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

Mary Cryan
School of Informatics
University of Edinburgh
Edinburgh EH9 3JZ, UK

Martin Dyer
School of Computing
University of Leeds
Leeds LS2 9JT, UK

Leslie Ann Goldberg
Department of Computer Science
University of Warwick
Coventry CV4 7AL, UK

Mark Jerrum
School of Informatics
University of Edinburgh
Edinburgh EH9 3JZ, UK

Russell Martin
Department of Computer Science
University of Liverpool
Liverpool L69 7ZF, UK

November 26, 2005

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 employs Markov chain methods indirectly. They leave open the question as to whether a natural Markov chain on such tables mixes rapidly. Here 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 [19, 20] for the multidimensional knapsack problem, we show that this chain mixes rapidly. We then apply the comparison method of Diaconis and Saloff-Coste [7] to show that the 2×2 chain is also rapidly mixing.

1 Introduction

Given two vectors 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

*This work was partially supported by the EPSRC grant "Sharper Analysis of Randomised Algorithms: a Computational Approach", the EPSRC grant GR/R44560/01 "Analysing Markov-chain based random sampling algorithms" and the IST Programme of the EU under contract numbers IST-1999-14186 (ALCOM-FT) and IST-1999-14036 (RAND-APX).

A preliminary version of this paper appeared in *Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science*, 2002, pp. 711–720.

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. No technique currently exists for polynomial-time sampling when the row and column sums can be arbitrary. In this paper we consider a particular restriction, namely, the case in which the number of rows is a constant. We focus on the Markov chain Monte Carlo (MCMC) method for sampling, which has already been successfully used to sample contingency tables, for other restrictions of the problem (see [8, 15, 2, 11]). We prove that a natural Markov chain, which we refer to as $\mathcal{M}_{2\times 2}$, is rapidly mixing when the number of rows is constant.

Before we give details of previous work on the MCMC method for sampling contingency tables, we will first discuss recent work on approximate counting of contingency tables, when the number of rows is constant. Cryan and Dyer [3] recently gave a fully polynomial randomized approximation scheme (*fpras*) for approximately counting contingency tables in this setting (i.e. for approximating $|\Sigma_{r,c}|$ with given r, c). It was previously shown by Dyer et al. [12] that the problem of *exact* counting is $\sharp P$-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* (fully polynomial almost uniform sampler) (see Jerrum et al. [17]). Contingency tables are not known to be self-reducible - it is true that the existence of an fpaus for almost-uniform sampling of contingency tables does imply an fpras for approximately counting contingency tables (see, for example, [11]), but the other direction is not known to hold. It is shown in [3] that the fpras does imply a sampling algorithm, though this algorithm depends on ϵ^{-1} rather than on $\log \epsilon^{-1}$. Recently Dyer [10] developed an elegant dynamic programming technique for contingency tables with a constant number of rows. He applied this technique to design two algorithms: an fpaus for uniformly sampling contingency tables with a constant number of rows; an fpras for approximately counting the number of contingency tables when the number of rows is constant. The running times of his algorithms significantly improve on the results in [3].

The algorithm in [3] is a mixture of dynamic programming and volume estimation, and uses Markov chain methods only indirectly. The sampling algorithm of [10] is based on dynamic programming, and samples are generated by a probabilistic traceback through the dynamic programming table. It does not use Markov chain methods at all. Therefore the question still remains as to whether the 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.

