Exact Sampling and Approximate Counting Techniques

Mark Huber*
School of Operations Research and Industrial Engineering
Cornell University
Ithaca, NY 14850
mhuber@orie.cornell.edu

Abstract

We present two algorithms for uniformly sampling from the proper colorings of a graph using k colors. We use exact sampling from the stationary distribution of a Markov chain with states that are the k-colorings of a graph with maximum degree Δ. As opposed to approximate sampling algorithms based on rapid mixing, these algorithms have termination criteria that allow them to stop on some inputs much more quickly than in the worst case running time bound. For the first algorithm we show that when $k > \Delta(\Delta + 2)$, the algorithm has an upper limit on the expected running time that is polynomial. For the second algorithm we show that for $k > r\Delta$, where r is an integer that satisfies $r^r > n$, the running time is polynomial. Previously, Jerrum showed that it was possible to *approximately* sample uniformly in polynomial time from the set of k-colorings when $k \geq 2\Delta$, but our algorithm is the first polynomial time *exact* sampling algorithm for this problem. Using approximate sampling, Jerrum also showed how to approximately count the number of k-colorings. We give a new procedure for approximately counting the number of k-colorings that improves the running time of the procedure of Jerrum by a factor of $(m/n)^2$ when $k \geq 2\Delta$, where n is the number of nodes in the graph to be colored and m is the number of edges. In addition, we present an improved analysis of the chain of Luby and Vigoda for exact sampling from the independent sets of a graph. Finally, we present the first polynomial time method for exactly sampling from the sink free orientations of a graph. Bubley and Dyer showed how to approximately sample from this state space in $\Theta(m^3 \ln(\epsilon^{-1}))$ time, our algorithm takes $O(m^4)$ expected time.

*This work was partially supported by an Office of Naval Research Fellowship and NSF grants CCR-9307391, CCR-9700029, and DMS-9505155, and ONR grant N00014-96-1-00500

1 Introduction

Recently a number of exciting results have appeared in the area of Monte Carlo Markov Chain (MCMC) theory. One such result is the procedure of Propp and Wilson [10] known as coupling from the past (CFTP), which allows us to sample directly from the stationary distribution of certain Markov chains without visiting each state in the chain. Many chains that arise naturally out of statistical mechanics and approximate counting problems have a number of states exponential in the size of the input. Although this makes it impossible to efficiently compute the entire stationary distribution, using CFTP we can still sample efficiently from the stationary distribution.

The state space we are primarily interested in sampling from here is the set of proper colorings of a graph $G = (V, E)$ using k colors. A proper coloring of a graph G is an assignment of colors to nodes so that no two neighboring nodes receive the same color. This state space is a special case of a framework from statistical mechanics known as the Potts model.

The ability to sample efficiently from state spaces such as the Potts model leads to better approximate counting algorithms and has applications in statistical mechanics (see [1]). The k-coloring problem is of interest in complexity theory. Jerrum, Valiant, and Vazirani [6] showed that for a class of problems which includes k-colorings that a method for efficient approximate sampling from the state space could be used to construct an efficient method for approximating the size of the state space. Counting the number of k-colorings of a graph is a $\sharp P$-complete problem, making it unlikely that an efficient algorithm will be found to solve it exactly.

In the next section we describe the Potts model in more detail, after which we present a brief description of CFTP, along with our first algorithm for exact sampling from the k-colorings of a graph. In section 5 we present our exact k-coloring sampling algorithm, which is the first to run in polynomial time. We then present a second exact sampling algorithm for the k-coloring chain that uses both CFTP and rejection sampling, and has polynomial running time. We then briefly discuss the extension of these methods to the gen-

eral Potts model. We then present an algorithm for approximately counting the number of k-colorings that improves upon the running time of the previous method (due to Jerrum) by a factor of $O(m^2/n^2)$. Finally we present the first polynomial time method for exact sampling from the sink free orientations of a graph. This algorithm has a longer running time than the previous approximate sampling algorithm due to Bubley and Dyer [2].

We use the total variation distance to quantify what we mean by approximate and exact sampling. If the distributions p and π put probability mass on a finite set, the total variation distance between them is

$$|p - \pi|_{TV} = \frac{1}{2} \sum_x |p(x) - \pi(x)|.$$

In approximate sampling, the goal is to prove that the total variation distance between the algorithm's distribution and the desired distribution is smaller than some fixed $\epsilon > 0$. In exact sampling, the total variation distance is zero, that is, we are exactly sampling from the desired distribution.

Many algorithms in this area are based on sampling from the stationary distribution of a Markov chain, and for convenience we measure the running times of these algorithms by the number of steps that need to be taken in a Markov chain. Let $P_{x,t}$ be the probability distribution of a particle on a Markov chain that started at state x and ran for t steps, and let $\tau_x(\epsilon)$ be the smallest t for which $|P_{x,t} - \pi|_{TV} \leq \epsilon$. The mixing time of the chain is $\tau(\epsilon) = \max_x \tau_x(\epsilon)$. We will say that a chain is rapidly mixing if $\tau(\epsilon)$ is bounded above by a polynomial in n and $\ln(1/\epsilon)$, where n is a variable that parameterizes the size of the chain.

Previously, Jerrum [5] exhibited a chain for k-colorings that was rapidly mixing provided that $k \geq 2\Delta$, where Δ is the maximum degree of the graph G that we are coloring. His algorithm took $O\left(\frac{k}{k-2\Delta} \ln\left(\frac{1}{\epsilon}\right) n \ln(n)\right)$ steps in the Markov chain to ensure that the total variation distance was below ϵ. We present an algorithm for which we can show the following.

Theorem 1 *Suppose that $k > \Delta(\Delta + 2)$. Then our first algorithm is an exact sampling algorithm for which the running time is a random variable T that satisfies*

$$E[T] \leq 8 \left(\frac{k - \Delta}{k - \Delta(\Delta + 2)}\right) n \ln(n).$$

and $P(T > \delta E[T]) < (1/4)^\delta$.

Our second algorithm has a running time that is better than the first when $\Delta^\Delta > n$.

Theorem 2 *Suppose that $k > r\Delta$, where r is an integer. Our second algorithm is also an exact sampling algorithm for which the running time is a random variable T satisfying $P(T > \delta E[T]) < (1/4)^\delta$, where*

$$E[T] \leq \frac{k}{k - \Delta} n \ln(n) \left(1 + \left(\frac{\Delta}{k}\right)^{(r-2)}\right)^n.$$

These algorithms are weaker than Jerrum's in that they require more colors to run in polynomial time. However, they are exact sampling algorithms whereas Jerrum's method only generates an approximate sample. Moreover, these algorithms might finish running before the bounds given on the the running time would indicate. In the algorithms which rely on rapid mixing, the algorithm must always take the same worst case amount of steps in the Markov chain. In algorithms like ours, which are based on CFTP, we have termination criteria that allow us to end the algorithm before the worst case analysis would indicate.

