From the Proposition above, each edge from S to $\overline{S}$ lies on at most $3n^4N$ canonical paths. So the number of edges from S to $\overline{S}$ must be at least

$$\frac{1}{48n^4N}N(N - |S|) \geq \frac{1}{48n^4N}\frac{N|S|}{2} = \frac{|S|}{96n^4}$$

This proves the result. □

These results will be useful later on in arguing that the matching generator will approximate a random sampler quickly. However, the arguments about canonical paths only deal with one case of the matching generator - when it tries to run on the set $M_n \cup M_{n-1}$. The most natural question to ask here is "Can the argument be extended to arbitrary $M_r \cup M_{r-1}$?" Most steps of the argument would have an analogue for $M_r \cup M_{r-1}$. Admittedly, it would be unlikely that the injective map f would have an injective analogue g for this case, but it it possible to set an upper bound on the pre-image of g.

Unfortunately, such a project has small chance of success. This is because for $x', y' \in M_r$, $x' \triangle y'$ could well contain odd-length paths. As we saw in Proposition 2.2, these must be dealt with in pairs when making the transition from x' to y'. The nice thing about the canonical path argument was that we were able to give a rule for the ordering of the cycles in $x' \triangle y'$. This was important in arguing that the map f was injective. Suppose we did define some canonical path between x' and y', with the transition u to v lying on it.

Suppose further that we managed to find a function g analogous to f in this case. Then it would be difficult to tell from $g(x', y')$ and u, v in what order the odd-length paths were dealt with; it would even be difficult to tell which odd-length paths were meant to form pairs. Thus it would be hard to tell which were the edges of x' and which were the edges of y' along these paths. It is very difficult to set a bound on the pre-image of g in these situations.

We will therefore continue with the analysis of the generator algorithm, confining ourselves to the case of $M_n \cup M_{n-1}$ only. Once we have done this, we will show that this case can be used, with some manipulation, to provide an almost uniform sampler for $M_r \cup M_{n-1}$.

Chapter 3

Properties of Markov Chains

In this chapter, we wish to investigate how the Markov Chain behaves as the walk on $\mathcal{Q}$ progresses. In particular, we wish to establish two things. To start with, we want to establish that as the walk progresses, a matching in $M_n \cup M_{n-1}$ becomes as likely as any other to be the current state. So if we finish at a given point far enough into the walk, taking the matching we end with will be like sampling almost uniformly at random from $M_n \cup M_{n-1}$.

To make subsequent discussion easier, we adopt two conventions. These are

1. The elements of $V(\mathcal{Q})$ are matchings of G. We will assign each matching in $V(\mathcal{Q})$ a unique number from $[N]$, and may identify the matching with that number. So $V(\mathcal{Q}) = [N]$

2. Any vectors introduced are assumed to be row vectors, unless stated otherwise. Subsequently, matrix-vector multiplication involves right-multiplying a vector by a matrix.

The second convention will affect what definition of "eigenvector" we use. We will take it that $\mathbf{v}$ is an eigenvector of a matrix $\mathbf{A}$ if there is a scalar λ such that $\mathbf{vA} = \lambda\mathbf{v}$. Most matrices we introduce will be symmetric however, so the use of this non-standard definition will make little difference.

We now introduce some new concepts and definitions.

3.1 The Transition Probability Matrix

3.1.1 The Transition Probability Matrix Introduced

Definition 3.1. Let Markov chain $\mathcal{M}$ use the set of states $S =: \{s_1, s_2, \dots, s_k\}$. The *transition probability matrix* $\mathbf{P}$ for $\mathcal{M}$ is the $|S| \times |S|$ matrix such that each entry $p_{i,j} = \mathbf{Pr}(X_{t+1} = s_j | X_t = s_i)$.

So the transition probability matrix $\mathbf{P}$ in the present case is the $N \times N$ matrix where each entry $p_{i,j}$ is the combined weight of all edges from vertex

i to vertex j in $\mathcal{Q}$. If no edges exist between i and j, $p_{i,j}$ is set as 0. Note that if $i \neq j$ then $p_{i,j} = \frac{1}{2|E|}$ or $p_{i,j} = 0$. If $i = j$ then $p_{i,j} = \frac{a}{2|E|}$ where a is the number of self-loops at i.

We can relate $\mathbf{P}$ to a specialized adjancency matrix of $\mathcal{Q}$. Recall that for each vertex in $\mathcal{Q}$, half the edges connected to it accounted for the "lazy decision" to not doing anything for one step. Let $\mathbf{Q}$ be the $N \times N$ outdegree matrix of $\mathcal{Q}$ that ignores these "lazy" edges. In other words, $q_{i,j}$ is the number of edges that leave i for j and $q_{i,i}$ is the number of self-loops at i that do not reflect the decision to do nothing. Hence, for all i, $\sum_j q_{i,j} = |E|$. The outdegree matrix of $\mathcal{Q}$ that does account for the "lazy decision" will be $|E|\mathbf{I}_N + \mathbf{Q}$. Given how $\mathbf{P}$ is defined, it is easy to see that $\mathbf{P} = \frac{1}{2|E|}(|E|\mathbf{I}_N + \mathbf{Q})$.

Any edge $\overrightarrow{ij}$ for $i \neq j$ in $\mathcal{Q}$ has an opposing edge $\overrightarrow{ji}$, so $p_{ij} = p_{ji}$ for all i, j. Thus $\mathbf{P}$ is symmetric. Given that each $i \in V(\mathcal{Q})$ has $2|E|$ edges that start at it,

$$\sum_j p_{i,j} = 2|E|\frac{1}{2|E|} = 1$$

A matrix where the rows all sum to 1 in this way is called a *stochastic* matrix. Since $\mathbf{P}$ is symmetric, $\sum_i p_{i,j} = \sum_i p_{j,i} = 1$. Thus the columns of $\mathbf{P}$ each add to 1 as well. $\mathbf{P}$ is then properly described as *doubly stochastic*.

It is worthwhile to note here what an effect the matrix $\mathbf{P}$ has. Suppose $\mathbf{q}$ is a row vector with $|\mathbf{q}| = N$, containing a probability distribution for the vertices of $\mathcal{Q}$ at time t. So we may say that the probability that i is the current vertex is $\mathbf{q}(i)$. Now, $\mathbf{qP}$ will give a new probability distribution for the vertices, since for each i,

$$\mathbf{Pr}(X_{t+1} = i) = \sum_{j=1}^{N} \mathbf{Pr}(X_{t+1} = i | X_t = j)\mathbf{Pr}(X_t = j) = \sum_{j=1}^{N} p_{i,j}\mathbf{q}(j),$$

where each X_t is a state of the Markov chain.

So, if $\mathbf{q}$ gives the probability distribution at one step, $\mathbf{qP}$ gives the probability distribution at the next step. But this gives us a way to calculate the probability distribution at an arbitrary step. Suppose we start the Markov Chain at a state chosen according to the probability distribution $\mathbf{q}_0$, and want to know the probability distribution after t steps. Then we need only start with $\mathbf{q}_0$, and for each new step, right-multiply the current vector by $\mathbf{P}$ to get $\mathbf{q}_0\mathbf{P}\ldots\mathbf{P}$ where $\mathbf{P}$ appears t times in the multiplication. Thus, after t steps, the chain will have probability distribution $\mathbf{q}_0\mathbf{P}^t$.

3.1.2 The Eigenvalues of the Transition Probability Matrix

In this part, our major task will be to place bounds on the eigenvalues of $\mathbf{P}$. This will come in useful in proving that our matching generator approximates a random sampler of $M_n \cup M_{n-1}$. These bounds may seem to be proved in a roundabout manner, but a more direct proof of these bounds is slightly more difficult.

Proposition 3.1. $|E|$ is an eigenvalue of $\mathbf{Q}$ with associated N-dimensional eigenvector $\mathbf{v} := (1, 1, \ldots, 1)$.

Proof. Let $\mathbf{w} := \mathbf{vQ}$. Then for each w_i,

$$w_i = \sum_{j=1}^{N} v_j q_{ji} = \sum_{j=1}^{N} q_{ji} \quad [\text{as each } v_j = 1] \, .$$

So

$$w_i = \sum_{j=1}^{N} q_{ji} = |E| = |E| v_i .$$

Hence $\mathbf{vQ} = |E|\mathbf{v}$. $\square$

The next Proposition is proved using a standard sort of proof. An analogue for the result and proof can be found in [1].

Proposition 3.2. If μ is an eigenvalue of $\mathbf{Q}$ then $|\mu| \leq |E|$.

Proof. Since μ is an eigenvector of $\mathbf{Q}$, both μ, and consequently its associated eigenvector $\mathbf{v}$ are non-zero. Let $\mathbf{x} := \frac{1}{||\mathbf{v}||}\mathbf{v}$. $\mathbf{x}$ must be non-zero as well, so there is a non-zero entry x_i in $\mathbf{x}$ of largest absolute value. Then we will have:

$$\begin{aligned}
|\mu||x_i| &= |\mu x_i| \\
&= \left| \sum_{j=1}^{N} x_j q_{j,i} \right| \quad [\text{as } \mu\mathbf{x} = \mathbf{xQ}] \\
&\leq \sum_{j=1}^{N} |x_j||q_{j,i}| \quad [\text{by the triangle inequality}] \\
&\leq \sum_{j=1}^{N} |x_i||q_{j,i}| \quad [\text{as } |x_i| \geq |x_j| \text{ for all } j] \\
&= |x_i| \sum_{j=1}^{N} |q_{j,i}| \\
&= |x_i| \sum_{j=1}^{N} q_{j,i} \quad [\text{as each } q_{j,i} \geq 0] \\
&= |E||x_i|
\end{aligned}$$

Dividing through by $|x_i|$ gives $|\mu| \leq |E|$. $\square$

Proposition 3.3. $\mathbf{v}$ is an eigenvector of $\mathbf{Q}$ if and only if it is an eigenvector of $\mathbf{P}$. Furthermore, if μ is the eigenvalue of $\mathbf{Q}$ associated with $\mathbf{v}$ then $\lambda := \frac{1}{2}\left(1 + \frac{\mu}{|E|}\right)$ is the eigenvalue of $\mathbf{P}$ associated with $\mathbf{v}$.

Proof. Assuming $\mathbf{v}$ is an eigenvector of $\mathbf{Q}$ with eigenvalue μ, we have

$$\begin{aligned}\mathbf{vQ} = \mathbf{v}\mu \quad &\Leftrightarrow \quad \frac{1}{2}\mathbf{vI}_N + \frac{1}{2|E|}\mathbf{vQ} = \frac{1}{2}\mathbf{vI}_N + \frac{1}{2|E|}\mathbf{v}\mu \\ &\Leftrightarrow \quad \mathbf{vP} = \mathbf{v}\left(\frac{1}{2} + \frac{\mu}{2|E|}\right)\end{aligned}$$

This establishes the result. $\square$

Using Propositions 3.2 and 3.3, we see that if λ is an eigenvalue of $\mathbf{P}$, and μ the corresponding eigenvalue for $\mathbf{Q}$,

$$|\lambda| = \left|\frac{1}{2}\left(1 + \frac{\mu}{|E|}\right)\right| \leq \frac{1}{2}\left(1 + \frac{|\mu|}{|E|}\right) \leq \frac{1}{2}\left(1 + \frac{|E|}{|E|}\right) = 1.$$

Since $|E|$ is an eigenvalue of $\mathbf{Q}$, $\frac{1}{2}\left(1 + \frac{|E|}{|E|}\right) = 1$ must be an eigenvalue of $\mathbf{P}$, the largest such eigenvalue. Also, suppose we take $\mathbf{v} := \left(\frac{1}{N}, \frac{1}{N}, \ldots, \frac{1}{N}\right) \in \mathbb{R}^N$. For each x_i in $\mathbf{x} = \mathbf{vP}$, $x_i = \sum_j \frac{1}{N} p_{j,i} = \frac{1}{N}\sum_j p_{j,i} = \frac{1}{N}$. So $\mathbf{v}$ is an eigenvector of $\mathbf{P}$ associated with eigenvalue 1.

For a lower bound on an eigenvalue λ of $\mathbf{P}$, we reason as follows:

$$\begin{aligned}-|E| \leq \mu \quad &\Rightarrow \quad \frac{1}{2} + \frac{-|E|}{2|E|} \leq \lambda \\ &\Rightarrow \quad 0 \leq \lambda\end{aligned}$$

It is well-known that the eigenvalues of $\mathbf{P}$ will be the roots of the polynomial in z, $det(z\mathbf{I} - \mathbf{P})$, a polynomial of order N. Thus the polynomial has N roots and $\mathbf{P}$ has N, not necessarily distinct, eigenvalues.

So we may order the eigenvalues of $\mathbf{P}$ as follows. If λ_1 is the largest eigenvalue, λ_2 the second largest, and so on, we have $1 = \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_N \geq 0$.

3.2 Classifications of Markov Chains

Definition 3.2. A Markov Chain $\mathcal{M}$ is *irreducible* if its underlying graph is strongly connected.

We have seen above that $\mathcal{Q}$ is strongly connected in Proposition 2.2. The Markov Chain we are concerned with must be irreducible, then.

Now, it is useful in analyzing the behaviour of a Markov Chain to look at the probability that it will return to a place that it has left. To this end, we introduce some mathematical shorthand. We let r_{ij}^t denote the probability that the first time the Markov Chain reaches state j is time t, after starting at state i. So $r_{ij}^t = \mathbf{Pr}[X_t = j, \text{ and } X_u \neq j \ \forall 1 \leq u \leq t-1 | X_0 = i]$.

Definition 3.3. A state i of a Markov Chain is said to be *transient* if $\sum_{t>0} r_{ii}^t < 1$. If $\sum_{t>0} r_{ii}^t = 1$ then i is said to be *persistent.*

So in a transient Markov chain it is not guaranteed that, once the chain has entered a particular state, that it will ever return to that state, even in an infinite run. Conversely, a persistent Markov Chain, if allowed to continue forever, will repeat any state it enters. However, note that there is no guarantee for a finite-length Markov Chain. It is possible to distinguish bewteen two types of persistent Markov Chain. To do this, we will introduce another piece of mathematical shorthand. We say that h_{ij} is the expected number of steps a Markov Chain will make to reach state j, starting from state i. So, $h_{ij} = \sum_{t>0} tr_{ij}^t$.

Definition 3.4. A persistent state i of a Markov Chain is said to be *null persistent* if $h_{ii} = \infty$ and is said to be *non-null persistent* if $h_{ii} < \infty$.

So if a state i is non-null persistent if we can start off its Markov Chain at i and point to some future point in time as the expected time before which the Chain will have repeated the starting state. i will be null persistent if we know that the Markov Chain starting with i must return to i in an infinite run, but we cannot have a finite "temporal window" in which we expect this to happen. So i will repeat, we just cannot estimate when.

Definition 3.5. A state i of a Markov Chain is *periodic* if there exists an integer $z > 1$ such that if $\mathbf{Pr}[X_{t+d} = i | X_t = i] > 0$ then z divides d. If there is no such integer z then i is *aperiodic.* A Markov Chain is said to be *aperiodic* if all of its states are aperiodic.

Thus a Markov Chain only gets the chance to enter a periodic state at regular intervals.

Definition 3.6. A state is said to be *ergodic* if it is aperiodic and non-null persistent. A Markov Chain is said to be *ergodic* if all of its states are ergodic.

Definition 3.7. A probability distribution π for a Markov Chain with transition probability matrix $\mathbf{P}$ is a *stationary distribution* for that chain if $\pi = \pi\mathbf{P}$.

So if a Markov Chain ever reaches such a distribution π, it remains with this distribution.

We have now introduced enough concepts to draw some conclusions about the Markov Chain under consideration. Recall that for our sampling algorithm to work, the process had to eventually reach a state where the chances of having the current matching as the current matching was close to $\frac{1}{|M_n \cup M_{n-1}|}$. By using a major theorem stated below, we will be able to see that this does happen.

3.3 The Matching Generator is Near-uniform

The following theorem is actually part of the Fundamental Theorem of Markov Chains. The proof of the Fundamental Theorem is, unfortunately, quite lengthy, so the result is stated here without proof. The statement of this Theorem is based on the statement of it in [10].

Theorem 3.1. Let $\mathcal{M}$ be a finite Markov Chain with N states. Furthermore, suppose that $\mathcal{M}$ is irreducible and aperiodic. Then:

1. All states of $\mathcal{M}$ are ergodic (and so $\mathcal{M}$ is ergodic)
2. $\mathcal{M}$ has a unique stationary distribution π where each $\pi_i > 0$ for $1 \leq i \leq N$

Suppose $\mathcal{M}$ is the Markov Chain that models our matching generator. We have already seen that Proposition 2.2 implies that $\mathcal{M}$ is irreducible. It is obviously finite as the number of matchings that can be generated is finite. Also, $\mathcal{M}$ must be aperiodic as each vertex in $\mathcal{Q}$ has a self-loop. A state was defined to be periodic if it only had the chance to enter its Markov chain at regular intervals greater than 1. But self-loops ensure any state can be immediately repeated within $\mathcal{M}$. Hence $\mathcal{M}$ must be aperiodic by definition. Therefore the above theorem applies to $\mathcal{M}$. $\mathcal{M}$ is ergodic and has a unique stationary distribution π.

Since $\mathbf{P}$ is the transition probability matrix for $\mathcal{M}$, $\pi\mathbf{P} = \pi$. So π is an eigenvector of $\mathbf{P}$ associated with eigenvalue $\lambda_1 = 1$. But then, from previous working [page 27], it must be that $\pi = \left(\frac{1}{N}, \ldots, \frac{1}{N}\right)$ as this gives a valid probability distribution. Hence by Theorem 3.1 the unique stationary distribution of $\mathcal{M}$ is the uniform distribution. If we manage to show that $\mathcal{M}$ tends towards this distribution no matter the starting state, then we have what we want.

Thus we need to show that as the chain moves through different probability distributions, these distributions get "closer" to the stationary (uniform) one. So we need some way to measure the "distance" between two distributions.

The Markov Chain we are dealing with starts at some fixed state i. This is equivalent to having an initial probability distribution vector $\mathbf{q}$ where $q_i = 1$ and $q_j = 0$ for $j \neq i$. The distribution vector after t steps will be $\mathbf{q}\mathbf{P}^t$. If we denote the (j,k)-th entry of $\mathbf{P}^t$ by $p_{j,k}^{(t)}$ then the j-th entry of $\mathbf{q}\mathbf{P}^t$ will be $p_{i,j}^{(t)}$. This is because $\mathbf{q}$ reflects the chain starting at state i. Given this, the following measure makes some sense.

Definition 3.8. The *relative pointwise distance* $\Delta(t)$ of a Markov chain $\mathcal{M}$ from π at time t is defined to be

$$\Delta(t) = \max_{i,j} \frac{|p_{i,j}^{(t)} - \pi_j|}{\pi_j}.$$

This, of course, is not the only possible distance-measuring function that could be used, but it suits our purposes.

The relative pointwise distance is a refinement of the standard pointwise distance function. The pointwise distance function takes the largest difference between corresponding entries of two vectors as their distance. Instead, the relative pointwise distance function considers each difference between entries in proportion to the entry in π, and takes the largest of these. It considers each possible starting state i as well, so if $\Delta(t)$ decreases as t increases, our chain must converge to the stationary distribution, no matter what the starting state. Obviously, if $\Delta(t) = 0$ then the chain must have reached the stationary distribution by time t, as the entries of the current probability distribution vector will not differ from the stationary probability distribution vector.

So to show that our Markov chain tends to the uniform distribution it will suffice to show that $\Delta(t) \to 0$ as $t \to \infty$. We will use a feature of the Markov chain to do this. Suppose a Markov chain has reached an invariant distribution, and so has reached some sort of equilibrium. Yet while it moves within this distribution, it may be that it will tend to go a certain way down a path rather than take the reverse direction. For instance, it may tend to move clockwise around a particular cycle rather than anticlockwise. A patient observer familiar with the chain would then be able to tell if they were watching a normal running of the chain, or watching the events in reverse. The following definition attempts to single out Markov Chains where this does *not* happen.

Definition 3.9. An ergodic Markov chain $\mathcal{M}$ with transition probability matrix $\mathbf{P}$ and stationary distribution π is said to be *time reversible* if, for all i, j; $\pi_i p_{i,j} = \pi_j p_{j,i}$

Our Markov Chain $\mathcal{M}$ is obviously time reversible. $p_{i,j} = p_{j,i}$ for all i, j and $\pi_i = \frac{1}{N}$ for all i. Thus for all i, j, $\pi_i p_{i,j} = \pi_j p_{j,i}$.

Let $\mathbf{D}$ be the $N \times N$ diagonal matrix with $D_{i,i} = \sqrt{\pi_i}$. Thus $\mathbf{D}^{-1}$ is an $N \times N$ diagonal matrix with $D_{i,i} = \frac{1}{\sqrt{\pi_i}}$. Then the following lemma holds. It, and the lemma that follows it, were stated by Jerrum and Sinclair [5].

Lemma 3.1. Let $\mathcal{M}$ be an ergodic Markov chain with transition matrix $\mathbf{P}$. $\mathcal{M}$ is time reversible if and only if $\mathbf{C} := \mathbf{DPD}^{-1}$ is symmetric.

Proof. Since $\mathcal{M}$ is ergodic, $\mathcal{M}$ is irreducible. No matter the current state in an irreducible Markov chain, one can always get to another state by a sequence of transitions. Thus, starting from any state, any vertex has a chance of being visited. Therefore every vertex has a chance of appearing under the stationary distribution. We infer that each $\pi_i > 0$ in π.

$$\begin{aligned} \text{Each } p_{i,j}\pi_i = p_{j,i}\pi_j \quad &\Leftrightarrow \quad \sqrt{\pi_i}p_{i,j}\frac{1}{\sqrt{\pi_j}} = \sqrt{\pi_j}p_{j,i}\frac{1}{\sqrt{\pi_i}} \\ &\Leftrightarrow \quad c_{i,j} = c_{j,i}. \end{aligned}$$

This establishes the claim. □

Lemma 3.2. Let $\mathcal{M}$ be the Markov chain that models matching generation and let $\mathbf{C}$ be as above. Then $\mathbf{C}$ and $\mathbf{P}$ share the same eigenvalues. Also, $\mathbf{v}$ where $v_i := \sqrt{\pi_i}$ is an eigenvector for $\mathbf{C}$. Its associated eigenvalue is 1.

Proof. By a standard result in Linear Algebra [8], since $\mathbf{C} = \mathbf{DPD}^{-1}$, $\mathbf{C}$ and $\mathbf{P}$ share the same eigenvalues.
Let $\mathbf{v} := \mathbf{C}\pi^T$. Then for each v_i,

$$\begin{aligned} v_i &= \sum_{j=1}^{N} c_{i,j}\sqrt{\pi_j} \\ &= \sum_{j=1}^{N} \frac{\sqrt{\pi_i}}{\sqrt{\pi_j}} p_{i,j}\sqrt{\pi_j} \\ &= \sqrt{\pi_i}\sum_{j=1}^{N} p_{i,j} \\ &= \sqrt{\pi_i} \end{aligned}$$

This is what we wanted. □

We are now in a position to state and prove the major result of this section. This was stated and proved by Jerrum and Sinclair [5].

Theorem 3.2. Let $\mathcal{M}$ be a time-reversible Markov chain with transition matrix $\mathbf{P}$ and stationary distribution π. If $\mathbf{P}$ has eigenvalues $1 = \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_N$ then the relative pointwise distance of $\mathcal{M}$ at time t is bounded by

$$\Delta(t) \leq \frac{\lambda_2^t N}{\min_j \pi_j}.$$

Proof. Take $\mathbf{C} := \mathbf{DPD}^{-1}$, as before. $\mathbf{C}$ is obviously real-valued. Using the Lemmas above, $\mathbf{C}$ is symmetric and has the same eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_N$ as $\mathbf{P}$. It is a standard result in Linear Algebra that for any symmetric, real-valued, $n \times n$ matrix $\mathbf{A}$, there is a set of n unit, pairwise-orthogonal eigenvectors $\mathbf{e}_1, \ldots, \mathbf{e}_n$ of $\mathbf{A}$ that form a basis for $\mathbb{R}^n$ [10]. Then there is an orthonormal basis for $\mathbb{R}^N$ consisting of unit eigenvectors $\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_N$ of $\mathbf{C}$, where each $\mathbf{e}_i$ has associated eigenvalue λ_i.

For each $\mathbf{e}_i$, make an $N \times N$ matrix $\mathbf{E}_i := \mathbf{e}_i^T\mathbf{e}_i$. We will show that $\mathbf{C} = \sum_i \lambda_i \mathbf{E}_i$. For all $\mathbf{e}_i$,

$$\begin{aligned}
\mathbf{e}_i \sum_j \lambda_j \mathbf{E}_j &= \sum_j \lambda_j \mathbf{e}_i \mathbf{e}_j^T \mathbf{e}_j \\
&= \lambda_i \mathbf{e}_i \quad [\text{as } \mathbf{e}_i\mathbf{e}_i^T = 1 \text{ and for } j \neq i, \mathbf{e}_i\mathbf{e}_j^T = 0] \\
&= \mathbf{e}_i \mathbf{C}
\end{aligned}$$

So $\mathbf{e}_i \left(\sum_j \lambda_j \mathbf{E}_j - \mathbf{C}\right) = 0$ for all i. But $\mathbf{e}_1, \ldots, \mathbf{e}_N$ was a basis for $\mathbb{R}^N$, so $\sum_j \lambda_j \mathbf{E}_j - \mathbf{C}$ maps $\mathbb{R}^N$ to 0. Thus, $\sum_j \lambda_j \mathbf{E}_j - \mathbf{C}$ is the zero map and hence the zero matrix. Therefore, $\mathbf{C} = \sum_j \lambda_j \mathbf{E}_j$.