We give here the first proof of rapid mixing for a natural Markov chain when the number of rows m is a constant. This Markov chain, which we refer to as $\mathcal{M}_{2\times 2}$, was introduced by Dyer and Greenhill [11]. During a step of the chain, a 2×2 subtable is selected uniformly at random and is updated randomly. The subtable is updated according to the "heat bath" method. In particular, a new subtable is chosen from the conditional distribution (the uniform distribution, conditioned on the configuration outside of the subtable). In order to analyse $\mathcal{M}_{2\times 2}$, we first introduce an alternative heat bath chain, $\mathcal{M}_{\mathrm{HB}}$, which randomly updates a larger subtable. In particular, for a constant d_m which will be defined later, it updates a subtable of size $m \times (2d_m + 1)$ (also following the "heat bath" method of selecting a new subtable chosen uniformly at random, conditioned on the configuration outside of the subtable). We use the multicommodity flow technique of Sinclair [24] to analyse the mixing time of $\mathcal{M}_{\text{HB}}$. Using techniques developed by Morris and Sinclair [20] (see also [19]), we show that this chain mixes in time polynomial in the number of columns and the logarithm of the table sum. In Section 5 we compare $\mathcal{M}_{\text{HB}}$ to $\mathcal{M}_{2\times 2}$ and hence show that $\mathcal{M}_{2\times 2}$ is also rapidly mixing. This is the first proof that any chain converges in polynomial time even when the number of columns, as well as the number of rows, is constant. Establishing mixing in this case is one step of our proof. (See Pak [22] for an approach to this problem not using MCMC.) We note further that our results provide an alternative (and very different) *fpras* for counting contingency tables to that of Cryan and Dyer[3].

We now 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 [5] 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 [6]. See also Mount [21] for additional information, and De Loera and Sturmfels [4] for the current limits of exact counting methods.

With the exception of [1, 3] (and a recent result of Dyer [10]), 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. 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 [6] and Diaconis and Saloff-Coste [8], although it is mentioned in [6] 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 is not better than pseudopolynomial in the input. That is, the mixing time is at least a polynomial in N (rather than a polynomial in $\log N$). For a discussion of pseudopolynomial time and approximation algorithms, see Chapter 8 of [26].

Later Diaconis and Saloff-Coste [8] 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 (polynomial in the number of rows, the number of columns and the logarithm of the table sum) 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 the 2×2 heat-bath chain $\mathcal{M}_{2\times 2}$ and showed that for two-rowed tables, the 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.

Our paper can properly be viewed as extending Dyer and Greenhill's results to any constant number of rows. Thus, our main result is that $\mathcal{M}_{2\times 2}$ is rapidly mixing for any constant number of rows. First, however, in Section 4, we examine $\mathcal{M}_{\mathrm{HB}}$. Theorem 7 shows that this chain is rapidly mixing. Theorem 8 of Section 5 bounds the mixing time of the $\mathcal{M}_{2\times 2}$ in terms of the mixing time of $\mathcal{M}_{\mathrm{HB}}$. Combining the two theorems gives the main result.

2 Definitions

First, we define the Markov chain $\mathcal{M}_{2\times 2}$. The state space is $\Sigma_{r,c}$. Given a contingency table $X \in \Sigma_{r,c}$, a move is made as follows. With probability $1/2$, the chain stays at state X. With the remaining probability, a 2×2 submatrix is chosen as follows. A pair of rows i_1, i_2 is chosen uniformly at random and a pair of columns j_1, j_2 is chosen uniformly at random. The sub-matrix $X[i_1, j_1], X[i_1, j_2], X[i_2, j_1], X[i_2, j_2]$ is then replaced with a sub-matrix chosen uniformly at random from the set of 2×2 matrices with row sums $X[i_1, j_1] + X[i_1, j_2]$ and $X[i_2, j_1] + X[i_2, j_2]$ and column sums $X[i_1, j_1] + X[i_2, j_1]$ and $X[i_1, j_2] + X[i_2, j_2]$.

The self-loop probability $1/2$ in the definition of $\mathcal{M}_{2\times 2}$ is introduced for a technical reason – it simplifies the comparison of $\mathcal{M}_{2\times 2}$ and $\mathcal{M}_{\mathrm{HB}}$ in Section 5 by ensuring that the eigenvalues of the transition matrix of $\mathcal{M}_{2\times 2}$ are not negative.

