Rapidly Mixing Markov Chains for Sampling Contingency Tables with a Constant Number of Rows *

Mary Cryan † University of Leeds

Martin Dyer† University of Leeds

Leslie Ann Goldberg‡ University of Warwick

Mark Jerrum§ University of Edinburgh

Russell Martin‡ University of Warwick

Abstract

*We consider the problem of sampling almost uniformly from the set of contingency tables with given row and column sums, when the number of rows is a constant. Cryan and Dyer [3] have recently given a fully polynomial randomized approximation scheme (*fpras*) for the related counting problem, which only employs Markov chain methods indirectly. But they leave open the question as to whether a natural Markov chain on such tables mixes rapidly. Here we answer this question in the affirmative, and hence provide a very different proof of the main result of [3]. We show that the "2×2 heat-bath" Markov chain is rapidly mixing. We prove this by considering first a heat-bath chain operating on a larger window. Using techniques developed by Morris and Sinclair [20] (see also Morris [19]) for the multidimensional knapsack problem, we show that this chain mixes rapidly. We then apply the comparison method of Diaconis and Saloff-Coste [8] to show that the 2×2 chain is rapidly mixing. As part of our analysis, we give the first proof that the 2×2 chain mixes in time polynomial in the input size when both the number of rows and the number of columns is constant.*

1 Introduction

Given two lists of positive integers, $r = (r_1, \ldots, r_m)$ and $c = (c_1, \ldots, c_n)$, an $m \times n$ matrix $[X[i, j]]$ of non-negative integers is a *contingency table* with row sums r and column sums c if $\sum_{j=1}^{n} X[i, j] = r_i$ for every row i and $\sum_{i=1}^{m} X[i, j] = c_j$ for every column j. We write $\Sigma_{r,c}$ to denote the set of all contingency tables with row sums r and column sums c. We assume that $\sum_{i=1}^{m} r_i = \sum_{j=1}^{n} c_j$ (since otherwise $\Sigma_{r,c} = \emptyset$) and denote by N the common total, called the *table sum*.

In this paper, we consider the problem of sampling contingency tables almost uniformly at random. Unfortunately, no general technique currently exists for polynomial time sampling of general contingency tables with arbitrary row and column sums. Here we consider a particular restriction, the case where the number of rows is a constant. We give an *fully-polynomial almost-uniform sampler (fpaus)* for this restriction. An *fpaus* is defined to be an algorithm which accepts an instance of the contingency tables problem together with a maximum error probability $\epsilon \in (0, 1)$, and outputs a random element of $\Sigma_{r,c}$ subject to the following two conditions: (i) the output distribution of the fpaus must lie within total variation distance ϵ of the uniform distribution; (ii) the running time of the fpaus must be bounded by a polynomial in the size of the input (in this case n and $\log N$) and in $\log \epsilon^{-1}$.

We first review recent work on approximate counting of contingency tables, and discuss the connection between approximate counting and almost-uniform sampling for contingency tables. Cryan and Dyer [3] recently gave a *fully polynomial randomized approximation scheme (fpras)* for *counting* contingency tables when the number of rows is constant. It was previously shown by Dyer et al. [12] that the problem of *exact* counting is #P-Complete, even when there are only two rows (Barvinok [1] gave a polynomial-time algorithm to exactly count contingency tables when the number of rows and the number of columns is constant). It is well-known that for all *self-reducible problems*, finding an fpras for approximate counting is equivalent to finding an fpaus for almost-uniform sampling (see Jerrum et al. [17]). However, the contingency tables problem is not known to

*Supported by the EPSRC grants "Sharper Analysis of Randomised Algorithms: a Computational Approach" and "Analysing Markov-chain based random sampling algorithms" and by the EC IST projects RAND-APX and ALCOM-IT. A full version (with proofs included) is available from `http://www.dcs.warwick.ac.uk/~leslie/papers`

† `(maryc|dyer)@comp.leeds.ac.uk`. School of Computing, University of Leeds, Leeds LS2 9JT, United Kingdom.

‡ `(leslie|martin)@dcs.warwick.ac.uk`. Department of Computer Science, University of Warwick, Coventry CV4 7AL, United Kingdom.

§ `mrj@dcs.ed.ac.uk`. Division of Informatics, University of Edinburgh, Edinburgh EH9 3JZ, United Kingdom.

be self-reducible. The existence of an fpaus for almost-uniform sampling of contingency tables does imply an fpras for approximately counting contingency tables (see for example Dyer and Greenhill [11]), but the other direction is not known to hold. Therefore the algorithm in [3] does not necessarily imply an fpaus for almost-uniform sampling, though it does imply a sampling algorithm that depends on ϵ^{-1} rather than $\log \epsilon^{-1}$. Moreover, the algorithm in [3] is a mixture of dynamic programming and volume estimation, and uses Markov chain methods only indirectly. So [3] leaves open the question as to whether the Markov chain Monte Carlo (MCMC) method can be applied *directly* to this problem. In addition to its intrinsic interest, this question has importance for two reasons. Firstly, previous research in this area has routinely adopted the MCMC approach. Secondly, the MCMC method is more convenient, and has been more widely applied, for practical applications of sampling.

In this paper we will give the first proof of rapid mixing for a natural Markov chain when the number of rows m is a constant. We first review previous work on the MCMC method for sampling contingency tables.

Contingency tables are important in applied statistics, where they are used to summarize the results of tests and surveys. The conditional volume test of Diaconis and Efron [6] is perhaps the most soundly based method for performing tests of significance in such tables. The Diaconis-Efron test provides strong motivation for the problem of efficiently choosing a contingency table with given row and column sums uniformly at random. Other applications of counting and sampling contingency tables are discussed by Diaconis and Gangolli [7]. See also Mount [21] for additional information, and De Loera and Sturmfels [5] for the current limits of exact counting methods.

With the exception of [1] and [3], most previous work on sampling contingency tables applies the MCMC method, as described in the survey of Jerrum and Sinclair [16]. This method, which has been used to solve many different sampling problems, is based on a very simple idea. Suppose that we have a Markov chain on a finite set of discrete structures Ω, defined by the transition matrix P. If the Markov chain is *ergodic*, then it will converge to a unique stationary distribution ϖ on Ω, regardless of the initial state. This gives a nice method for sampling from the distribution ϖ: starting in any state, we run the Markov chain for some "sufficiently long" number of steps. Then the final state is taken as a sample from ϖ. The key issue with using the MCMC method is determining how long the chain takes to converge to its stationary distribution.

The first explicit definition of Markov chains for uniformly sampling contingency tables apparently occurs in the papers of Diaconis and Gangolli [7] and Diaconis and Saloff-Coste [9], although it is mentioned in [7] that this chain had already been used by practitioners. A single step of the chain is generated as follows: an ordered pair of rows i_1, i_2 are chosen uniformly at random from all rows of the table, and an ordered pair of columns j_1, j_2 are chosen uniformly at random from all columns, giving a 2×2 submatrix. The entries of the 2×2 submatrix are modified as follows:

$$\begin{aligned} X'[i_1, j_1] &= X[i_1, j_1] + 1 \qquad & X'[i_1, j_2] &= X[i_1, j_2] - 1 \\ X'[i_2, j_1] &= X[i_2, j_1] - 1 \qquad & X'[i_2, j_2] &= X[i_2, j_2] + 1 \end{aligned}$$

If modifying the matrix results in a negative value for any $X'[i, j]$, the move is not carried out. Diaconis and Gangolli proved that this Markov chain is ergodic, and the stationary distribution of the chain is uniform on $\Sigma_{r,c}$. They did not attempt to bound the mixing time of the chain, but it is clear that the mixing time can never be better than pseudopolynomial in the input.