Note that if $i \neq j$, $\mathbf{E}_i\mathbf{E}_j = \mathbf{e}_i^T\mathbf{e}_i\mathbf{e}_j^T\mathbf{e}_j = \mathbf{e}_i^T(0)\mathbf{e}_j = 0$. Also, for all $m \in \mathbb{N}$, $\mathbf{E}_i^m = (\mathbf{e}_i^T\mathbf{e}_i)^m = \mathbf{e}_i^T(\mathbf{e}_i\mathbf{e}_i^T)^{m-1}\mathbf{e}_i = \mathbf{e}_i^T(1)^{m-1}\mathbf{e}_i = \mathbf{E}_i$. So all $\mathbf{E}_i$ are mutually orthogonal and idempotent. Hence, using the orthogonality of the matrices, we have $\mathbf{C}^t = \left(\sum_j \lambda_j \mathbf{E}_j\right)^t = \sum_j \lambda_j^t \mathbf{E}_j^t$. The idempotency of each $\mathbf{E}_j$ implies that $\mathbf{C}^t = \sum_j \lambda_j^t \mathbf{E}_j$. Therefore we have

$$\mathbf{P}^t = \left(\mathbf{D}^{-1}\mathbf{C}\mathbf{D}\right)^t = \mathbf{D}^{-1}\mathbf{C}^t\mathbf{D} = \mathbf{D}^{-1}\left(\sum_j \lambda_j^t \mathbf{E}_j\right)\mathbf{D} = \sum_j \lambda_j^t \mathbf{D}^{-1}\mathbf{E}_j\mathbf{D}.$$

Recall that $d_{i,i} = \sqrt{\pi_i}$. Given how each $\mathbf{E}_i$ was defined, we have that

$$\begin{aligned}
p_{i,j}^{(t)} &= \sum_{k=1}^{N} \lambda_k^t \frac{1}{\sqrt{\pi_i}} (E_k)_{i,j} \sqrt{\pi_j} \quad [\text{where } (E_k)_{i,j} \text{ is an entry of } \mathbf{E}_k] \\
&= \sqrt{\frac{\pi_j}{\pi_i}} \sum_{k=1}^{N} \lambda_k^t (e_k)_i (e_k)_j \quad [\text{where } (e_k)_i \text{ is the } i\text{-th entry of } \mathbf{e}_k] \\
&= \sqrt{\frac{\pi_j}{\pi_i}} \lambda_1^t (e_1)_i (e_1)_j + \sqrt{\frac{\pi_j}{\pi_i}} \sum_{k=2}^{N} \lambda_k^t (e_k)_i (e_k)_j \\
&= \sqrt{\frac{\pi_j}{\pi_i}} (1)^t (\sqrt{\pi_i})(\sqrt{\pi_j}) + \sqrt{\frac{\pi_j}{\pi_i}} \sum_{k=2}^{N} \lambda_k^t (e_k)_i (e_k)_j \\
&\qquad [\text{as each } (e_1)_i = \sqrt{\pi_i} \text{ (Lemma 3.2)}] \\
&= \pi_j + \sqrt{\frac{\pi_j}{\pi_i}} \sum_{k=2}^{N} \lambda_k^t (e_k)_i (e_k)_j
\end{aligned}$$

Applying the definition of relative pointwise distance then, we get that

$$\begin{aligned}
\Delta(t) &= \max_{i,j}\frac{|p_{i,j}^{(t)}-\pi_j|}{\pi_j} \\
&= \max_{i,j}\frac{\left|\pi_j+\sqrt{\frac{\pi_j}{\pi_i}}\sum_{k=2}^{N}\lambda_k^t(e_k)_i(e_k)_j-\pi_j\right|}{\pi_j} \\
&= \max_{i,j}\frac{\left|\sum_{k=2}^{N}\lambda_k^t(e_k)_i(e_k)_j\right|}{\sqrt{\pi_i}\sqrt{\pi_j}} \\
&\leq \max_{i,j}\frac{\left|\sum_{k=2}^{N}\lambda_2^t(e_k)_i(e_k)_j\right|}{\sqrt{\pi_i}\sqrt{\pi_j}} \quad [\text{as } \lambda_2 \geq \lambda_k \geq 0 \ \ \forall k \geq 2] \\
&= \lambda_2^t \max_{i,j}\frac{\sum_{k=2}^{N}|(e_k)_i||(e_k)_j|}{\sqrt{\pi_i}\sqrt{\pi_j}} \\
&\leq \lambda_2^t\frac{\max_{i,j}\sum_{k=2}^{N}|(e_k)_i||(e_k)_j|}{\min_{i,j}\sqrt{\pi_i}\sqrt{\pi_j}}
\end{aligned}$$

Now, each $|(e_k)_i| \leq 1$ so $\max_{i,j}\sum_{k=2}^{N}|(e_k)_i||(e_k)_j| \leq \sum_{k=2}^{N}1 < N$. Also, suppose $i = a, j = b$ provide $\min_{i,j}\sqrt{\pi_i}\sqrt{\pi_j}$. If, without loss of generality, $\sqrt{\pi_a} < \sqrt{\pi_b}$ then $(\sqrt{\pi_a})^2 < \sqrt{\pi_a}\sqrt{\pi_b}$, a contradiction. So $\sqrt{\pi_a} = \sqrt{\pi_b}$ and we may take $\min_{i,j}\sqrt{\pi_i}\sqrt{\pi_j} = \min_j \pi_j$. Thus,

$$\Delta(t) \leq \frac{\lambda_2^t N}{\min_j \pi_j}.$$

This is what we wanted. □

In the context of the Markov chain we are considering, $\min_j \pi_j = \frac{1}{N}$, so $\Delta(t) \leq \lambda_2^t N^2$. Suppose that $\lambda_2 < 1$. Obviously then, as $t \to \infty$, $\lambda_2^t \to 0$ and $\Delta(t) \to 0$. But then this means that our Markov chain tends to the uniform distribution, no matter what the starting state of the chain. So eventually, stopping the chain and taking the final matching generated will be almost like sampling from $|M_n \cup M_{n-1}|$ at random. We need to show that the supposition $\lambda_2 < 1$ is correct. Recall from Proposition 3.3 that there must be eigenvalue μ_2 of matrix $\mathbf{Q}$ such that $\lambda_2 = \frac{1}{2}\left(1+\frac{\mu_2}{|E|}\right)$. If we show that $\mu_2 < |E|$, we're done. This is what the next proposition achieves. Its proof is based on an idea in [1].

Proposition 3.4. The eigenvalue $\mu_1 = |E|$ $\mathbf{Q}$ has multiplicity 1.

Proof. Let $\mathbf{x} \in \mathbb{R}^N$ be an eigenvector of $\mathbf{Q}$ associated with μ_1. Recall that Proposition 3.1 implies that one such eigenvector is $(1, 1, \ldots, 1)$. $\mathbf{x}$ is non-zero by definition, and so has a non-zero entry x_i of largest absolute value.

Now the i-th column of $\mathbf{Q}$ contains information about the $|E|$ "non-lazy" edges beginning at vertex i of $\mathcal{Q}$. Let a be the number of non-negative entries in column i of $\mathbf{Q}$ and let $J := \{j_1, j_2, \ldots, j_a\}$ be the set of all j_k such that $q_{j_k,i} > 0$. Then

$$|E|x_i = \sum_j x_j q_{j,i} = \sum_{k=1}^{a} x_{j_k} q_{j_k,i}.$$

So, since $\sum_j q_{j,i} = |E|$, the final sum can be seen as the addition of $|E|$ terms, where any x_{j_k} appears in the sum $q_{j_k,i}$ times. So we have a series of $|E|$ terms, summing to $|E|x_i$. Now x_i was selected because of its largest absolute value. Thus, if any $x_{j_k} \neq x_i$, $\sum_{k=1}^{a} x_{j_k} q_{j_k,i} < |E|x_i$. Hence each $x_{j_k} = x_i$.

If we assign each x_l to vertex $l \in V(\mathcal{Q})$, then by the above reasoning we get that i and all its out-neighbours j have the same value $x_j = x_i$. But then a similar line of reasoning to the above implies that each out-neighbour k of an out-neighbour j of i is assigned the same value $x_k = x_j = x_i$. We can continue in this way, and since $\mathcal{Q}$ is strongly connected we can conclude that each vertex will be assigned the same value x_i.

Therefore $\mathbf{x} = a\pi$ for some scalar a, which establishes the Proposition. $\square$

Since eigenvalue $|E|$ of $\mathbf{Q}$ has multiplicity 1, $\mu_2 < |E|$, which is sufficient to show that $\lambda_2 < 1$. So the Markov chain does tend towards the stationary distribution. What we need to do now is show that the Markov chain approaches the stationary distribution quickly.

Chapter 4

The Speed of Convergence

Measuring the speed of our generator essentially involves placing an upper bound on the number of steps it needs to make before it gets close enough to the uniform distribution (We are still taking the relative pointwise distance as the distance between two vectors). What we accept as close enough is, as far as the theory is concerned, arbitrary. We can take what suits our purposes. Just how good a random sampler our matching generator is depends on how long we wish to let it run for. But suppose we are happy for the algorithm to finish when the relative pointwise distance $\Delta(t)$ goes below $\frac{1}{n^4}$. This is a good way to start, as it makes the required accuracy a polynomial in the input length. At first sight, it seems to force $\Delta(t)$ very close to 0 as well. If the number of steps needed to reach this point is polynomial in the input length, then we have succeeded in showing what we wanted to show.

4.1 Calculating the Number of Steps

So how many steps τ do we need to make before $\Delta(\tau) \leq \frac{1}{n^4}$? For any $t \in \mathbb{R}$, $1 + t \leq e^t$ [10]. Thus,

$$\lambda_2 = 1 + (\lambda_2 - 1) \leq e^{\lambda_2 - 1}$$

and so Theorem 3.2 together with the fact that $\pi_i = \frac{1}{N}$ gives

$$\Delta(t) \leq \lambda_2^t N^2 \leq (e^{-(1-\lambda_2)})^t N^2$$

Let $\tau \geq 8\frac{(\ln n)(\ln N)}{1-\lambda_2}$. This will give

$$\Delta(\tau) \leq (e^{-(1-\lambda_2)})^{8\frac{(\ln n)(\ln N)}{1-\lambda_2}} N^2 = (e^{-(4(\ln n)2(\ln N))})N^2 = \frac{1}{n^4}.$$

If we manage to show that the lower bound given for τ is polynomial in n, then we will have done what we set out to do. It is not immediately obvious that the lower bound is so. It is expressed in terms of N and λ_2 as well as n.

However, N must always be less than the total number of matchings $|\bigcup_{i=1}^{n} M_i|$ of G. But the set $\bigcup_{i=1}^{n} M_i$ is a subset of the power set $\mathcal{P}(E(G))$ of $E(G)$, which is in turn a subset of $\mathcal{P}(E(K_{n,n}))$. So $N \leq |\mathcal{P}(E(K_{n,n}))| = 2^{|E(K_{n,n})|} = 2^{n^2}$. Transferring this across to our lower bound, we get that $\ln N \leq \ln 2^{n^2} = n^2 \ln 2$, which is certainly a polynomial expression in n.

4.2 The Conductance of a Markov Chain

Unfortunately, obtaining a similar polynomial expression to substitute for λ_2 is a trickier affair. We will do so by relating both λ_2 and N to the value $\Phi_{\mathcal{Q}}$ - the conductance of $\mathcal{Q}$. We introduce conductance as follows.

Definition 4.1. Let S be a subset of $V(\mathcal{Q}) = [N]$. The *capacity* of S is

$$C_S := \sum_{i \in S} \pi_i$$

This is simple enough to understand. The capacity is simply the chance, under the stationary distribution, that the current matching in the Markov chain is to be found in S.

Definition 4.2. Let S be a subset of $V(\mathcal{Q})$. The *ergodic flow* out of S is

$$F_S := \sum_{i \in S} \sum_{j \in \overline{S}} \pi_i p_{i,j}$$

Thus the ergodic flow is the probability under the stationary distribution that the next move of the Markov chain will take it from S to $\overline{S}$.

Definition 4.3. Let S be a subset of $V(\mathcal{Q})$. The *conductance* of S is

$$\Phi_S := \frac{F_S}{C_S}$$

Hence the conductance of S is the probability under the stationary distribution that the Markov chain will leave S, given that it is currently in S. The conductance of the entire graph $\mathcal{Q}$ is taken to be the minimum conductance of all subsets of its vertex set. More formally,

Definition 4.4. The *conductance* of $\mathcal{Q}$ is defined as

$$\Phi_{\mathcal{Q}} := \min_{S \subset V(\mathcal{Q})} \Phi_S$$

As has been noted before, in $\mathcal{Q}$ for each edge $\overrightarrow{ij}, i \neq j$, there is an edge $\overleftarrow{ij}$. This combined with the fact that $\pi_1 = \pi_2 = \ldots = \pi_N$ (by Proposition 3.4) and $p_{i,j} = p_{j,i}$ (noted on page 25) means that for any set $S \subset V(\mathcal{Q}), \Phi_S = \Phi_{\overline{S}}$. Thus, in finding the minimum conductance for $\Phi_{\mathcal{Q}}$, we need only range over all subsets S where $|S| \leq \frac{1}{2}|V(\mathcal{Q})| = \frac{N}{2}$. So for each such S, $C_S = \sum_{i \in S} \pi_i \leq \frac{N}{2}\left(\frac{1}{N}\right) = \frac{1}{2}$. Hence $\Phi_{\mathcal{Q}} = \min_{S \subset V(\mathcal{Q}), C_S \leq \frac{1}{2}} \Phi_S$.

We are now in a position to relate λ_2 to $\Phi_{\mathcal{Q}}$. The following Lemma was stated and proved by Jerrum and Sinclair [5].

Lemma 4.1.

$$\lambda_2 \leq 1 - \frac{\Phi_{\mathcal{Q}}^2}{2}.$$

Proof. Let matrix $\mathbf{B} := \mathrm{I}_N - \mathbf{P}$, where $\mathbf{P}$ is the transition probability matrix as defined in Definition 3.1. The following facts about the entries of $\mathbf{B}$ will come in useful later on. $b_{j,i} = 0 - p_{j,i} = -p_{j,i} < 0$ if $i \neq j$. On the diagonal, $b_{i,i} = 1 - p_{i,i} = \sum_{j \neq i} p_{i,j} \geq 0$.

Let $\mathbf{x}$ be an eigenvector of $\mathbf{P}$ associated with λ_2. Observe that

$$\begin{aligned} \lambda_2 \sum_i x_i &= \sum_i \sum_j x_j p_{j,i} \\ &= \sum_j \sum_i x_j p_{j,i} \\ &= \sum_j x_j \sum_i p_{j,i} \\ &= \sum_i x_i \end{aligned}$$

So $\lambda_2 \sum_i x_i - \sum_i x_i = (\lambda_2 - 1)\sum_i x_i = 0$. Proposition 3.4 implies that $\lambda_2 \neq 1$ so $\lambda_2 - 1 \neq 0$. Therefore $\sum_i x_i = 0$.

Let $S := \{i \in V(\mathcal{Q}) : x_i > 0\} \subset V(\mathcal{Q})$. Without loss of generality, we suppose that $C_S \leq \frac{1}{2}$, otherwise substitute $-\mathbf{x}$ for $\mathbf{x}$.

It is easy to see that

$$\mathbf{xB} = \mathbf{x}(\mathbf{I}_N - \mathbf{P}) = \mathbf{xI}_N - \mathbf{xP} = \mathbf{x} - \lambda_2 \mathbf{P} = (1 - \lambda_2)\mathbf{x} \tag{4.1}$$

Define a new vector $\mathbf{y}$ by

$$y_i = \begin{cases} \frac{x_i}{\pi_i}, & \text{if } i \in S; \\ 0, & \text{if } i \notin S. \end{cases}$$

If need be, reassign the numbers of $V(\mathcal{Q})$ so that $y_1 \geq y_2 \geq \ldots \geq y_N$. Given that some, but not all, x_i are positive, there must be an $0 \leq r < N$ so that $y_1 \geq y_2 \geq \ldots \geq y_r > 0 = y_{r+1} = \ldots = y_N$. This also means that $S = \{1, \ldots, r\}$.

We can right-multiply 4.1 above by $\mathbf{y}^T$ to get

$$\begin{aligned}
& \mathbf{xBy}^T = (1-\lambda_2)\mathbf{xy}^T \\
\Leftrightarrow\quad & \sum_i \left(\sum_j x_j b_{j,i} \right) y_i = (1-\lambda_2) \sum_i x_i y_i \\
\Leftrightarrow\quad & \sum_i \sum_j x_j b_{j,i} y_i = (1-\lambda_2) \sum_{i \in S} (\pi_i y_i)\, y_i \\
\Leftrightarrow\quad & \sum_i \sum_j x_j b_{j,i} y_i = (1-\lambda_2) \sum_{i \in S} \pi_i y_i^2 \qquad (4.2)
\end{aligned}$$

Concentrating on the left-hand side of (2),

$$\sum_i \sum_j x_j b_{j,i} y_i = \sum_{i \in S} \sum_j x_j b_{j,i} y_i$$

as $y_i = 0$ if $i \notin S$. Let us turn our attention to the second summation, which is over $j \in [N]$. If $j \notin S$ then $x_j \leq 0$. It also implies that $i \neq j$ as $i \in S$ always. Thus, $b_{j,i} \leq 0$. This then implies that $x_j b_{j,i} y_i \geq 0$ as $y_i > 0$ by definition. We can conclude that $j \notin S$ gives a positive addend, and so get the following inequality.

$$\begin{aligned}
\sum_i \sum_j x_j b_{j,i} y_i &\geq \sum_{i \in S} \sum_{j \in S} x_j b_{j,i} y_i \\
&= \sum_{i \in S} \sum_{j \in S, j \neq i} \pi_j \left(\frac{x_j}{\pi_j} \right) (-p_{j,i}) y_i + \sum_{i \in S} \pi_i \left(\frac{x_i}{\pi_i} \right) (1 - p_{i,i}) y_i
\end{aligned}$$

The substitutions for each $p_{i,j}$ follow from the nature of the entries of $\mathbf{B}$ noted at the beginning of the proof. After further substitutions and simplifications, we get

$$\begin{aligned}
\sum_i \sum_j x_j b_{j,i} y_i &\geq \sum_{i \in S} \sum_{j \in S, j \neq i} -(\pi_j p_{j,i}) y_j y_i + \sum_{i \in S} \pi_i y_i \left(\sum_{j \neq i} p_{i,j} \right) y_i \\
&= \sum_{i \in S} \sum_{j \in S, j \neq i} -(\pi_i p_{i,j}) y_j y_i + \sum_{i \in S} \sum_{j \neq i} \pi_i p_{i,j} y_i^2 \\
&\quad [\text{as } \pi_i p_{i,j} = \pi_j p_{j,i}]
\end{aligned}$$

With regard to the first sum on the right-hand side, we may drop the requirement that $i, j \in S$. This is because if i or $j \notin S$ then $y_i = 0$ or $y_j = 0$, respectively. So simply taking all $j \neq i$ pairs will yield the same sum. Also, since $-(\pi_i p_{i,j}) y_j y_i = -(\pi_j p_{j,i}) y_i y_j$, each addend is repeated somewhere in the sum. So instead of summing over all $j \neq i$ pairs, we may sum over all

$i < j$ pairs and multiply the result by 2. Thus $\sum_{i \in S} \sum_{j \in S, j \neq i} -(\pi_i p_{i,j}) y_j y_i = 2\sum_{i<j} -(\pi_i p_{i,j}) y_j y_i$.

With regard to the second sum, on the right-hand side, once again we may drop the requirement that $i \in S$ as $y_i^2 = 0$ if $i \notin S$. Suppose $j \notin S$. Then $j > i$. In this case, $\pi_i p_{i,j} y_i^2 = \pi_i p_{i,j}(y_i^2 + y_j^2)$. Suppose $j \in S$. Then both $\pi_i p_{i,j} y_i^2$ and $\pi_j p_{j,i} y_j^2 = \pi_i p_{i,j} y_j^2$ will appear in the sum. But then we may include both terms at once in the summation by making $j > i$ and replacing $\pi_i p_{i,j} y_i^2$ with $\pi_i p_{i,j}(y_i^2 + y_j^2)$. So $\sum_{i \in S} \sum_{j \neq i} \pi_i p_{i,j} y_i^2 = \sum_{i<j} \pi_i p_{i,j}(y_i^2 + y_j^2)$.

The inequality then becomes

$$\begin{aligned} \sum_i \sum_j x_j b_{j,i} y_i &\geq 2\sum_{i<j} -(\pi_i p_{i,j}) y_j y_i + \sum_{i<j} \pi_i p_{i,j}(y_i^2 + y_j^2) \\ &= \sum_{i<j} \pi_i p_{i,j}(y_i - y_j)^2 \qquad (4.3) \end{aligned}$$

Using the fact that $\sum_{i \in S} \pi_i y_i^2 \neq 0$, we can thus rearrange equation 4.2 to get

$$\begin{aligned} 1 - \lambda_2 &= \frac{\sum_i \sum_j x_j b_{j,i} y_i}{\sum_{i \in S} \pi_i y_i^2} \\ &\geq \frac{\sum_{i<j} \pi_i p_{i,j}(y_i - y_j)^2}{\sum_{i \in S} \pi_i y_i^2} \quad \text{[by 4.3]} \qquad (4.4) \end{aligned}$$

To aid estimation of 4.4, we introduce the following sum:

$$\begin{aligned} \sum_{i<j} \pi_i p_{i,j}(y_i + y_j)^2 &= \sum_{i<j} \pi_i p_{i,j}(y_i^2 + 2y_i y_j + y_j^2) \\ &\leq 2\sum_{i<j} \pi_i p_{i,j}(y_i^2 + y_j^2) \\ &\quad \text{[as } (y_i - y_j)^2 = y_i^2 - 2y_i y_j + y_j^2 \geq 0 \Rightarrow y_i^2 + y_j^2 \geq 2y_i y_j] \\ &= 2\sum_{i \in S} \sum_{i<j} \pi_i p_{i,j}(y_i^2 + y_j^2) \\ &\quad \text{[since if } i \notin S, \text{ it follows that } j \notin S \text{ and so } y_i^2 + y_j^2 = 0] \\ &= 2\sum_{i \in S} \sum_{j \neq i} \pi_i p_{i,j} y_i^2 \\ &\quad \text{[for similar reasons as before]} \\ &= 2\sum_{i \in S} \pi_i y_i^2 \sum_{j \neq i} p_{i,j} \\ &\leq 2\sum_{i \in S} \pi_i y_i^2 \end{aligned}$$

Since $2\sum_{i \in S} \pi_i y_i^2 > 0$, we may divide through by this sum to get

$$\frac{\sum_{i<j} \pi_i p_{i,j}(y_i + y_j)^2}{2\sum_{i \in S} \pi_i y_i^2} \leq 1$$

Combining this inequality with 4.4, we get

$$
\begin{aligned}
1-\lambda_2 &\geq \left(\frac{\sum_{i<j}\pi_i p_{i,j}(y_i-y_j)^2}{\sum_{i\in S}\pi_i y_i^2}\right)\left(\frac{\sum_{i<j}\pi_i p_{i,j}(y_i+y_j)^2}{2\sum_{i\in S}\pi_i y_i^2}\right)\\
&\geq \frac{\left(\sum_{i<j}\pi_i p_{i,j}(y_i-y_j)(y_i+y_j)\right)^2}{2\left(\sum_{i\in S}\pi_i y_i^2\right)^2}\\
&\quad \text{[where the numerator is found by the Cauchy-Schwarz inequality]}\\
&= \frac{1}{2}\left(\frac{\sum_{i<j}\pi_i p_{i,j}(y_i^2-y_j^2)}{\sum_{i\in S}\pi_i y_i^2}\right)^2 \qquad (4.5)
\end{aligned}
$$

So if we manage to get $\Phi_{\mathcal{Q}}$ as a lower bound for the quotient in 4.5, we're done.

For every $0 \leq k \leq N$, define $S_k := \{1, \ldots, k\}$. With regard to the numerator of the quotient in 4.5,

$$
\begin{aligned}
\sum_{i<j}\pi_i p_{i,j}(y_i^2-y_j^2) &= \sum_{i<j}\left(\pi_i p_{i,j}\sum_{k=i}^{j-1}(y_k^2-y_{k+1}^2)\right)\\
&= \sum_{i<j}\sum_{k=i}^{j-1}\pi_i p_{i,j}(y_k^2-y_{k+1}^2)
\end{aligned}
$$

Trying to find a way to re-express the right-hand side directly is difficult. Things will be easier if we relate the summation back to the graph $\mathcal{Q}$. When does a $\pi_i p_{i,j}(y_k^2-y_{k+1}^2)$ term contribute something towards the overall sum? Obviously, it is necessary that $p_{i,j} \neq 0$. In other words, there must be an edge $\overrightarrow{ij}$ in $\mathcal{Q}$. Indeed, summing over all pairs $i < j$ is equivalent to summing over all edges $\overrightarrow{ij} \in E(\mathcal{Q})$ where $i < j$.

Consider each $k < N$. In view of the previous discussion, the number of $\pi_i p_{i,j}(y_k^2-y_{k+1}^2)$ terms that appear in the sum will simply be the number of edges that pass from S_k to $\overline{S_k}$. Hence, for fixed k, the contribution towards the overall sum will be

$$
\sum_{i\in S_k}\sum_{j\in\overline{S_k}}\pi_i p_{i,j}(y_k^2-y_{k+1}^2) = (y_k^2-y_{k+1}^2)\sum_{i\in S_k}\sum_{j\in\overline{S_k}}\pi_i p_{i,j}
$$

Taking the sum over all possible k then, we get the original equality to be

$$\begin{aligned}
\sum_{i<j} \pi_i p_{i,j}(y_i^2 - y_j^2) &= \sum_{k=1}^{N-1}(y_k^2 - y_{k+1}^2)\sum_{i\in S_k}\sum_{j\in \overline{S_k}} \pi_i p_{i,j} \\
&= \sum_{k=1}^{r}(y_k^2 - y_{k+1}^2)\sum_{i\in S_k}\sum_{j\in \overline{S_k}} \pi_i p_{i,j} \\
& \quad [\text{since if } k > r \text{ then } y_k = y_{k+1} = 0] \\
&= \sum_{k=1}^{r}(y_k^2 - y_{k+1}^2)F_{S_k} \quad (6)
\end{aligned}$$

It may be observed here that since $k \leq r$, the only S_k that are included in (6) are those S_k that are subsets of S.