Next we define the Markov chain $\mathcal{M}_{\mathrm{HB}}$. In Section 4 we will define a constant d_m (which depends upon m but not upon n or on the input vectors r and c). The state space is $\Sigma_{r,c}$. Given a contingency table $X \in \Sigma_{r,c}$, a move is made as follows. With probability $3/4$, the chain stays at state X. With the remaining probability, an $m \times (2d_m + 1)$ submatrix is chosen as follows. A set of $2d_m + 1$ columns $j_1, \ldots, j_{2d_m+1}$ is chosen uniformly at random from all columns of the table. The submatrix involving these columns is then replaced with submatrix chosen uniformly at random from the set of all $m \times (2d_m + 1)$ matrices with the same row and column sums as the chosen submatrix.

The self-loop probability 3/4 in the definition of $\mathcal{M}_{\mathrm{HB}}$ is again introduced for a technical reason – it ensures that the eigenvalues of the transition matrix of $\mathcal{M}_{\mathrm{HB}}$ are all at least 1/2, which is useful in the comparison of $\mathcal{M}_{2\times 2}$ and $\mathcal{M}_{\mathrm{HB}}$. It is not necessary to make the self-loop probability be 3/4 — anything greater than 1/2 suffices.

3 Background

In this section we summarize the techniques that we will use to bound the mixing time of $\mathcal{M}_{\mathrm{HB}}$. 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 [9].

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 is familiar: 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]). It is well-known that any ergodic Markov chain has a unique *stationary distribution*, that is, there is a unique distribution ϖ on Ω such that $\varpi P = \varpi$. Furthermore, for any choice of initial state $x \in \Omega$ and any state $y \in \Omega$, $P^t(x, y) \to \varpi(y)$ as $t \to \infty$ (see Chapter 6 of Grimmett and Stirzaker [13] for details). 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 $\mathrm{V}(\varpi, P^t(x)) = (1/2)\sum_{y\in\Omega} |\varpi(y) - P^t(x, y)|$. For any $0 < \epsilon < 1$ and any start state x, let $\tau_x(\epsilon)$ be defined as $\tau_x(\epsilon) = \min\{t : \mathrm{V}(\varpi, P^t(x)) \leq \epsilon\}$. 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 $P(u, v) > 0$. We call this graph the *Markov kernel.* 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 each path $p \in \mathcal{P}_{x,y}$ has a positive weight α_p, and 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 multi-commodity flow $\mathcal{F}$ by $\mathcal{L}(\mathcal{F}) = \max_{x,y} \max\{|p| : p \in \mathcal{P}_{x,y}\}$, 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]. See also the closely-connected work of Diaconis and Stroock [9]. Note that all logarithms in this paper are taken to be natural logarithms.

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. Suppose that the eigen-*

values of P are non-negative. Let $\mathcal{F}$ be a multicommodity flow on the graph G_Ω. Then the mixing time of the chain is bounded above by

$$\tau(\epsilon) \quad \leq \quad |\Omega|^{-1}\mathcal{L}(\mathcal{F}) \max_e \frac{\mathcal{F}(e)}{P(e)}(\log|\Omega| + \log \epsilon^{-1}). \tag{1}$$

Two key ingredients of our analysis of $\mathcal{M}_{\mathrm{HB}}$ in Section 4 are the "balanced almost-uniform permutations" and the "strongly balanced permutations" used by Morris and Sinclair [20] for their analysis of the multidimensional knapsack problem. 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 main idea is this: Given x and y we will use a permutation π of the columns of x to define a path of contingency tables from x to y. We will route flow from x to y along this path. Actually, π will be chosen from a distribution and 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, $\pi(i)$ will denote the original column (in $\{1,\ldots,n\}$) which gets put in position i by the permutation. Thus, $\pi\{1,\ldots,k\} = \{\pi(1),\ldots,\pi(k)\}$ denotes the set of original columns that get put in the first k positions by the permutation.