In addition, our algorithms are exact samplers, and so the running time does not depend on ϵ, making them faster than Jerrum's method by a factor of $\ln(1/\epsilon)$. Unlike Jerrum's method, however, the running time of these algorithms is random, and to ensure that the algorithm terminates with a probability of at least $1 - \delta$, it is necessary to run for an extra factor of $\ln(1/\delta)$ time. Note that ϵ, which bounds the total variation, will usually be much smaller than δ which bounds the probability our algorithm does not complete on schedule. Since these sampling algorithms are often run many times (for example in the counting applications) the running time is often even more closely concentrated around the expected running time.

In the chains we consider here, there is a color set C, a vertex set V, and the state space of the Markov chain is $\Omega \subseteq C^V$. For example, in the k-colorings of a graph, V is the vertex set of our graph $G = (V, E)$, and Ω is the set of proper colorings of the nodes of V. Some examples of chains in this class include the hard core gas model and the sink free orientations of a graph, which are discussed in section 2.

2 Models

The problem of sampling from the k-colorings of a graph is a special case of the Potts model. The Potts model, developed in 1952 [9], is a framework where we are given k colors with which to color an underlying graph $G = (V, E)$, either properly or improperly. For simplicity, here we will only discuss the Potts model with no external field. Let x be such a coloring, and let S denote the number of edges in the graph whose endpoints are colored by x with the same color. The Hamiltonian of a state x, denoted $H(x)$, is defined as $H(x) = -JS$, where J is a constant. Then we wish to sample from the Gibbs distribution where the relative probability of state x is $\exp(-\beta H(x))$, where β is a positive constant known as the inverse temperature. The Potts model, like all the models we will discuss, comes from statistical mechanics where it is used to model material behavior (see [12]).

If the parameter J is negative, then higher probability is assigned to states x where the endpoints of edges are colored the same way. This is often referred to as the ferromagnetic Potts model, and algorithms for exact sampling from this distribution are known [10]. If J is positive, however, then states x where endpoints of edges are identically colored have low probability. This is known as the antiferro-

magnetic Potts model.

The parameter β is inversely proportional to the temperature, so as $\beta \to \infty$ the temperature goes to zero. As the temperature decreases, the antiferromagnetic Potts model is more likely to select a proper coloring of the graph, where no two adjacent nodes are given the same color. Therefore it is customary to consider the problem of uniformly sampling from the proper k-colorings of a graph as the antiferromagnetic Potts model with zero temperature.

2.1 The Ising Model

The Potts model is a generalization of an earlier model called the Ising Model. In the Ising Model the numbers of colors is 2. In this special case our algorithm reduces to algorithms previously developed by Propp and Wilson [10] (in the ferromagnetic case) and Häggström and Nelander [3] (in the antiferromagnetic case).

In combinatorial terms the Ising Model may be thought of as sampling from weighted cuts of a graph. When the temperature is low, with high probability the sampled cut will be the maximum cut in the graph. Finding the maximum cut in a graph is an NP-complete problem, and so in general it will not be possible to efficiently sample when the temperature drops below a threshold value. In section 6.1 we show bounds on the temperature which guarantee that the algorithm runs in polynomial time.

2.2 The Hard Core Gas Model.

In this model we have a graph $G = (V, E)$, which is usually a lattice. The color set $C = \{0, 1\}$. A node colored 1 indicates that a gas molecule occupies that node, and a node colored 0 indicates that that node is empty. The state space consists of all colorings (placement of gas molecules) such that no two adjacent nodes are colored 1, that is, no two gas molecules are next to one another. This is equivalent to the state space being the independent sets of the graph, where a node is colored 1 if it is part of the independent set and colored 0 otherwise.

Exact sampling algorithms have been given for this problem by Häggstöm and Nelander [3] and more recently by Luby and Vigoda [7]. In section 5 we bound the running time of the procedure of Häggstöm and Nelander and give an improved analysis of the method of Luby and Vigoda.

2.3 Sink Free Orientations of a Graph.

An orientation of an undirected graph $G = (N, E)$ is an assignment of a direction to each edge in the graph. A sink free orientation is an orientation in which no node in N has outdegree 0. This can fit our framework by considering the "vertex set" V to be the set of edges in G and the color set to be $\{0, 1\}$, where 0 indicates one direction for the edge and 1 indicates the other. Then as before the state space is a subset of C^E.

Bubley and Dyer proved a chain for this problem was rapidly mixing using path coupling [2]. In section 8 we present the first polynomial time algorithm for sampling exactly uniformly from this state space.

3 Coupling From the Past

The procedure that we use is based on coupling from the past (CFTP), a technique developed by Propp and Wilson [10] that gives an exact random sample from a Markov chain. In CFTP, we assume that a particle has been running on the Markov chain since time $-\infty$. We are concerned with the location of the particle at time 0. The particle has been moving on the chain for all time, and so intuitively one would believe that the particle at time 0 is distributed according to the stationary distribution. In fact Propp and Wilson were able to show that this is true.

The idea behind CFTP is to start at time $-T$ with a separate particle for each possible state in the chain. Then run the chain forward in time as a coupling process. That is, if two particles collide in the chain, from then on they move together as one particle. We will refer to such a collision as a coalescence. If all the particles coalesce into one particle by time 0, then Propp and Wilson showed that the coalesced particle has the stationary distribution. [10].

To make CFTP work for a specific Markov chain, two issues need to be addressed. Since the number of states of the Markov chain may be exponentially large in the input, we cannot track all of the particles to see when coalescence occurs. We need a method to efficiently determine when only one particle remains so that we know when to stop the algorithm. In addition, we would like a polynomial upper bound on the expected time needed for all the particles to coalesce, so that we have a provably efficient algorithm.

3.1 Our Framework.

We will be considering Markov chains with a state space satisfying $\Omega \subseteq C^V$, where C is a color set and V is a vertex set. Our approach to showing coalescence will be to keep track of what information we know about the particle in the chain. At the beginning, the particle may be any one of the set of proper colorings, and so for each vertex we do not know what color the vertex is. We can record this information (or lack of information) by saying that the set of possible colors for each vertex is C.

As we move in the Markov chain, we gain information about the set of possible colors at a vertex. Suppose a step in the chain changes a particular vertex v to either blue or red. The color of v is still indeterminate, but there are fewer possibilities for the color of v than at the start of the process. If the set of possible colors for v is only a single color, say {green}, then we know that vertex v is colored green. If all of the vertices in the graph have only a single possible color, then all of the particles have coalesced into a single particle and we are done.

Let $\mathcal{M}_1$ be the Markov chain with state space $\Omega \subseteq C^V$, and let X denote a particle on this chain. We now desire a second chain whose states record the set of possible colors

for vertices of the graph. We will call such a chain a bounding chain. This bounding chain $\mathcal{M}_2$ will have state space $\Omega' = (2^C)^V$, that is, at each vertex v we assign a set of possible colors. We create a particle W on this chain $\mathcal{M}_2$ that satisfies $X(v) \in W(v)$ for all v. To ensure that this is true even when we know nothing about X, we start W at state C^V.