By definition of $\Phi_{\mathcal{Q}}$, $\Phi_{\mathcal{Q}} \leq \Phi_{S_k} = \frac{F_{S_k}}{C_{S_k}}$ for each S_k. Therefore, we have that $F_{S_k} \geq \Phi_{\mathcal{Q}} C_{S_k}$. (6) then becomes

$$\begin{aligned}
\sum_{i<j} \pi_i p_{i,j}(y_i^2 - y_j^2) &\geq \sum_{k=1}^{r}(y_k^2 - y_{k+1}^2)\Phi_{\mathcal{Q}} C_{S_k} \\
&= \Phi_{\mathcal{Q}} \sum_{k=1}^{r}(y_k^2 - y_{k+1}^2) C_{S_k} \\
&= \Phi_{\mathcal{Q}} \sum_{k=1}^{r}(y_k^2 - y_{k+1}^2) \sum_{i=1}^{k} \pi_i
\end{aligned}$$

In the present inequality, each summand in the sum on the right-hand side has the form $(y_k^2 - y_{k+1})^2 \pi_i$. It is clear from the right-hand side that i never rises above r. For each $i' \leq r$, there is a $(y_k^2 - y_{k+1})^2 \pi_{i'}$ term for all $i' \leq k \leq r$. This is because $\pi_{i'}$ only gets the chance to appear in the overall summation when k reaches i'. So for fixed i, the contribution towards the overall sum is

$$\sum_{k=i}^{r}(y_k^2 - y_{k+1}^2)\pi_i = \pi_i \sum_{k=i}^{r}(y_k^2 - y_{k+1}^2).$$

The present inequality then becomes

$$\begin{aligned}
\sum_{i<j} \pi_i p_{i,j}(y_i^2 - y_j^2) &\geq \Phi_{\mathcal{Q}} \sum_{i=1}^{r} \pi_i \sum_{k=i}^{r} (y_k^2 - y_{k+1}^2) \\
&= \Phi_{\mathcal{Q}} \sum_{i\in S} \pi_i \sum_{k=i}^{r} (y_k^2 - y_{k+1}^2) \quad [\text{as } S = \{1, \ldots, r\}] \\
&= \Phi_{\mathcal{Q}} \sum_{i\in S} \pi_i (y_i^2 - y_{i+1}^2 + \ldots + y_r^2 - y_{r+1}^2) \\
&= \Phi_{\mathcal{Q}} \sum_{i\in S} \pi_i (y_i^2 - y_{r+1}^2) \\
&= \Phi_{\mathcal{Q}} \sum_{i\in S} \pi_i y_i^2 \quad [\text{as } y_{r+1} = 0]
\end{aligned}$$

Then dividing the inequality through by $\sum_{i\in S} \pi_i y_i^2$ will give the lower bound we wanted for 4.5. Therefore

$$1 - \lambda_2 \geq \frac{\Phi_{\mathcal{Q}}^2}{2}$$

as desired. □

We can immediately conclude that $\frac{1}{1-\lambda_2} \leq \frac{2}{\Phi_{\mathcal{Q}}^2}$. We need to relate $\Phi_{\mathcal{Q}}$ to n now. This is done by the following lemma.

Lemma 4.2. For the Markov chain $\mathcal{M}$ on the digraph $\mathcal{Q}$ under consideration, $\Phi_{\mathcal{Q}} \geq \frac{1}{192n^6}$.

Proof. Let $S \subset V(\mathcal{Q})$. By Proposition 2.5, the number of edges $e(S, \overline{S})$ going from S to $\overline{S}$ is at least $\frac{|S|}{96n^4}$. Since each edge has the same probability, namely $\frac{1}{2|E|}$,

$$F_S = e(S, \overline{S}) \frac{1}{N} \frac{1}{2|E|} \geq \frac{|S|}{192n^4 N|E|}.$$

It is easily seen that $C_S \geq \frac{|S|}{N}$, so

$$\Phi_S \geq \frac{|S|}{192n^4 N|E|} \frac{N}{|S|} = \frac{1}{192n^4|E|}$$

But $|E| \leq n^2$ as $G \subseteq K_{n,n}$. This means that for any set $S \subset V(\mathcal{Q})$,

$$\Phi_S \geq \frac{1}{192n^6}.$$

This proves the claim. □

Altogether then, Lemmas 4.1 and 4.2 imply that $\frac{1}{1-\lambda_2} \leq 2(192n^6)^2 = 73728n^{12}$. This means that, going back to the number of steps τ needed, the following Proposition holds:

Proposition 4.1. $\tau = \mathrm{O}(n^{15})$. In particular, running the Markov chain for $\tau = \lceil 600000(\ln n)(n^{14}) \rceil$ steps yields a relative pointwise distance $\Delta(\tau) \leq \frac{1}{n^4}$ to the uniform distribution.

Proof. If $\tau \geq 600000(\ln n)(n^{14})$ then

$$\begin{aligned} \tau &\geq 8(\ln n)(n^2 \ln 2)(73728n^{12}) \\ &\geq 8(\ln n)(\ln N)(73728n^{12}) \quad \text{[as noted in Section 4.1]} \\ &\geq 8\frac{\ln n \ln N}{1-\lambda_2} \end{aligned}$$

which, as observed in Section 4.1, gave a suitable accuracy for the matching generator. So the suggested value for τ works. □

4.3 The Case of $M_r \cup M_{r-1}$ for Arbitrary r

We have now shown that the matching generator is an efficient, almost-uniform sampler, but only for the case of $M_n \cup M_{n-1}$. We must now show that this remains the case for $M_r \cup M_{r-1}$ where $r < n$. Following Motwani and Raghavan [10], this is done by modifying the algorithm slightly. In essence, we change the problem of sampling from $M_r \cup M_{r-1}$ into a problem of sampling from perfect and near-perfect matchings in a new graph G'.

Let $0 < r < n$. Recall that the graph G we are working with is bipartite with vertex classes X and Y, both of size n. Given this G, we define a new bipartite graph G' by adding $n - r$ new vertices to each vertex class X and Y. We then connect every new vertex $x' \in X$ and $y' \in Y$ to every old vertex $y \in Y$ and $x \in X$, respectively. Note that now $|X| = |Y| = 2n - r$ and $|G'| = |G| + 2n - 2r \leq 4n$. So G' may be created in time polynomial in $|G|$. Note also, that the notation M_r continues to mean the set of all matchings of size r in G. It is now easy to prove the following Propositions:

Proposition 4.2. G' contains a perfect matching if and only if G contains a matching of size r. Furthermore, for each $m \in M_r$, there are exactly $((n-r)!)^2$ perfect matchings in G'.

Proof. Let m' be a perfect matching in G'. Then all of the $n-r$ new vertices in X are matched to $n - r$ old vertices in Y. The situation is symmetric for Y. Thus X and Y each have r old vertices that must be matched using edges in G. This proves the forward implication in the first statement.

For the reverse implication, let m be a matching of size r in G. Then X and Y each have $n - r$ unmatched old vertices. But each vertex class has $n - r$ new vertices that are connected to every old vertex in the other class. Thus each unmatched old vertex can be matched to a new vertex in the opposite class. This gives a perfect matching in G'.

For the second statement, let $m \in M_r$ and build a perfect matching in G' from m. There are exactly $(n-r)!$ ways to match the unmatched old vertices in X to the new vertices in Y. The situation is similar for the unmatched old vertices in y. Thus, given m, we can generate exactly $((n-r)!)^2$ perfect matchings in G'. □

For the next proposition, a *near-perfect* matching is a matching that covers all but 2 vertices of G.

Proposition 4.3. G' contains a near-perfect matching if and only if G contains a matching of size $r + 1$, r or $r - 1$. Furthermore,

1. For each $m \in M_{r+1}$ there are exactly $((n-r)!)^2$ near-perfect matchings in G'.
2. For each $m \in M_r$ there are exactly $2(n - r)((n - r)!)^2$ near-perfect matchings in G'.
3. For each $m \in M_{r-1}$ there are exactly $(n-r+1)^2((n-r)!)^2$ near-perfect matchings in G'.

Proof. First we prove the forward implication in the first statement. Let m be a near-perfect matching in G'. There are three possible cases:

1. There are two new vertices $x' \in X, y' \in Y$ that are unmatched in m.
2. There is an old vertex in one class and a new vertex in the other class that are unmatched in m.
3. There are two old vertices $x \in X, y \in Y$ that are unmatched in m.

For Case 1, there are $n - r - 1$ new vertices in X that are connected to $n - r - 1$ old vertices in Y. Similarly, there are $n - r - 1$ new vertices in Y that are connected to $n - r - 1$ old vertices in X. This leaves X and Y each containing $r + 1$ old vertices that must be matched using edges in G.

For Case 2, we suppose, without loss of generality, that the unmatched old vertex is $x \in X$ and the unmatched new vertex is $y' \in Y$. The other case will be symmetric. So all $n - r$ new $x' \in X$ must be matched to $n - r$ old $y \in Y$. But this forces the other r old vertices in Y to be matched to r old vertices in X using edges in G.

For Case 3, each of the $n-r$ new vertices in one class must be matched by m to an old vertex to the opposing class. But each class has an unmatched old vertex. So there must be $r - 1$ old vertices in X that are matched to old vertices in Y using edges in G.

For the reverse implication, we consider the three cases in turn. We will prove the claims of the second statement as we do so. Let $m \in M_{r+1}$. This leaves $n - r - 1$ old vertices in each class that can be matched to $n - r - 1$ new vertices in the opposing class. Two new vertices must be left unmatched. Thus, we can build a near-perfect matching in G' from m. Furthermore, to build such a matching, we must choose the two new vertices to be unmatched. There are $n-r$ possibilities for each class, and so there are $(n-r)^2$ ways to choose the unmatched vertices. For each of these choices, there are $(n-r-1)!$ ways to match the available old vertices in X to the available new vertices in Y. The situation is similar for the available old vertices in Y. So altogether, we can build $(n-r)^2((n-r-1)!)^2 = ((n-r)!)^2$ near-perfect matchings in G' from m

Let $m \in M_r$. We can connect every available old vertex in X, say, to every new vertex in Y. But then it is possible to connect all but one available old vertex in Y to $n - r - 1$ new vertices in X. This will give a near-perfect matching in G'. Furthermore, to build this matching we chose one of $(n-r)!$ ways to connect the available old vertices of X to the new vertices of Y. Once this was done, we chose a new $x' \in X$ and an old unmatched $y' \in Y$ to remain unmatched. There were $(n-r)^2$ possible choices here. Once this choice was made, we then had to match the remaining vertices in one of $(n-r-1)!$ ways. Thus, we could have built $(n-r)!(n-r)^2(n-r-1)! = (n-r)((n-r)!)^2)$ near-perfect matchings in this way. But our decision to cover all the old vertices of X was arbitrary. We could have done the same for Y. Thus, given m, we can build $2(n-r)((n-r)!)^2$ near-perfect matchings in G'.

Let $m \in M_{r-1}$. We can connect every new vertex in one class to an available old vertex in the opposing class. But there are $n - r + 1$ old vertices in each class that are unmatched by m. This leaves two old vertices unmatched. Hence, given m, we can build a near-perfect matching in G'. Furthermore, we can build this matching by selecting two available old vertices to remain unmatched. There are $(n - r + 1)^2$ possibilities here. Once this is done, we can connect the remaining vertices. There are, using similar reasoning to the above, $((n-r)!)^2$ possible ways to do this. So, given M, we can build $(n-r+1)^2((n-r)!)^2$ near-perfect matchings in G'. □

Let $R_1(r)$ be the number of perfect matchings in G' and $R_2(r)$ the number of near-perfect matchings in G'. Given Propositions 4.2 and 4.3, $R_1(r) = ((n-r)!)^2|M_r|$ and $R_2(r) = ((n-r)!)^2(|M_{r+1}| + 2(n-r)|M_r| + (n-r+1)^2|M_{r-1}|)$. Therefore,

$$\frac{R_1(r)}{R_2(r)} = \frac{|M_r|}{|M_{r+1}| + 2(n-r)|M_r| + (n-r+1)^2|M_{r-1}|}.$$

If we divide both the numerator and the denominator on the right-hand side by $|M_r|$, we get

$$\frac{R_1(r)}{R_2(r)} = \frac{1}{\frac{|M_{r+1}|}{|M_r|} + 2(n-r) + (n-r+1)^2 \frac{|M_{r-1}|}{|M_r|}}.$$

Rearranging gives

$$\frac{|M_r|}{|M_{r-1}|} = (n-r+1)^2 \left[\frac{R_2(r)}{R_1(r)} - 2(n-r) - \frac{|M_{r+1}|}{|M_r|} \right]^{-1}. \tag{4.6}$$

This is all very well, but how does this formula help us in finding $\frac{|M_r|}{|M_{r-1}|}$? Suppose we run our modified algorithm on a suitable $n \times n, (0,1)$ matrix $\mathbf{A}$. By the previous chapter, the algorithm will find an estimate for $\frac{|M_n|}{|M_{n-1}|}$. It will also find estimates for the values $\frac{R_1(2)}{R_2(2)}, \frac{R_1(3)}{R_2(3)}, \ldots, \frac{R_1(n-1)}{R_2(n-1)}$. Then 4.6 applied to $\frac{R_1(n-1)}{R_2(n-1)}$ and $\frac{|M_n|}{|M_{n-1}|}$ will give an estimate for $\frac{|M_{n-1}|}{|M_{n-2}|}$. Then 4.6 can be used again with $\frac{R_1(n-2)}{R_2(n-2)}$ to give an estimate for $\frac{|M_{n-2}|}{|M_{n-3}|}$, and so on until an estimate of $\frac{|M_2|}{|M_1|}$ is reached. This gives us a suitable way around the original problem of sampling from $M_r \cup M_{r-1}$. It is important to note that using the matching generator to sample directly from $M_r \cup M_{r-1}$ could well be accurate and efficient. It is just that we are unable to show this.

Chapter 5

The Running Time of the Approximation Scheme

We have seen that the matching generator will sample almost uniformly from $M_r \cup M_{r-1}$ in time polynomial in n. But this was part of a wider project: showing that the permanent-estimation algorithm was a FPRAS. We now need to show that we need only a comparatively small number of samples to estimate $|M_n|$. To accomplish this, we will find the following mathematical tools useful.

5.1 A Mathematical Toolkit: The Markov Inequality, Chernoff Bounds and Sampling Techniques

The following are all results in Probability Theory. They will be helpful in establishing how many samples we need from $M_r \cup M_{r-1}$ to get a good estimate of $\frac{|M_r|}{|M_{r-1}|}$.

5.1.1 The Markov Inequality

This is a fairly basic result in Probability Theory. It is quoted in many a textbook in this subject area. Essentially, it gives the lower bound on the likelihood that a non-negative random variable is larger than a given positive value. The bound it establishes is not always tight. It is stated here since Chernoff bounds, some of which will be useful later, are derived from it. As a point of clarification, we denote by $\mathbb{E}(X)$ the *expected value of* X.

Theorem 5.1. (The Markov Inequality) Let X be a random variable that can assume only non-negative values. Let $a > 0$. Then

$$\mathbf{Pr}(X \geq a) \leq \frac{\mathbb{E}(X)}{a}$$

.

Proof. (From [9]) Let

$$I := \begin{cases} 1, & \text{if } X \geq a; \\ 0, & \text{if } X < a. \end{cases} .$$

Since $X \geq 0$ and $a > 0$, if $I = 0$ then $I < \frac{X}{a}$. Also, if $I = 1$ then $X \geq a$ and $I = 1 \leq \frac{X}{a}$. Thus,

$$I \leq \frac{X}{a}.$$

Now,

$$\begin{aligned} \mathbb{E}(I) &= 0 \cdot \mathbf{Pr}(X < a) + 1 \cdot \mathbf{Pr}(X \geq a) \\ &= \mathbf{Pr}(X \geq a) \end{aligned}$$

So,

$$\mathbf{Pr}(X \geq a) = \mathbb{E}(I) \leq \mathbb{E}\left(\frac{X}{a}\right) = \frac{\mathbb{E}(X)}{a}.$$

□

5.1.2 Chernoff Bounds

The term "Chernoff bounds" refers to a family of bounds derived from the Markov inequality. What unites them is the technique used to derive them: all their derivations involve applying the Markov inequality to the random variable e^{tX} (where $t \in \mathbb{R}$ is usually chosen to suit the needs of the moment), rather than to X itself. For instance, if $X \geq a$ and $t > 0$ then $e^{tX} \geq e^{ta}$, as the exponential function is an increasing function. Then we have

$$\mathbf{Pr}(X \geq a) = \mathbf{Pr}(e^{tX} \geq e^{ta}) \leq \frac{\mathbb{E}\left(e^{tX}\right)}{e^{ta}},$$

where the inequality follows from the Markov inequality. But this gives us a Chernoff bound on $\mathbf{Pr}(X \geq a)$; namely,

$$\mathbf{Pr}(X \geq a) \leq \frac{\mathbb{E}\left(e^{tX}\right)}{e^{ta}}.$$

A well-chosen t can give a fairly tight bound. For this reason, a Chernoff bound may be preferable to the Markov inequality in assessing the likelihood of an outcome. The following Chernoff bounds will be useful in the analysis of our sampling procedure. All the Chernoff bounds in this section, and their proofs, are taken from [9].

Theorem 5.2. Let $X_1, \ldots, X_n$ be independent, binary, random variables such that $\mathbf{Pr}(X_i = 1) = p_i > 0$ for all i. Let $X := \sum_{i=1}^{n} X_i$ and let $\mu := \mathbb{E}(X)$. Then for any $\delta > 0$,

$$\mathbf{Pr}(X \geq (1+\delta)\mu) < \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\mu}.$$

Proof. For any $t > 0$,

$$\mathbf{Pr}(X \geq (1+\delta)\mu) \leq \frac{\mathbb{E}\left(e^{tX}\right)}{e^{t(1+\delta)\mu}},$$

from the Chernoff bound given at the beginning of this section.

Now,

$$e^{tX} = \exp\left(t\sum_{i=1}^{n} X_i\right) = \prod_{i=1}^{n} e^{tX_i}.$$

The X_i are independent, so the e^{tX_i} are mutually independent. Thus we have

$$\mathbb{E}\left(e^{tX}\right) = \mathbb{E}\left(\prod_{i=1}^{n} e^{tX_i}\right) = \prod_{i=1}^{n} \mathbb{E}\left(e^{tX_i}\right).$$

For each e^{tX_i}, $e^{tX_i} = e^{t\cdot 0} = 1$ with probability $\mathbf{Pr}(X_i = 0) = 1 - p_i$. Similarly, the probability that $e^{tX_i} = e^t$ is p_i. So,

$$\mathbb{E}\left(e^{tX}\right) = \prod_{i=1}^{n} \left((1-p_i) + e^t p_i\right) = \prod_{i=1}^{n} \left(1 + p_i\left(e^t - 1\right)\right).$$

We now use the fact that $1 + x \leq e^x$ for any $x \in \mathbb{R}$ to get

$$\mathbb{E}\left(e^{tX}\right) \leq \prod_{i=1}^{n} e^{p_i(e^t-1)} = \exp\left(\sum_{i=1}^{n} p_i\left(e^t - 1\right)\right) = \exp\left(\left(e^t - 1\right)\mathbb{E}(X)\right) = e^{(e^t-1)\mu}.$$

Going back to our original inequality, this gives

$$\mathbf{Pr}(X \geq (1+\delta)\mu) \leq \frac{e^{(e^t-1)\mu}}{e^{t(1+\delta)\mu}}.$$

Since $\delta > 0$, we can set $t := \ln(1+\delta) > 0$ and get

$$\mathbf{Pr}(X \geq (1+\delta)\mu) \leq \frac{e^{\delta\mu}}{(1+\delta)^{(1+\delta)\mu}} = \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\mu}.$$

$\square$

Theorem 5.3. Let $X_1, \ldots, X_n$ be independent, binary, random variables such that $\mathbf{Pr}(X_i = 1) = p_i > 0$ for all i. Let $X := \sum_{i=1}^{n} X_i$ and let $\mu := \mathbb{E}(X)$. Then for any $0 < \delta \leq 1$,

$$\mathbf{Pr}(X \geq (1+\delta)\mu) \leq e^{-\frac{1}{3}\mu\delta^2}.$$

Proof. Given Theorem 5.2, it is sufficient to show that

$$\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}} \leq e^{-\frac{1}{3}\delta^2}.$$

Take the function

$$f(\delta) := \delta - (1+\delta)\ln(1+\delta) + \frac{\delta^2}{3}.$$

If we take first and second derivatives, we get

$$\begin{aligned} f'(\delta) &= -\ln(1+\delta) + \frac{2}{3}\delta, \\ f''(\delta) &= -\frac{1}{1+\delta} + \frac{2}{3}. \end{aligned}$$

For $0 < \delta \leq \frac{1}{2}$, $f''(\delta) \leq 0$ and for $\frac{1}{2} < \delta < 1$, $f''(\delta) > 0$. So $f'(\delta)$ first decreases as δ goes from 0 to $\frac{1}{2}$. It then increases for the remainder of the interval. $\ln(2) > \frac{2}{3}$, so $f'(1) < 0$. This, combined with the fact that $f'(0) = 0$ means that $f'(\delta) \leq 0$ for $0 \leq \delta \leq 1$. Thus, $f(\delta)$ is decreasing on the interval in question. But $f(0) = 0$, so $f(\delta) \leq 0$.

However, if we take the exponential function of both sides, we get

$$\exp\left(\delta - (1+\delta)\ln(1+\delta) + \frac{\delta^2}{3}\right) \leq 1.$$

Since the exponential function is increasing, the inequality is preserved. Simplifying, we have

$$\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}} e^{\frac{1}{3}\delta^2} \leq 1.$$

Multiplying both sides by $e^{-\frac{1}{3}\delta^2}$ will give us what we want. □

Theorem 5.4. Let $X_1, \ldots, X_n$ be independent, binary, random variables such that $\mathbf{Pr}(X_i = 1) = p_i > 0$ for all i. Let $X := \sum_{i=1}^{n} X_i$ and let $\mu := \mathbb{E}(X)$. Then for any $0 < \delta < 1$,

$$\mathbf{Pr}(X \leq (1-\delta)\mu) \leq \left(\frac{e^{-\delta}}{(1-\delta)^{(1-\delta)}}\right)^{\mu}.$$

Proof. Here, we take $t := \ln(1-\delta) < 0$, since $0 < \delta < 1$. So if $X \leq a$ then $e^{tX} \geq e^{ta}$. Using by now familiar reasoning, we have

$$\begin{aligned} \mathbf{Pr}(X \leq (1-\delta)) &= \mathbf{Pr}(e^{tX} \geq e^{t(1-\delta)\mu}) \\ &\leq \frac{\mathbb{E}\left(e^{tx}\right)}{e^{t(1-\delta)\mu}} \\ &\leq \frac{e^{(e^t-1)\mu}}{e^{t(1-\delta)\mu}} \\ &= \left(\frac{e^{-\delta}}{(1-\delta)^{(1-\delta)}}\right)^{\mu} \end{aligned}$$

This establishes the inequality. $\square$

Theorems 5.2, 5.3 and 5.4 can be seen as a build-up to the following result:

Theorem 5.5. Let $X_1, \ldots, X_n$ be independent, binary, random variables such that $\mathbf{Pr}(X_i = 1) = p_i > 0$ for all i. Let $X := \sum_{i=1}^{n} X_i$ and let $\mu := \mathbb{E}(X)$. Then for any $0 < \delta < 1$,

$$\mathbf{Pr}(|X - \mu| \geq \delta\mu) \leq 2e^{-\frac{1}{3}\mu\delta^2}.$$

Proof.

$$\begin{aligned}
\mathbf{Pr}(|X-\mu| \geq \delta\mu) &= \mathbf{Pr}(X - \mu \leq -\delta\mu \text{ or } \delta\mu \leq X - \mu) \\
&= \mathbf{Pr}(X \leq (1-\delta)\mu) + \mathbf{Pr}(X \geq (1+\delta)\mu) \\
&\leq \left(\frac{e^{-\delta}}{(1-\delta)^{(1-\delta)}}\right)^{\mu} + \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\mu}
\end{aligned}$$

Now,

$$\begin{aligned}
\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}} > \frac{e^{-\delta}}{(1-\delta)^{(1-\delta)}} &\Leftrightarrow e^{2\delta} > (1+\delta)^{(1+\delta)}(1-\delta)^{(\delta-1)} \\
&\Leftrightarrow 2\delta > \ln(1+\delta) + \delta\ln(1-\delta^2) - \ln(1-\delta) \\
&\Leftrightarrow 2\delta > \sum_{i=1}^{\infty} \frac{(-1)^{i-1}\delta^i}{i} + \sum_{i=1}^{\infty} \frac{-\delta^{2i+1}}{i} + \sum_{i=1}^{\infty} \frac{\delta^i}{i} \\
&\Leftrightarrow 2\delta > 2\delta - \sum_{i=1}^{\infty} \frac{\delta^{2i+1}}{i(2i+1)}
\end{aligned}$$

which clearly holds as $0 < \delta < 1$. So this means that

$$\begin{aligned}
\mathbf{Pr}(|X-\mu| \geq \delta\mu) &< 2\left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\mu} \\
&\leq 2e^{-\frac{1}{3}\mu\delta^2} \quad \text{[by the proof of Theorem 5.3]}
\end{aligned}$$

$\square$

5.2 Bounds on the Number of Samples Needed

We now seek to use the Chernoff bounds to show that we need not take many samples to get a good approximation of $\frac{|M_r|}{|M_{r-1}|}$.