Later Diaconis and Saloff-Coste [9] considered the case when the numbers of rows and columns are both constant and proved that, in this case, their chain converges in time quadratic in the table sum. Hernek [15] considered the case when the table has two rows and proved that the same chain mixes in time polynomial in the number of columns and the table sum. Chung et al. [2] showed that a slightly modified version of the Diaconis and Saloff-Coste chain converges in time polynomial in the table sum, the number of rows and the number of columns, provided that all row and column sums are sufficiently large.

The first truly polynomial-time algorithm for sampling contingency tables was given by Dyer, Kannan and Mount [12]. They took a different approach to the sampling problem, considering $\Sigma_{r,c}$ as the set of integer points within a convex polytope. They used an existing algorithm for sampling continuously from a convex polytope, combined with a rounding procedure, to sample integer points from inside the polytope. For any input with row sums of size $\Omega(n^2 m)$ and column sums of size $\Omega(nm^2)$, their algorithm converges to the uniform distribution on $\Sigma_{r,c}$ in time polynomial in the number of rows, the number of columns, and the logarithm of the table sum. Their result was later refined by Morris [18], who showed that the result also holds when the row sums are $\Omega(n^{3/2} m \log m)$ and the column sums are $\Omega(m^{3/2} n \log n)$.

Using different techniques, Dyer and Greenhill [11] considered the problem of sampling contingency tables when the table has only two rows. They considered a natural 2×2 "heat-bath" chain. In the two-row case, a single step of the Dyer-Greenhill chain at contingency table X is performed as follows: two columns j_1, j_2 are chosen uniformly at random from all columns, giving a 2×2 submatrix (since the table only has two rows) with column sums c_{j_1}, c_{j_2} and a pair of induced row sums s_1, s_2. A 2×2 submatrix is then chosen uniformly at random from the set of all tables with the induced row sums (s_1, s_2) and column sums (c_{j_1}, c_{j_2}). The j_1th and j_2th columns of X are replaced by this new subtable. Dyer and Greenhill showed that for two-rowed tables, their chain converges to the uniform distribution on $\Sigma_{r,c}$ in time that is polynomial in the number of columns and the logarithm of the table sum.

For tables with more than two rows, a single step of the Dyer-Greenhill chain is performed as follows: two rows i_1, i_2 are chosen uniformly at random from all rows, and two columns j_1, j_2 are chosen uniformly at random from all columns. This gives a a 2×2 submatrix with a pair of induced row sums s_1, s_2 and a pair of induced column sums b_1, b_2. A 2×2 submatrix is then chosen uniformly at random from the set of all tables with the induced row sums (s_1, s_2) and induced column sums (b_1, b_2), and the 2×2 submatrix is replaced by this new subtable. We refer to this chain as $\mathcal{M}_{2\times 2}$.

Our main result is that $\mathcal{M}_{2\times 2}$ is rapidly mixing when the number of rows is constant, and therefore we extend Dyer and Greenhill's results to any constant number of rows. However, in Section 3, we first analyse a chain which randomly modifies a $m \times (2d_m + 1)$ subtable, where d_m is a constant we will describe later. For each m we will define a constant d_m which depends only on m. A step of the chain, starting from a contingency table, X, is as follows. First, $(2d_m + 1)$ columns $j_1, \ldots, j_{2d_m+1}$ are chosen uniformly at random from the columns of X. Then the induced row sums $s_1, \ldots, s_m$ are calculated and the chosen columns of X are replaced with a subtable selected uniformly at random from all tables with row sums $s_1, \ldots, s_m$ and column sums $c_{j_1}, \ldots, c_{j_{2d_m+1}}$. We will refer to this chain as $\mathcal{M}_{\mathrm{HB}}$. The fact that a step of the chain can be carried out in polynomial time follows from Pak[22].

In Section 3 we prove that $\mathcal{M}_{\mathrm{HB}}$ is rapidly mixing. We use the multicommodity flow technique of Sinclair [24] to analyse the mixing time of this $m \times (2d_m + 1)$ "heat-bath" chain. Using techniques developed by Morris and Sinclair [20] (see also Morris [19]), we are able to show that this chain mixes in time polynomial in the number of columns and the logarithm of the table sum. In Section 4 we compare this chain to the Dyer-Greenhill chain $\mathcal{M}_{2\times 2}$ and hence show that $\mathcal{M}_{2\times 2}$ is also rapidly mixing. It may be observed that no proof was previously known that the Dyer-Greenhill (or any other) chain converges in polynomial time even when the number of columns, as well as the number of rows, is constant. Establishing this fact is one step of our proof. (See Pak [22] for an approach to this problem not using MCMC.)

Theorem 6 proves that $\mathcal{M}_{\mathrm{HB}}$ is rapidly mixing. Theorem 7 bounds the mixing time of $\mathcal{M}_{2\times 2}$ in terms of the mixing time of $\mathcal{M}_{\mathrm{HB}}$. Combining the two theorems gives the main result. We note that our results provide a very different *fpras* for this problem to that of Cryan and Dyer[3].

A full version of this paper (with proofs included) is available online [4].

2 Technical Background

In this section we summarize the techniques that we will use to bound the mixing time of our heat-bath chain. Our analysis is carried out using the multicommodity flow approach of Sinclair [24] for bounding the mixing time of a Markov chain. Sinclair's result builds on some earlier work due to Diaconis and Stroock [10].

In this section, and throughout the rest of the paper, we will use $[n]$ to denote the set $\{1, \ldots, n\}$, when n is a positive integer. We will use w^i to denote the ith component of a multidimensional weight vector w.

The setting for the multicommodity approach is as follows: we have a finite set Ω of discrete structures, and a transition matrix P on the state space Ω. It is assumed that the Markov chain defined by P is *ergodic*, that is, it satisfies the properties of *irreducibility* and *aperiodicity* (see Grimmett and Stirzaker [13]). Then the Markov chain has a unique stationary distribution ϖ, that is, a unique distribution ϖ on Ω satisfying $\varpi P = \varpi$. Sinclair also assumes that the Markov chain is *reversible* with respect to its stationary distribution, that is, $\varpi(x)P(x,y) = \varpi(y)P(y,x)$ for all $x, y \in \Omega$.

For any start state x, we define the *variation distance* between the stationary distribution and a walk of length t by

$$\mathrm{V}(\varpi, P^t(x)) \quad = \quad (1/2) \sum_{y \in \Omega} |\varpi(y) - P^t(x,y)|.$$

For any $0 < \epsilon < 1$ and any start state x, let $\tau_x(\epsilon)$ be defined as

$$\tau_x(\epsilon) \quad = \quad \min\{t : \mathrm{V}(\varpi, P^t(x)) \le \epsilon\}.$$

The *mixing time* of the chain is given by the function $\tau(\epsilon)$, defined as $\tau(\epsilon) = \max\{\tau_x(\epsilon) : x \in \Omega\}$.

The multicommodity flow approach is defined in terms of a graph G_Ω defined by the Markov chain. The vertices of G_Ω are the elements of Ω, and the graph contains an edge $(u \to v)$ for every pair of states such that $P(u,v) > 0$.