We will run particle X on $\mathcal{M}_1$ and particle W on $\mathcal{M}_2$ simultaneously. The particle X is unknown until the algorithm terminates, but we know the state of W at all times. If $|W(v)| = 1$ for all v, then knowing W will enable us to determine X. (In physics, the entropy of a system is the logarithm of the number of states the particle may be in. In our case, the entropy is $\sum_v \ln(|W(v)|)$ and we wish for the entropy to go to 0, indicating that only one state is possible for the particle.)

Define the set D to be $\{v : |W(v)| > 1\}$. D is the set of vertices where X is not determined by W. Let $A = V \setminus D$, so A is the set of vertices where W determines exactly what X is. The algorithm terminates when $|A| = n$.

3.2 Monotonicity and Antimonotonicity.

Propp and Wilson observed that CFTP could always be applied to the class of monotone Markov chains, and Häggström and Nelander extended their results to the case of antimonotone Markov chains. Suppose we denote our color set by $C = \{1, \ldots, n\}$. Without going into the definitions of monotone and antimonotone, we remark that both Propp and Wilson and Häggström and Nelander's results for chains on C^V may be regarded as using a bounding chain where each $W(v)$ is an interval $\{L(v), \ldots, U(v)\}$.

Unfortunately neither the k-coloring chain (with $k \geq 4$) nor the sink free orientations chain are monotone or antimonotone, so these previous methods do not apply.

4 The Bounding Chain for k-colorings

Our bounding chain will give information about a particle moving according to the heat bath Markov chain of Salas and Sokal [11] that has as its state space the proper k-colorings of a graph G. In one step of the heat bath chain, we choose a vertex uniformly at random and choose a color for the vertex uniformly at random from those which make a proper coloring. The vertex is then changed to the new color. This chain is symmetric and so has a stationary distribution that is uniform over all k-colorings.

Jerrum showed that a different k-coloring chain known as the Glauber dynamics chain is rapidly mixing. Salas and Sokal showed independently of Jerrum that both the Glauber dynamics chain and the heat bath chain were rapidly mixing for $k \geq 2\Delta$ using a technique known as Dobrushin uniqueness [11].

We now present our method of determining when coalescence at time 0 has occurred for the heat bath chain. Let $\mathcal{M}_1$ be the heat bath chain. We now describe our bounding chain $\mathcal{M}_2$. An alternate way of viewing the heat bath chain

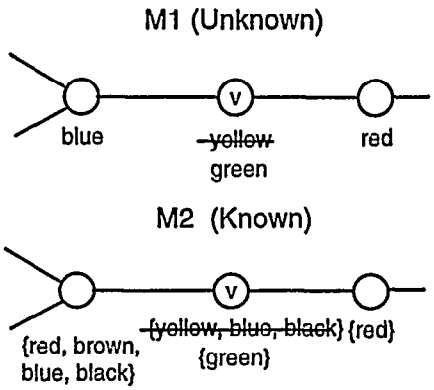

Figure 1: One Step in M_1 and M_2

is the following. Select a vertex uniformly at random. Then randomly choose colors from C without repetition until we get a color that is not blocked, that is, until we choose a color such that no neighbor of V has that same color. Then switch V to that color.

This equivalent view of the heat bath chain is how we find $s(W)$ from W. First select a vertex v uniformly at random. We set $s(W)(V \setminus v) = W(V \setminus v)$, and so we are only concerned with the value of $s(W)$ at v. We know the color of some of the neighbors of v, those neighbors which lie in A. Let $B = \{c : \{v, w\} \in E, W(w) = \{c\}\}$, that is, B is the set of colors that we know for sure we cannot choose for v. Let $F = \{c : \{v, w\} \in E, c \in W(w)\}$, that is, F is the set of colors that might possibly be blocked for v. Now from the colors in $C \setminus B$, choose colors uniformly at random until we either get a color in $C \setminus F$ or we have chosen $\Delta + 1$ colors. Let $s(W)(v)$ be the set of colors chosen.

There is guaranteed to be at least color is $s(W)(v)$ that is not blocking, since we selected colors until either we had $\Delta + 1$ different colors or the color selected was in $C \setminus F$. Let $s(X)(v)$ be the *first* color selected that was not blocking. Then we have maintained that for each step of the Markov chain, $X(v) \in W(v) \Rightarrow s(X)(v) \in s(W)(v)$ for all v. Figure 4 shows one step for $\mathcal{M}_1$ and $\mathcal{M}_2$.

5 Running Time of Algorithm

For the algorithm to terminate, we desire $|A| = n$, that is, $|W(v)| = 1$ for all v. Let A_t be A after t time steps. Then $|A_t|$ is a random walk on the numbers $\{0, \ldots, n\}$, with 0 being a reflecting barrier (since we always know the color of at least 0 nodes) and n being an absorbing barrier (once all the colors are known, they stay known). We wish to determine how long it takes before we hit n (this event will happen in finite time with probability 1 as long as $P(|A_{t+1}| > |A_t|) > 0$ when $A_t < n$). Let w_i denote the number of times that $P(|A_t| = i$. We will develop bounds on the w_i that will allow us to bound the overall expected time until coalescence occurs.

5.1 Random Walk with Absorbing and Reflecting Barriers.

Our approach will be to show that the random walk is a submartingale and then apply several theorems from martingale theory. Recall that the random variables $X_1, X_2, \ldots$ form a submartingale if $E[X_{t+1} - X_t | X_0, \ldots X_t] \geq 0$. The random variable τ is a stopping time for the submartingale if for each n we can determine whether $\tau \geq n$.

Three results from martingale theory will be needed [8].

Lemma 1 *(Martingale Stopping Theorem) Let X_t be a submartingale and let τ be a stopping time for X_t satisfying $P(\tau < \infty) = 1$ and $E|X_\tau| < \infty$. Then $E[X_\tau] \geq E[X_0]$.*

A well known corollary of this result is Wald's Lemma, which we will utilize later.

Lemma 2 *(Wald's Lemma) Let $X_1, X_2, \ldots$ be independent, identically distributed random variables with stopping time τ satisfying $E[\tau] < \infty$. Then*

$$E\left[\sum_{t=1}^{\tau} X_t\right] = E[\tau]E[X_1].$$

For any real x, let $x^+ = \max\{x, 0\}$.

Lemma 3 *(Upcrossing Inequality) Suppose $X_1, \ldots, X_n$ is a submartingale and let $U(\alpha, \beta)$ be the number of times that $X_i \leq \alpha \leq \beta \leq X_{i+1}$. Then*

$$EU(\alpha, \beta) \leq \frac{1}{\beta - \alpha}[E(X_n - \alpha)^+ - E(X_1 - \alpha, 0)^+].$$

It is easy to extend this result from fixed n to arbitrary stopping times that are finite with probability one. We are now able to state our result.