Definition 2 (Morris and Sinclair [20, Definition 3.2]) *Let σ be a random variable taking values in S_n (i.e., σ is a permutation of $\{1,\ldots,n\}$) and let $\lambda \in \mathbb{R}$. Then σ is a λ-uniform permutation if*

$$\Pr[\sigma\{1,\ldots,k\} = U] \quad \leq \quad \lambda \times \binom{n}{k}^{-1}$$

for every k with $k \in [n]$ and every $U \subseteq \{1,\ldots,n\}$ of cardinality k.

Definition 3 (Morris and Sinclair [20, Definition 5.1]) *Let $w_1,\ldots,w_n \in \mathbb{R}^d$ be any d-dimensional* weights *satisfying $\sum_{j=1}^n w_j = 0$ (i.e. $\sum_{j=1}^n w_j^i = 0$ for every $i \in [d]$). A permutation σ of $1,\ldots,n$ is ℓ-balanced if*

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

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

Checking the definition above, we see that a balanced permutation is one in which the partial sums of the weights (in each dimension) do not vary too much, where the factor ℓ gives us a bound on this variation. Morris and Sinclair showed how to construct balanced almost-uniform permutations when d is constant. (See also Theorem 3.2 in [19].)

Theorem 4 (Morris and Sinclair [20, Theorem 5.3]) *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 $\mathbb{R}^d$, there exists a g_d-balanced, $p_d(n)$-uniform permutation.*

The key points which we should keep in mind are (1) the distribution which Morris constructs is "nearly" uniform (and has a fair amount of entropy) and (2) the permutations satisfy some sort of balance property on multi-dimensional weights. Roughly, one should think of these weights as corresponding to the columns of our contingency tables – the multiple dimensions come from having multiple rows. Loosely speaking, the "balance" property of these permutations will be used to construct our multicommodity flow to generate a path (or a family of paths) of contingency tables to get from X to Y, each pair of tables along this path differing by a single move of $\mathcal{M}_{\mathrm{HB}}$.

Note that the construction of the g_d-balanced, $p_d(n)$-uniform permutation of Morris and Sinclair is carried out by induction on the number of dimensions d. It is clear from the construction that g_d will be no greater than $4^{d+1}-1$, though it is not easy to see a way of obtaining a smaller constant. We mention this because g_m will appear in the exponent of n when we bound the mixing time of our Markov chains (where m is the number of rows).

The "almost uniform" property will help ensure that the flow $\mathcal{F}(e)$ through any edge in G_Ω will not be too large (cf. Theorem 1). As mentioned before, we actually use a combination of permutations, one of which is balanced and almost-uniform, and a second type called "strongly balanced."

Definition 5 (Morris and Sinclair [20, Definition 5.4]) *Let $w_1, \ldots, w_n \in \mathbb{R}^d$ be any d-dimensional weights. Define $\mu = (\mu^1, \ldots, \mu^d)$ to be the vector of means of the w_j weights ($\mu^i = (\sum_{j=1}^n w_j^i)/n$ for all i). A permutation σ of $1, \ldots, n$ is strongly ℓ-balanced if for all $k \in [n]$ and all $i \in [d]$, there exists a set $S \subseteq [n]$ with $|S \oplus \{1, \ldots, k\}| < \ell$ such that $(\sum_{j=1}^k w_{\sigma(j)}^i - k\mu^i)$ and $(\sum_{j \in S} w_{\sigma(j)}^i - k\mu^i)$ have opposite signs (or either is 0).*

The main difference between ordinary balance and "strong balance" is that the definition of ordinary balance requires that the prefix sum $\sum_{j=1}^k w_{\sigma(j)}^i$ should be "close" to $k\mu$ for every k. However strong balance requires that for every prefix k and every row i, it should be possible to find a small number of columns so that removing those columns changes the sign of $(\sum_{j=1}^k w_{\sigma(j)}^i - k\mu^i)$.