For any $x, y \in \Omega$, a *unit flow* from x to y is a set $\mathcal{P}_{x,y}$ of simple directed paths of G_Ω from x to y, such that (i) each path $p \in \mathcal{P}_{x,y}$ has a positive weight α_p, and (ii) the sum of the α_p over $p \in \mathcal{P}_{x,y}$ is 1. A *multicommodity flow* is a family of unit flows $\mathcal{F} = \{\mathcal{P}_{x,y} : x, y \in \Omega\}$ containing a unit flow for every pair of states from Ω. The important properties of a multicommodity flow are the maximum flow passing through any edge and the maximum length of a path in the flow. We define the length $\mathcal{L}(\mathcal{F})$ of the multicommodity flow $\mathcal{F}$ by

$$\mathcal{L}(\mathcal{F}) \quad = \quad \max_{x,y} \max\{|p| : p \in \mathcal{P}_{x,y}\},$$

where $|p|$ denotes the length of p. For any edge e of G_Ω, we define $\mathcal{F}(e)$ to be the sum of the α_p weights over all p such that $e \in p$ and $p \in \mathcal{P}_{x,y}$ for some $x, y \in \Omega$.

The following theorem is an amalgamation of the results of Sinclair [24]:

Theorem 1 (Sinclair [24]) *Let P be the transition matrix of an ergodic, reversible Markov chain on Ω whose stationary distribution is the uniform distribution. Let $\mathcal{F}$ be a multicommodity flow on the graph G_Ω. Then the mixing time of the chain is bounded above by*

$$\tau(\epsilon) \leq 2|\Omega|^{-1}\mathcal{L}(\mathcal{F}) \max_e \tfrac{\mathcal{F}(e)}{P(e)}(\log|\Omega| + \log \epsilon^{-1}) \qquad (1)$$

Two key ingredients of our analysis of the large heat-bath chain in Section 3 are the "balanced almost-uniform permutations" and the "strongly balanced permutations" used by Morris and Sinclair [20, 19] for the analysis of the multidimensional knapsack problem. These balanced permutations were devised by Morris and Sinclair [20, 19] for arranging multidimensional weights. A balanced permutation of a list of multidimensional weights is any arrangement of the weights so that the total of every prefix of length k is "close" to $k\mu$, where μ is the multidimensional mean of the weights. The particular types of balanced permutations that we will use are defined below.

Definition 2 (Morris [19] Definition 3.1)
Let $w_1, \ldots, w_n \in \mathbf{R}^d$ be any d-dimensional weights *with the d-dimensional mean μ. A permutation σ of $[n]$ is ℓ-balanced if*

$$|\sum_{j=1}^{k} w^i_{\sigma(j)} - k\mu^i| \quad \leq \quad \ell M_i$$

for all $i \in [d]$, $k \in [n]$, where $M_i = \max_{1\leq j\leq n} |w^i_j - \mu^i|$.

Definition 3 (Morris [19] Definition 3.3)
Let $w_1, \ldots, w_n \in \mathbf{R}^d$ be any d-dimensional weights with the d-dimensional mean μ. A permutation σ of $[n]$ is strongly ℓ-balanced if for all $k \in [n]$ and all $i \in [d]$, there exists a set $S \subseteq [n]$ with $|S \oplus [k]| < \ell$ such that $(\sum_{j=1}^{k} w^i_{\sigma(j)} - k\mu^i)$ and $(\sum_{j\in S} w^i_{\sigma(j)} - k\mu^i)$ have opposite signs (or either is 0).

The concept of a balanced permutation is closely related to the concept of a strongly balanced permutation, but the strongly balanced permutation has an extra property: to change the sign of $\sum_{j=1}^{k} w^i_{\sigma(j)} - k\mu^i$, we can achieve this by adding and deleting a constant number of weights. However, for ℓ-balanced permutations (not necessarily strongly balanced permutations) Morris and Sinclair [20] (see also Morris [19]) were able to construct random permutations which are ℓ-balanced and which are also closely related to uniform random permutations. Their construction of "balanced almost-uniform permutations" is as follows.

Theorem 4 (Morris [19] Theorem 3.2) *For every positive integer d, there exists a constant g_d and a polynomial function p_d such that for any set of weights $\{w_j\}_{j=1}^n$ in $\mathbf{R}^d$, there exists a g_d-balanced, $p_d(n)$-uniform permutation.*

The key points to keep in mind are (1) the distribution of the g_d-balanced permutation σ is closely related to the uniform distribution (the prefix probabilities for every $\sigma\{1, \ldots, k\}$ are not too large) (2) the permutations satisfy the balance property of Definition 2. Morris and Sinclair [20, 19] also adapted a result of Steinitz [25] (see also Grinberg and Sevast'yanov [14]) to show that

Theorem 5 (Morris [19] Lemma 3.4) *For any sequence $\{w_j\}_{j=1}^n$ in $\mathbf{R}^d$, there exists a strongly $16d^2$-balanced permutation.*

We will use an interleaving of a balanced almost-uniform permutation and a strongly balanced permutation to spread flow between each pair of states $x, y \in \Sigma_{r,c}$. The interleaving will allow us to construct a permutation π which is strongly balanced and which also has some of the random properties of the p_d-uniform permutation of Theorem 5.

The multidimensional weights that we consider will correspond to the columns of a contingency table, where the multiple dimensions come from having multiple rows.

The main idea is this: Given x and y we will use the permutation π of the columns of the table to define a path of contingency tables from x to y. We will route flow from x to y along this path. The amount of flow routed along the path corresponding to π will be proportional to the probability with which π is generated. We will use the following notation. If π is a permutation of the n columns of a contingency table, $\pi(j)$ will denote the original column (in $[1, \ldots, n]$) which is the jth column to be altered on the path from x to y. When column j is altered on the path from x to y, we will roughly think of column j of the current table as being replaced by its value in y. This is not as straightforward as it might appear, and more details are given in Section 3. The expression $\pi\{1, \ldots, k\}$ will denote the first k columns to be altered on the path from x to y.

3 Analysis of the generalized chain

Let $r = (r_1, \ldots, r_m)$ be a list of row sums and $c = (c_1, \ldots, c_n)$ a list of column sums. Let the state space Ω be $\Sigma_{r,c}$. Recall that N is the table sum $\sum_{i=1}^{m} r_i$.

Let g_m be the constant of Theorem 4 for balanced almost-uniform permutations for dimension m. Let $d_m = 2m(3g_m + 1) + 1 + 34m^2$. Let $\mathcal{M}_{\mathrm{HB}}$ be the heat-bath Markov-chain with window-size $m \times (2d_m + 1)$ which was introduced at the end of Section 1. Let P_{HB} be the transition matrix of this chain. In this section, we prove the following theorem.

Theorem 6 *The mixing time τ_{HB} of $\mathcal{M}_{\mathrm{HB}}$ is bounded from above by a polynomial in n, $\log N$ and $\log \epsilon^{-1}$.*

In order to prove Theorem 6, we will show how to define a multicommodity flow $\mathcal{F}$ such that the total flow along any transition (ω, ω'') is at most $2fn^{2d_m+1}P_{\mathrm{HB}}(\omega, \omega'')$, where f is an expression that is at most $\mathrm{poly}(n)|\Omega|$. We will ensure that $\mathcal{L}(\mathcal{F})$ is bounded from above by a polynomial in n. Theorem 6 will then follow from (1). First, we will define a multicommodity flow $\mathcal{F}^*$ in which the total flow through any state ω is at most f. We will construct $\mathcal{F}$ by modifying $\mathcal{F}^*$. The construction of $\mathcal{F}^*$ uses the method of Morris and Sinclair [20, 19].

Let k be the index of the largest column sum c_k. Let X and Y be contingency tables in Ω. Let X_j denote the jth column of X. We will show how to route a unit of flow from X to Y.