Theorem 3 *Let N_t be a random walk on $\{0, \ldots, n\}$ with n an absorbing state and 0 a reflecting state. Suppose that $N_0 = 0$, $|N_{t+1} - N_t| \leq s$, $P(N_{t+1} \neq N_t) > 0$, and w_i is the expected number of times that $N_t = i$.*

1. *If $E(N_{t+1} - N_t) \geq 0$ then*

$$w_i \leq \frac{2s(n-i)}{P(N_{t+1} \neq N_t | N_t = i)}.$$

2. *If $E(N_{t+1} - N_t) \geq \ell > 0$ for all $N_t < n$, then $\sum_{i=0}^{n} w_i \leq n/\ell$.*

Note that which conclusion of the theorem is stronger will depend on whether $P(N_{t+1} \neq N_t)$ or ℓ is smaller.

Proof. Let τ be the first time $N_t = n$. Then N_t forms a submartingale and τ is a stopping time. Also note that

$$\min_{N_t < n} P(N_{t+1} \neq N_t) > 0 \Rightarrow E(\tau) < \infty,$$

a fact that we will need later.

We first show part 2. Suppose $E(N_{t+1} - N_t) \geq \ell > 0$ for all $N_t < n$. Then $N_t - \ell t$ is a martingale for $t \leq \tau$, and so we may apply the martingale stopping theorem to state that $E[N_\tau - \tau\ell] \geq E[N_0] \geq 0$. Since $N_\tau = n$ and ℓ is a constant, we have that $E[\tau] < n/\ell$ which completes the proof.

For part 1, we consider what can happen so that $N_t = i$. Either $N_{t-1} = i$ or a step was taken. Let v_i denote the expected number of times $N_t = i$ and $N_{t-1} \neq i$. Then by Wald's Lemma we have that $w_i = v_i / P(N_{t+1} \neq N_t | N_t = i)$. To determine the v_i, we note that if we took a step to get to i, then it was either an upcrossing or a downcrossing. As in the upcrossing inequality, say that N_t upcrosses (α, β) if $N_{t-1} \leq \alpha < \beta \leq N_t$, and N_t downcrosses (α, β) when $N_{t-1} \geq \beta > \alpha \geq N_t$. Since $N_0 = 0$ and $N_\tau = n$, the number of downcrossings $D(\alpha, \beta)$ that occur are bounded above by the number of upcrossings $U(\alpha, \beta)$. Using the upcrossing inequality is simple for intervals of length 1.

$$EU(i, i+1) \leq \frac{1}{1}[E(n-i) - 0] = n - i.$$

N_t is a random walk on the integers where $|N_t - N_{t-1}| \leq s$, and so

$$\begin{aligned} v_i &\leq \sum_{j=i-s}^{i-1} EU(j, j+1) + \sum_{j=i}^{i+s-1} ED(j, j+1) \\ &\leq \sum_{j=i-s}^{i+s-1} EU(j, j+1) \\ &\leq \sum_{j=i-s}^{i+s-1} n - j \\ &\leq 2s(n-i)\ . \square \end{aligned}$$

Intuitively, our theorem says that the expected time spent at high levels such as $n-1$ or $n-2$ is small since it is likely to bump into the absorption state at n after a short while. More time is spent at low levels such as 1 or 2, since the walk may spend considerable time there before moving upwards. Another thing to note is that the proof of part 2 of the theorem ignored the fact that 0 is a reflecting barrier. That is why part 1 may give a stronger bound when ℓ is very small. Part 2 of this theorem has the following well known variant, which is proved in a similar fashion.

Theorem 4 *Let N_t be a random walk on $\{0, \ldots, n\}$ with n an absorbing state and 0 a reflecting state. Suppose that $N_0 = 0$, $|N_{t+1} - N_t| \leq s$, $P(N_{t+1} \neq N_t) > 0$, and w_i is the expected number of times that $N_t = i$. If $E(N_{t+1} - N_t | N_t = i) \geq \ell_i > 0$ for all $N_t < n$, then $\sum_{i=0}^{n} w_i \leq \sum_{i=0}^{n} 1/\ell_i$.*

5.2 Application to the Hard Core Gas Model.

The results of the preceding section can be used to analyze algorithms for exact sampling from the hard core gas model (see section 2.2), one due to Häggström and Nelander [3], and the other due to Luby and Vigoda [7]. Häggström and

Nelander did not analyze their method's running time in [3], but our results can be used to show that the method is polynomial when $\lambda \leq 1/(\Delta - 1)$. The bound on the running time of the method of Luby and Vigoda is somewhat faster, so we only describe our analysis of their method. For their method: $E(N_{t+1} - N_t) \geq 0$, $P(N_{t+1} \neq N_t) = O(i/n)$, and $|N_{t+1} - N_t| \leq 2$. They then used a weaker martingale result to conclude that the running time was bounded above by $O(n^3)$. Using Thm 3, the expected running time is bounded above by $\sum_{i=0}^{n} 2 \cdot 4i/(i/n) = 8n^2$, and so in fact the running time of their algorithm is $O(n^2)$.

5.3 Analyzing our Algorithm.

Now we are ready to show that our algorithm runs quickly when the number of colors is large. Since we are using the CFTP methodology, we only need to bound the expected running time until coalescence to bound the expected running time of the algorithm. Jerrum's proof that the Glauber dynamics chain was rapidly mixing [5] will form the template for our proof.

Proof of Thm 1 We wish to find $E(|A_{t+1}| - |A_t|)$ so that we may apply Thm 3. Let W_t be our particle after t time steps, and as before let $A_t = \{v : |W_t(v)| = 1\}$, and set $D_t = V \setminus A_t$. Our chain has coalesced to a single particle when $|A_t| = n$, and so we wish to determine the probabilities that N_t increases and decreases in size at each step. We will then apply our random walk result to bound the expected running time.

Let $\delta = |A_{t+1}| - |A_t|$. For the set A_t to increase in size after one step of the chain ($\delta = 1$), we must have selected a vertex in D_t, and then picked a color that was not blocked by the neighboring nodes. For node v, let $d(v)$ be the number of neighbors of v that are in D_t. Each of the $d(v)$ nodes could be one of at most $\Delta + 1$ colors, since $|W(v)| \leq \Delta + 1$ for all $v \in D$. Let $b(v)$ be the number of colors known to be in use by neighbors of v, so $b(v) = |\{c : \{v, w\} \in E \text{ and } w \in A\}|$. The total number of colors that may be in use by neighbors of v is $d(v)(\Delta + 1) + b(v)$. The probability that v goes from D_t to A_t is the probability that a node is D_t is chosen, and then a free color is picked from C.

$$\begin{aligned} P(\delta = 1) &\geq \sum_{v \in D_t} \frac{1}{n} \cdot \frac{k - b(v) - d(v)(\Delta + 1)}{k - b(v)} \\ &\geq \sum_{v \in D_t} \frac{1}{n} \cdot \frac{k - \Delta - d(v)(\Delta + 1)}{k - \Delta} \\ &= \frac{|D_t|}{n} \left[1 - \sum_{v \in D_t} \frac{d(v)(\Delta + 1)}{k - \Delta} \right]. \end{aligned}$$