Definition 5.1. A sampling procedure is said to sample from a set $A \cup B$ with *error* d if, for any sample s, the probability that $s \in A$ is such that $\left|\mathbf{Pr}(s \in A) - \frac{|A|}{|A \cup B|}\right| \leq d$.

Suppose we have a set $A \cup B$ and a procedure that allows us to sample from this set with error d. Let $R := \frac{|A|}{|A \cup B|}$ and take m independent samples from $A \cup B$. Let $X := \sum_{i=1}^{m} X_i$ be the sum of m independent random variables with each

$$X_i := \begin{cases} 1, & \text{if sample } i \text{ is in } A; \\ 0, & \text{if sample } i \text{ is not in } A. \end{cases}$$

With this set-up, we can establish the next Theorem.

Theorem 5.6. Let $\epsilon, \delta > 0$ with $d < \frac{\epsilon}{2}$. Let X be as described and suppose that $\omega > \frac{3}{R(2\epsilon)^2} \ln\left(\frac{2}{\delta}\right)$. Then $\mathbf{Pr}(|X - R\omega| \geq \epsilon R\omega) < \delta$.

Proof. Let $\mu := \mathbb{E}(X)$. This means that $R\omega(1-d) \leq \mu \leq R\omega(1+d)$. So μ lies in an interval of length $2dR\omega < \epsilon R\omega$. So, using the error bound on μ, we have

$$\mathbf{Pr}(|X - R\omega| \geq \epsilon R\omega) \leq \mathbf{Pr}(|X - \mu| \geq 2\epsilon\mu).$$

But then we can apply the Chernoff bound of Theorem 5.5 to get

$$\mathbf{Pr}(|X - \mu| \geq 2\epsilon\mu) \leq 2e^{-\frac{1}{3}R\omega(2\epsilon)^2} < \delta$$

□

But Theorem 5.6 gives us an idea of how large a sample we need to force a certain level of accuracy when estimating a single value, $\frac{|M_r|}{M_{r-1}|}$ for some r. We need to consider the cumulative effect of all errors on the final estimate of $\frac{|M_n|}{|M_1|}$. Theorem 5.8 will accomplish this. However, the proof of this theorem depends on some inequalities, which we prove beforehand.

Proposition 5.1. For all $x \in \mathbb{R}, k \in \mathbb{N}$,

$$(1-x)^k \geq 1 - 2kx.$$

Proof. The proof will be by induction on k.

Suppose $k = 1$. Then $(1-x)^k = 1 - x \geq 1 - 2kx$, and the statement is satisfied in this case.

So suppose $k > 1$ and the statement holds for all $k' < k$. Then

$$\begin{aligned} (1-x)^k &= (1-x)^{k-1} - x(1-x)^{k-1} \\ &\geq 1 - 2(k-1)x - x(1-2(k-1)x) \quad \text{[by inductive assumption]} \\ &\geq 1 - 2kx \end{aligned}$$

Therefore the result holds by induction. □

Proposition 5.2. For all $x \geq 0, k \in \mathbb{N}$ with $xk \leq \frac{1}{2}$,

$$(1+x)^k \leq 1 + 2kx.$$

Proof. The proof will be by induction on k.

Suppose $k = 1$. Then $(1+x)^k = 1 + x \leq 1 + 2kx$, and the statement is satisfied in this case.

So suppose $k > 1$ and the statement holds for all $k' < k$. Now when $x = 0$,$(1+x)^k = 1+2kx$. Now if $f(x) := 1+2kx$ and $g(x) := (1+x)^k$ then $f'(x) = k$ and

$$\begin{aligned} g'(x) &= k(1+x)^{k-1} \\ &\leq k(1+2kx) \\ &\quad \text{[since } (1+x)^{k-1} \leq 1+2(k-1)x \leq 1+2kx \text{ by assumption]} \\ &\leq 2k. \quad \left[\text{since } kx \leq \frac{1}{2}\right] \end{aligned}$$

So comparing derivatives we have $g'(x) \leq f'(x)$ for x as specified. So since $f(0) = g(0)$ this means that $(1+x)^k \leq 1+2kx$ for x, k as given. Therefore the result holds by induction. □

The following theorem follows easily from the fact that $\mathbf{Pr}(A \cup B) = \mathbf{Pr}(A) + \mathbf{Pr}(B) - \mathbf{Pr}(A \cap B)$. It is mentioned in [9].

Theorem 5.7. (*The union bound*) If $E_1, \ldots, E_n$ is a sequence of events then

$$\mathbf{Pr}\left(\bigcup_{i=1}^{n} E_i\right) \leq \sum_{i=1}^{n} \mathbf{Pr}(E_i).$$

Now recall that the algorithm finds estimates for $r_1, \ldots, r_{n-1}$ for $\frac{|M_2|}{|M_1|}, \ldots, \frac{|M_n|}{|M_{n-1}|}$ respectively. Its final output is the estimated permanent

$$|M_1| \prod_{i=1}^{n-1} r_i$$

while the actual permanent is

$$|M_1| \prod_{i=1}^{n-1} \frac{|M_{i+1}|}{|M_i|}.$$

Let

$$R' := \prod_{i=1}^{n-1} \left(r_i \frac{|M_i|}{|M_{i+1}|} \right).$$

So R' is the estimated permanent divided by the actual permanent. So obviously the closer R' is to 1, the more accurate the estimate. The following Theorem uses R' place bounds on the cumulative errors of $r_1, \ldots, r_{n-1}$. The Theorem and its proof are drawn from [9].

Theorem 5.8. Let $s := n-1$. Suppose that for each $i \in \{1, \ldots, s\}$, r_i is such that

$$\mathbf{Pr}\left(\left|\frac{|M_{i+1}|}{|M_i|} - r_i\right| \leq \frac{\epsilon}{2s}\frac{|M_{i+1}|}{|M_i|}\right) \geq 1 - \frac{\delta}{s}$$

for some $\frac{\epsilon}{2s}, \frac{\delta}{s} > 0$. Then

$$\mathbf{Pr}(|R' - 1| \leq \epsilon) \geq 1 - \delta.$$

Proof. The condition on r_i is equivalent to

$$\mathbf{Pr}\left(\left|\frac{|M_{i+1}|}{|M_i|} - r_i\right| > \frac{\epsilon}{2s}\frac{|M_{i+1}|}{|M_i|}\right) < \frac{\delta}{s}.$$

Using the union bound,

$$\mathbf{Pr}\left(\forall i, \left|\frac{|M_{i+1}|}{|M_i|} - r_i\right| > \frac{\epsilon}{2s}\frac{|M_{i+1}|}{|M_i|}\right) < (n-1)\frac{\delta}{s} = \delta.$$

So the probability that $\left|\frac{|M_{i+1}|}{|M_i|} - r_i\right| \leq \frac{\epsilon}{2s}\frac{|M_{i+1}|}{|M_i|}$ for all i is at least $1-\delta$. Hence $\left(1 - \frac{\epsilon}{2s}\right)\frac{|M_{i+1}|}{|M_i|} \leq r_i$ or $r_i \leq \left(1 + \frac{\epsilon}{2s}\right)\frac{|M_{i+1}|}{|M_i|}$ with probability at least $1-\delta$. So

$$1 - \frac{\epsilon}{2s} \leq r_i \frac{|M_i|}{|M_{i+1}|} \leq 1 + \frac{\epsilon}{2s}$$

with probability at least $1-\delta$. Therefore

$$\begin{aligned} 1-\epsilon &\leq 1 - \frac{\epsilon}{2s} \\ &\leq \left(1 - \frac{\epsilon}{2s}\right)^s \quad \text{[by Proposition 5.1]} \\ &\leq \prod_{i=1}^{n-1}\left(r_i \frac{|M_i|}{|M_{i+1}|}\right) = R'. \end{aligned}$$

and

$$\begin{aligned} 1+\epsilon &\geq 1 + \frac{\epsilon}{2s} \\ &\geq \left(1 + \frac{\epsilon}{2s}\right)^s \quad \text{[by Proposition 5.2]} \\ &\geq \prod_{i=1}^{n-1}\left(r_i \frac{|M_i|}{|M_{i+1}|}\right) = R'. \end{aligned}$$

with probability at least $1-\delta$. This proves the theorem. $\square$

With regard to the current problem, suppose we use the matching generator to produce ω samples from $M_n \cup M_{n-1}$. Then $\frac{X}{\omega}$ is an estimate for $\frac{|M_n|}{|M_n \cup M_{n-1}|}$. By the same token, $\frac{\omega - X}{\omega}$ estimates $\frac{|M_{n-1}|}{|M_n \cup M_{n-1}|}$. If $\epsilon_1, \delta_1 > 0$ then by Theorem 5.6 we can choose $\omega > \frac{3}{R(2\epsilon)^2} \ln\left(\frac{2}{\delta}\right)$ large enough so that

$$\mathbf{Pr}\left(\left|X - \frac{|M_n|}{|M_n \cup M_{n-1}|}\omega\right| \geq \epsilon_1 \frac{|M_n|}{|M_n \cup M_{n-1}|}\omega\right) < \delta_1.$$

Notice this automatically gives an estimate for $|M_{n-1}|$ with

$$\mathbf{Pr}\left(\left|(\omega - X) - \frac{|M_{n-1}|}{|M_n \cup M_{n-1}|}\omega\right| \geq \epsilon_1 \frac{|M_{n-1}|}{|M_n \cup M_{n-1}|}\omega\right) < \delta_1.$$

But this means that the estimate of $\frac{|M_n|}{|M_{n-1}|}$, namely $\frac{X}{\omega - X}$, is such that

$$\mathbf{Pr}\left(\left|\frac{X}{\omega - X} - \frac{|M_n|}{|M_{n-1}|}\right| \geq 2\epsilon_1 \frac{|M_n|}{|M_{n-1}|}\right) < 2\delta_1.$$

Even given the modified algorithm, we may assume that the error bound on each estimate of $\frac{|M_r|}{|M_{r-1}|}$ remains $2\epsilon_1$. This is because of the nature of the formula 4.6. In 4.6, the error bound for $\frac{|M_r|}{|M_{r-1}|}$ will bounded by the error bounds for $\frac{R_2(r)}{R_1(r)}$ and $\frac{|M_{r+1}|}{|M_r|}$. So by Theorem 5.8, if the algorithm calculates $R := \prod_{i=2}^{n} \frac{|M_i|}{|M_{i-1}|}$ then

$$\mathbf{Pr}\left(\left|R - \frac{|M_n|}{|M_1|}\right| \geq \frac{\epsilon_1}{n-1}\frac{|M_n|}{|M_1|}\right) \leq 2\delta_1.$$

Now Theorem 5.6 seems to provide a useful lower bound for the number of samples needed. However, the lower bound includes the value R, one of the quantities the process of sampling is trying to uncover. Before we can apply Theorem 5.6 then, we must set an upper bound on R. This will lead to a suitable lower bound for the number of samples ω. In our modified algorithm, we are always trying to sample perfect and near-perfect matchings. Now, suppose the matching generator is sampling from graph G' with vertex class size $n' \leq n^2$. Each perfect matching contains n' edges, and so has n' near-perfect matchings as subsets. Now consider a pair of edges from a perfect matching. There will be $\binom{n'}{2}$ such pairs. Bearing Proposition 2.1 in mind, these edges lie on at most 2 augmenting paths of length 3. Therefore there are at most 2 near-perfect matchings that augment to the perfect matching using the edges chosen. Now Proposition 2.1 implies that each near-perfect matching can augment to a perfect matching. So for each perfect matching, there are at most $n' + 2\binom{n'}{2} = (n')^2 \leq n^4$ near-prefect matchings. Thus, if R_i is the number of perfect matchings in the graph and r_i the number of near-perfect matchings, then $r_i \leq n^4 R_i$ and $\frac{1}{n^4} \leq \frac{R_i}{r_i}$.

Using the nomenclature above, we have $\alpha = \frac{1}{n^4}$ and $d = \frac{1}{n^4}$. Altogether then, if we take $\delta > 0, \epsilon > \frac{2(n-1)}{n^4} = 2(n-1)d$ and let $\epsilon_1 := \frac{\epsilon}{n-1}, \delta_1 := 2\delta$ then

$$\omega > \frac{3n^4}{2\epsilon_1^2} \ln\left(\frac{2}{\delta_1}\right)$$

will give an (ϵ, δ) bound on the final result by Theorems 5.6 and 5.8. We will have to take ω samples from $n-1$ sets, so altogether the entire algorithm will run in time polynomial in $|G| = n^2, \frac{1}{\epsilon}, \ln\left(\frac{1}{\delta}\right)$. The matching generator runs in polynomial time for each sample. Hence the algorithm forms a FPRAS.

Chapter 6

A Second Problem: Counting Colourings

Having examined an approximation scheme for the permanent of a matrix, we go on to consider another such scheme for a different problem: that of counting the number of k-colourings in a graph. We begin by analyzing an algorithm designed by Jerrum [4]. It approaches the problem in a similar to the scheme that approximate the permanent. It attempts to reach a suitable estimate for the number of k-colourings by sampling from a series of modified graphs. The bulk of this part will be devoted to showing that the sampling method Jerrum uses is efficient and almost-uniform. Having done this we can examine a modification to the sampling algorithm suggested by Thomas Hayes and Eric Vigoda [3]. After this we can then show that, for both sampling procedures, the number of samples needed to estimate the number of k-colourings is polynomially bounded. We begin with some basic definitions.

Definition 6.1. Let C be a finite set. A *colouring* of a graph G is a function $f : V(G) \to C$ such that if x and y are adjacent vertices of G then $f(x) \neq f(y)$.

The "colours" C of $V(G)$ could be anything, but C is usually taken to be $[k] = \{1, \ldots, k\}$ for some $k \in \mathbb{N}$. Also, while it is more accurate to describe such an f as a colouring of $V(G)$, it is often said that f is a colouring of G.

Definition 6.2. $f : V(G) \to C$ is a *k-colouring* of G if $|C| = k$ for some $k \in \mathbb{N}$. The *chromatic number* $\chi(G)$ of G is the smallest $k \in \mathbb{N}$ such that G has a k-colouring.

Definition 6.3. We let $\Omega_k(G)$ denote the set of all k-colourings of a graph G.

Suppose we have $k \in \mathbb{N}$ and we wish to work out how many k-colourings a given graph G has. We take $n := |G|$. Without explicit construction of every possible k-colouring, it is hard to see how to do this. The task of explicit construction is, of course, incredibly inefficient. As for the case with the permanent of a (0,1) matrix, if we are willing to settle for an approximate value then, given a bound on k, there is an FPRAS available that will estimate the number of k-colourings of G. The following is dedicated to the description and analysis of such a scheme.

6.1 The Strategy Behind the Scheme

The strategy here is similar to the strategy involved in estimating $|M_n|$ in the first problem. The idea is to find a sequence of ratios that ultimately relate the unknown quantity (ie. the number of k-colourings) to a known quantity. This is done by manipulating the base graph G itself.

By the definition of "colouring", if xy is an edge of G then x and y must have different colours. Suppose that G is an edgeless graph on n vertices. Then the number of k-colourings of G is easy to work out. Any vertex can be any of the k colours, so the number of k-colourings in this case will then be $|\Omega_k(G)| = k^n$. But suppose G does have some edges. $|\Omega_k(G)|$ is now not so easy to figure out. Some arrangements of edges will have complex ramifications for the possible colourings on G.

Let xy be an edge of G and consider the graph $G-xy$. $G-xy$ will have more k-colourings than G, since we are now able to colour x and y with the same colour. Any colouring of G will still be a valid colouring of $G-xy$, though. Using this fact, we will be able to estimate the ratio $\frac{|\Omega_k(G)|}{|\Omega_k(G-xy)|}$ by taking random samples from $\Omega_k(G-xy)$ and seeing if each sample is a valid colouring for G as well as $G-xy$. The proportion of those that are will give the fraction.

Building on this, we let $m := e(G)$ and produce a sequence of subgraphs $G = G_m \supseteq G_{m-1} \supseteq \ldots \supseteq G_0$ by successively removing edges from G until we have an edgeless G_0. Then if by sampling we manage to estimate $\frac{|\Omega_k(G_i)|}{|\Omega_k(G_{i-1})|}$ for each i, we will be able to estimate

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

as we know that $|\Omega_k(G_0)| = k^n$. That being said, $|\Omega_k(G_1)|$ is easy to work out as well. Let x be an endpoint of the edge in $G-1$ We may colour the other $n-1$ vertices of G any colour, but x may not receive the colour of its neighbour. Hence $|\Omega_k(G_1)| = k^{n-1}(k-1)$. We may thus take $|\Omega_k(G)|$ to be

$$|\Omega_k(G)| = \frac{|\Omega_k(G_m)|}{|\Omega_{m-1}|}\cdots\frac{|\Omega_k(G_2)|}{|\Omega_k(G_1)|}k^{n-1}(k-1).$$

This is the idea behind the scheme, then. Given a graph G, we remove all but one of its edges. Then we restore the edges of G one-by-one, allowing us to estimate $\frac{|\Omega_k(G_i)|}{|\Omega_k(G_{i-1})|}$ for each $i \geq 2$ by sampling. Simple arithmetic then gives us an estimation for $|\Omega_k(G)|$. This is described more formally in the following section.

6.2 An Approximation Scheme for $|\Omega_k(G)|$

We suppose that $k \geq 2\Delta(G)+1$. Now $\chi(G) \leq \Delta(G)+1$ since having $\Delta(G)+1$ colours ensures that no matter what colours appear in the neighbourhood of a vertex $x \in V(G)$, there is always a colour available to assign to x. Thus, there will always be a k-colouring of G.

This immediately suggests that we can get a k-colouring of any G_r in a greedy fashion. One can consider each vertex v in turn and assign it the lowest-valued colour c such that c has not appeared in v's neighbourhood. It is easy to see that one needs at most $\Delta(G)+1$ colours to do this. The result will still be a k-colouring of the graph, though.

The following randomized algorithm, when given graph G and integer k as inputs, estimates the number of k-colourings of G. The edge set $E(G)$ of G is taken to be $\{e_1, e_2, \ldots, e_m\}$. The set of colours is taken to be C.

1. Let $G_m := G$ and inductively define $G_{i-1} := G_i - e_i$ for all $1 \leq i \leq m$. Let $r = 1$ and start with G_r.

2. Let x be an endpoint of e_1. For all the other $n-1$ vertices of G, assign them a colour uniformly at random. For x choose a colour uniformly at random from the $k-1$ colours not assigned to its neighbour. Let f be the k-colouring so created.

3. For each G_r with $r \in \{1, \ldots, m-1\}$, estimate $\frac{|\Omega_k(G_{r+1})|}{|\Omega_k(G_r)|}$ by generating λ k-colourings of G_r as follows:

   (a) Obtain a $(\Delta(G)+1)$-colouring of G_r in a greedy fashion, as described above.

   (b) Select a vertex $v \in V(G)$ uniformly at random.

   (c) Select a colour $c \in C$ uniformly at random.

   (d) If $f(y) \neq c$ for all $y \in N_{G_r}(v)$ then let $f(v) := c$, otherwise let $f(v)$ remain unchanged.

   (e) Do this for τ steps and take the final f as a sampled k-colouring of G_r. Take note as to whether f is a genuine colouring of G_{r+1} or not.

   (f) Repeat this entire procedure from (b) until ω k-colourings have been generated. For each step repetition, start with the same initial f. If a is the number of generated k-colourings of G_r that are genuine colourings of G_{r+1} as well, then $\frac{|\Omega_k(G_{r+1})|}{|\Omega_k(G_r)|} \approx \frac{a}{\tau}$.

4. Increment r by 1 and repeat for the new G_r until the case of $r = m-1$ has been dealt with.

5. Output the estimate of $\frac{|\Omega_k(G_m)|}{|\Omega_{m-1}|} \cdots \frac{|\Omega_k(G_2)|}{\Omega_k(G_1)|} k^{n-1}(k-1)$.

Now, as for the the problem of estimating the permanent, we seek to show that the colouring generator is efficient. To do this, we will again make use of Markov chains. The approach will be slightly different to the first problem, though. Having done this, we can then show that the entire sampling procedure is efficient and reasonably accurate. It is to the first task that we now turn.

Chapter 7

Modelling Colouring Generation

In this chapter we seek to show that the method used to sample from the k-colourings of a graph - namely the procedure used to generate new colourings - is an efficient, almost-uniform sampler. In particular, Jerrum's argument to this effect will be presented. His argument does attempt to do this directly, as for the proof that the matching generator of the first part was an efficient, almost-uniform sampler. Rather, Jerrum makes use of a simple, yet powerful, result: the Coupling Lemma. The first part of this chapter is largely devoted to the theoretical background behind the Coupling Lemma. Once this result has been introduced, we cam consider Jerrum's argument proper. To begin with though, we satisfy ourselves that it is appropriate to use Markov chains to model the colouring generator, and show that the Fundamental Theorem of Markov Chains applies.

Our colouring generator starts off with a "seed" k-colouring f and then proceeds to make a series of random changes to it. This corresponds to a sequence of colourings, each differing from its predecessor in at most one vertex. The algorithm has been written in such a way that the probability that a colouring f' appears at a particular point in the sequence depends only upon the previous colouring f. f' cannot follow f if f' differs from f in more than one vertex, say. Any colourings that appear prior to f will have no effect on the probability of f' appearing at that point. It seems as if Markov chains are a suitable way to model this generator, but we need to check that the generator fulfills the necessary requirements.

Recall Definition 2.1 of a Markov chain. Applying this definition to the current case, we can clearly take the set S of possible states to be $\Omega_k(G_r)$. It should be equally clear that we can treat the sequence of colourings that the generator goes through as a sequence of random variables. These variables take their values from $S = \Omega_k(G_r)$. Given the memoryless nature of the generator, we can take the probability function P on $S \times S$ to be defined by $P((s_i, s_j)) = \mathbf{Pr}(X_{t+1} = s_j | X_t = s_i)$. Our generator therefore fulfills the requirements to be modelled by a Markov chain. Let $\mathcal{M}$ be this chain. We assign each k-colouring in S a unique number from $[|S|]$ and identify each colouring by its number.

It turns out we already know enough about the generator to draw some conclusions about $\mathcal{M}$. The next two propositions are simple results noted by Jerrum [4].

Proposition 7.1. $\mathcal{M}$ is irreducible.

Proof. Let $\mathcal{Q}$ be the directed multigraph that underlies $\mathcal{M}$. So $V(\mathcal{Q}) = \Omega(G_r)$ for a running of the generator on G_r. $\overrightarrow{ff'} \in E(\mathcal{Q})$ if and only if f and f' disagree on exactly one vertex of G_r.

Let $f, f' \in \Omega(G_r)$ be arbitrary and let $v_1, \ldots, v_n$ be an ordering of the vertices of G_r. We want to manipulate f one vertex at a time to resemble f'. This will show that there is a path from f to f' in $\mathcal{Q}$. For increasing i consider $f(v_i)$ and $f'(v_i)$. If $f(v_i) = f'(v_i)$ then pass to the next i. If not, we may suppose that $f(v_h)$ has been altered to $f'(v_h)$ for all $h < i$. Consider each $v_j \in N_{G_r}(v_i)$ where $j > i$. If $f(v_j) = f'(v_i)$ then select a $c \in C$ such that $c \neq f'(v_i)$ and c does not colour any vertex in $N_{G_r}(v_j)$. Since $k \geq 2\Delta(G)+1$ there must be such a c. Then let $f(v_j) := c$. Once this has been done then $f(v_j) \neq f'(v_i)$ for all $j > i$. By assumption, $f(v_h) = f'(v_h) \neq f'(v_i)$ for $h < i$, as f' is a genuine colouring of G_r. Hence no neighbour of v_i has colour $f'(v_i)$. So we may let $f(v_i) := f'(v_i)$ and continue with the next i. This will eventually alter f to be f'. Hence there is a directed path from f to f' in $\mathcal{Q}$.

Therefore, $\mathcal{Q}$ is strongly connected and $\mathcal{M}$ is irreducible. □

Proposition 7.2. $\mathcal{M}$ is aperiodic.

Proof. To see this, we need only note that for each $f \in \Omega_k(G_r)$ there is a non-zero probability that the generator will select both a vertex and the colour it currently has at random. But that means it is possible for any f to be immediately repeated within the chain $\mathcal{M}$, and thus $\mathcal{M}$ is aperiodic. □

Theorem 3.1 stated that a finite, irreducible and aperiodic Markov chain was ergodic and has a unique stationary distribution. Propositions 7.1 and 7.2 now immediately imply that $\mathcal{M}$ has a stationary distribution π.

Let $\mathbf{P}$ be the transition probability matrix for $\mathcal{M}$. Recall that its rows each add to 1. Suppose the generator can make a transition from colouring f to colouring f'. A move from f to f' involves giving a random vertex a random colour. But that means that the generator can return to f by changing the colour of that vertex back. Both these transitions will have the same probability. So $\sum_{i=1}^{|S|} p_{i,j} = \sum_{j=1}^{|S|} p_{i,j} = 1$. So if we take $|S|$-dimensional vector $\mathbf{v} := \left(\frac{1}{|S|}, \ldots, \frac{1}{|S|}\right)$ then for each v_i

$$v_i\mathbf{P} = \sum_{i=1}^{|S|} \frac{1}{|S|} p_{i,j} = \frac{1}{|S|} \sum_{i=1}^{|S|} p_{i,j} = v_i.$$

We must conclude that $\pi = \mathbf{v}$. Notice that π is the uniform distribution on S, which will be of great use to us.

Once again, we must use a Markov chain analysis to show that the colouring generator has two properties. First, we must show that as the generator continues it gets closer to behaving like a uniform sampler. Second, we will need to show that the generator gets within a suitable error-range from the uniform distribution quickly. We did the same for the matching generator in the first problem. However, there we proved that the matching generator had these properties in a more-or-less direct fashion. Here we will use a so-called "coupling argument" to prove that the colouring generator has both these properties, in one fell swoop.