The rough idea is as follows. We first define the notion of a *column constrained table*, which is a set of n columns which have the correct column sums for $\Sigma_{r,c}$, but may violate the row sum constraints. We will choose a permutation π from an appropriate distribution. π will be a permutation of most of the columns of the table. The permutation π will define a path $X = Z_0, \ldots, Z_{n'}$ (for some $n' < n$) of column constrained tables, where each table Z_h contains the column Y_j for $j \in \pi\{1, \ldots, h\}$ and X_j for all other j (so at each point, we swap another column of X for the same column of Y). In Step 1 we show that the balance properties of π ensure that for any Z_h, we can bring all the row sums of Z_h below r_i by deleting a constant number of columns. Then in Step 2, we show how to use this fact to define a path $X = Z'_0, \ldots, Z'_{n'+1} = Y$ where each Z'_h is in Ω and there is a transition in $\mathcal{M}_{\mathrm{HB}}$ from each Z'_h to Z'_{h+1}. The amount of flow that we route along this path will be proportional to the probability with which π is chosen.

Let R_i^X be the set of indices for the $3g_m + 1$ largest entries of row i of X. Let R_i^Y be the set of indices for the $3g_m + 1$ largest entries of row i of Y. Let R be the union of all the R_i^X and R_i^Y sets together with the index k. The cardinality of R is at most $2m(3g_m + 1) + 1$. The columns in R are "reserved" columns that we identify before permuting the columns. We will not permute these columns — we need them for something else. For every row i, $M_i = \min\{\max\{X[i,j] : j \notin R\}, \max\{Y[i,j] : j \notin R\}\}$. Define $L_i = \{j : j \notin R, X[i,j] > M_i\} \cup \{j : j \notin R, Y[i,j] > M_i\}$. $L = \cup_{i=1}^m L_i$. $S = [n] - (L \cup R)$. For every column $j \in [n] - R$, define the m-dimensional weight $w_j = Y_j - X_j$. Let μ be the m-dimensional vector representing the mean of the $w_{j \in [n]-R}$. Note that $\mu^i = (\sum_{j \in R} X[i,j] - \sum_{j \in R} Y[i,j])/(n - |R|)$.

Let π_1 be a strongly $16m^2$-balanced permutation on the set of weights $\{w_j\}_{j \in L}$. Let π_2 be a g_m-balanced $p_m(|S|)$-uniform permutation on $\{w_j\}_{j \in S}$. π_2 is a random variable. Interlacing π_1 and π_2 in the same way as Morris [19], we get a permutation π on $\{w_j\}_{j \in [n]-R}$ satisfying inequalities (3.8) and (3.9) of Morris on page 35. That is, for every prefix h (h is the index of a column), every dimension i (i is the index of a row), we have sets of column indices $V_{i,h}$ and $W_{i,h}$ such that $V_{i,h}$ differs from $\{1, \ldots, h\}$ by at most $17m^2$ indices and $W_{i,h}$ differs from $\{1, \ldots, h\}$ by at most $17m^2$ indices and

$$\sum_{j \in V_{i,h}} w^i_{\pi(j)} \leq (h-1)\mu^i + 3g_m M_i \qquad (2)$$

$$\sum_{j \in W_{i,h}} w^i_{\pi(j)} \geq (h-1)\mu^i - 3g_m M_i \qquad (3)$$

Let $n' = n - |R|$. We define the path of tables $X = Z_0, Z_1, ..., Z_{n'}$ as follows. For every h, Z_h contains the columns X_j for $j \in R \cup \pi\{h+1, \ldots, n'\}$ and columns Y_j for $j \in \pi\{1, \ldots, h\}$. $Z_{n'}$ differs from Y by at most $2m(3g_m + 1) + 1$ columns.

$Z_0, \ldots, Z_{n'}$ may not be contingency tables in $\Sigma_{r,c}$ since they may not satisfy the row constraints. Thus, we cannot use this path as the path from X to Y. Nevertheless, we can base our path on these tables. In particular, we introduce the notation $(J(X), J(Y))$ to denote a set containing columns from X and from Y: for any $J(X) \subseteq [n]$ and $J(Y) \subseteq [n]$, $(J(X), J(Y))$ contains X_j for every $j \in J(X)$ and Y_j for $j \in J(Y)$. Let $J_h(X) = R \cup \pi\{h+1, \ldots, n'\}$ and $J_h(Y) = \pi\{1, \ldots, h\}$. Therefore Z_h is the set of columns $(J_h(X), J_h(Y))$. For any set of columns $(J(X), J(Y))$, we represent the "row sum" for row i by $row^i(J(X), J(Y))$, which has the value $\sum_{j \in J(X)} X[i,j] + \sum_{j \in J(Y)} Y[i,j]$. Note that Z_h satisfies all the column sums for $\Sigma_{r,c}$ (though some rows i may have $row^i(J(X), J(Y)) \neq r_i$), so Z_h is a column constrained table.

Step 1: We show we can modify Z_h by "deleting" at most d_m columns (including all of the X_j columns for $j \in R$) to bring the row sum for every row i below $r_i(1 - 1/n)$. We also show a dual result - if we "add" at most d_m columns to Z_h this brings the row sum for every row i above $r_i(1 + 1/n)$. Let $V_{i,h}$ and $W_{i,h}$ be defined as for (2) and (3).

Let $B_{i,h} = R \cup (\pi\{1, \ldots, h\} \oplus \pi\{V_{i,h}\})$.

Let $C_{i,h} = R \cup (\pi\{1, \ldots, h\} \oplus \pi\{W_{i,h}\})$.

Finally, define

$$D_h \quad =_{def} \quad (\cup_i B_{i,h}) \cup (\cup_i C_{i,h})$$

Consider Z_h with all of the columns in D_h "deleted". This is the table

$$Z^*_h \quad =_{def} \quad (J_h(X) - D_h, J_h(Y) - D_h)$$

In the full version we show that $row^i(J_h(X) - D_h, J_h(Y) - D_h) \leq r_i(1 - 1/n)$ for all i. Also define

$$\bar{Z}^*_h \quad =_{def} \quad (J_h(X) \cup D_h, J_h(Y) \cup D_h)$$

In the full version we show that $row^i(J_h(X)\cup D_h, J_h(Y)\cup D_h) \geq r_i(1+1/n)$ for all i.

Note that $|D_h| = d_m = 2m(3g_m+1)+1+34m^2$ (as defined previously). Also D_h contains all of R, including the index k.

Step 2: Now we show how to convert Z_h into a element of Ω. We focus on the "deleted" columns D_h, and show that by changing only the entries of the columns in D_h, we can obtain a contingency table $Z'_h \in \Sigma_{r,c}$. We will also show a dual result: that if we define $\bar{Z}_h$ to be the set of columns which contains X_j for every Y_j column in Z_h and contains Y_j for every X_j column in Z_h, we can show the same result for $\bar{Z}_h$ (we can construct a $\bar{Z}'_h$ in $\Sigma_{r,c}$ by changing d_m columns).

First let $\widehat{r}_i = row^i(J_h(X) - D_h, J_h(Y) - D_h)$, the partial row sum for row i of Z_h with the D_h columns removed. Define $s_i = r_i - \widehat{r}_i$ for all i, the sum for row i of the subtable that was removed from Z_h. Let $N_h = \sum_{i=1}^m s_i = \sum_{j\in D_h} c_j$, by construction. We have two cases.