Now for the bad case, where A_t decreases in size. For this to happen, we must choose a vertex in A_t, and then we must pick a color that is blocked by a neighbor in D_t. There are $d(v)(\Delta + 1)$ such colors, and so

$$\begin{aligned} P(\delta = -1) &\leq \sum_{v \in A_t} \frac{1}{n} \cdot \frac{d(v)(\Delta + 1)}{k - b(v)} \\ &\leq \sum_{v \in A_t} \frac{1}{n} \cdot \frac{d(v)(\Delta + 1)}{k - \Delta}. \end{aligned}$$

Let

$$\ell = E(\delta) = P(\delta = 1) - P(\delta = -1).$$

Something interesting interesting happens when we perform this subtraction. The expression for $P(\delta = 1)$ contains a term like $-(\Delta + 1)/(k - \Delta) \sum_{v \in D_t} d(v)$ and $-P(\delta = -1)$ contains a term like $-(\Delta + 1)/(k - \Delta) \sum_{v \in A_t} d(v)$. Combining these terms yields the sum over $D_t \cup A_t$ which is all nodes. The sum of the $d(v)$ over all nodes is just the number of edges incident to the set D_t, which is bounded above by $|D_t|\Delta$. Hence

$$E(\delta) \geq \frac{|D_t|}{n} \cdot \frac{k - \Delta - \Delta(\Delta + 1)}{k - \Delta}.$$

This will be greater than 0 precisely when $k > \Delta(\Delta + 2)$. When this occurs, from Thm 3 we have that the expected time until $|A_t| = n$ is

$$\frac{k - \Delta}{k - 2\Delta - \Delta^2} n \ln(n).$$

Utilizing the CFTP methodology completes the algorithm and its proof. □.

6 The Second Algorithm for k-Colorings

Our second algorithm for k-colorings combines the ideas of the bounding chain and rejection sampling. It runs in $O\left(\frac{k}{k-\Delta} n \ln(n) \left(1 + \left(\frac{\Delta}{k}\right)^r\right)^n\right)$ time. Here $r = \lceil k/\Delta \rceil - 3$, so that $k > (r + 2)\Delta$. When $r^r > n$, this is an improvement over the first algorithm. Even when $r^r < n$, this provides a bound (albeit exponential) on the running time. The first algorithm is not guaranteed to complete at all when the running time is not polynomial. The trivial rejection exact sampling algorithm for the k-coloring problem would be to choose a coloring uniformly at random from all of the k^n colorings (either proper or improper) and if it is proper, keep it. This algorithm requires roughly $(1 + \Delta/k)^n$ samples before a proper coloring will be found, where each sample takes time $O(n)$.

The outline of our procedure is as follows. We will take exact samples from a state space Ω' that contains the set Ω of proper colorings. The samples from Ω' will be obtained using CFTP using $O(\frac{k}{k-\Delta} n \ln(n))$ steps of the Markov chain. We will then show that the probability that a uniform sample from Ω' lies in Ω is at least $(1 + (\Delta/k)^r)^{(-n)}$, and so the expected number of samples from Ω' that we need to take before getting a sample in Ω is $(1 + (\Delta/k)^r)^n$.

The improvement comes from a modification of our first algorithm. Instead of letting the size of $W(v)$ grow to $\Delta + 1$, now we will force $|W(v)| \leq r + 1$. The technique we will use involves adding a dummy color, which we will call white.

Unlike other colors, we will allow both endpoints of an edge to be colored white.

Our chain works as follows. Pick a node uniformly at random. With probability at most b_p change the color of the node to white. If we don't change the color of the node to white, pick a color from those colors that don't neighbor our chosen node, and change the node to that color. This chain is a slight modification of the heat bath chain. Let W_i denote those colorations that have i nodes colored white. We are interested, of course, in obtaining a sample from W_0. This chain is reversible with all states in a class W_i having same stationary probability. We derive $\pi(W_i)$ by noting:

$$\begin{aligned}
\pi(W_i)b_p\frac{n-i}{n} &\leq \pi(W_{i+1})(1-b_p)\frac{i+1}{n} \\
\pi(W_{i+1}) &\leq \pi(W_i)\frac{b_p(n-i)}{(1-b_p)(i+1)} \\
\pi(W_i) &\leq \pi(W_0)\left(\frac{b_p}{1-b_p}\right)^n\binom{n}{i}.
\end{aligned}$$

Hence

$$\begin{aligned}
\pi(W_0) &\leq \frac{\pi(W_0)}{\sum_{i=0}^n \pi(W_i)} \\
&= (1+b_p/(1-b_p))^{-n},
\end{aligned}$$

and $(1+b_p)^n$ upper bounds the expected number of samples needed before we get a sample from W_0.

We now set $b_p = (\Delta/k)^r$. We can simulate this value for b_p in the chain as follows. Choose a node uniformly at random. Choose r colors one after another uniformly at random. If all r colors chosen are blocked, then set the color to white. Let $b(v)$ denote the number of blocked colors at our chosen node v. The probability that all r colors will be blocking is $(b(v)/k)^r \leq b_p$. If we do not turn the node white, we assign it the first color out of the r we chose that makes a proper coloring for the graph.

To take one step in $\mathcal{M}_2$, we choose a node v at random, then r colors. If the first color c is not blocking, we set $W(v) = c$. Otherwise we let $W(v)$ be the set of r colors plus white. The total number of blocking colors around any one node is now not $(\Delta+1)\Delta$, but is instead just $r\Delta$.

We may now prove Thm 2. Since $\left(1+\left(\frac{\Delta}{k}\right)^r\right)^n$ samples are taken, we need only show that the time needed to take each sample is $n\ln(n)k/(k-\Delta)$. The proof is quite similar to that of Thm 1, except that now everywhere we used $\Delta+1$ as the maximum size of $W(v)$, we may use r instead (technically, the maximum size of $W(v)$ will be $r+1$, but one of those colors will be white and so the maximum number of blocking colors is r). The same procedure employed in the proof of thm 1 may be used again here, yielding

$$E(|A_{t+1}| - |A_t|) = \frac{(n-|A_t|)\Delta}{n(k-\Delta)}.$$

From our section on random walks, we have that the running time is bounded above by $\sum_{i=1}^n \frac{n}{i}\cdot\frac{k-\Delta}{k} \leq \frac{k-\Delta}{k}n\ln(n)$, which completes the proof of Thm 2. □

6.1 The Antiferromagnetic Potts Model with Positive Temperature

We have presented two algorithms for the antiferromagnetic Potts model with zero temperature. Both of these algorithms run faster when the temperature is positive. When the temperature of the Potts model is above 0, i.e., $\beta < \infty$, improper colorings are possible, they merely have a lower probability of occurring. As stated in section 1, the relative probability of an coloring with $H(x)$ edges having both endpoints sharing a color is $\exp(-J\beta H(x))$. Therefore the probability that a node moves from D to A is greater, since even if we pick a blocking color there is a small chance that the chain will accept it. If the temperature is high enough, it will run quickly even if k is small.