7.1 Coupling of Markov Chains

Given a Markov chain $\mathcal{M}$, a coupling of $\mathcal{M}$ is essentially two copies of $\mathcal{M}$ set off at the same time, usually from differing starting positions. How these copies behave with respect to each other will have implications for the properties of the generator of question. The following will formalize this. The theoretical background described here is largely drawn from the book by Michael Mitzenmacher and Eli Upfal [9].

Definition 7.1. Let $\mathcal{M}$ be a Markov chain on a state space S. A *coupling* of $\mathcal{M}$ is a Markov chain $\mathcal{Z} = (\mathcal{X}, \mathcal{Y})$ on a state space $S \times S$ with $\mathcal{X}$ and $\mathcal{Y}$ Markov chains such that

$$\begin{aligned}
\mathbf{Pr}(X_{t+1} = x_2 \mid Z = (x_1, y)) &= \mathbf{Pr}(M_{t+1} = x_2 \mid M_t = x_1) \\
\mathbf{Pr}(Y_{t+1} = y_2 \mid Z = (x, y_1)) &= \mathbf{Pr}(M_{t+1} = y_2 \mid M_t = y_1)
\end{aligned}$$

So a coupling $\mathcal{Z}$ of $\mathcal{M}$ is a pair of Markov chains $\mathcal{X}$ and $\mathcal{Y}$, both defined on $\mathcal{M}$'s state space and with transition probabilities as for $\mathcal{M}$.

Ultimately, we wish to use a coupling of $\mathcal{M}$ to show that $\mathcal{M}$ approaches a uniform probability distribution quickly. As in the first problem, this means that we will need some way to measure the distance between two probability distributions on S. Once we have stated the means of measurement, we will prove a useful equality for this distance.

Definition 7.2. Given two probability distribution vectors $\mathbf{u}, \mathbf{v}$ on S, we take the *total variation distance* between $\mathbf{u}$ and $\mathbf{v}$ to be

$$||\mathbf{u} - \mathbf{v}||_d := \frac{1}{2}\sum_{i \in S} |u_i - v_i|.$$

As the total variation distance is half the L_1-norm it is obviously a valid distance function. At this point, we also introduce some notation. We take u_A to be the probability, under probability distribution $\mathbf{u}$, that the current colouring f is in $A \subseteq S$. Then $u_A = \sum_{i \in A} u_i$.

It turns out that the distance between two distributions can be restated in terms of subsets of S. This lemma is found in [9].

Lemma 7.1. Let u, v be probability distributions on S. Then

$$||\mathbf{u} - \mathbf{v}||_d = \max_{A \subseteq S} |u_A - v_A|.$$

Proof. Let $S^+ := \{i \in S : u_i - v_i \geq 0\}$ and let $S^- := \{i \in S : u_i - v_i < 0\}$. Note that S^+ and S^- partition S.

Let $A \subseteq S$ and consider $\sum_{i \in A}(u_i - v_i)$. Consider each $i \in A$. If $i \in S^+$ then the summand $u_i - v_i \geq 0$. If $i \in S^-$ then the summand $u_i - v_i < 0$. Given this, it becomes clear that

$$\max_{A \subseteq S}(u_A - v_A) = \max_{A \subseteq S}\left[\sum_{i \in A}(u_i - v_i)\right] = \sum_{i \in S^+}(u_i - v_i) = u_{S^+} - v_{S^+}$$

and that similarly

$$\max_{A \subseteq S}(v_A - u_A) = v_{S^-} - u_{S^-}.$$

Now, $|u_{S^+} - v_{S^+}| = \left|\sum_{i \in S^+}(u_i - v_i)\right|$. But each summand is non-negative, so $\left|\sum_{i \in S^+}(u_i - v_i)\right| = \sum_{i \in S^+}|u_i - v_i|$. For similar reasons,

$$|v_{S^-} - u_{S^-}| = \sum_{i \in S^-}|v_i - u_i| = \sum_{i \in S^-}|u_i - v_i|.$$

Altogether then, we have that

$$\begin{aligned}
2\max_{A \subseteq S}|u_A - v_A| &= 2|u_{S^+} - v_{S^+}| \\
&= |u_{S^+} - v_{S^+}| + |v_{S^-} - u_{S^-}| \\
&= \sum_{i \in S}|u_i - v_i| \\
&= 2||\mathbf{u} - \mathbf{v}||_d
\end{aligned}$$

which is sufficient to show the result. □

In order for the coupling argument to proceed, we need some concepts that link the running time of a Markov chain to the distance function above. Let

$$Q := \{\mathbf{q} \in \mathbb{R}^{|S|} : q_i = 1 \text{ for some unique } i \text{ and } q_j = 0 \text{ for all } j \neq i\}.$$

So Q is the set of all probability distribution vectors that correspond to $\mathcal{M}$ starting off at a definite state $i \in S$. Let $\mathbf{q}_k$ denote the vector $\mathbf{q} \in Q$ that corresponds to $\mathcal{M}$ starting at state k. If $\mathcal{M}$ begins at state k then the probability distribution after t steps will be $\mathbf{q}_k\mathbf{P}^t$. Now we can state the following definitions, which explicitly link running time to distribution distance.

Definition 7.3. Let $\mathcal{M}$ start at state k and run for t steps - so having probability distribution vector $\mathbf{q}_k\mathbf{P}^t$. Denote by $\Delta_k(t)$ the distance between this vector and the stationary distribution. Thus, $\Delta_k(t) = ||\mathbf{q}_k\mathbf{P}^t - \pi||_d$.

Definition 7.4. Let $\Delta(t) := \max_{k\in S}\Delta_k(t)$ be the greatest distance after time t between the stationary distribution and all distributions that start at a definite state.

Definition 7.5. Let $\epsilon > 0$ and let $\phi_k(\epsilon) := \min\{t : \Delta_k(t) \leq \epsilon\}$ record the first step where the distance between $\mathbf{q}_k\mathbf{P}^t$ and π becomes less than or equal to ϵ.

Definition 7.6. The *mixing time* of $\mathcal{M}$ is taken to be $\phi(\epsilon) := \max_{k\in S}\phi_k(\epsilon)$

The type of Markov chain we are interested in are those that approach the uniform distribution. It will turn out that $\mathcal{M}$ will be such a chain, but this will be proved below. For $\mathcal{M}$ then, $\phi(\epsilon)$ records the step by when, no matter which definite state k we start $\mathcal{M}$ at, the probability distribution vector must have got close to π. Hence the term "mixing time".

Definition 7.7. If $\phi(\epsilon)$ for $\mathcal{M}$ is a polynomial in $\log_2\frac{1}{\epsilon}$ and the size of the input problem, then $\mathcal{M}$ is said to be *rapidly mixing*.

Thus a rapidly mixing Markov chain has both the properties we are looking for in the present case: we can get it arbitrarily close to the uniform (stationary) distribution relatively quickly.

We now state and prove the major result of this section. This will drive our use of couplings to analyze $\mathcal{M}$'s behaviour.

Lemma 7.2. (The Coupling Lemma) Suppose that Markov chain $\mathcal{M}$ has a coupling $\mathcal{Z} = (\mathcal{X}, \mathcal{Y})$. If X_t in $\mathcal{X}$ and Y_t in $\mathcal{Y}$ have probability distribution vectors $\mathbf{u}$ and $\mathbf{v}$ respectively, then

$$||\mathbf{u} - \mathbf{v}||_d \leq \mathbf{Pr}(X_t \neq Y_t).$$

Proof. [1]Let $\mathbf{q} \in \mathbb{R}^{|S|^2}$ be the probability distribution vector for $\mathcal{Z}$ at time t. Note that each entry of $\mathbf{q}$ will then be non-negative. Let $D \subseteq S \times S$ be the set $\{(i,i) : i \in S\}$, with D^c its complement in $S \times S$. Then $\mathbf{Pr}(X_t \neq Y_t) = q_{D^c}$. Also, $u_i = q_{\{i\}\times S}$ and $v_i = q_{S\times\{i\}}$ for all $i \in S$.

Then, for any $A \subseteq S$,

$$\begin{aligned}
||\mathbf{u} - \mathbf{v}||_d &= \frac{1}{2}\sum_{i\in S}|u_i - v_i| \\
&= \frac{1}{2}\sum_{i\in S}\left|q_{\{i\}\times\{i\}} + q_{(\{i\}\times S)\cap D^c} - \left(q_{\{i\}\times\{i\}} + q_{(S\times\{i\})\cap D^c}\right)\right| \\
&\leq \frac{1}{2}\left(\sum_{i\in S}|q_{(\{i\}\times S)\cap D^c}| + \sum_{i\in S}|q_{(S\times\{i\})\cap D^c}|\right) \\
&\qquad \text{[by the triangle inequality]} \\
&= \frac{1}{2}(q_{D^c} + q_{D^c})
\end{aligned}$$

We may take the result as proven since $\mathbf{Pr}(X_t \neq Y_t) = q_{D^c}$. □

This may seem like a trivial result, but it has one important aspect: it allows us to link the total variation distance to the running time of a Markov chain. The Coupling Lemma essentially says that the total variation distance between the probability distributions of $\mathcal{X}$ and $\mathcal{Y}$ at time t is bounded above by the probability that $\mathcal{X}$ and $\mathcal{Y}$ have not met at time t. The full power of this result only becomes clear when we consider the following theoretical situation.

For the next result, we consider a special sort of coupling. Let $\mathcal{Z} = (\mathcal{X}, \mathcal{Y})$ be a coupling of Markov chain $\mathcal{M}$. We use $\mathcal{Z}$ to define a new coupling $\mathcal{Z}' = (\mathcal{X}, \mathcal{Y}')$ where $Y_0' = Y_0$ and for all $t > 0$,

$$Y_t' := \begin{cases} Y_t, & \text{if } Y_{t-1}' \neq X_{t-1}\ ; \\ X_t, & \text{if } Y_{t-1}' = X_{t-1}. \end{cases}$$

Hence $\mathcal{Y}'$ is so defined that it follows $\mathcal{Y}$ until $\mathcal{X}$ and $\mathcal{Y}$ agree, after which point it follows $\mathcal{X}$. Call such a coupling a *tandem coupling.* For a coupling $\mathcal{Z}$, the time $t_s := \min\{t : X_t = Y_t\}$ when $\mathcal{X}$ and $\mathcal{Y}$ first meet is called $\mathcal{Z}$'s *stopping time.* Theorem 7.1 is then an easy corollary to the Coupling Lemma.

Theorem 7.1. Let coupling $\mathcal{Z} = (\mathcal{X}, \mathcal{Y}')$ of Markov chain $\mathcal{M}$ be a tandem coupling. Suppose that X_0 is chosen uniformly at random and $Y_0 = \mathbf{q}$ for some $\mathbf{q} \in Q$. If t_s is $\mathcal{Z}$'s stopping time then for all t,

$$||\mathbf{q}\mathbf{P}^t - \pi||_d \leq \mathbf{Pr}(t_s > t).$$

[1]This proof is taken from a lecture given by Dr. Nikolaos Fountoulakis.

Proof. Clearly for any t, $\mathbf{Pr}(X_t \neq Y_t) = \mathbf{Pr}(t_s > t)$. Also, X_t has probability distribution π for all t. Using Lemma 7.2 we get

$$||\mathbf{q}\mathbf{P}^t - \pi||_d \leq \mathbf{Pr}(X_t \neq Y_t) = \mathbf{Pr}(t_s > t).$$

□

Notice that $\mathcal{X}$ starts off in the same way that $\mathcal{M}$ does for a colouring generator. During its run, it behaves in accordance with how $\mathcal{M}$ would behave in the same situation - it obeys the same transition probabilities. What is helpful about this theorem is that it enables us to examine the behaviour of the Markov chain by examining the behaviour of the algorithm it models, not by examining the chain itself. This is sometimes an easier task. Also, suppose we create a tandem coupling of $\mathcal{M}$ such that $\mathbf{Pr}(t_s > t)$ gets small quickly for increasing t. Then, by Theorem 7.1, $\mathcal{M}$ itself must tend to the uniform distribution, and it must do so quickly. It is to such a task that we now turn. In so doing we can begin to examine Jerrum's argument.

7.2 A Coupling for $\mathcal{M}$

Given Markov chain $\mathcal{M}$, the most obvious coupling $\mathcal{Z} = (\mathcal{X}, \mathcal{Y})$ of $\mathcal{M}$ is one where $\mathcal{X}$ and $\mathcal{Y}$ act independently of one another. That is, at time t, $\mathcal{X}$ and $\mathcal{Y}$ each make a random decision for X_{t+1} and Y_{t+1} respectively, based on $\mathcal{M}$'s transition probabilities. Hence, one step of $\mathcal{Z}$ will involve two random decisions. However, suppose we set up $\mathcal{Z}$ so that the transitions of $\mathcal{X}$ and $\mathcal{Y}$ at each step are based on a *single* random decision. Indeed, suppose further that $\mathcal{Z}$ is created in such a way that once $\mathcal{X}$ and $\mathcal{Y}$ meet, they continue together. In other words, $\mathcal{Z}$ behaves like a tandem coupling. We may "rig" such a $\mathcal{Z}$ so that, far from $\mathcal{X}$ and $\mathcal{Y}$ behaving independently of one another, they will tend towards each other.

How is this to be done? The general idea is to set up $\mathcal{Z}$ so that once $\mathcal{X}$ and $\mathcal{Y}$ agree on the colour of a vertex, it becomes difficult for them to disagree on it later.

7.2.1 The Coupling Defined

A First Attempt

Suppose we have $\mathcal{Z} = (\mathcal{X}, \mathcal{Y})$ where X_0 is chosen from all k-colourings uniformly at random, and where Y_0 is always some definite k-colouring of G. So $Y_0 = \mathbf{q}$ for some $\mathbf{q} \in \mathcal{Q}$. Obviously this set-up is a purely theoretical construct. It would be extremely difficult to choose a random colouring for X_0 in real life. What this coupling will be used for is to show that the value of Y_t becomes more and more random for increasing t.

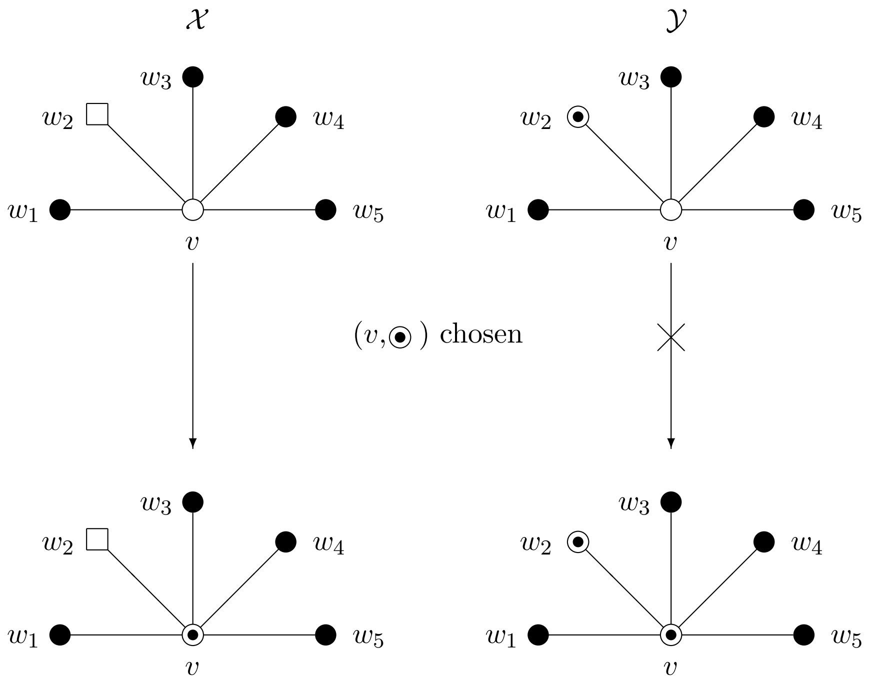

Figure 7.1: A problematic situation

Considering Theorem 7.1, we can do this if we manage to produce a situation in which $\mathcal{Y}$ approaches $\mathcal{X}$ quickly. So the idea will be to 'nudge' the colourings X_t and Y_t closer together for increasing t, but not so as to violate $\mathcal{Z}$'s being a coupling. The basic idea behind this approach is simple: instead of letting $\mathcal{X}$ and $\mathcal{Y}$ behave independently, we base the behaviour of $\mathcal{X}$ and $\mathcal{Y}$ on a single random decision, as mentioned above.

Suppose at each step t we choose a vertex $v \in V(G)$ and a colour $c \in C$. If c does not appear in the neighbourhood of v under X_t then assign c to v in X_{t+1}. Similarly, if assigning c to v in Y_{t+1} results in a genuine colouring, do so. This appears to give a procedure whereby $\mathcal{X}$ and $\mathcal{Y}$ may gradually approach each other. After all, the procedure is attempting to give each chosen vertex the same colour in X_{t+1} and Y_{t+1}. After some thought, a potential problem becomes apparent. Suppose the chosen colour c appears in v's neighbourhood under X_t, but not under Y_t (or *vice versa*)? v would be given colour c in Y_{t+1} but not in X_{t+1}. If X_t and Y_t already disagreed on v, then nothing is lost. X_{t+1} and Y_{t+1} will disagree on the same vertex. However, if X_t and Y_t *agreed* on v, then X_{t+1} and Y_{t+1} will disagree on v, and $\mathcal{X}$ and $\mathcal{Y}$ will diverge slightly (see Figure 7.1).

It is interesting to investigate whether this is a major problem. After all, if the problem outlined above is a rare event, then maybe $\mathcal{X}$ and $\mathcal{Y}$ still converge, and do so quickly. To investigate further, the following sets and parameters will be useful.

We take $A_t := \{v \in V(G_r) : X_t(v) = Y_t(v)\}$ to be the set of all vertices of G_r where $\mathcal{X}$ and $\mathcal{Y}$ agree at step t. $D_t := V(G_r) \backslash A_t$ is the set of all vertices where $\mathcal{X}$ and $\mathcal{Y}$ disagree at step t. Also, we let $m_t := |\{ad \in E(G_r) : a \in A_t \text{ and } d \in D_t\}|$ be the number of edges of G_r that join A_t to D_t. Finally, we let $d'_t(v) := |\{w \in N_{G_r}(v) : w \in D_t\}|$ if $v \in A_t$, and $d'_t(v) := |\{w \in N_{G_r}(v) : w \in A_t\}|$ if $v \in D_t$. So $d'_t(v)$ is the number of edges joined to v that have one endpoint in A_t and the other in D_t. Note that

$$\sum_{v \in A_t} d'_t(v) = m_t = \sum_{v \in D_t} d'_t(v).$$

Now, the colour of at most one vertex is altered at every step of the coupling, so $|D_{t+1}| = |D_t| - 1, |D_t|$, or $|D_t| + 1$. The following bounds on the likely outcome are easily derived.

Proposition 7.3. For $\mathcal{Z}$ as outlined, $\mathbf{Pr}(|D_{t+1}| = |D_t| + 1) \leq \frac{2m_t}{nk}$

Proof. Suppose v, c were the vertex and colour, respectively, chosen for step t. For it to be the case that $|D_{t+1}| = |D_t| + 1$, $X_t(v) = Y_t(v)$ and $X_{t+1}(v) \neq Y_{t+1}(v)$. That is, $v \in A_t$ and $v \in D_{t+1}$. Also v changed colour only in one of X_{t+1}, Y_{t+1}. This means that there is a $w \in N_{G_r}(v)$ such that $X_t(w) = c$ or $Y_t(w) = c$, but not both. Clearly, there are at most $2d'_t(v)$ such colours c.

So, out of all the nk vertex-colour combinations (v, c), there will be at most $\sum_{v \in A_t} 2d'_t(v)$ such combinations that lead to an increase in the number of disagreeing vertices.

Thus, $\mathbf{Pr}(|D_{t+1}| = |D_t| + 1) \leq \frac{1}{nk}\left(2\sum_{v \in A_t} d'_t(v)\right) = \frac{2m_t}{nk}$. □

Proposition 7.4. For $\mathcal{Z}$, $\mathbf{Pr}(|D_{t+1}| = |D_t| - 1) \geq \frac{k - 2\Delta(G)}{nk}|D_t| + \frac{m_t}{nk}$ if $|D_t| > 0$.

Proof. Let v be the vertex and c the colour chosen for step t. For $|D_{t+1}| = |D_t| - 1$, it must be that $v \in D_t$ and $v \in A_{t+1}$. So the procedure was able to colour v with c in both X_{t+1} and Y_{t+1}. Hence for all $w \in N_{G_r}(v)$, $X_t(w) \neq c$ and $Y_t(w) \neq c$. There are at most $\Delta(G)$ colours in $N_{G_r}(v)$ under X_t, similarly for $Y_t(v)$. So c must be one of at least $k - 2\Delta(G) + d'_t(v)$ colours, where the $d'_t(v)$ term is to ensure that colours appearing in both colourings of $N_{G_r}(v)$ are not counted (ie. taken away) twice.

Therefore,

$$\mathbf{Pr}(|D_{t+1}| = |D_t| - 1) \leq \frac{1}{nk} \sum_{v \in D_t} (k - 2\Delta(G) + d'_t(v) = \frac{k - 2\Delta(G)}{nk}|D_t| + \frac{m_t}{nk}.$$

□

At step t, D_t is expected to gain $1 \times \mathbf{Pr}(|D_{t+1}| = |D_t|+1) \leq \frac{2m_t}{nk}$ vertices [by Proposition 7.3]. Also, it is expected to have a 'negative increment' of $-1 \times \mathbf{Pr}(|D_{t+1}| = |D_t|-1) \leq -\left(\frac{k-2\Delta(G)}{nk}|D_t| + \frac{m_t}{nk}\right)$ vertices [by Proposition 7.4]. So the following Proposition is easily derived. A similar result is found in [9].

Proposition 7.5. $\mathbb{E}(|D_{t+1}|) \leq \left(1 - \frac{k-2\Delta(G_r)}{nk}\right)|D_t| + \frac{m_t}{nk}$.

Proof. Suppose $|D_t| \neq 0$ or n. Using the expected gains and losses for D_t derived from Propositions 7.3 and 7.4,

$$\mathbb{E}(|D_{t+1}|) \leq |D_t| + \frac{2m_t}{nk} - \left(\frac{k-2\Delta(G)}{nk}|D_t| + \frac{m_t}{nk}\right).$$

After re-arranging and cancelling, the result follows easily.

If $|D_t| = 0$ then D_t can only increase in size. Thus, since $|D_t| = m_t = 0$

$$\mathbb{E}(|D_{t+1}|) \leq |D_t| + \frac{2m_t}{nk} \leq \left(1 - \frac{k-2\Delta(G)}{nk}\right)|D_t| + \frac{m_t}{nk}.$$

Also, if $|D_t| = n$ then $D_t = V(G_r)$ and D_t can only decrease. Hence,

$$\mathbb{E}(|D_{t+1}|) \leq |D_t| - \left(\frac{k-2\Delta(G)}{nk}|D_t| + \frac{m_t}{nk}\right) \leq \left(1 - \frac{k-2\Delta(G)}{nk}\right)|D_t| + \frac{m_t}{nk}.$$

□

The situation does not look promising. It would be preferable to have $\mathbb{E}(|D_{t+1}|) = \alpha|D_t|$ for some $0 < \alpha < 1$, so that D_t will tend to shrink as t increases. However, this is not so. It is hard to determine the value of the summand $\frac{m_t}{nk}$, and so arguably it is hard to determine whether D_t will shrink.

One can see that the procedure gave too large a bound to $\mathbf{Pr}(|D_{t+1}| = |D_t| + 1)$. This introduced the 'rogue summand' in the expression for $\mathbb{E}(|D_{t+1}|)$. There is little hope of a more sophisticated analysis improving the bound. After all, if for all $v \in A_t$ the disagreeing neighbours of v use $2d'_t(v)$ colours in both colourings, then the bound will be tight. The procedure as it stands thus gives $v \in A_t$ too much scope to join D_{t+1} in the next step. So the issue that was initially singled out as a potential problem did turn out to be a major problem. We must therefore aim to improve the procedure. We will therefore leave this line of inquiry and attempt a more sophisticated approach that should give us a less unwieldy bound on $\mathbb{E}(|D_{t+1}|)$. This will be the approach of Jerrum to the problem [4].

Reconsidering the Approach

When regarding what went wrong with the first attempt at a coupling, it becomes obvious that we must make our procedure select candidate colours more intelligently, but just as randomly. If the chosen $v \in D_t$, then the procedure may continue as initially defined. This did not cause a problem in the eventual bound of $\mathbb{E}(|D_t|)$. It is when $v \in A_t$ that we must be more careful.

Suppose we have $v \in A_t$. Consider the neighbourhood of v. Also, consider the colours that X_t and Y_t use to colour this neighbourhood. Now, there could well be colours that X_t uses to colour $N_{G_r}(v)$ but that Y_t does not use in $N_{G_r}(v)$, and *vice versa*. We let

$$C_{X_t} := \{c \in C : X_t(u) = c \text{ for some } u \in N_{G_r}(v) \text{ and } \forall w \in N_{G_r}(v), Y_t(w) \neq c\}$$

be the set of all such colours for X_t. We let C_{Y_t} be defined similarly.

The failure of the initial procedure was due to the cases when $v \in A_t$, and $c \in C_{X_t}$ or $c \in C_{Y_t}$. What we would like to do is set things up so that when the procedure attempts colour v in X_{t+1} with a colour $c \in C_{X_t}$, then the procedure attempts to colour v in Y_{t+1} with a colour $c' \in C_{Y_t}$. Thus v's colour will not change in either X_{t+1} and Y_{t+1}, and so v remains in A_{t+1}. This is accomplished by using the following "swapping" function.