First suppose $N_h < 2(md_m)^2$. It is well-known that whenever the total of the row sums equals the total of the column sums, there is at least one contingency table satisfying these row and column sums (see Diaconis and Gangolli [7]). For this case we choose *any* set of modified values $Z'_h[i,j]$ for $j \in D_h$ such that $\sum_{i=1}^m Z'_h[i,j] = c_j$ for all $j \in D_h$ and $\sum_{j\in D_h} Z'_h[i,j] = s_i$ for all $1 \leq i \leq m$. Note that because $N_h < 2(md_m)^2$ we have $s_i < 2(md_m)^2$ for all i and therefore $r_i < 2n(md_m)^2$ for all i.

Alternatively, assume that $N_h \geq 2(md_m)^2$. As above, we are guaranteed that there is some set of $Z'_h[i,j]$ values for $j \in D_h$ that satisfy the row and column sums. But, for this case, we will need something stronger – we show we can modify the values of $Z_h[i,j]$ for the $j \in D_h$ columns in a *structured way* to obtain a subtable Z'_h satisfying the induced row sums s_i and the column sums c_j.

We already know that k is the index of the largest c_j for $j \in D_h$. Let ℓ be the index of the biggest s_i value. For every $i \neq \ell$ and every $j \in D_h - \{k\}$, we define $a_{i,j}$ in terms of the overall row sums and the column sums.

$$a_{i,j} =_{def} \lfloor \min\{r_i, c_j\}/n(d_m)^2 \rfloor$$

Since $Z_h[i,j]$ is either $X[i,j]$ or $Y[i,j]$, we know $Z_h[i,j] \leq \min\{r_i, c_j\}$. Therefore we can write

$$Z_h[i,j] = Q[i,j](a_{i,j}+1) + R[i,j]$$

for $Q[i,j], R[i,j]$ non-negative integers, $Q[i,j] < n(d_m)^2$ and $0 \leq R[i,j] \leq a_{i,j}$ (unique), for every $i \neq \ell$ and every $j \in D_h - \{k\}$. We will show that by changing only the values of the $Q[i,j]$ to new values $Q'[i,j]$, we can obtain a subtable Z'_h satisfying the row sums s and the column sums.

Our analysis in the full version will use the fact that only the $Q[i,j]$ values are changed to derive an upper bound on the number of tables Z_h correspond to a particular Z'_h. This will be necessary to bound the congestion in our multi-commodity flow.

It is well-known (see Dyer et al. [12]) that the row and column sums are satisfied by any integer matrix Z'_h which has $Z'_h[i,j] \geq 0$ for all i and also satisfies the following inequalities:

$$\sum_{i\neq\ell} Z'_h[i,j] \quad\leq\quad c_j \quad \text{for } j \in D_h - \{k\} \tag{4}$$

$$\sum_{j\in D_h-\{k\}} Z'_h[i,j] \quad\leq\quad s_i \quad \text{for } i \neq \ell \tag{5}$$

$$\sum_{i\neq\ell}\sum_{j\in D_h-\{k\}} Z'_h[i,j] \quad\geq\quad N_h - s_\ell - c_k \tag{6}$$

Now define the $Q'[i,j]$ in terms of the induced row sums and the original column sums:

$$Q'[i,j] \quad =_{def} \quad \lfloor s_i c_j / N_h(a_{ij}+1) \rfloor$$

for all $i \neq \ell$ and all $j \in D_h - \{k\}$. Let $Z'_h[i,j] = Q'[i,j](a_{i,j}+1) + R[i,j]$. Note that $Q'[i,j] \geq 0$ for all i,j. In the full version, we prove that equalities (4), (5) and (6) are satisfied.

Therefore, in parallel with our path of column constrained tables, we have a path $X = Z'_0, Z'_1, \ldots, Z'_h, \ldots, Z'_{n'}$ such that Z'_h differs from Z_h in only d_m columns and $Z'_0, Z'_1, \ldots$ are true contingency tables. We can add another step to change $Z'_{n'}$ (using one step of the Markov chain) into Y. The amount of flow from X to Y that is routed along this path will be proportional to the probability that π is chosen.

Analysis of flow: In the full version, we show that the flow through any state $Z' \in \Omega$ is at most $\mathrm{poly}(n)|\Omega|$. In the application of Morris and Sinclair [20, 19] this is already sufficient to prove polynomial time mixing, since the term $P(e)$ in the denominator of (1) is only polynomially small. However, for our heat-bath chain P_{HB}, it may be exponentially small, and further argument is required to establish rapid mixing.

To this end, let $e = (\omega, \omega')$ $(\omega, \omega' \in \Omega^2)$ be a (directed) transition of our heat-bath chain, with transition probability $P_{\mathrm{HB}}(e)$. Suppose that f_e units of flow are shipped along e in the multi-commodity flow defined above. We will disperse the flow through e by sending it from ω to ω' via a "random destination" ω''.

Let B be the set of columns on which ω and ω' disagree and let W be the set of all size $m \times (2d_m+1)$ heat-bath windows which include B. Let Ω'' be the set of all contingency tables ω'' such that

1. There is a $U \in W$ which contains all the columns on which ω and ω'' differ, and

2. There is a $U' \in W$ which contains all the columns on which ω' and ω'' differ.

For each $\omega'' \in \Omega''$, we will route $f_e/|\Omega''|$ flow from ω to ω' via ω''. Note that this construction doubles the length of our flow paths, but no more.

In the full version, we show that the total flow in the new multicommodity flow along transition (ω, ω'') is at most $2fn^{2d_m+1}P_{\mathrm{HB}}(\omega, \omega'')$. This is now sufficient for the right hand side of (1) to be polynomially bounded, since the (possibly small) $P_{\mathrm{HB}}(e)$ term cancels. This completes the proof of Theorem 6.

4 Mixing of the 2×2 chain

Theorem 6 shows that the Markov chain $\mathcal{M}_{\mathrm{HB}}$ is rapidly mixing. In this section we use the comparison method of Diaconis and Saloff-Coste [8] (see also Randall and Tetali [23], Vigoda [26]) to show that the 2×2 chain $\mathcal{M}_{2\times 2}$ is also rapidly mixing.

4.1 Setting up the comparison

We briefly describe the comparison method of Diaconis and Saloff-Coste. Suppose that we have two ergodic reversible Markov chains $\mathcal{M}$ and $\mathcal{M}'$, both of which converge to the uniform distribution on the same state space Ω. The comparison method is used whenever we already have an upper bound for the mixing time $\tau_{\mathcal{M}}$ of the chain $\mathcal{M}$, and we would like to bound the mixing time $\tau_{\mathcal{M}'}$ of $\mathcal{M}'$. We will take the same approach as Vigoda [26], and we will use a multicommodity flow approach to bound $\tau_{\mathcal{M}'}$ in terms of $\tau_{\mathcal{M}}$. Let $P_{\mathcal{M}}$ denote the transition matrix of the chain $\mathcal{M}$ and $P_{\mathcal{M}'}$ denote the transition matrix of the chain $\mathcal{M}'$. Let $E(P_{\mathcal{M}})$ be the *kernel* of the Markov chain $\mathcal{M}$, that is, $E(P_{\mathcal{M}}) = \{(x, y) : P_{\mathcal{M}}(x, y) > 0\}$. Let $E(P_{\mathcal{M}'})$ be the kernel of $\mathcal{M}'$. Let $G_{\mathcal{M}'}$ be the graph whose nodes are the elements of Ω, and whose edges are the pairs (x, y) such that $(x, y) \in E(P_{\mathcal{M}'})$.