Specifically, the heat bath chain for positive temperature works as follows. For $\mathcal{M}_1$, pick a vertex $v \in_R V$, and choose a color $c \in C$ randomly with relative probabilities $\exp(-J\beta m(c,v))$, where $m(c,v)$ is the number of neighbors of v colored c. For $\mathcal{M}_2$, we also start by choosing $v \in_R V$, and setting $W(v) = \emptyset$. What happens next depends on the number of colors.

If $k \geq \Delta+1$, we then pick colors one at a time uniformly at random from C and add them to $W(v)$, stopping when either we have chosen $\Delta+1$ different colors, or a $(0,1)$ uniform random variable we choose along with each color lies below $\exp(-J\beta m(c,v))$.

Suppose $k \leq \Delta$. The worst that can happen is that the unknown nodes all block our chosen color. So we choose one $(0,1)$ uniform random number, U, and then assign an interval to each color c of length

$$\frac{\exp(-J\beta(m(c,v)+d(v)))}{\exp(-J\beta(m(c,v)+d(v))) + \sum_{c_1\neq c}\exp(-J\beta m(c,v))}.$$

The length of this interval is the minimum probability that color c will be chosen. If U falls into the interval associated with color c, set $W(v) = c$. Otherwise set $W(v) = C$, the whole color set.

Theorem 5 *Using these bounding chains, the algorithm run in random time T satisfying $P(T > \delta E[T]) < (1/2)^\delta$. Let $\ell = \Delta\frac{k}{1+(k-1)e^{J\beta}} + 1 - \Delta$. If $\ell \geq 0$, (equivalently, if $J\beta < \ln\left(1+\frac{k/(k-1)}{\Delta-1}\right)$), then*

$$E[T] \leq \min\left\{\frac{n\ln n}{\ell}, 2n^2e^{-\Delta}\right\}.$$

Proof. We consider the case where $k \leq \Delta$. As before, let $\delta = |A_{t+1}| - |A_t|$. In order for δ to be 1, the chosen vertex v must lie in D_t and the random $(0,1)$ uniform U must lie in one of the color intervals. Similarly for $\delta = -1$, $v \in A_t$ and U lies outside the color intervals. The intervals are disjoint, and so their total length is the sum of their individual lengths. The length of the interval for c can be lower bounded by

$$e^{-J\beta d(v)}\frac{e^{-J\beta m(c,v)}}{\sum_c e^{-J\beta m(c,v)}},$$

making $e^{-J\beta d(v)}$ a lower bound on the sum of the lengths of the intervals. Hence

$$\begin{aligned} E[\delta] &= P(\delta = 1) - P(\delta = -1) \\ &\leq \sum_{v \in D_t} \frac{1}{n} e^{-J\beta d(v)} - \left[\sum_{v \in A_t} \frac{1}{n} \left(1 - e^{-J\beta d(v)}\right) \right] \\ &\leq \sum_{v \in V} \frac{1}{n} e^{-J\beta d(v)} - \frac{n - |D_t|}{n} \\ &\leq \frac{1}{n} \left(|D_t| \Delta e^{-J\beta} + n - |D_t| \Delta - n + |D_t| \right) \\ &\leq \frac{|D_t|}{n} \left(\Delta e^{-J\beta} + 1 - \Delta \right). \end{aligned}$$

A more careful analysis shows that the sum of the lengths of intervals is bounded above by $\frac{k}{1+(k-1)\exp(J\beta d(v))}$. Also note $P(A_{t+1} \neq A_t) \geq (n - |A_t|)\exp(-\Delta)$ Use of Thm 3 completes the proof.

In the case where $k = 2$, we have Propp and Wilson's exact sampler for the Ising model [10]. This shows that their method runs in polynomial time for $J\beta \leq \ln(1 + 2/(\Delta - 1)) \leq 2/(\Delta - 1)$. Although previous results on the mixing time of Markov chains for the Ising model are known (see [14], [4], and [13]) this is, as far as we know, the first upper bound on running times for an exact sampler for this problem.

7 Counting the Number of k-colorings

Determining the number of proper k-colorings of a graph is a ♯P-complete problem, so it is unlikely that an exact method that runs in polynomial time will be discovered. However, methods for approximating the number of k-colorings exist. Jerrum, Valiant, and Vazirani [6] showed that for the class of self-reducible problems, the problem of approximately counting could be reduced to the problem of taking a polynomial number of approximate samples. In fact, they show that the number of samples needed to come up with an approximation that comes within $1 \pm \epsilon$ of the true answer with probability at least 3/4 is polynomial in the input and $1/\epsilon$. Such a method is known as a fully polynomial randomized approximation scheme, or fpras.

Jerrum [5] applied their procedure to the specific problem of counting k-colorings. He showed that when $k > 2\Delta$, that a fpras existed with running time $O(\frac{m^2}{\epsilon^2} \cdot \frac{k}{k-2\Delta} \ln\left(\frac{4\epsilon}{m}\right))$. We now present a fpras that takes fewer samples than Jerrum's algorithm by a factor of m^2/n^2

Theorem 6 *When $k > 2\Delta$, a fpras exists for counting the number of k-colorings of a graph that has running time*

$$O\left(\frac{n^2}{\epsilon^2} \cdot \frac{k}{k - 2\Delta} \ln\left(\frac{\epsilon}{n}\right)\right).$$

Proof. Our method recursively reduces the problem by removing a single node at each of n steps. At each step, we will take n^2/ϵ^2 samples, and each sample can be taken in $\frac{k}{k-\Delta}\frac{k}{k-2\Delta} \ln\left(\frac{4\epsilon}{n}\right))$ time.

Suppose we choose a node v, and then color the graph with our node deleted. Returning v to the graph, we may complete the coloring in $k - b(v)$ ways, where $b(v)$ is the number of colors used to color nodes adjacent to v. Since $1 \leq b(v) \leq \Delta$, the number of ways to color the graph is just $C_G = \sum_{i=1}^{\Delta}(k - i)C_i$, where C_i is the number of ways to color the graph without our node using only i colors on the nodes adjacent to v. Let $C_{G\setminus\{v\}}$ be the number of ways to color the graph with v deleted. Then

$$\frac{C_G}{C_{G\setminus\{v\}}} = \sum_{i=1}^{\Delta}(k - i)\frac{C_i}{C_{G\setminus\{v\}}}.$$

Since $C_i/C_{G\setminus\{v\}}$ is just the probability that a random coloring of $G \setminus \{v\}$ uses i colors on the nodes adjacent to v, we have that $\sum_{i=1}^{\Delta} k\frac{C_i}{C_{G\setminus\{v\}}} = k$ and $\sum_{i=1}^{\Delta} i\frac{C_i}{C_{G\setminus\{v\}}}$ is just the expected number of colors used on neighbors of v.

Suppose that the node set of G is $\{v_1, v_2, \ldots, v_n\}$. Let $R_i = C_{G\setminus\{v_1,\ldots,v_{i-1}\}}/C_{G\setminus\{v_1,\ldots,v_i\}}$ (using the convention that $C_{G\setminus\{v_1,\ldots,v_n\}} = 1$). Then $\prod_{i=1}^{n} R_i = C_G$, and we have reduced our problem to estimating each of the R_i values. Let b_i be the number of colors neighboring node v_i in a uniformly random coloring of $\{G \setminus \{v_1, v_2, \ldots, v_{i-1}\}\}$. By what we showed above the problem of estimating R_i is exactly the problem of estimating $k - E(b_i)$. We can estimate these values using sampling.