Suppose at step t the coupling has chosen vertex $v \in A_t$ to re-colour. This automatically defines the sets C_{X_t}, C_{Y_t} for this step. Now either $|C_{X_t}| \geq |C_{Y_t}|$ or $|C_{X_t}| < |C_{Y_t}|$. Suppose, without loss of generality, that $|C_{X_t}| < |C_{Y_t}|$. The other case will be similar. Take a $C'_{Y_t} \subseteq C_{Y_t}$ such that $|C'_{Y_t}| = |C_{X_t}|$. But then we can map each $c_1 \in C_{X_t}$ to a unique $c_2 \in C_{Y_t}$. That is, we can produce a bijection between C_{X_t} and C'_{Y_t}.

Now suppose the coupling chose colour c for step t. If $c \notin C_{X_t} \cup C'_{Y_t}$ then we set $c' := c$ and attempt to colour v with c in X_{t+1} and with c' in Y_{t+1}. If $c \in C_{X_t} \cup C'_{Y_t}$ however, we take c' to be the colour that maps to c. More precisely, if $c \in C_{X_t}$ then c' is the colour that c maps to in C'_{Y_t}. Also, if $c \in C'_{Y_t}$ then c' is the colour in C_{X_t} that maps to c. Once again, the procedure will then attempt to colour v with c in X_{t+1} and with c' in Y_{t+1}.

So how does this revised procedure work in practice? If we consider the example shown in Figure 7.2 things may become a little clearer. The coupling has reached step t and has chosen vertex v to re-colour with colour 3. Note that the colourings currently agree on v. Now $C_{X_t} = \{2, 3, 4\}$, $C_{Y_t} = \{6, 7, 8, 9\}$ and the procedure has chosen the correspondences $2 \leftrightarrow 6, 3 \leftrightarrow 7, 4 \leftrightarrow 8$. So c remains 3 but c' is set as 7. But v cannot be coloured with 3 in X_{t+1} and with 7 in Y_{t+1} as these colours already appear in the respective colourings of the neighbourhood. So the colourings remain in agreement on v in step $t + 1$. Note that under the old procedure, v would have remained the same colour in X_{t+1}, but would have been re-coloured in Y_{t+1}.

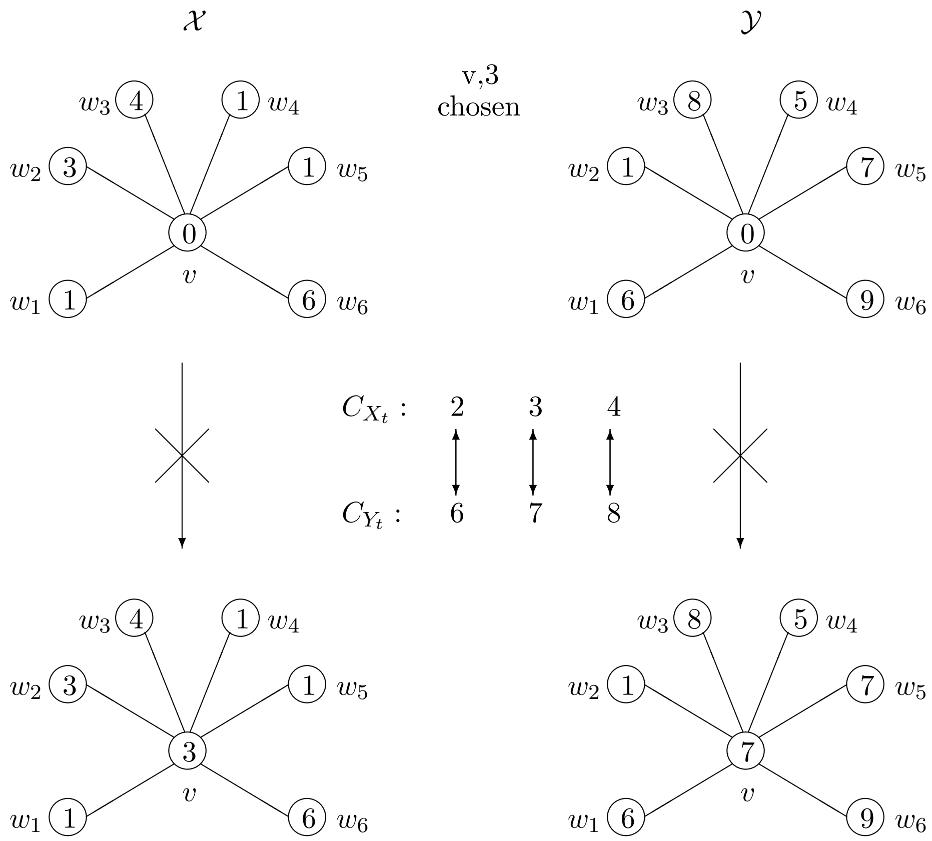

Figure 7.2: The revised coupling

More generally, it is easily seen that if the chosen colour c is in C_{X_t} and there is a corresponding c' in C_{Y_t}, then v will remain the same colour, and so $|D_{t+1}| = |D_t|$. However, things may still go "wrong", even with this new procedure. To see why this is so, we consider all the possibilities for c. Without loss of generality, assume that $|C_{X_t}| \leq |C_{Y_t}|$. The other case will be similar. Now $C_{X_t}, C'_{Y_t}, C_{Y_t} \backslash C'_{Y_t}$ and $C \backslash (C_{X_t} \cup C_{Y_t})$ partition C. The possible cases will then be:

1. $c \in C_{X_t}$:

   This has already been considered. In this case $v \in A_{t+1}$.

2. $c \in C \backslash (C_{X_t} \cup C_{Y_t})$:

   Then c either appears, or does not appear, in both colourings of $N_{G_r}(v)$. In both cases, $v \in A_{t+1}$.

3. $c \in C'_{Y_t}$:

   Then c' is taken to be the colour in C_{X_t} that maps to c. The procedure will then attempt to colour v with c in X_{t+1}. Since $c \in C_{Y_t}$, this can be done. Similarly, v can be coloured with $c' \in C_{X_t}$ in Y_{t+1}. But $c \neq c'$, so $v \in D_{t+1}$.

4. $c \in C_{Y_t} \backslash C'_{Y_t}$:

   Then there is no corresponding $c' \in C_{X_t}$. Thus the procedure sets $c' := c$. But v can be coloured with c in X_{t+1}, but not in Y_{t+1}. Hence, $v \in D_{t+1}$.

So the revised procedure can still produce the undesired behaviour, but only if $c \in C_{Y_t}$. Note that if $|C_{X_t}| > |C_{Y_t}|$ then the undesired behaviour will arise if $c \in C_{X_t} \backslash C'_{X_t} \cup C_{Y_t}$.

Suppose $|C_{X_t}| \leq |C_{Y_t}|$. It is easy to see that if $c \in C_{Y_t}$ then any vertex $w \in N_{G_r}(v)$ coloured with c must be in D_t, by definition of C_{Y_t}. For every $c \in C_{Y_t}$ there must be at least one such w, so $|C_{Y_t}| \leq d'_t(v)$.

Suppose now that $|C_{X_t}| > |C_{Y_t}|$. Similarly to the above, if $c \in C_{X_t} \backslash C'_{X_t} \cup C_{Y_t}$, then every $w \in N_{G_r}(v)$ coloured with c is in D_t, and for every such c there must be at least one such w. Therefore, as $|C'_{X_t}| = |C_{Y_t}|$,

$$|C_{X_t} \backslash C'_{X_t} \cup C_{Y_t}| = |C_{X_t} \backslash C'_{X_t}| + |C_{Y_t}| = |C_{X_t} \backslash C'_{X_t}| + |C'_{X_t}| = |C_{X_t}|.$$

As for $|C_{Y_t}|$, we have $|C_{X_t}| \leq d'_t(v)$.

We now have the requisite understanding of the revised procedure to improve the bound on $\mathbf{Pr}(|D_{t+1}| = |D_t| + 1)$ given in Proposition 7.3.

Proposition 7.6. Given the revised procedure, $\mathbf{Pr}(|D_{t+1}| = |D_t| + 1) \leq \frac{m_t}{nk}$.

Proof. As for Proposition 7.3, for it to be the case that $|D_{t+1}| = |D_t| + 1$, $v \in A_t$ and $v \in D_{t+1}$. But as just observed, the number of colours which lead to this happening is at most $\mathbf{max}\{|C_{X_t}|, |C_{Y_t}\}| \leq d'_t(v)$. So following the reasoning of Proposition 7.3, we have that

$$\mathbf{Pr}(|D_{t+1}| = |D_t| + 1) \leq \frac{1}{nk}\left(\sum_{v \in A_t} d'_t(v)\right) = \frac{m_t}{nk}.$$

□

The bound in Proposition 7.4 is unproblematic, so it is not revised here.

Proposition 7.7. Given that the number of disagreeing vertices at time t is $|D_t|$, $\mathbb{E}(|D_{t+1}|) \leq \left(1 - \frac{k-2\Delta(G)}{nk}\right)|D_t|$.

Proof. The proof of this proposition is structurally identical to the proof of Proposition 7.5. However, Proposition 7.6 lessens the bound of $\mathbf{Pr}(|D_{t+1}| = |D_t| + 1)$ in Proposition 7.3 by $\frac{m_t}{nk}$. This will give

$$\mathbb{E}(|D_{t+1}|) \leq \left(1 - \frac{k - 2\Delta(G)}{nk}\right)|D_t|,$$

the result we want. □

Given that $k \geq 2\Delta(G) + 1$ and $\frac{k-2\Delta(G)}{nk} > 0$, we expect $|D_t|$ to decrease with increasing t. We could immediately apply this result to find the mixing time of $\mathcal{M}$, but first it would be wise to allay a possible concern.

The concern is that $\mathcal{Z}$ under the revised procedure no longer behaves as a coupling. Definition 7.1 required that each chain $\mathcal{X}, \mathcal{Y}$ in $\mathcal{Z}$ behave as copies of $\mathcal{M}$ in their own right. In the simple coupling, when both $\mathcal{X}, \mathcal{Y}$ behaved independently of one another, this was fairly obvious. But for our present coupling, the behaviour of the chains $\mathcal{X}$ and $\mathcal{Y}$ is far from independent. However, if we examine exactly how c' was chosen at each step, then we will see that both $\mathcal{X}$ and $\mathcal{Y}$ have the desired properties.

Obviously, $\mathcal{X}$ acts as a faithful copy of $\mathcal{M}$. For any step, the vertex-colour pair (v, c) it attempts to use has always been chosen uniformly at random. With regard to $\mathcal{Y}$, suppose $\mathcal{Z}$ chooses the vertex-colour pair (v, c) at time t. Then the colour we attempt to use for Y_{t+1} will be a function $g(c)$ of c. If $v \in D_t$, then g is clearly the identity function; whereas if $v \in A_t$ then there may be a subset of colours which g attempts to interchange. In either case, g is a bijection. Since the original colour c was chosen uniformly at random, $g(c)$ must be distributed uniformly. So $\mathcal{Y}$ acts as a faithful copy of $\mathcal{M}$.

We may now derive the following Proposition. For notational ease we let $\alpha := \frac{k-2\Delta(G)}{nk}$.

Proposition 7.8. For given $\epsilon > 0$, the mixing time $\phi(\epsilon)$ of $\mathcal{M}$ is at most $\frac{1}{\alpha}\ln\left(\frac{n}{\epsilon}\right)$.

Proof. From repeated applications of Proposition 7.7 it is easy to see that if $\mathcal{Z}$ starts with $|D_0|$ vertices disagreeing then for arbitrary t, $\mathbb{E}(|D_t|) \leq (1-\alpha)^t|D_0|$. But whatever colouring we start with, $|D_0| \leq n$ so $\mathbb{E}(|D_t|) \leq (1-\alpha)^t n \leq ne^{-\alpha t}$.

If we have $t \geq \frac{1}{\alpha}\ln\left(\frac{n}{\epsilon}\right)$, we get

$$\mathbf{Pr}(|D_t| > 0) \leq E(|D_t|) \leq ne^{-\alpha t} \leq \epsilon.$$

If t_s is the stopping time for $\mathcal{Z}$ then it easily follows that $\mathbf{Pr}(t_s > t) = \mathbf{Pr}(|D_t| > 0)$, since once $\mathcal{X}$ and $\mathcal{Y}$ meet, they continue together. By Theorem 7.1,

$$||\mathbf{q}\mathbf{P}^t - \pi||_d \leq \mathbf{Pr}(t_s > t) \leq \epsilon.$$

But the k-colouring we started with was arbitrary. The result follows by a simple application of the definition of mixing time (Definition 7.6). $\square$

So given an acceptable degree of sampling bias $\epsilon > 0$, we need only run the colouring generator for $\tau = \left\lceil \frac{1}{\alpha}\ln(n) + \frac{1}{\alpha}\ln\left(\frac{1}{\epsilon}\right)\right\rceil$ steps. Hence the running time t is polynomial in $n = |G|$ and $\log_2\left(\frac{1}{\epsilon}\right)$, and so $\mathcal{M}$ is rapidly mixing. This is the desired result.

Chapter 8

An Improvement on the Mixing Rate for Dense Triangle-free Graphs

The preceding chapter showed that we can sample almost uniformly, and in polynomial time, from the k-colourings of a graph G_r, where $k \geq 2\Delta(G)+1$. This was done by examining the behaviour of a coupling of a Markov chain; in particular, the Markov chain that modelled the colouring generator. Once an appropriate procedure had been found to drive the movement of the coupling, the major result of the chapter followed fairly easily. The aim of this chapter is to present a result of Hayes and Vigoda in [3]: that if the initial input graph G is triangle-free then the bound given on the mixing rate in the last chapter can be improved. To do this, they alter the algorithm slightly and use a more sophisticated analysis of the behaviour of the coupling.

Hayes and Vigoda also use a coupling argument to establish their claim. In the following chapter, this argument will be presented in detail. Their general approach is to argue that, in a coupling yet to be defined, the distance between any random state X_t and set state Y_t in the coupling will tend to decrease in the next step. Bounds on this behaviour then give a mixing rate for the coupling. To begin with, we introduce the modified algorithm and consider a possible reason for the restriction to triangle-free graphs. Both will contribute towards an increase in speed.

8.0.2 An Increase in Speed

In Jerrum's algorithm just presented, at each step a colour is chosen uniformly at random from the set of all colours C. This sometimes means that the algorithm is unable to do anything. This tends to happen when the algorithm is unable to give a chosen vertex the chosen colour. What Hayes and Vigoda suggest is that, at each step, the algorithm chooses a vertex v.

It then chooses a colour uniformly at random from all the colours available to v; that is, all colours that do not appear in v's neighbourhood. One of course expects this to increase the speed of the algorithm as it will not attempt to generate "impossible" colourings.

As an important aside, note that if the current colour c of v is regarded as an available colour then the new algorithm's Markov chain will be aperiodic (see Proposition 7.2). Also, we can assume that k is large enough so that the chain is irreducible (the proof of this would be similar to that of Proposition 7.1). For similar reasons to the original colour- sampling algorithm, the stationary distribution of this new chain will be the uniform distribution π (see page 63). Let $\mathcal{M}$ be the Markov chain of this new coupling.

Intuitively, why should the restriction to triangle-free graphs also contribute towards an increase in the speed of the algorithm? In a triangle-free graph, the neighbourhood of a vertex v is an independent set. So when colouring $N(v)$, the only restrictions to colour of a vertex $w \in N(v)$ will come from outside $N(v)$. This means that it is more likely that some neighbours of v will share the same colour. This in turn implies that v will probably have a larger palette to be coloured from than if some if its neighbours were adjacent. This is termed the "local uniformity property" in the literature. Lemma 8.1 encapsulates this. This lemma examines the $\mathcal{X}$ chain in the $\mathcal{Z}$ coupling. It tries to show that the probability of there being a vertex with few colours available to it is relatively small. Note that we no longer assume that $k \geq 2\Delta(G) + 1$. Instead we may assume a smaller lower bound on k. Indeed, subsequent reasoning will lead to the less restrictive requirement that $k > 1.764\Delta(G)$ for a large $\Delta(G)$.

It should be mentioned as a matter of course that restricting our attention to a triangle-free graph G does not raise any issues with regard to sampling from subgraphs of G. If G is triangle-free then any subgraph of G used for sampling will be triangle-free as well. Hence any results proved for G will apply to them as well.

Before Hayes and Vigoda's coupling argument can be presented, however, a lemma needs to be introduced that formalizes the intuition just mentioned: that triangle-free graphs will tend to offer a vertex more colours to choose from.

Before this lemma can be stated and proved, however, we need to introduce some new ideas and a key result.

8.0.3 Martingales

This section can be seen as a build-up to the Azuma-Hoeffding inequality. This result will be used to show that, in a triangle-free graph, a vertex is likely to have many colours available to it. The Azuma-Hoeffding inequality is based on the concept of Martingales. These are introduced here. Many of the ideas of this section are drawn from [9].

Definition 8.1. Suppose $W_0, W_1, \dots$ and $Z_0, Z_1, \dots$ are sequences of random variables such that

1. Z_n if a function of $W_0, W_1, \dots, W_n$,
2. $\mathbb{E}(|Z_n|) < \infty$ for all n, and
3. $\mathbb{E}(Z_{n+1} \mid W_0, W_1, \dots, W_n) = Z_n$ for all n.

Then $Z_0, Z_1, \dots$ is called a *martingale* with respect to the sequence $W_0, W_1, \dots$. If a sequence is a martingale with respect to itself then it is simply called a *martingale*.

The third condition in Definition 8.1 can be seen as the "major defining characteristic" of martingales. In a martingale, one would expect the next value in the sequence to be equal to the current value.

Definition 8.2. Let $W_0, W_1, \dots, W_n$ be a sequence of random variables and let Y be a random variable dependent on $W_0, W_1, \dots, W_n$ with $\mathbb{E}(|Y|) < \infty$. A sequence of random variables $Z_0, Z_1, \dots, Z_n$ is a *Doob martingale* with respect to $W_0, W_1, \dots, W_n$ if, for all $0 \le i \le n$,

$$Z_i = \mathbb{E}(Y \mid W_0, W_1, \dots, W_i).$$

It should be mentioned here that if A, B are random variables then $\mathbb{E}(A \mid B)$ is itself a random variable. It acts as a function of B, taking the value $\mathbb{E}(Y \mid B = b)$ when $B = b$. The same holds when B is a random vector rather than a random variable, as in Definition 8.2.

Although Definition 8.2 labels a given sequence $Z_0, Z_1, \dots Z_n$ as a "Doob martingale", it is not immediately obvious that all such sequences are in fact martingales. The following Proposition holds true, but is stated without proof.

Proposition 8.1. If $Z_0, Z_1, \dots, Z_n$ is a Doob martingale with respect to $W_0, W_1, \dots, W_n$ then it is a martingale with respect to the same sequence.

This is all very well, but what sort of object is a Doob martingale? Essentially, a Doob martingale records how the expectation of a random variable Y changes as more is known about the set that affects Y's value. As the sequence continues, it is as if more and more details trickle in about the nature of the set, and each new detail may lead to a reassessment of the likely value of Y.

With a view towards setting up Lemma 8.1, we take $A(X, v) := [k] \backslash X(N(v))$ to be set set of all colours in the palette $[k]$ that v can be given in colouring X. So essentially $A(X, v)$ is the set of all colours that do not appear in v's neighbourhood. Let $W_1, \dots, W_k$ be the set of indicator random variables for

$A(X, v)$ on $C = [k]$. That is, each $W_i \in \{0, 1\}$ and $W_i = 1$ if and only if $i \in A(X, v)$. Note that $|A(X, v)| = \sum_{i=1}^{k} W_i$ is a random variable.

In the meantime, for ease of notation, we let $f(W_1, \ldots, W_k) := A(X, v)$. Suppose $W_1, \ldots, W_i', \ldots, W_k$ is new sequence created by "flipping" the value of W_i for some i. Then $f(W_1, \ldots, W_i', \ldots, W_k) = f(W_1, \ldots, W_k) \pm 1$. So $|f(W_1, \ldots, W_k) - f(W_1, \ldots, W_i', \ldots, W_k)| \leq 1$.

Define a new random sequence by

$$U_0 := \mathbb{E}(f(W_1, \ldots, W_k))$$

and

$$U_m := \mathbb{E}(f(W_1, \ldots, W_k) \mid W_1, \ldots, W_m) \text{ for all } 1 \leq m \leq k.$$

By Definition 8.2, the sequence $U_0, \ldots, U_k$ is a Doob martingale with respect to $W_1, \ldots, W_k$. What is interesting is that since $|f(W_1, \ldots, W_k) - f(W_1, \ldots, W_i', \ldots, W_k)|$ is bounded, $|U_i - U_{i-1}|$ must be bounded as well for all i. This is stated and proved more formally below (from [9]).

Proposition 8.2. Let $U_0, \ldots, U_k$ be defined as above. Then for all i, there is a random variable B_i, dependent on $U_0, \ldots, U_{i-1}$, such that $B_i \leq U_i - U_{i-1} \leq B_i + 1$.

Proof. Let $\overline{W} := (W_1, \ldots, W_k)$ and let $\overline{W}_i := (W_1, \ldots, W_i)$.

Then $U_i - U_{i-1} = \mathbb{E}(f(\overline{W}) \mid \overline{W}_i) - \mathbb{E}(f(\overline{W}) \mid \overline{W}_{i-1})$ So the largest possible value of $U_i - U_{i-1}$ is

$$\max_{x \in \{0,1\}} \mathbb{E}(f(\overline{W}) \mid \overline{W}_{i-1}, W_i = x) - \mathbb{E}(f(\overline{W}) \mid \overline{W}_{i-1}),$$

while the smallest possible value of the same will be

$$\min_{y \in \{0,1\}} \mathbb{E}(f(\overline{W}) \mid \overline{W}_{i-1}, W_i = y) - \mathbb{E}(f(\overline{W}) \mid \overline{W}_{i-1}).$$

The difference between these two bounds will of course be

$$\begin{aligned}
& \max_{x \in \{0,1\}} \mathbb{E}(f(\overline{W}) \mid \overline{W}_{i-1}, W_i = x) - \min_{y \in \{0,1\}} \mathbb{E}(f(\overline{W}) \mid \overline{W}_{i-1}, W_i = y) \\
= & \max_{x,y \in \{0,1\}} \big(\mathbb{E}(f(\overline{W}) \mid \overline{W}_{i-1}, W_i = x) - \mathbb{E}(f(\overline{W}) \mid \overline{W}_{i-1}, X_i = y)\big) \\
= & \max_{x,y \in \{0,1\}} \big(\mathbb{E}(f(W_1, \ldots, x, \ldots, W_k) - f(W_1, \ldots, y, \ldots, W_k) \mid \overline{W}_{i-1}\big)
\end{aligned}$$

where x and y are swapped for W_i in $f(W_1, \ldots, W_k)$. The last equality follows from the linearity of expectations.

Now, the choices of colours for the neighbours of v are independent, as G is triangle-free. Therefore, $W_1, \ldots, W_k$ is a sequence of independent random variables. Recall that the values of $W_i, \ldots, W_k$ do not depend on the preceding values $W_1, \ldots, W_{i-1}$. So fix $\overline{W}_{i-1} := \mathbf{w} \in \{0,1\}^{i-1}$. Then the values $W_i, \ldots, W_k$ will not be influenced by $\mathbf{w}$, and

$$\begin{aligned}
& (\mathbb{E}(f(W_1, \ldots, x, \ldots, W_k) - f(W_1, \ldots, y, \ldots, W_k) \mid \mathbf{w}) \\
= & \sum_{z_i, \ldots, z_k \in \{0,1\}} (\mathbf{Pr}(W_i = z_i, \ldots, W_k = z_k)(f(\mathbf{w}, x, \ldots, z_k) - f(\mathbf{w}, y, \ldots, z_k))) \\
\leq & \; 1 \times \sum_{z_i, \ldots, z_k \in \{0,1\}} (\mathbf{Pr}(W_i = z_i, \ldots, W_k = z_k)) \\
= & \; 1
\end{aligned}$$

Thus, the maximum possible value for $U_i - U_{i-1}$ is larger than the minimum possible value by at most 1. Setting

$$B_i := \min_{y \in \{0,1\}} \mathbb{E}(f(\overline{W}) \mid \overline{W}_{i-1}, X_i = y) - \mathbb{E}(f(\overline{W}) \mid \overline{W}_{i-1})$$

will therefore give the bounds on $U_i - U_{i-1}$ that we want. □

The key result that we will use is stated here without proof. It is taken from [9]. Its proof is similar to that of a Chernoff bound.

Theorem 8.1. (*Azuma-Hoeffding Inequality*) Let $Z_0, \ldots, Z_n$ be a martingale such that for all i there exist random variables B_i (which may be functions of $Z_0, \ldots, Z_{i-1}$) with

$$B_i \leq |Z_i - Z_{i-1}| \leq B_i + 1.$$

Then for all $k > 0$ and any $\lambda > 0$,

$$\mathbf{Pr}(|Z_k - Z_0| \geq \lambda) \leq 2e^{-2\lambda^2/2k}.$$

Note we now have the background necessary to state and prove Lemma 8.1.

8.0.4 The Number of Colour Options for X_t

The following proposition will be needed for Lemma 8.1. Lemma 8.1 itself is found in [3].

Proposition 8.3. If $0 < p \leq 1$ then $(1-p)^{\frac{1}{p}} \geq \frac{1-p}{e}$.

Proof. By assumption,

$$\begin{aligned} p &\geq p - \sum_{i=2}^{\infty} \frac{p^i}{i(i-1)} \\ &= p - p^2 + \frac{p^2 - p^3}{2} + \frac{p^3 - p^4}{3} + \frac{p^4 - p^5}{4} + \dots \\ &= (p-1) \sum_{i=1}^{\infty} \frac{-p^i}{i} \\ &= (p-1)\ln(1-p). \end{aligned}$$

Then

$$\begin{aligned} & p + (1-p)\ln(1-p) \geq 0 \\ \Rightarrow\ & e^p (1-p)^{(1-p)} \geq 1 \\ \Rightarrow\ & e^p \geq (1-p)^{(p-1)} \\ \Rightarrow\ & e \geq (1-p)^{1-\frac{1}{p}} \\ \Rightarrow\ & (1-p)^{\frac{1}{p}} \geq \frac{1-p}{e}. \end{aligned}$$

□

Lemma 8.1. If G is a triangle-free graph, $0 < \beta \leq 1$ and $k \geq \Delta(G) + \frac{2}{\beta}$, then a random k-colouring X of $V(G)$ will have

$$\mathbf{Pr}\left(\exists v \in V(G) : |A(X,v)| < k(e^{-\Delta(G)/k} - \beta)\right) \leq 2ne^{-\beta^2 k/8},$$

where $n = |G|$.