Now suppose we define a *unit flow* $f_{x,y}$ on the graph $G_{\mathcal{M}'}$, for every pair of states $(x, y) \in E(P_{\mathcal{M}})$. That is, for every $(x, y) \in E(P_{\mathcal{M}})$, we construct a set $\Gamma_{x,y}$ of simple paths from x to y in the graph $G_{\mathcal{M}'}$. For every $\gamma \in \Gamma_{x,y}$, we assign some value $f_{x,y}(\gamma) \in [0, 1]$, so that

$$\sum_{\gamma \in \Gamma_{x,y}} f_{x,y}(\gamma) = 1. \tag{7}$$

For the comparison method, the important quantities are the quantities $A_{z,z'}$, defined for every $(z, z') \in E(P_{\mathcal{M}'})$ by

$$A_{z,z'} = \sum_{(x,y) \in E(P_{\mathcal{M}})} \sum_{\substack{\gamma \in \Gamma_{x,y} \text{ such} \\ \text{that } (z,z') \in \gamma}} |\gamma| f_{x,y}(\gamma) \frac{P_{\mathcal{M}}(x, y)}{P_{\mathcal{M}'}(z, z')},$$

where $|\gamma|$ denotes the length of the path γ.

Then the comparison theorem of Diaconis and Saloff-Coste [8] (see also Vigoda [26]), states that the mixing time $\tau_{\mathcal{M}'}(\epsilon)$ of the chain $\mathcal{M}'$ is

$$O(\tau_{\mathcal{M}}(\epsilon) \log(|\Omega|) \max_{(z,z') \in E(P_{\mathcal{M}'})} A_{z,z'}). \tag{8}$$

We will apply this theorem with $\mathcal{M}_{\mathrm{HB}}$ as the Markov chain whose mixing time is known (see Section 3) and $\mathcal{M}_{2\times 2}$ as the Markov chain whose mixing time we want to bound. For us the state space Ω is $\Sigma_{r,c}$. Recall that the transition matrix of $\mathcal{M}_{\mathrm{HB}}$ is denoted by P_{HB} and that the transition matrix of $\mathcal{M}_{2\times 2}$ is denoted by $P_{2\times 2}$. Since we represent contingency tables by X and Y, we will use (X, Y) to denote elements of $E(P_{\mathrm{HB}})$, and (Z, Z') to denote elements of $E(P_{2\times 2})$. We denote the mixing time of $\mathcal{M}_{\mathrm{HB}}$ by τ_{HB} and the mixing time of $\mathcal{M}_{2\times 2}$ by $\tau_{2\times 2}$.

In our construction of the flow, we will ensure that the length of each path $\gamma \in \Gamma_{X,Y}$ is bounded by a constant. Thus, the upper bound (8) of the theorem of Diaconis and Saloff-Coste tells us that to establish rapid mixing, we only need to concentrate on bounding $A_{Z,Z'}$ for every $(Z, Z') \in E(P_{2\times 2})$ (since $\log |\Omega| = \log |\Sigma_{r,c}| \in O(n(\log N))$ and τ_{HB} is bounded). Therefore we need only define $f_{X,Y}$ for every $(X, Y) \in E(P_{\mathrm{HB}})$ such that Equation (7) is satisfied and such that, for all $(Z, Z') \in E(P_{2\times 2})$, the following is satisfied:

$$\sum_{(X,Y) \in E(P_{\mathrm{HB}})} \sum_{\substack{\gamma \in \Gamma_{X,Y} \\ (Z,Z') \in \gamma}} f_{X,Y}(\gamma) \tfrac{P_{\mathrm{HB}}(X,Y)}{P_{2\times 2}(Z,Z')} \leq \mathrm{poly}(n). \tag{9}$$

It helps us to re-work Equation (9) before defining the flows. For $(X, Y) \in E(P_{\mathrm{HB}})$, let $\mathcal{W}(X, Y)$ be the set of all $m \times (2d_m + 1)$ "windows" such that X and Y agree outside of W, where a "window" is just a set of m rows and $2d_m + 1$ columns. Note that

$$P_{\mathrm{HB}}(X, Y) = \sum_{W \in \mathcal{W}(X,Y)} \frac{1}{\binom{n}{2d_m+1}} \frac{1}{|\Omega_X(W)|},$$

where $\Omega_X(W)$ is the set of all contingency tables that agree with X outside of W. We may view $P_{\mathrm{HB}}(X, Y)$ as an average of the quantities $1/|\Omega_X(W)|$ over all windows $W \in \mathcal{W}(X, Y)$. Therefore, we can pick some $W(X, Y) \in \mathcal{W}(X, Y)$ such that

$$P_{\mathrm{HB}}(X, Y) \leq \frac{1}{|\Omega_X(W(X, Y))|}. \tag{10}$$

The essential idea to keep in mind in what follows is that routing the unit flow $f_{X,Y}$ from X to Y is done using paths of contingency tables that differ from one another solely on (a part of) the chosen window $W(X, Y)$ satisfying (10).

For each $m \times (2d_m + 1)$ window W, let $E_W = \{(X,Y) \in P_{\mathrm{HB}} \mid W(X,Y) = W\}$. Later, when we define our flows, we do the following for every fixed window W: For every $(X,Y) \in E_W$, we define a flow $f_{X,Y}$ such that Equation (7) is satisfied. We also ensure that for all $(Z,Z') \in E(P_{2\times 2})$, the following is satisfied:

$$\sum_{(X,Y)\in E_W} \sum_{\substack{\gamma\in\Gamma_{X,Y}\\(Z,Z')\in\gamma}} f_{X,Y}(\gamma)\tfrac{P_{\mathrm{HB}}(X,Y)}{P_{2\times 2}(Z,Z')} \leq \mathrm{poly}(n). \quad (11)$$

Since there are only polynomially-many windows W, equation (11) implies equation (9), and ensures rapid mixing.

For each window W, Section 4.2 shows how to define a flow $f^*_{X,Y}$ for every $(X,Y) \in E_W$ such that

$$\sum_{\gamma\in\Gamma_{X,Y}} f^*_{X,Y}(\gamma) = 1$$

and the total flow through any contingency table $Z \in \Sigma_{r,c}$ is in $O(|\Omega_X(W)|)$. We define $f_{X,Y}$ by modifying $f^*_{X,Y}$. Let $f^*_{X,Y}(Z)$ denote the amount of flow passing through the contingency table Z in the flow $f^*_{X,Y}$. Let $f^*(Z) = \sum_{(X,Y)\in E_W} f^*_{X,Y}(Z)$. Similarly, let $f_{X,Y}(Z,Z')$ denote the amount of flow passing through the transition (Z,Z') in the flow $f_{X,Y}$. Let $f(Z,Z') = \sum_{(X,Y)\in E_W} f_{X,Y}(Z,Z')$. Our construction of $f_{X,Y}$ from $f^*_{X,Y}$ ensures that for every $(Z,Z') \in E(P_{2\times 2})$, we have

$$f(Z,Z') \leq 2f^*(Z)P_{2\times 2}(Z,Z')\binom{n}{2}\binom{m}{2}. \quad (12)$$

Thus, the left-hand-side of (11) is equal to

$$\begin{aligned} & \tfrac{f(Z,Z')}{P_{2\times 2}(Z,Z')}P_{\mathrm{HB}}(X,Y) \\ \leq\ & \tfrac{f(Z,Z')}{P_{2\times 2}(Z,Z')}\tfrac{1}{|\Omega_X(W)|} \\ \leq\ & \tfrac{2f^*(Z)\binom{n}{2}\binom{m}{2}}{|\Omega_X(W)|} \\ \leq\ & \mathrm{poly}(n), \end{aligned}$$

where the first inequality comes from Equation (10), the second comes from Equation (12) which we establish in the full version, and the third comes from the fact that $f^*(Z) \in O(|\Omega_X(W)|)$, which is established in Section 4.2. We will then have shown that Equation (11) is satisfied, as required, so the 2×2 heat bath chain is rapidly mixing on $\Sigma_{r,c}$.