Morris and Sinclair [20] adapted a result of Steinitz [25] (see also Grinberg and Sevast'yanov [14]) to show that

Theorem 6 (Morris and Sinclair [20, Lemma 5.5]) *For any sequence $\{w_j\}_{j=1}^n$ in $\mathbb{R}^d$, there exists a strongly $16d^2$-balanced permutation.*

4 Analysis of the generalized chain

We fix $r = (r_1, \ldots, r_m)$, the list of row sums, and $c = (c_1, \ldots, c_n)$, the list of column sums, and let Ω be the state space $\Sigma_{r,c}$ of $m \times n$ contingency tables with these row and column sums. Recall that N denotes the table sum $\sum_{i=1}^m r_i$.

Recall that g_m is the constant of Theorem 4 for balanced almost-uniform permutations for columns of dimension m. Let $d_m = 2m(3g_m + 1) + 1 + 34m^3$. We use P_{HB} for the transition matrix of the Markov chain $\mathcal{M}_{\mathrm{HB}}$ which was defined in Section 2.

In this section, our goal is to prove the following theorem.

Theorem 7 *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 7, we will show how to define a multicommodity flow $\mathcal{F}$ such that the total flow along any transition (ω, ω'') is at most $8fn^{2d_m+1}P_{\mathrm{HB}}(\omega, \omega'')$, where f is an expression that is at most $poly(n)\,|\Omega|$. We will also ensure that $\mathcal{L}(\mathcal{F})$ is bounded from above by a polynomial in n. Theorem 7 will then follow from (1) in Theorem 1. Constructing the flow $\mathcal{F}$ is done in a two stage process. In Subsection 4.1, we will define a multicommodity flow $\mathcal{F}^*$. Then in Subsection 4.2, we will first prove that the total flow through any state ω is at most f. Finally, also in Subsection 4.2, we will construct $\mathcal{F}$ by modifying $\mathcal{F}^*$.

In the first subsection we define the multicommodity flow we use in our application of Theorem 1. The construction uses the balanced, strongly-balanced, and almost-uniform properties of permutations of (some of) the columns of the contingency tables.

The second subsection shows that, with the multicommodity flow we define, each edge of the graph G_Ω is not too congested, and each path length is small. Theorem 7 then follows from Theorem 1.

4.1 Defining the flow

The construction of $\mathcal{F}$ in this subsection uses the methods of Morris and Sinclair [20] introduced in Section 3.

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 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 an $m \times n$ matrix that has 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. The distribution from which π is chosen will be defined in terms of an interlacing of the random balanced permutation of Theorem 4 and the strongly balanced permutation of Theorem 6. π will be a permutation of most of the columns of the table. The permutation π will define a path

$$Z_0 = X, \ldots, Z_{n'}$$

(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 the column X_j for all other j (so at each point, we swap another column of X for the same column of Y). In Subsection 4.1.1 we show that the balance properties of π ensure that for any Z_h, we can bring all the row sums below r_i by deleting a constant number of columns. Then in Subsection 4.1.2 we will show how to use this fact to define a path

$$X = Z'_0, \ldots, Z'_{n'+1} = Y$$

where each Z'_h is in $\Sigma_{r,c}$ and there is a transition in $\mathcal{M}_{\mathrm{HB}}$ from each Z'_h to Z'_{h+1}. The amount of flow that we will route along this path will be the proportional to the probability with which π is chosen.

4.1.1 A first step towards building paths

We start building our path(s) from X to Y by first defining a path of column constrained tables $Z_0 = X, \ldots, Z_{n'}$ using an interlacing of the random balanced permutation of Theorem 4 and the strongly balanced permutation of Theorem 6. In Subsection 4.1.2, we will show how to modify these columns, in a specific manner, to yield a new path of tables such that (i) the new tables are contingency tables (i.e. they satisfy the row sums as well as the column sums) and (ii) each successive pair along this path differs by a transition of $\mathcal{M}_{\mathrm{HB}}$.