Proof. Fix any vertex v. Let $W_{j,w}$ be the random indicator variable for the event that vertex w is coloured with colour j. $W_{j,w} = 1$ if this is the case. Note that $|A(X,v)| = \sum_{j\in[k]} \prod_{w\in N(v)} (1 - W_{j,w})$, since $\prod_{w\in N(v)}(1 - W_{j,w})$ will contribute 1 to the entire sum if and only if j does not appear in the neighbourhood of v.

Now, encode by F all the information about the colouring outside $N(v)$. So F records the colour of each $x \notin N(v)$. Now recall that G is triangle-free, and so each $w \in N(v)$ may choose from all its available colours independently of the rest of $N(v)$. For each colour j then, each $w \in N(v)$ can decide independently to accept j as its colour (This is assuming the background information contained in F). So for each j,

$$\mathbb{E}\left(\prod_{w\in N(V)} (1 - W_{j,w}) \mid F\right) = \prod_{w\in N(v)} \mathbb{E}(1 - W_{j,w} \mid F)$$

as $\mathbb{E}(AB) = \mathbb{E}(A)\mathbb{E}(B)$ for independent random variables A, B. Linearity of expectation then gives

$$\mathbb{E}(|A(X,v)| \mid F) = \sum_{j\in[k]} \left(\prod_{w\in N(v)} (1 - \mathbb{E}(W_{j,w} \mid F)) \right).$$

But each $w \in N(v)$ is able to choose a colour uniformly at random from $A(X,w)$. So for each w where $j \in A(X,w)$, $\mathbf{Pr}(W_{j,w} = 1 \mid F) = \frac{1}{|A(X,w)|}$. Note that $\mathbb{E}(W_{j,w} \mid F) = \mathbf{Pr}(W_{j,w} = 1 \mid F)$. We may ignore vertices w of $N(v)$ where $j \notin A(X,w)$ as they have $W_{j,w} = 0$ and so will contribute 1 to the product $\prod_{w\in N(v)}(1 - \mathbb{E}(W_{j,w} \mid F))$. Therefore,

$$\mathbb{E}(|A(X,v)| \mid F) = \sum_{j\in[k]} \left(\prod_{w\in N(v):j\in A(X,w)} \left(1 - \frac{1}{|A(X,w)|}\right) \right).$$

Now we may use the arithmetic-geometric mean inequality to get

$$\mathbb{E}(|A(X,v)| \mid F) \geq k \prod_{j\in[k]} \left(\prod_{w\in N(v):j\in A(X,w)} \left(1 - \frac{1}{|A(X,w)|}\right)^{\frac{1}{k}} \right).$$

But we can re-arrange the order of multiplication to get

$$\mathbb{E}(|A(X,v)| \mid F) \geq k \prod_{w\in N(v)} \prod_{j\in A(X,w)} \left(1 - \frac{1}{|A(X,w)|}\right)^{\frac{1}{k}}.$$

In the inner multiplication, the term being multiplied does not depend on j. In this sense, j is merely counting the number of members of $A(X,w)$. Hence

$$\mathbb{E}(|A(X,v)| \mid F) \geq k \prod_{w\in N(v)} \left(1 - \frac{1}{|A(X,w)|}\right)^{\frac{|A(X,w)|}{k}}$$

But $0 < 1 - \frac{1}{|A(X,w)|} \leq 1$ and $(1-p)^{\frac{1}{p}} \geq \frac{1-p}{e}$ for all $0 < p \leq 1$ by Proposition 8.3. So,

$$\mathbb{E}(|A(X,v)| \mid F) \geq k \prod_{w\in N(v)} \left(\frac{1 - 1/|A(X,w)|}{e}\right)^{\frac{1}{k}}.$$

Now $|A(X,w)| \geq k - \Delta(G) \geq \frac{2}{\beta}$ and $|N(v)| \leq \Delta(G)$. This implies

$$\mathbb{E}(|A(X,v)| \mid F) \geq ke^{-\Delta(G)/k} \left(1 - \frac{\beta}{2}\right)^{\Delta(G)/k}.$$

But $e^{-\Delta(G)/k} < 1$ so $e^{-\Delta(G)/k} - e^{-\Delta(G)/k}\frac{\beta}{2} \geq e^{-\Delta(G)/k} - \frac{\beta}{2}$ and

$$\mathbb{E}(|A(X,v)| \mid F) \geq k\left(e^{-\Delta(G)/k} - \frac{\beta}{2}\right)^{\Delta(G)/k} \geq k\left(e^{-\Delta(G)/k} - \frac{\beta}{2}\right).$$

The last inequality follows from the fact that $e^{-\Delta(G)/k} - \frac{\beta}{2} < 1$ and $\frac{\Delta(G)}{k} < 1$.

We know from Proposition 8.2 that we can use $|A(X,v)|$ to define a martingale $Z_0, \ldots, Z_k$ that satisfies the conditions of the Azuma-Hoeffding inequality. Take $\lambda := k\frac{\beta}{2} > 0$. By Theorem 8.1,

$$\mathbf{Pr}(|Z_k - Z_0| \geq \lambda) \leq 2e^{-\lambda^2/2k}.$$

In particular, $Z_0 = \mathbb{E}(|A(X,v)| \mid F)$ and $Z_k = |A(X,v)|$, by definition. So

$$\mathbf{Pr}\left(||A(X,v)| - \mathbb{E}(|A(X,v)| \mid F)| \geq k\frac{\beta}{2}\right) \leq 2e^{-(k\beta/2)^2/2k}.$$

Thus,

$$\mathbf{Pr}\left(|A(X,v)| - k\left(e^{-\Delta(G)/k} - \frac{\beta}{2}\right) \leq -k\frac{\beta}{2}\right) \leq 2e^{-\beta^2 k/8}.$$

Therefore,

$$\mathbf{Pr}\left(|A(X,v)| \leq k(e^{-\Delta(G)/k} - \beta)\right) \leq 2e^{-\beta^2 k/8}.$$

This holds for any vertex v. The bound, summed over all vertices, will then yield the result. □

8.1 A New Coupling

In proceeding with the argument, Hayes and Vigoda set up a new coupling $\mathcal{Z} = (\mathcal{X}, \mathcal{Y})$, where $\mathcal{X}, \mathcal{Y}$ are faithful copies of $\mathcal{M}$[1]. Throughout, we suppose that $\mathcal{X}$ starts at a colouring X_0 chosen uniformly at random, and that $\mathcal{Y}$ starts at a particular colouring Y_0.

The difficulty here is to define a procedure that takes $\mathcal{Z}$ from step to step. In $\mathcal{M}$, a vertex was chosen at random and then a colour chosen at random from all the colours currently available to v. If we let $\mathcal{X}$ and $\mathcal{Y}$ be two independent copies of $\mathcal{M}$ then $\mathcal{Z}$ will indeed be a valid coupling. But as noted in the previous chapter, this is unlikely to lead to a good bound on the mixing rate. It would be helpful if, as for Jerrum's coupling, we are able to base the movement of $\mathcal{X}$ and $\mathcal{Y}$ on a shared random decision at each step. But suppose vertex v was chosen at step t. According to the workings

[1]In actual fact, Hayes and Vigoda merely sketched the coupling. The details given here were suggested by Dr. Deryk Osthus.

of $\mathcal{M}$, we now need to choose a colour at random from all colours currently available to v. But this is where the problem lies. The colours available to v in X_t and Y_t are $A(X_t, v)$ and $A(Y_t, v)$, respectively. $A(X_t, v)$ and $A(Y_t, v)$ are likely to be non-equal sets, and of different sizes.

A procedure can be given that solves this problem. Suppose vertex v is chosen at random for step t. How $\mathcal{Z}$ behaves next depends on the comparative sizes of $|A(X_t, v)|$ and $|A(Y_t, v)|$. Let C_{X_t}, C_{Y_t} be as defined in the previous chapter (see page 71). The different cases are:

1. $|A(X_t, v)| = |A(Y_t, v)|$
   In this case there is an obvious bijection $g : A(X_t, v) \to A(Y_t, v)$ with $g(c) = c$ for all $c \in A(X_t, v) \cap A(Y_t, v)$. The coupling chooses a colour c uniformly at random from $A(X_t, v)$ and lets $X_{t+1}(v) := c$ and $Y_{t+1}(v) := g(c)$. There is an obvious similarity to Jerrum's coupling in this case. Note that

$$\mathbf{Pr}(g(c) \text{ chosen} \mid v) = \mathbf{Pr}(c \text{ chosen} \mid v) = \frac{1}{|A(X_t, v)|} = \frac{1}{|A(Y_t, v)|}.$$

2. $|A(X_t, v)| < |A(Y_t, v)|$
   For this step, $\mathcal{Z}$ will not choose a single colour, rather it will choose an ordered pair of colours. Now, it must be that $|C_{X_t}| < |C_{Y_t}|$. Let $p := \frac{1}{|A(Y_t,v)|}$. Let $g : A(X_t, v) \to A(Y_t, v)$ be an injection where $g(c) = c$ for all $c \in A(X_t, v) \cap A(Y_t, v)$. Also, let D be the set of all colours in C_{Y_t} not mapped to by g. Assign probabilities to elements of $A(X_t, v) \times A(Y_t, v)$ as follows.

   (a) For each $(c, g(c)) \in A(X_t, v) \times Im(g)$ let

$$\mathbf{Pr}((c, g(c))) := p.$$

   (b) For each $(c, d) \in A(X_t, v) \times D$ let

$$\mathbf{Pr}((c, d)) := \frac{p}{|A(X_t, v)|}.$$

   (c) Assign probability 0 to any other pair (c, c').

   Once a pair (a, b) has been chosen, v is assigned colour a in X_{t+1} and colour b in Y_{t+1}.

3. $|A(Y_t, v)| < |A(X_t, v)|$
   This case is symmetric to the one above. In this case g is an injection from $A(Y_t, v)$ to $A(X_t, v)$, and $p := \frac{1}{|A(X_t,v)|}$.

Note that in every case the probability functions defined are genuine probability functions. In each case the probabilities sum to 1. This is obvious for case 1, but for case 2 the sum of the probabilities is

$$\frac{|A(X_t,v)|}{|A(Y_t,v)|} + \frac{|A(X_t,v)||D|}{|A(X_t,v)||A(Y_t,v)|} = \frac{|A(X_t,v)||A(Y_t,v)|}{|A(X_t,v)||A(Y_t,v)|} = 1,$$

which follows as $|D| = |A(Y_t,v)| - |A(X_t,v)|$. Case 3 will be similar to case 2.

It must now be shown that this procedure enables $\mathcal{Z}$ to behave as a genuine coupling. We must show that $\mathcal{X}$ and $\mathcal{Y}$, each considered in isolation, behave as faithful copies of $\mathcal{M}$. Now if $|A(X_t,v)| = |A(Y_t,v)|$ then v is coloured $c \in A(X_t,v)$ with probability $\frac{1}{|A(X_t,v)|}$. Similarly, v is coloured $g(c) \in A(Y_t,v)$ with probability $\frac{1}{|A(Y_t,v)|}$. Thus in this case $\mathcal{X}$ and $\mathcal{Y}$ behave as faithful copies of $\mathcal{M}$.

Suppose $|A(X_t,v)| < |A(Y_t,v)|$ then. Suppose $c \in A(X_t,v)$. Now v can receive colour c in X_{t+1} only if one of the pairs $(c,g(c))$ or $(c,d) \in A(X_t,v) \times D$ was chosen. $(c,g(c))$ is chosen with probability $\frac{1}{|A(Y_t,v)|}$ while the probability of choosing a pair (c,d) is $\sum_{d\in D} \mathbf{Pr}((c,d)) = \frac{|D|}{|A(Y_t,v)||A(X_t,v)|}$. Thus v is coloured with c in X_{t+1} with total probability

$$\frac{1}{|A(Y_t,v)|}+\frac{|D|}{|A(Y_t,v)||A(X_t,v)|} = \frac{1}{|A(Y_t,v)|}+\frac{|A(Y_t,v)|-|A(X_t,v)|}{|A(Y_t,v)||A(X_t,v)|} = \frac{1}{|A(X_t,v)|}.$$

Hence $\mathcal{X}$ acts as a faithful copy of $\mathcal{M}$ in this case. Suppose $c \in A(Y_t,v)$. Note that $c \in Im(g)$ or $c \in D$. Suppose $v \in Im(g)$. Then v can receive colour c in Y_{t+1} if $c = g(c')$ for some $c' \in A(X_t,v)$ and $(c'c)$ was chosen with probability $\frac{1}{|A(Y_t,v)|}$. Suppose $c \in A(Y_t,v)\backslash Im(g) = D$ then. v will be coloured c in Y_{t+1} if a pair $(a,c) \in A(X_t,v) \times D$ is chosen. This will happen with probability

$$\sum_{a\in A(X_t,v)} \mathbf{Pr}((a,c)) = \frac{|A(X_t,v)|}{|A(Y_t,v)||A(X_t,v)|} = \frac{1}{|A(Y_t,v)|}.$$

Hence $\mathcal{Y}$ acts as a faithful copy of $\mathcal{M}$ in this case.

The case of $|A(Y_t,v)| < |A(X_t,v)|$ will be analogous to the previous case. So $\mathcal{X}$ and $\mathcal{Y}$ always behave as faithful copies of $\mathcal{M}$.

How the coupling runs may be made clearer if an example is considered. In Figure 8.1 the two colourings X_t and Y_t of the same graph are presented. If vertex w is chosen in this step then the choices of colour are straightforward. Colours 2,5,6, and 7 may be chosen for both chains, each with probability $\frac{1}{5}$. With probability $\frac{1}{5}$ colour 1 may be chosen for $\mathcal{X}$, which forces colour 3 to be chosen for $\mathcal{Y}$. If vertex v is chosen then the coupling chooses ordered pairs of colours at random. The pairs it has to choose from (that is, the ones with non-zero probability) are (2,2), (6,6), (7,7), (1,3), (2,5), (6,5), (7,5) and

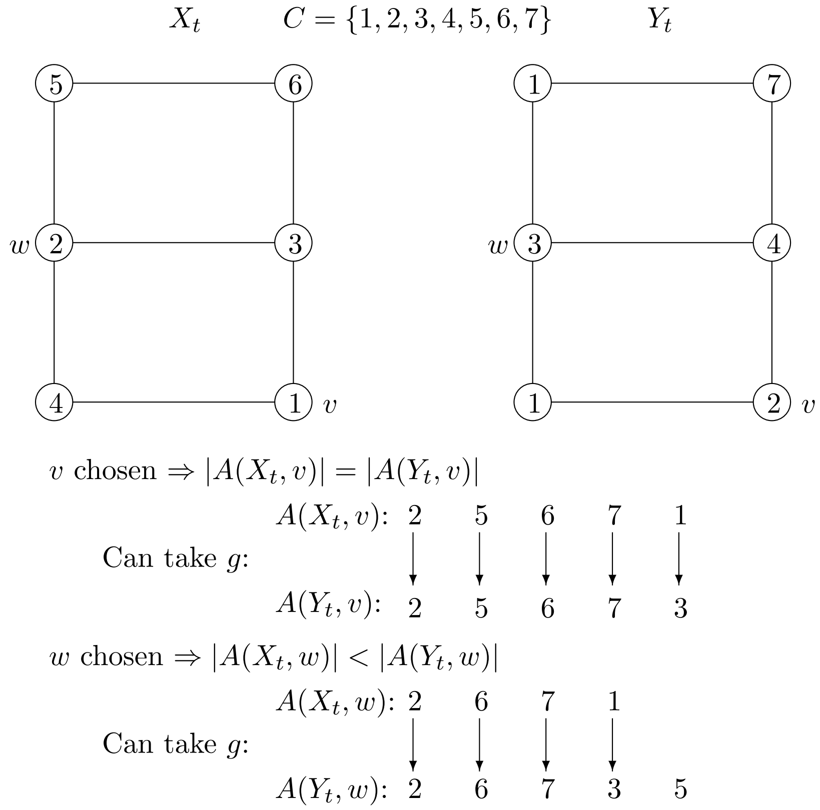

Figure 8.1: An example of Hayes and Vigoda's coupling

(1,5). The pairs (2,2), (6,6), (7,7) and (1,3) are all assigned probability $\frac{1}{5}$, while (2,5), (6,5), (7,5) and (1,5) are all assigned probability $\frac{1}{20}$. Note that this distribution has the desired properties. The probabilities sum to 1 and, for example, the probability of colour 5 being chosen for $\mathcal{Y}$ in this step is

$$\mathbf{Pr}((2,5)) + \mathbf{Pr}((6,5)) + \mathbf{Pr}((7,5)) + \mathbf{Pr}((1,5)) = 4\frac{1}{20} = \frac{1}{5} = \frac{1}{|A(Y_t, v)|}.$$

Also, note that the probability of colour 6 being chosen for $\mathcal{X}$ is

$$\mathbf{Pr}((6,6)) + \mathbf{Pr}((6,5)) = \frac{1}{5} + \frac{1}{20} = \frac{1}{4} = \frac{1}{|A(X_t, v)|}.$$

We may now continue with Hayes and Vigoda's argument [3]. We wish to show that, conditioned on the colours available to vertices under X_t, we expect X_{t+1}, Y_{t+1} to agree on more vertices than X_t, Y_t. This will be the purpose of Lemma 8.2. The following definitions are for ease of notation.

Definition 8.3. $\rho(X_t, Y_t) := |\{v \in V(G) : X_t(v) \neq Y_t(v)\}|$ is the number of vertices where X_t and Y_t disagree.

Definition 8.4. Suppose coupling $\mathcal{Z} = (\mathcal{X}, \mathcal{Y})$ reaches state (x, y). (x, y) is ϵ *distance decreasing* for $\mathcal{Z}$ if

$$\mathbb{E}(\rho(X_{t+1}, Y_{t+1}) \mid X_t = x, Y_t = y) < (1 - \epsilon)\rho(x, y).$$

With the new coupling, the next result follows easily.

Lemma 8.2. Let $0 < \beta < 1$ and suppose that $\forall v \in V(G), |A(X,v)| \geq \frac{\Delta(G)}{1-\beta}$ for colouring X. Then for any possible colouring Y,

$$\mathbb{E}(\rho(X_{t+1}, Y_{t+1}) \mid X_t = X, Y_t = Y) \leq \left(1 - \frac{\beta}{n}\right) \rho(X, Y).$$

Proof. Suppose v was the vertex chosen for step t. For v to receive the same colour c in X_{t+1} and Y_{t+1}, it must be that $c \in A(X_t, v) \cap A(Y_t, v)$. But whatever the relative sizes of $|A(X_t, v)|$ and $|A(Y_t, v)|$, it will always be that case that c is chosen for both chains with probability $\min\left\{\frac{1}{|A(X_t,v)|}, \frac{1}{|A(Y_t,v)|}\right\}$. So

$$\begin{aligned}
\mathbf{Pr}(X_{t+1}(v) \neq Y_{t+1}(v)) &= 1 - \mathbf{Pr}(c \in A(X_t, v) \cap A(Y_t, v)) \\
&= 1 - \frac{|A(X_t, v) \cap A(Y_t, v)|}{\max\{|A(X_t, v)|, |A(Y_t, v)|\}} \\
&= \frac{\max\{|A(X_t, v)|, |A(Y_t, v)|\} - |A(X_t, v) \cap A(Y_t, v)|}{\max\{|A(X_t, v)|, |A(Y_t, v)|\}}
\end{aligned}$$

For a colour to be available to v in X_t but not Y_t, say, it must be that a neighbour w of v has this colour in Y_t. Furthermore, w must be coloured differently in X_t, else this colour would be unavailable to v here as well. Let d_v be the number of neighbours of v receiving different colours in X_t and Y_t. Then $\max\{|A(X_t, v)|, |A(Y_t, v)|\} - |A(X_t, v) \cap A(Y_t, v)| \leq d_v$ and

$$\mathbf{Pr}(X_{t+1}(v) \neq Y_{t+1}(v)) \leq \frac{d_v}{\max\{|A(X_t, v)|, |A(Y_t, v)|\}}.$$

For a vertex w, let E_w be the event that chosen vertex v is w. By definition, we have,

$$\begin{aligned}
\mathbb{E}(\rho(X_{t+1}, Y_{t+1})) &= \sum_{w \in V} [\mathbf{Pr}(X_{t+1}(w) \neq Y_{t+1}(w) \mid E_w)\mathbf{Pr}(E_w) \\
&\quad + \mathbf{Pr}(X_{t+1}(w) \neq Y_{t+1}(w) \mid \neg E_w)\mathbf{Pr}(\neg E_w)] \\
&\quad [\text{since } \mathbf{Pr}(B) = \mathbf{Pr}(B \mid A)\mathbf{Pr}(A) + \mathbf{Pr}(B \mid \overline{A})\mathbf{Pr}(\overline{A})]
\end{aligned}$$

But if vertex w is not v then w's colour is fixed for this step. So $\mathbf{Pr}(X_{t+1}(w) \neq Y_{t+1}(w) \mid \neg E_w)$ is 0 if X_t and Y_t agree on w. It is 1 otherwise. So each time $\mathbf{Pr}(X_{t+1}(w) \neq Y_{t+1}(w) \mid \neg E_w) = 1$, $\mathbf{Pr}(\neg E_w) = \frac{n-1}{n}$ will be contributed to the overall sum. This will obviously happen at most $\rho(X_t, Y_t)$ times. Hence,

$$\mathbb{E}(\rho(X_{t+1}, Y_{t+1})) \leq \sum_{w \in V} \left[\mathbf{Pr}(X_{t+1}(w) \neq Y_{t+1}(w) \mid E_w)\frac{1}{n}\right] + \frac{n-1}{n}\rho(X_t, Y_t)$$

Using the bound derived for $\mathbf{Pr}(X_{t+1}(v) \neq Y_{t+1}(v))$ above, we get

$$\mathbb{E}(\rho(X_{t+1}, Y_{t+1})) \leq \frac{n-1}{n}\rho(X,Y) + \frac{1}{n}\sum_{w \in V(G)} \frac{d_w}{\max\{|A(X_t,w)|, |A(Y_t,w)|\}}.$$

Now the sum will count the number of ordered pairs of vertices (w, w') where X_t and Y_t disagree on w. But then we can find an upper bound for this figure by considering each w' where X_t and Y_t disagree, and then count the number of edges connected to w'. w' has at most $\Delta(G)$ edges connected to it. Thus, $\sum_{w \in V(G)} d_w \leq \rho(X_t, Y_t)\Delta(G)$. Also, by assumption, $|A(X,w)| \geq \frac{\Delta(G)}{1-\beta}$ for all w, so $\max\{|A(X_t,w)|, |A(Y_t,w)|\} \geq \frac{\Delta(G)}{1-\beta}$. Altogether, this gives

$$\mathbb{E}(\rho(X_{t+1}, Y_{t+1})) \leq \frac{n-1}{n}\rho(X,Y) + \frac{1}{n}\frac{\rho(X,Y)\Delta(G)}{\Delta(G)/(1-\beta)} = \left(1 - \frac{\beta}{n}\right)\rho(X,Y).$$

□

The message of this lemma is clear: If we manage to show that any colouring X_t in $\mathcal{X}$ allows any vertex a wide enough choice of colours, then the two constituent chains of $\mathcal{Z}$ will tend to converge in the next step. This sort of result is promising in terms of showing that $\mathcal{X}$ and $\mathcal{Y}$ will tend to couple, but to show that this happens quickly requires more work. With this in mind, we prove the following theorem of Hayes and Vigoda.