By considering all $m \times (2d_m + 1)$ sized windows which contain the two columns on which Z and Z' differ, we can see that for each $(Z,Z') \in E(P_{2\times 2})$, we have

$$A_{Z,Z'} \leq C\binom{n}{2}\binom{n-2}{2d_m-1}$$

where the constant C accounts for the maximum length of any (X,Y) path for $(X,Y) \in E(P_{\mathrm{HB}})$, and the constant factors arising in the bound for the flow $f^*(Z)$ over a single $m \times (2d_m + 1)$ window W. Therefore, we have the following theorem:

Theorem 7 *The mixing time $\tau_{2\times 2}(\epsilon)$ of the Markov chain $\mathcal{M}_{2\times 2}$ is*

$$O(\tau_{HB}(\epsilon)\log(|\Sigma_{r,c}|)n^{2d_m+1}).$$

Therefore by Theorem 6, the mixing time of $\mathcal{M}_{2\times 2}$ is bounded by a polynomial in n, $\log N$ and $\log \epsilon^{-1}$.

4.2 Defining $f^*(X,Y)$

In this section we outline a method for defining a flow for every $(X,Y) \in E_W$ such that $\sum_{\gamma\in\Gamma_{X,Y}} f^*_{X,Y}(\gamma) = 1$ and the total flow through any contingency table Z, due to pairs in E_W, is in $O(|\Omega_X(W)|)$. The full details are shown in the full paper, but are omitted here due to space considerations.

Throughout the entire section, we focus on some fixed $m \times (2d_m + 1)$ sized window W of the larger $m \times n$ table. Without loss of generality (and to make our notation simpler in what follows), we assume that W includes the first $2d_m + 1$ columns of the table. This window W has induced row sums ρ_i (for $i \in [m]$) and induced column sums ζ_j (for $j \in [2d_m + 1]$). For convenience we also set $\delta = 2d_m + 1$.

Let $\rho = (\rho_1, \ldots, \rho_m)$, $\zeta = (\zeta_1, \ldots, \zeta_\delta)$ be the lists of induced row and column sums. Let $\Sigma_{\rho,\zeta}$ denote the set of $m\times\delta$ contingency tables with row sums ρ and column sums ζ, and let N_W denote the table sum. Let $\Upsilon, \Psi \in \Sigma_{\rho,\zeta}$. We show how to route a unit of flow between Υ and Ψ using a path of contingency tables that differ by 2×2 heat bath moves. This flow lifts in the obvious fashion to transitions $(X,Y) \in E(P_{\mathrm{HB}})$, giving us the flow $f^*_{X,Y}$ required in the previous section. In other words, we simply use the exact same sequence of 2×2 transitions on the window $W(X,Y)$, keeping everything outside this window fixed (where X and Y agree anyway).

If $N_W < (2m\delta)^2$ then $|\Sigma_{\rho,\zeta}| \in O(1)$, so it doesn't really matter how we route flow between Υ and Ψ. For example, it suffices to fix each square in the contingency table in lexicographic order. Each path in the resulting flow is of length $O(1)$ and there are $O(1)$ pairs (Υ, Ψ) of contingency tables, so the desired bound is easily established. Thus, from now on we assume $N_W \geq (2m\delta)^2$ and we show how to construct a flow between Υ and Ψ in $\Sigma_{\rho,\zeta}$.

Without loss of generality we may assume that the row totals are sorted into non-descending order and that the column totals are also sorted into non-descending order. Therefore ρ_m is the largest row sum and ζ_δ is the largest column sum.

As we did in section 3, we view the space $\Sigma_{\rho,\zeta}$ of contingency tables as the $(m-1)(\delta-1)$-dimensional space of integer matrices Φ that satisfy $\Phi[i,j] \geq 0$ for all $i \in [m-1]$

and all $j \in [\delta - 1]$ and also satisfy inequalities analogous to (4), (5) and (6) (see the full version for details). Let

$$\alpha_{i,j} = \left\lfloor \frac{\min\{\rho_i, \zeta_j\}}{m^2\delta^2} \right\rfloor \quad (13)$$

for all $i \in [m-1]$, $j \in [\delta - 1]$.

For any contingency table $\Phi \in \Sigma_{\rho,\zeta}$, and any $i \in [m-1]$, $j \in [\delta - 1]$, we can write

$$\Phi[i,j] = Q[i,j](\alpha_{i,j} + 1) + R[i,j],$$

for a unique integer $R[i,j]$ satisfying $0 \leq R[i,j] \leq \alpha_{i,j}$, and a unique integer $Q[i,j]$.

Let

$$Q^*[i,j] = \left\lfloor \frac{\rho_i \zeta_j}{N_W(\alpha_{i,j}+1)} \right\rfloor \quad (14)$$

for all $i \in [m-1]$, $j \in [\delta - 1]$.

If Φ is a contingency table such that $Q[i,j] = Q^*[i,j]$ for every $i \in [m-1]$, $j \in [\delta-1]$, then we say that Φ belongs to the *inner domain* of $\Sigma_{\rho,\zeta}$.

The following lemma is crucial in our method of defining the flow, because it tells us that for any original contingency table $\Phi \in \Sigma_{\rho,\zeta}$, there is a contingency table Φ^* in the inner domain which has the same set of remainders $\bmod(\alpha_{i,j} + 1)$ as Φ.

Lemma 8 *Let $\Phi^*[i,j]$ be defined by $\Phi^*[i,j] = Q^*[i,j](\alpha_{i,j} + 1) + R[i,j]$, for any non-negative integers $R[i,j] \leq \alpha_{i,j}$, for all $i \in [m-1]$, $j \in [\delta - 1]$. Then $\Phi^* \in \Sigma_{\rho,\zeta}$.*

Consider $\Upsilon, \Psi \in \Sigma_{\rho,\zeta}$. We will route flow from Υ to Ψ via two points in the inner domain.

For every $i \in [m-1]$, $j \in [\delta - 1]$, let $R_\Upsilon[i,j]$ be the unique integer such that $\Upsilon[i,j] \bmod (\alpha_{i,j}+1) = R_\Upsilon[i,j]$ and $0 \leq R_\Upsilon[i,j] \leq \alpha_{i,j}$. Let Υ^* be the point in the inner domain with the remainders $R_\Upsilon[i,j]$; that is, $\Upsilon^*[i,j] = Q^*[i,j](\alpha_{i,j}+1)+R_\Upsilon[i,j]$ for every $i \in [m-1]$, $j \in [\delta-1]$. By Lemma 8, $\Upsilon^* \in \Sigma_{\rho,\zeta}$.

For every $i \in [m-1]$, $j \in [\delta - 1]$, let $R_\Psi[i,j]$ be the unique integer such that $\Psi[i,j] \bmod (\alpha_{i,j}+1) = R_\Psi[i,j]$ and $0 \leq R_\Psi[i,j] \leq \alpha_{i,j}$. Let Ψ^* be the point in the inner domain with the remainders $R_\Psi[i,j]$; that is, $\Psi^*[i,j] = Q^*[i,j](\alpha_{i,j}+1)+R_\Psi[i,j]$ for every $i \in [m-1]$, $j \in [\delta-1]$. By Lemma 8, $\Psi^* \in \Sigma_{\rho,\zeta}$.