As with Jerrum's method, we will use approximate sampling to draw samples from the set of all proper k-colorings. The running time for his sampler is $\frac{k}{k-2\Delta} n \ln\left(\frac{n}{\epsilon'}\right)$, where the total variation distance between the distribution the sample is drawn from and the uniform distribution is ϵ'. Denote the number of samples which we take by t.

Suppose we take $t = \lceil 17m\Delta^2/((k - \Delta)\epsilon)^2 \rceil$ samples at each step i, and let $b_i^{(j)}$ be the number of blocking colors used in sample j. Let $Z_i = 1/(kt)\sum_{j=1}^{t} b_i^{(j)}$, so that our estimate for R_i will be $\hat{R}_i = k(1 - Z_i)$, and our estimate for C_G will be $\hat{C}_G = \prod_{i=1}^{n} \hat{R}_i = k^n \prod_{i=1}^{n}(1 - Z_i)$.

Since $b_i^{(j)} \leq \Delta$, we have that that

$$\begin{aligned} \mathrm{Var} Z_i &\leq \frac{1}{(kt)^2} t[\Delta^2 - (Eb_i)^2] \\ &\leq \frac{1}{t}\left(\frac{\Delta}{k}\right)^2. \end{aligned}$$

Similarly, we know that

$$E(1 - Z_i) = 1 - Eb_i^{(j)} \geq 1 - \Delta/k,$$

and so

$$\begin{aligned} \frac{\mathrm{Var}(1 - Z_i)}{[E(1 - Z_i)]^2} &\leq \frac{1}{t}\frac{\left(\frac{\Delta}{k}\right)}{(1 - \Delta/k)^2} \\ &= \frac{1}{t}\left(\frac{\Delta}{k - \Delta}\right)^2 \end{aligned}$$

$$\leq \epsilon^2/(17m).$$

Let $ERROR$ be the event where $|\hat{C}_G - E\hat{C}_G| \geq \epsilon E\hat{C}_G/2$ From Chebyshev's inequality, the independence of the Z_i's, *and the fact that* $1 + x \leq e^x$ we have that

$$\begin{aligned}
P(ERROR) &\leq \frac{4\mathrm{Var}\left(\prod_{i=1}^n (1-Z_i)\right)}{\epsilon^2 \prod_{i=1}^n E(1-Z_i)^2} \\
&= \frac{4E\left(\prod_{i=1}^n (1-Z_i)^2\right)}{\epsilon^2 \prod_{i=1}^n E(1-Z_i)^2} - \frac{1}{\epsilon^2} \\
&= \frac{4}{\epsilon^2}\left[\prod_{i=1}^n \left(1 + \frac{\mathrm{Var}(1-Z_i)}{E(1-Z_i)^2}\right) - 1\right] \\
&\leq \frac{4}{\epsilon^2}\left[\left(1 + \frac{\epsilon^2}{17m}\right)^m - 1\right] \\
&\leq \frac{4}{\epsilon^2}\left[\exp\left(\epsilon^2/17\right) - 1\right] \\
&< 1/4,
\end{aligned}$$

when $\epsilon < 1$.

Now $E\hat{C}_G = \prod_{i=1}^n E\hat{R}_i$, and $C_G = \prod_{i=1}^n R_i$. If $\frac{\epsilon}{9m}$ is the total variation, then $|E\hat{R}_i - R_i| \leq \epsilon R_i/(4n)$ for all i and $|E\hat{C}_G - C_G| \leq \epsilon C_G/3$ for $\epsilon \leq 1$. We know that $|\hat{C}_G - E\hat{C}_G| \leq \epsilon E\hat{C}_G$ with probability at least 3/4. Taken together, we have (for $\epsilon \leq 1$) that $|\hat{C}_G - C_G| \leq \epsilon C_G$ with probability at least 3/4, which completes the proof. □

The previous theorem used the approximate sampling technique of Jerrum so that we could have $k > 2\Delta$. If $k > \Delta(\Delta+2)$, we may use our exact sampling algorithm. A similar proof shows that

Theorem 7 *When $k > \Delta(\Delta+2)$, there exists a fpras for counting the number of k-colorings of a graph which runs in $O\left(\frac{n^2}{\epsilon^2} \cdot \frac{k-\Delta}{k-\Delta(\Delta+2)}\right)$ time.*

8 Sink Free Orientations of a Graph

Bubley and Dyer [2] showed using path coupling that the heat bath chain for finding a random sink free orientation of a graph is rapidly mixing. In the heat bath chain, an edge is chosen uniformly at random, and its direction is chosen uniformly from the set of acceptable directions (those that do not create a sink in the orientation). They also noted that the problem of finding a sink free orientation can be reduced to finding a satisfying assignment in a Twice-SAT problem. In Twice-SAT, each literal appears exactly twice. Regular edges have one of two orientations. Suppose that we have some edges which either have both orientations or none. Then this generalized sink free orientation problem is equivalent to Twice-SAT. The chain of Bubley and Dyer works for the generalized sink free orientation problem and hence for Twice-SAT as well. Our algorithm only applies to the regular sink free orientation problem.

The chain of Bubley and Dyer required $O(m^3 \ln(\epsilon^{-1}))$ steps to find an approximate sample with total variation distance bounded above by ϵ. Our method takes longer for large ϵ, but we note that unless $\epsilon \leq 1/2^m$, their method does not guarantee that a particular sink free orientation will be sampled with positive probability. When $\epsilon \leq 1/2^m$, their running time is comparable to our expected running time.

Theorem 8 *Suppose Bubley and Dyer's chain for approximate sampling from the sink free orientations of a graph is ergodic. Then our algorithm exactly samples from the sink free orientations of a graph, with random running time T satisfying $E[T] = O(m^4)$ and $P(T > \delta E[T]) < (1/2)^\delta$.*

We outline the proof and our algorithm simultaneously. The idea is to run the chain in two alternating phases. In Phase I the chain is run normally. Bubley and Dyer [2] showed that any two particles on the chain in Phase I will coalesce in $O(m^3)$ time with probability at least 1/2. In Phase II, the chain will be run in such a way that all of the possible orientations will coalesce down to two particles with some constant positive probability in $O(m^4)$ time. By running the chain in Phase II followed by Phase I, all the particles will have coalesced down to one particle with some constant positive probability.

Let A_t denote the edges we know the direction of at time t, and set $D_t = m \setminus A_t$. We will show that the probability that $|A_t|$ goes to m as t goes to infinity is positive. This is different from the k-coloring problem in that both 0 and m are absorbing states. That is, if ever $|A_t| = 0$, we cannot gain any information by running the chain since we can never tell if switching the direction of an edge is a valid move. Alternatively, if $|A_t| = m$, then all of the edges are known and they will stay known for all times after t.