Theorem 8.2. Suppose $0 < \epsilon, \delta < 1$. Let X_T, Y_T be the states of a coupling at step T with

$$\mathbf{Pr}((X_t, Y_t) \text{ is not } \epsilon \text{ distance decreasing}) \leq \delta.$$

Then

$$\mathbf{Pr}(X_T \neq Y_T) \leq \left((1-\epsilon)^T + \frac{\delta}{\epsilon}\right)|G|.$$

Proof. Let A be the event that (X_t, Y_t) is ϵ distance decreasing.

$$\begin{aligned}\mathbb{E}(\rho(X_{t+1}, Y_{t+1}) - (1-\epsilon)\rho(X_t, Y_t)) &= \mathbf{Pr}(A)\mathbb{E}(\rho(X_{t+1}, Y_{t+1}) - (1-\epsilon)\rho(X_t, Y_t) \mid A) + \\ &\quad \mathbf{Pr}(\neg A)\mathbb{E}(\rho(X_{t+1}, Y_{t+1}) - (1-\epsilon)\rho(X_t, Y_t) \mid \neg A)\end{aligned}$$

But if A holds, then $\rho(X_{t+1}, Y_{t+1}) < (1-\epsilon)\rho(X_t, Y_t)$ so $\mathbb{E}(\rho(X_{t+1}, Y_{t+1}) - (1-\epsilon)\rho(X_t, Y_t) \mid A) < 0$. Thus,

$$\mathbb{E}(\rho(X_{t+1}, Y_{t+1}) - (1-\epsilon)\rho(X_t, Y_t)) \leq \mathbf{Pr}(\neg A)\mathbb{E}(\rho(X_{t+1}, Y_{t+1}) - (1-\epsilon)\rho(X_t, Y_t) \mid \neg A).$$

$\mathbf{Pr}(\neg A) < \delta$ by assumption and clearly $\rho(X_{t+1}, Y_{t+1}) - (1-\epsilon)\rho(X_t, Y_t) \leq \rho(X_{t+1}, Y_{t+1})$ so

$$\mathbb{E}(\rho(X_{t+1}, Y_{t+1}) - (1-\epsilon)\rho(X_t, Y_t)) \leq \delta\mathbb{E}(\rho(X_{t+1}, Y_{t+1})) \leq \delta|G|.$$

The last inequality holds because $\rho(X_{t+1}, Y_{t+1}) \leq |G|$ always. So, using linearity of expectation, we can rearrange to get

$$\mathbb{E}(\rho(X_{t+1}, Y_{t+1})) \leq (1-\epsilon)\mathbb{E}(\rho(X_t, Y_t)) + \delta|G|. \tag{8.1}$$

Considering the first steps of the coupling, we get

$$\mathbb{E}(\rho(X_1, Y_1)) \leq (1-\epsilon)\mathbb{E}(\rho(X_0, Y_0)) + \delta|G| \leq (1-\epsilon)|G| + \delta|G|.$$

Using this, and the recursive nature of inequality 8.1, we are able to derive an upper bound for each $\mathbb{E}(\rho(X_t, Y_t))$. For step t, suppose

$$\mathbb{E}(\rho(X_t, Y_t)) \leq (1-\epsilon)^t|G| + \delta|G|\left(\frac{1-(1-\epsilon)^t}{\epsilon}\right).$$

Then

$$\begin{aligned}
\mathbb{E}(\rho(X_{t+1}, Y_{t+1})) &\leq (1-\epsilon)\left[(1-\epsilon)^t|G| + \delta|G|\left(\frac{1-(1-\epsilon)^t}{\epsilon}\right)\right] + \delta|G| \\
&\quad \text{[by inequality 8.1]} \\
&= (1-\epsilon)^{t+1}|G| + \delta|G|\left(\frac{1-(1-\epsilon)^{t+1}}{\epsilon}\right)
\end{aligned}$$

But

$$\mathbb{E}(\rho(X_1, Y_1)) \leq (1-\epsilon)|G| + \delta|G| = (1-\epsilon)|G| + \delta|G|\left(\frac{1-(1-\epsilon)^1}{\epsilon}\right).$$

Thus, by induction,

$$\mathbb{E}(\rho(X_t, Y_t)) \leq \left((1-\epsilon)^t + \frac{\delta}{\epsilon}\right)|G|$$

for all t. Using Markov's inequality, we derive that

$$\begin{aligned}
\mathbf{Pr}(X_T \neq Y_T) &= \mathbf{Pr}(\rho(X_T, Y_T) \geq 1) \\
&\leq \mathbb{E}(\rho(X_T, Y_T)) \\
&\leq \left((1-\epsilon)^T + \frac{\delta}{\epsilon}\right)|G|
\end{aligned}$$

Hence the result holds. □

Theorem 8.2 makes an interesting suggestion. For Jerrum's argument, it was argued in general that the coupled chains would tend to converge at every step. In contrast, Theorem 8.2 introduces the idea that an upper bound may be found for $\mathbf{Pr}(X_T \neq Y_T)$, even if there is a small subset of coupled states from which $\mathcal{X}$ and $\mathcal{Y}$ will tend to diverge.

Now, Theorem 8.2 has another theorem that follows as a corollary. The new theorem will be of great importance in the overall argument. It must be borne in mind that in what follows "$||X_t - Y_t||_d$" is taken to mean the total variation distance between the *probability distributions* of X_t and Y_t, not any distance between X_t and Y_t themselves. The form $||X_t - Y_t||_d$ is certainly technically imprecise, but it what follows technical precision may hinder rather than aid understanding.

Theorem 8.3. Let $\epsilon, \delta > 0$. Let S be a set of colourings with the following property: for all $X \in S$, (X, Y) is ϵ distance decreasing for all colourings Y. Suppose further that

$$\pi(S) \geq 1 - \frac{\epsilon}{8|G|}$$

where π is the stationary distribution of $\mathcal{X}$. Then if the coupling starts at (X_0, Y_0) and $T \geq \frac{\lceil \ln(40|G|) \rceil \lceil \ln(1/\delta) \rceil}{\epsilon}$, then

$$||X_T - \pi||_d \leq \delta.$$

Proof. Recall that X_0 is chosen randomly according to π, and so each subsequent X_t can be seen as having the same probability distribution. Therefore, for any step t,

$$\mathbf{Pr}((X_t, Y_t) \text{ not } \epsilon \text{ distance decreasing}) \leq \mathbf{Pr}(X_t \notin S) = 1 - \pi(S).$$

Let $T' := \left\lceil \frac{\ln(40|G|)}{\epsilon} \right\rceil$. Since $X_{T'}$ will have distribution π, $||Y_{T'} - X_{T'}||_d = ||Y_{T'} - \pi||_d$. Then

$$\begin{aligned}
||Y_{T'} - \pi||_d &\leq \mathbf{Pr}(X_{T'} \neq Y_{T'}) \quad \text{[by the Coupling Lemma]} \\
&\leq \left((1-\epsilon)^{T'} + \frac{1 - \pi(S)}{\epsilon} \right) |G| \quad \text{[by Theorem 8.2]} \\
&\leq \left(\exp(-\epsilon T') + \frac{1}{8|G|} \right) |G| \quad \text{[since } (1-x)^t \leq e^{-xt}\text{]} \\
&\leq \left(\frac{1}{40|G|} + \frac{1}{8|G|} \right) |G| = \frac{3}{20} \quad \text{[by value of T']}
\end{aligned}$$

Recall X_0 is arbitrary. For any possible Y_0, we will have (X_0, Y_0) with

$$\begin{aligned}
||X_{T'} - Y_{T'}||_d &\leq ||X_{T'} - \pi||_d + ||\pi - Y_{T'}||_d \quad \text{[since } || \; ||_d \text{ is a norm]} \\
&< \frac{3}{10} < \frac{1}{e}.
\end{aligned}$$

Note that $T = T' \lceil \ln(1/\delta) \rceil$. So if the coupling runs for T steps, it can be seen as a sequence of $\lceil \ln(1/\delta) \rceil$ couplings of length T'. But then in the final step of each "sub-coupling", the total variation distance between the probability distributions must be bounded above $\frac{1}{e}$.

Using the Coupling Lemma, it is easy to see that

$$||Y_T - \pi||_d \leq \max_{X_0} \mathbf{Pr}(Y_T \neq X_T),$$

where X_0 ranges over all possible starting states for $\mathcal{X}$. If the coupled chains meet by step T, then they must meet in one of the $\lceil \ln(1/\delta) \rceil$ sub-couplings. $\mathcal{Z}$ is a tandem coupling, so the probability that the chains remain apart by the end of the i-th subcoupling is

$$\mathbf{Pr}(X_{iT'} \neq Y_{iT'} \mid X_{(i-1)T'} \neq Y_{(i-1)T'}).$$

That is, it is the probability that the chains will not meet in this subcoupling given that they have not met in the previous one. The proof that $||X_{T'} - Y_{T'}||_d < \frac{1}{e}$ means that we need not worry about whether $X_{iT'}$ has a uniform probability distribution. The $\frac{1}{e}$ bound holds for any starting pair $(X_{(i-1)T'}, Y_{(i-1)T'})$. Therefore,

$$\begin{aligned} ||Y_T - \pi||_d &\leq \prod_{i=1}^{\lceil \ln(1/\delta) \rceil} \mathbf{Pr}(X_{iT'} \neq Y_{iT'} \mid X_{jT'} \neq Y_{jT'} \forall j < i) \\ &\quad \text{[repeatedly applying the rule } \mathbf{Pr}(A) = \mathbf{Pr}(A \mid B)\mathbf{Pr}(B)\text{]} \\ &\leq \prod_{i=1}^{\lceil \ln(1/\delta) \rceil} \mathbf{Pr}(X_{iT'} \neq Y_{iT'} \mid X_{(j-1)T'} \neq Y_{(j-1)T'}) \\ &\quad \text{[by the observation above]} \\ &\leq \left(\frac{1}{e}\right)^{\lceil \ln(1/\delta) \rceil} \\ &\leq \delta \end{aligned}$$

□

Hayes and Vigoda [3] draw all their reasoning together in the next result:

Theorem 8.4. Let $0 < \eta < 1$ and let $\alpha \in \mathbb{R}$ such that $\alpha - e^{\frac{1}{\alpha}} = 0$. Suppose G is a triangle-free graph with $|G| = n$. Let $k \geq \max\left\{(1+\eta)\alpha\Delta(G), 288\frac{\ln(96n^3/\eta)}{\eta^2}\right\}$ and let X_0 be any k-colouring of G. Then for the Markov chain $\mathcal{M}$ under consideration and for any $\delta > 0$, if $T \geq 6n\frac{\lceil \ln(40n) \rceil \lceil \ln(1/\delta) \rceil}{\eta}$ then

$$||X_T - \pi||_d \leq \delta.$$

Proof. Let $\beta = \frac{\eta}{6}$. Now $k \geq 288\frac{\ln(96n^3/\eta)}{\eta^2}$ and $k \geq (1+\eta)\alpha$. Note that since $\alpha - e^{\frac{1}{\alpha}} = 0$, $\alpha = 1.763\ldots < 2$. Given β,

$$\begin{aligned} k &\geq 288\frac{\ln\left(\frac{96n^3}{\eta}\right)}{\eta^2} = 8\frac{\ln\left(\frac{16n^3}{\beta}\right)}{\beta^2} \geq \frac{8}{\beta^2} \quad \left[\text{as } \frac{16n^3}{\beta} > e\right] \\ &\geq \frac{8}{\beta}. \end{aligned}$$

Similarly,

$$k \geq (1+\eta)\alpha\Delta(G) > \alpha\Delta(G) > \frac{3}{2}\Delta(G).$$

So, if $\Delta(G) \geq \frac{4}{\beta}$, then

$$k \geq \frac{3}{2}\Delta(G) \geq \Delta(G) + \frac{1}{2}\frac{4}{\beta} = \Delta(G) + \frac{2}{\beta}.$$

If $\Delta(G) < \frac{4}{\beta}$,

$$k \geq \frac{8}{\beta} = \frac{6}{\beta} + \frac{2}{\beta} > \Delta(G) + \frac{2}{\beta}.$$

Altogether then, $k \geq \Delta(G) + \frac{2}{\beta}$.

Let S be the set of all k-colourings Y of G such that, for any vertex $v \in V(G)$, $|A(X_0, v)| \geq k(e^{-\Delta(G)/k} - \beta)$. Given the bound on k, we may apply Lemma 8.1 to get

$$\mathbf{Pr}\left(\exists v \in V(G) : |A(X_0, v)| < k(e^{-\Delta(G)/k} - \beta)\right) \leq 2ne^{-\beta^2 k/8}.$$

So, under π, the vertices of S must contribute probabilities to there *not* being such a v. Therefore,

$$\pi(S) \geq 1 - 2ne^{-\beta^2 k/8}.$$

As $k \geq 8\frac{\ln\left(\frac{16n^3}{\beta}\right)}{\beta^2}$, we may further conclude that

$$\pi(S) \geq 1 - 2\frac{\beta}{16n^2} = 1 - \frac{\beta}{n}\frac{1}{8|G|}.$$

Now, using the known bounds on α, β and η,

$$\begin{aligned} (1+\eta)(1-\beta)(1-\alpha\beta) &> (1+\eta)(1-\frac{\eta}{6})(1-\frac{\eta}{3}) \\ &= 1 + \frac{1}{2}\eta - \frac{4}{9}\eta^2 + \frac{1}{18}\eta^3 \\ &> 1. \end{aligned}$$

The last inequality follows since the negation would require $\frac{1}{18}\eta^2 - \frac{4}{9}\eta + \frac{1}{2}$ to be less than 0 for $0 < \eta < 1$. This is not the case, so the inequality holds.

Now, $0 < \eta = 6\beta < 1$ and $\alpha < 2$ implies that $\alpha(1+\eta) > 1$. So $\alpha(1+\eta)(1-\beta) > 1-\beta$. Therefore,

$$
\begin{aligned}
\frac{\Delta(G)}{1-\beta} &\leq \frac{k}{\alpha(1+\eta)(1-\beta)} \\
&= \frac{k(1-\alpha\beta)}{\alpha(1+\eta)(1-\beta)(1-\alpha\beta)} \\
&< \frac{k(1-\alpha\beta)}{\alpha} \quad [\text{as } (1+\eta)(1-\beta)(1-\alpha\beta) > 1] \\
&= k\left(\frac{1}{\alpha} - \beta\right) \\
&= k(e^{-\frac{1}{\alpha}} - \beta) \quad [\text{given } \alpha = e^{\frac{1}{\alpha}}] \\
&< k(e^{-\frac{\Delta(G)}{k}} - \beta) \quad [\text{since } k \geq (1+\eta)\alpha\Delta(G) > \alpha\Delta(G)] \\
&\leq |A(X,v)| \text{ for all } X \in S \text{ and } v \in V(G).
\end{aligned}
$$

Then for any $X \in S$ that appears in the new coupling $\mathcal{Z}$, the pair (X,Y) it forms with a colouring Y will be $\frac{\beta}{n}$ distance decreasing. This is by Lemma 8.2. A straight-forward application of Theorem 8.3 then gives the result. □

Whilst Theorem 8.4 can be seen as the major result of this chapter and establishes a polynomial lower bound on the number of steps τ that $\mathcal{M}$ needs to make, the conditions it sets down are complex and unwieldy. The following theorem can be seen as a corollary of Theorem 8.4, one that seeks to set down more straightforward conditions for G and k to satisfy. We suppose again that G is triangle-free with $n = |G|$.

Theorem 8.5. If $\Delta(G) > 90\ln(96n^3)$ with $k > 1.764\Delta(G)$ then there is an $0 < \eta < 1$ with $\Delta(G) > 90\frac{\ln(96n^3/\eta)}{\eta^3}$ such that $\mathcal{M}$ has mixing time $T \geq 6n\frac{\lceil \ln(40n)\rceil \lceil \ln(1/\epsilon)\rceil}{\eta}$ for $\epsilon > 0$.

Proof. The inequality $\Delta(G) > 90\ln(96n^3)$ is strict. So by taking successive values of η that approach 1, one can see that if $\Delta(G) > 90\ln(96n^3)$ then there is obviously an $0 < \eta < 1$ such that

$$\Delta(G) > 90\frac{\ln(96n^3/\eta)}{\eta^3} \geq 180\frac{\ln(96n^3/\eta)}{\eta^2+\eta^3}.$$

But then

$$k > 1.764\Delta(G) > \alpha\Delta(G) > 288\frac{\ln(96n^3/\eta)}{\eta^2+\eta^3}.$$

So

$$k > (1+\eta)\alpha\Delta(G) > 288\frac{\ln(96n^3/\eta)}{\eta^2}.$$

Hence k satisfies the conditions in Theorem 8.4. The result follows by an application of Theorem 8.4 and the definition of mixing time. □

So suppose we have a graph G of large maximum degree such that it can fulfil the conditions in Theorem 8.5. If $\epsilon > 0$ is an acceptable degree of sampling bias we may run the generator of Hayes and Vigoda for $\tau \geq n^{\frac{\lceil \ln(40n) \rceil \lceil \ln(1/\epsilon) \rceil}{\eta}}$ steps, where η is dependent on $\Delta(G)$. This will give a sample chosen almost uniformly at random. But for η, the claim that the number of steps needed is polynomial in $n = |G|$ and $\frac{1}{\epsilon}$ would be clear. However, if we assume the difference between $\Delta(G)$ and $90 \ln(96n^3)$ is large then we need not take a small value for η. Indeed, if we set the difference between and $\Delta(G)$ and $90 \ln(96n^3)$ large enough, we will be able to bound η from below by a constant. This would make the lower bound for τ a polynomial in $n = |G|$ and $\frac{1}{\epsilon}$. So we have what we want, if only for a select group of graphs.

Chapter 9

The FPRAS for the Colouring-Counting Problem

In the previous two chapters, we have seen two sampling algorithms, each of which is able to sample almost uniformly from a particular graph in an efficient manner. Certainly the second algorithm was only proved efficient for triangle-free graphs, so we henceforth assume that this algorithm is not used to sample from any other sort of graph. What we must now do is show, as for the problem of computing the permanent, that the entire approximation algorithm, of which the sampling procedures only form a part, is efficient. That is, we must show that the algorithm forms an FPRAS. This will be done in a similar manner to the algorithm for the first problem. Indeed, this chapter will use many of the same ideas as Chapter 5.

To begin with, it will help to set a bound on the number of samples needed to supply a reasonable estimate of $\frac{|\Omega_k(G_{r+1})|}{|\Omega_k(G_r)|}$ for some r. Recall that in the set-up of Theorem 5.6 the sample space $A \cup B$ had m samples taken from it. R was defined to be the ratio $\frac{|A|}{|A \cup B|}$ and $X = \sum_{i=1}^{m} X_i$ was the sum of m independent random variables where each

$$X_i = \begin{cases} 1, & \text{if sample } i \text{ is in } A; \\ 0, & \text{if sample } i \text{ is not in } A. \end{cases}$$

With regard to the present algorithm, recall that we are sampling from a graph G_r and are recording if the sample is a genuine k-colouring of G_{r+1} as well. But if we take A to be the set of samples that are colourings of both G_r and G_{r+1} then $A \cup B$ will be the set of all k-colourings of G_r. More importantly, Theorem 5.6 is now clearly relevant to our situation. We suppose that we sample from the k-colourings of a graph G_r with sampling error $d = \frac{1}{n^4}$.

But as for the FPRAS for approximating the permanent, before we can apply Theorem 5.6 we must supply an upper bound for R. But clearly $|\Omega_k(G_{r+1})| = |A| \leq |A \cup B| = |\Omega_k(G_r)|$, so $R = \frac{|A|}{|A \cup B|} \leq 1$. So if we have an allowable sampling error set by $\epsilon_1, \delta_1 > 0$ with $\epsilon_1 > \frac{2}{n^4} = 2d$ then taking $\omega > \frac{3}{4\epsilon_1^2} \ln\left(\frac{2}{\delta}\right)$ will give the desired number of samples by Theorem 5.6.

Now we must consider the cumulative effects of all the sampling errors. That is, we require an analogue of Theorem 5.8. But while Theorem 5.8 was stated in the context of the problem of computing the permanent, a careful consideration of its proof reveals that it need not be context-specific. It turns out that Theorem 5.8 has a direct analogue in the present case. Suppose we take r_i as the estimated value for $\frac{|\Omega_k(G_{i+1})|}{|\Omega_k(G_i)|}$ and let $R' = \prod_{i=1}^{m} \left(r_i \frac{|\Omega_k(G_i)|}{|\Omega_k(G_{i+1})|} \right)$ (recall that $m = |E(G)|$). Then the analogue of Theorem 5.8 is the following.

Theorem 9.1. Suppose that for each $i \in \{1, \ldots, m\}$, r_i is such that

$$\mathbf{Pr}\left(\left| \frac{|\Omega_k(G_{i+1})|}{|\Omega_k(G_i)|} - r_i \right| \leq \frac{\epsilon}{2m} \frac{|\Omega_k(G_{i+1})|}{|\Omega_k(G_i)|} \right) \geq 1 - \frac{\delta}{m}$$

for some $\frac{\epsilon}{2m}, \frac{\delta}{m} > 0$. Then

$$\mathbf{Pr}(|R' - 1| \leq \epsilon) \geq 1 - \delta.$$

Proof. The proof is similar to the proof of Theorem 5.8. □

So altogether, we may take $\delta > 0$ and $\epsilon > \frac{4}{n^2} = 2n^2 \frac{2}{n^4} \geq 2m(2d)$. The last inequality holds because the number of edges in G is $m \leq n^2 = |G|^2$. So if $\delta_1 = \frac{\delta}{m}$ and $\epsilon_1 = \frac{\epsilon}{2m}$ with $\omega > \frac{3}{4\epsilon_1^2} \ln\left(\frac{2}{\delta_1}\right)$, Theorems 5.6 and 9.1 imply that this will give

$$\mathbf{Pr}(|R' - 1| \leq \epsilon) \geq 1 - \delta.$$

But the algorithm will sample from at most m graphs. So at most $m\omega \leq n^2\omega$ samples will be taken in all. Note ω has a polynomial lower bound. The sampling process itself, whether one uses the algorithm of Jerrum, or Hayes and Vigoda, takes polynomial time. Hence the scheme presented to approximate the number of k-colourings of G is an FPRAS.

Chapter 10

Conclusion

10.1 General Observations

Having considered all three randomized algorithms presented, one can see that each is a fully polynomial randomized approximation scheme. While the two counting problems were very different in nature, the randomized algorithms used to estimate solutions shared some common ideas. For instance, all of them used a sampling approach to provide an estimate. Each tried to use a series of samples to compare an unknown quantity of the input to a known, or easily discovered, quantity. For example, in the case of the estimation of a permanent, the algorithm tried to compare the number of perfect matchings in a graph to the number of edges in the same graph. In the colouring problem, the algorithms attempted to compare the number of valid k-colourings in the full input graph G to the number of k-colourings in an edgeless graph of order $|G|$. This suggests that there may be many counting problems where a similar approach may offer a good estimate as a solution.

However, the proofs that each algorithm was able to sample almost uniformly at random from a given set were quite different. In the case of the Jerrum and Sinclair's algorithm to estimate the permanent, the algorithm was shown to sample almost uniformly in a direct fashion. A bound was given on the conductance of the associated Markov chain. In comparison, the algorithms estimating the number of k-colourings in a graph were shown to sample almost uniformly, but this was not proved directly. Instead, the Coupling Lemma was used to prove that the algorithms were almost uniform samplers. All one needed to do was produce a coupling where the two chains will tend to meet quickly, and this turned out to be sufficient to prove the claim. The Coupling Lemma is a very surprising result. It allows one to place an upper limit on the mixing time, even if one uses a coupling where the random decisions are "skewed" so that the constituent chains will meet quickly.

10.2 History and Outlook

As noted in chapter 1, it was Valiant who showed that the problem of computing the permanent of a matrix is **#P**-complete [11]. The fastest known deterministic algorithm for computing the permanent of a matrix was given by H. Ryser in 1963 [7]. However, it runs in time $O(n2^n)$ where n is the number of rows of the square matrix concerned. Whilst better than considering all $n!$ summands in the definition of the permanent, it is still far from efficient.

The randomized algorithm for the restricted problem of estimating the permanent of a dense $(0,1)$ matrix was shown to be an FPRAS by Jerrum and Sinclair in 1989 [6]. An earlier attempt had been made. A. Broder had attempted to show that the algorithm was an FPRAS in 1986 [10]. His approach was to use a coupling argument. Unfortunately, Broder's proof contained a fatal error, as noted in [6]. Jerrum, Sinclair and Vigoda have gone on to prove a more general result. In a 2004 paper [7] they managed to produce an FPRAS that is able to estimate the permanent of an $n \times n$ matrix with non-negative entries. This naturally raises the question of whether it is possible to find another scheme that will be able to deal with matrices containing negative entries as well. That is, the question is whether or not there is an FPRAS that can be estimate the permanent of an arbitrary real matrix. However, in the same paper, Jerrum, Sinclair and Vigoda managed to show that if there was such a scheme then one could compute the permanent of a $(0,1)$ matrix in polynomial time. But computing the permananent of a $(0,1)$ matrix is **#P**-complete. So whilst it is still possible that such a scheme exists, it is unlikely that there is such a scheme.

Jerrum published his randomized algorithm for estimating the number of k-colourings in a graph in 1994 [4]. He mentions a number of open problems at the end of his paper. One is to do with the bound on k. Jerrum set a lower bound of $2\Delta(G)+1$ on k. Jerrum mentions that it is possible to show that if $\Delta(G) \leq k \leq 2\Delta(G)$ then his algorithm will sample almost uniformly from the set of all k-colourings. That is, the colouring generator will tend towards the uniform distribution across all colours. However, it is not clear that it will do so in polynomial time. This is the problem that is still open. Vigoda was able to produce another algorithm for the same problem that had a polynomial mixing rate for $k > \frac{11}{6}\Delta(G)$ [2]. It worked for any graph G. Hayes and Vigoda, as has been seen, managed to push the bound on k down further, but only for triangle-free graphs. Similarly, Molloy in 2004 was able to improve the bound to $k > 1.489\Delta(G)$, but only for graphs of relatively large girth [2]. So the problem of a definite lower bound for k for arbitrary graphs is still open.

Bibliography

[1] N. Biggs, *Algebraic Graph Theory* (Cambridge, Cambridge University Press, 1993)

[2] A. Frieze and E. Vigoda, http://www.math.cmu.edu/ af1p/colouringsurvey.pdf (accessed 3.4.07)

[3] T. Hayes and E. Vigoda, "Coupling with the Stationary Distribution and Improved Sampling for Colorings and Independent Sets", *The Annals of Applied Probability* **16(3)**,1297 (2006)

[4] M. Jerrum, "A Very Simple Algorithm for Estimating the Number of k-Colourings of a Low-Degree Graph", *Random Structures and Algorithms* **7(2)**, 157 (1995)

[5] M. Jerrum and A Sinclair, "Approximate Counting, Uniform Generation and Rapidly Mixing Markov Chains", *Information and Computation* **82**, 93 (1989)

[6] M. Jerrum and A. Sinclair, "Approximating the Permanent", *SIAM Journal on Computing* **18(6)**, 1149 (1989)

[7] M. Jerrum, A. Sinclair and E. Vigoda, "A Polynomial-Time Approximation Algorithm for the Permanent of a Matrix with Nonnegative Entries", *Journal of the ACM* **15(4)**,671 (2004)

[8] R. Kaye and R. Wilson, *Linear Algebra* (Oxford, Oxford University Press, 1998)

[9] M. Mitzenmacher and E. Upfal, *Probability and Computing* (Cambridge, Cambridge University Press, 2005)

[10] R. Motwani and P. Raghavan, *Randomized Algorithms* (Cambridge, Cambridge University Press, 1995)

[11] L.G. Valiant, "The Complexity of Computing the Permanent", *Theoretical Computer Science* **8**,189 (1974)