The routing from Υ to Ψ proceeds in two stages. In the first phase of the routing, we route flow from Υ to Υ^*. We do this routing from Υ to Υ^* by changing only four of the $Q[i,j]$ values by ± 1 (on some 2×2 subwindow) at each step. In the full version we show that we can construct a short path from Υ to Υ^* using these moves. Note that in the first phase we never change the remainders $R_\Upsilon[i,j]$.

By defining a similar path between Ψ and Ψ^* and then reversing all the edges, we can route flow from Ψ and Ψ^* in a similar way.

The length of the path between Υ and Υ^* (and hence between Ψ^* and Ψ) in phase one is shown to be at most $2(m^2\delta^2)^{m\delta}$. Also, the amount of flow passing through any contingency table $\Phi \in \Sigma_{\rho,\zeta}$ due to "phase 1" flow is at most $|\Sigma_{\rho,\zeta}|(m^2\delta^2)^{2m\delta}$.

In the second phase of the routing we route flow from Υ^* to Ψ^* by changing the $R_\Upsilon[i,j]$ to the $R_\Psi[i,j]$ values. These remainders are "fixed" in order, i.e. we construct a series of new tables by first changing $R_\Upsilon[1,1]$ to the value $R_\Psi[1,1]$. Then, from that resulting table we change the value $R_\Upsilon[1,2]$ to $R_\Psi[1,2]$. In general, we consider the lexicographic ordering $(1,1),(1,2),\ldots,(1,\delta - 1),(2,1),\ldots,(m-1,1),\ldots,(m-1,\delta-1)$. We "fix" the remainders, one at a time, in this order to define a path between Υ^* and Ψ^*. This defines a path of length at most $(m-1)(\delta-1)$ for the second phase of the routing. Lemma 8 guarantees that each state in the path is in $\Sigma_{\rho,\zeta}$. The flow through any contingency table Φ due to "phase 2" flow is shown to be bounded above by $|\Sigma_{\rho,\zeta}|(m\delta)(m^2\delta^2)^{2m\delta}$.

Now we combine phases 1 and 2, and lift it back to the set of $m \times n$ contingency tables $\Sigma_{r,c}$. Recall that we need only consider each $m \times (2d_m + 1)$ sized window W individually, where ρ and ζ represent the induced row and column sums, respectively, of this window. The length of any path between $X, Y \in \Sigma_{r,c}$ that differ only on W is at most $4(m\delta)^{2m\delta} + m\delta$. Also, the flow through any table $Z \in \Sigma_{r,c}$, due to the window W, is bounded by $|\Sigma_{\rho,\zeta}|(2(m\delta)^{4m\delta} + m\delta(m^2\delta^2)^{2m\delta})$.

This establishes the criteria for the length of paths between X and Y and for the flow $f^*_{X,Y}$ we required in section 4.1. We then modify $f^*_{X,Y}$, as was outlined in that section, to give the new flow $f_{X,Y}$ that satisfies Theorem 7.

References

[1] A.I. Barvinok, A polynomial-time algorithm for counting integral points in polyhedra when the dimension is fixed. *Mathematics of Operations Research*, **19**(4), pp. 769–779, 1994.

[2] F.K.R. Chung, R.L. Graham and S.-T. Yau, On sampling with Markov chains. *Random Structures & Algorithms*, **9**(1-2), pp. 55–77, 1996.

[3] M. Cryan and M. Dyer, A polynomial-time algorithm to approximately count contingency tables when the number of rows is constant. *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, pp. 240–249, 2002.

[4] M. Cryan, M. Dyer, L.A. Goldberg, M. Jerrum and R. Martin, Rapidly mixing Markov chains for sampling contingency tables with a constant number of rows (full version). Available online from `http://www.dcs.warwick.ac.uk/~leslie/papers/contingency.ps`

[5] J.A. De Loera and B. Sturmfels, Algebraic unimodular counting. Preprint, Department of Mathematics, University of California, Davis, 2001.

[6] P. Diaconis and B. Efron, Testing for independence in a two-way table: new interpretations of the chi-square statistic (with discussion). *Annals of Statistics*, **13**, pp. 845–913, 1995.

[7] P. Diaconis and A. Gangolli, Rectangular arrays with fixed margins, in: D. Aldous, P.P. Varaiya, J. Spencer and J.M. Steele (Eds.), *Discrete Probability and Algorithms*, IMA Volumes on Mathematics and its Applications, **72**, Springer, New York, pp. 15–41, 1995.

[8] P. Diaconis and L. Saloff-Coste, Comparison theorems for reversible Markov chains. *The Annals of Applied Probability*, **3**(3), pp. 696–730, 1993.

[9] P. Diaconis and L. Saloff-Coste, Random walk on contingency tables with fixed row and column sums. Technical Report, Department of Mathematics, Harvard University, 1995.

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

[11] M. Dyer and C. Greenhill, Polynomial-time counting and sampling of two-rowed contingency tables. *Theoretical Computer Science*, **246**, pp. 265–278, 2000.

[12] M. Dyer, R. Kannan and J. Mount, Sampling contingency tables. *Random Structures & Algorithms*, **10**(4), pp. 487–506, 1997.

[13] G.R. Grimmett and D.R. Stirzaker, *Probability and Random Processes*, Oxford University Press, Oxford, 1992.

[14] V.S. Grinberg and S.V. Sevast'yanov, Value of the Steinitz constant. *Funktsional. Anal. i Prilozhen.*, **14**(2), pp. 56–57, 1980.

[15] D. Hernek, Random generation of $2 \times n$ contingency tables. *Random Structures & Algorithms*, **13**(1), pp. 71–79, 1998.

[16] M. Jerrum and A. Sinclair, Markov chain Monte Carlo method: an approach to approximate counting and integration. D.S. Hochbaum (Ed.), *Approximation Algorithms for NP-Hard Problems*, PWS, Boston, pp. 482–520, 1997.

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

[18] B. Morris, Improved bounds for sampling contingency tables. *3rd International Workshop on Randomization and Approximation Techniques in Computer Science*, volume 1671 of *Lecture Notes in Computer Science*, pp. 121–129, 1999.

[19] B. Morris, *Random Walks in Convex Sets*. PhD thesis, Department of Statistics, University of California, Berkeley, 2000.

[20] B. Morris and A.J. Sinclair, Random walks on truncated cubes and sampling 0-1 knapsack solutions. *Proceedings of the 40th IEEE Symposium on Foundations of Computer Science*, pp. 230–240, 1999. Full version available from `http://www.cs.berkeley.edu/~sinclair/knapsack.ps`, June 2002.

[21] J. Mount, *Application of Convex Sampling to Optimization and Contingency Table Generation*. PhD thesis, Technical report CMU-CS-95-152, Computer Science Department, Carnegie Mellon University, 1995.

[22] I. Pak, On sampling integer points in polyhedra. Preprint, Department of Mathematics, Yale University, 2000.

[23] D. Randall and P. Tetali, Analyzing Glauber dynamics by comparison of Markov chains. *Journal of Mathematical Physics* **41**(3), pp. 1598–1615, 2000.

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

[25] E. Steinitz, bedingt konvergente Reihen und konvexe Systeme, *J. reine angew. Math.*, **143**, pp. 128–175, 1913.

[26] E. Vigoda, Improved bounds for sampling colorings. *Journal of Mathematical Physics* **41**(3), pp. 1555–1569, 2000.