Suppose that $|A_0| \geq 1$. As before, let $\delta = |A_{t+1}| - |A_t|$. We will show that $E[\delta] \geq 0$. Consider an edge a in A_t that is leaving node v. If any other edge is known to be leaving v, then v will not be a sink even if edge a flips direction, and so a cannot move from A_t to D_{t+1}. Similarly, if all the other edges adjacent to v are known to be entering v, then a cannot flip direction, and cannot move from A_t to D_{t+1}. The only way for a to switch from A_t to D_{t+1} is for v to be adjacent to some unknown edge b. If we select b and change its direction so that it enters v, that is an acceptable move since a leaves v. This moves b from D_t to A_{t+1}. The probability of selecting b and its good direction in this manner is equal to the probability of selecting a and its bad direction, and so if we only change these two edges we would have that $P(\delta = 1) = P(\delta = -1)$. Since we have said that a is the only known edge that leaves b, we only use the edge b in this manner once. Therefore, we have that overall $P(\delta = 1) \geq P(\delta = -1)$, and so $|A_t|$ is a submartingale.

Let S be he starting time for a run of Phase II. Let a be any edge of the chain. Now this edge has one of two directions, either 0 or 1. Partition the set of particles into two classes based on the direction of a at time S, so $X_1 = \{X | X(a) = 1 \text{ at time } S\}$ and $X_2 = \{X | X(a) = 0 \text{ at } S\}$.

Throughout Phase II, we will force the two classes to evolve independently. Particles within each class will couple, pairs from different classes will not. First choose an edge

b uniformly at random, and then pick two independent random variables uniformly from $[0,1]$. All particles in class X_1 take a step in the chain according to the first random number, and all particles in class X_2 take a step according to the second random number. If $|A_t^1|$ is the number of known particles at time t in class 1, and $|A_t^2|$ is the number of known particles in class 2, then at the start we have that $|A_t^1| = |A_t^2| = 1$, since we know the direction of edge a for each class. Each $|A_t^i|$ is a submartingale which is absorbed at m and has positive probability of increasing by 1 at each time step, hence $P(A_t^i = m) \to L \geq 1/m$ as $t \to \infty$ for $i \in \{1,2\}$.

Whenever either $|A_t^1|$ or $|A_t^2|$ equals 0, we have to start over by picking an edge a and resetting classes X_1 and X_2 so that $|A_t^1| = |A_t^2| = 1$. There is exactly one run where both hit m. To bound the time it takes for this good event to occur, we examine a related martingale $N_t = (N_t^1, N_t^2)$ where $P(N_{t+1}^i = N_t^i + 1) = P(N_{t+1}^i = N_t^i - 1) = 1/2$, and if either $N_t^1 = 0$ or $N_t^2 = 0$, $N_{t+1}^1 = N_{t+1}^2 = 1$. Finally, let m be an absorbing state for each N_t^i. Since we know that $|A_t^i|$ is a martingale with unit changes, the N_t lower bounds the value of $|A_t^i|$ if the size of A_t^i changed at every time step.

Let $GOOD(j)$ denote the event where $N_t^i = j$ reaches m before hitting 0. Then $P(GOOD(1)) = 1/m$, and the expected running time for both N_t^1 and N_t^2 to hit m is $1/m^2$. Hence the expected running time (by Wald's Lemma) is m^2 times the expected running time for one of the N_t^i to hit 0 given that we know one hits 0, plus the expected time needed for both to hit m given that they both hit m.

Knowing that N_t^1 ends at m means that at each step it is even more likely to go up. In particular, it is still a submartingale, and we can use Theorem 3 to conclude that the expected number of times $N_t^1 = i$ is at most $2(n-i)$.

Similarly, the probability that $\delta = -1$ given that it ends at 0 is greater than the old probability that $\delta = -1$, and so N_t^1 is then a supermartingale. That makes $n - N_t^1$ a submartingale, and the Upcrossing Inequality can be used once more to conclude that the expected number of times $N_t^i = j$ is at most 2. There are m steps between 0 and m, and so the total number of changes is $2m$. The difference between N_t^i and $|A_t^i|$ is that $|A_t^i|$ doesn't always change at every time step. However, as long as there exists an unknown edge, there also exists at least one edge which when selected changes the size of $|A_t^i|$. Therefore the $P(|A_t^i|)$ changes is at least $1/m$, and the expected number of time steps between each change is m. Hence the total number of steps taken in this part of the algorithm is $2m^4$. The remaining run only takes $2m^3$ steps, so the total number of steps needed is $O(m^4)$.

By then running Phase I for $O(m^3)$ more time we then coalesce the two particles from Phase I into one state, distributed according to the stationary distribution. The running time for the approximate counting algorithm follows directly from the analysis of Bubley and Dyer, substituting our sampler running time for theirs.

References

[1] R. J. Baxter. *Exactly Solved Models in Statistical Mechanics*. Academic Press, London, 1982.

[2] Russ Bubley and Martin Dyer. Path coupling, Dobrushin uniqueness, and approximate counting. School of Computer Studies Research Report Series 97.04, University of Leeds, 1997.

[3] Olle Häggström and Karin Nelander. Exact sampling from anti-monotone systems, 1997. Preprint.

[4] Salvatore Ingrassia. On the rate of convergence of the Metropolis algorithm and Gibbs sampler by geometric bounds. *The Annals of Applied Probability*, 4(2):347–389, 1994.

[5] M. Jerrum. A very simple algorithm for estimating the number of k-colourings of a low-degree graph. *SIAM Journal on Computing*, 22:1087–1116, 1995.

[6] M. Jerrum, L. Valiant, and V. Vazirani. Random generation of combinatorial structures from a uniform distribution. *Theoretical Computer Science*, 43:169–188, 1986.

[7] Michael Luby and Eric Vigoda. Approximately counting up to four (extended abstract). In *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*, pages 682–687, 1997.

[8] Sidney C. Port. *Theoretical Probability for Applications*. Wiley-Interscience, 1994.

[9] R. B. Potts. Some generalised order-disorder transformations. *Proceedings of the Cambridge Philosophical Society*, 48:106–109, 1952.

[10] James G. Propp and David B. Wilson. Exact sampling with coupled Markov chains and applications to statistical mechanics. *Random Structures and Algorithms*, 9(1 and 2):223–252, 1996.

[11] J. Salas and A. D. Sokal. Absence of phase transition for antiferromagnetic potts models via the Dobrushin uniqueness theorem. *Journal of Statistical Physics*, 86(3–4):551–579, 1997.

[12] Barry Simon. *The Statistical Mechanics of Lattice Gasses*, volume 1. Princeton University Press, 1993.

[13] Daniel W. Stroock and Boguslaw Zegarlinski. The logarithmic Sobolev inequality for discrete spin systems on a lattice. *Communications in Mathematical Physics*, 149(1):175–193, 1992.

[14] Lawrence E. Thomas. Bound on the mass gap for finite volume stochastic Ising models at low temperature. *Communications in Mathematical Physics*, 126(1):1–11, 1989.