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 $R = \left(\cup_i R_i^X\right) \cup \left(\cup_i R_i^Y\right) \cup \{k\}$ be the union of all the R_i^X and R_i^Y sets, together with the index k (which was defined earlier to be the index of the largest column sum).

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 do not permute these columns — we need them for something else. For every row i, define

$$\begin{aligned} M_i &= \min\{\max\{X[i,j] : j \notin R\}, \max\{Y[i,j] : j \notin R\}\} \\ L_i &= \{j : j \notin R, X[i,j] > M_i\} \cup \{j : j \notin R, Y[i,j] > M_i\}. \end{aligned}$$

Note that by definition of M_i, we either have $\{j : j \notin R, X[i,j] > M_i\} = \emptyset$ or $\{j : j \notin R, Y[i,j] > M_i\} = \emptyset$, so each L_i corresponds to a set of columns of X *or* a set of columns of Y, but not both. Also from their definitions, we see that $M_i \leq X[i,j]$, for all $j \in R_i^X$, and $M_i \leq Y[i,j]$, for all $j \in R_i^Y$. Set $L = \cup_{i=1}^m L_i$ and $S = [n] - (L \cup R)$.

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

$$\mu^i = \frac{\sum_{j \in [n]-R} Y[i,j] - X[i,j]}{n - |R|} = \frac{\sum_{j \in R} X[i,j] - Y[i,j]}{n - |R|}.$$

Let π_1 be a strongly $16m^2$-balanced permutation on the set of weights $\{w_j\}_{j \in L}$. This exists by Theorem 6. Let π_2 be a g_m-balanced $p_m(|S|)$-uniform permutation on $\{w_j\}_{j \in S}$. This exists by Theorem 4. π_2 is a random permutation. We interlace π_1 and π_2 in the same way as Morris and Sinclair [20] do to get a permutation π on $\{w_j\}_{j \in [n]-R}$. For the benefit of the reader, we restate the rule for performing this interlacing: Suppose that $\pi(1), \pi(2), \ldots, \pi(h)$ have already been assigned and that $\pi\{1, 2, \ldots, h\} = \pi_1\{1, \ldots, h_1\} \cup \pi_2\{1, \ldots, h_2\}$. Then either $\frac{h_1}{h} \leq \frac{|L|}{|L|+|S|}$ or $\frac{h_2}{h} < \frac{|S|}{|L|+|S|}$. We define $\pi(h+1)$ by

$$\pi(h+1) = \begin{cases} \pi_1(h_1 + 1) & \text{if } \frac{h_1}{h} \leq \frac{|L|}{|L|+|S|} \\ \pi_2(h_2 + 1) & \text{if } \frac{h_2}{h} < \frac{|S|}{|L|+|S|}. \end{cases}$$

This new permutation π satisfies inequalities (5.8) and (5.9) in Morris and Sinclair [20], reproduced as inequalities (2) and (3) below and proved in [20]. (See also inequalities (3.8) and (3.9) in [19].) These inequalities state that for every prefix h (h is the index of a column) and every dimension i (i is the index of a row), there exist 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

$$\sum_{j \in V_{i,h}} w^i_{\pi(j)} \leq (h-1)\mu^i + 3g_m M_i \quad \text{for every } i = 1, \ldots, m \tag{2}$$

$$\sum_{j \in W_{i,h}} w^i_{\pi(j)} \geq (h-1)\mu^i - 3g_m M_i \quad \text{for every } i = 1, \ldots, m. \tag{3}$$

The sets $V_{i,h}$ and $W_{i,h}$ will play an important role for us later.

Now let $n' = n - |R|$. For the permutation π constructed above 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 $j \in$

$R \cup \pi\{h+1,\dots,n'\}$ and columns Y_j for $j \in \pi\{1,\dots,h\}$. We see that $Z_{n'}$ differs from Y by at most $2m(3g_m+1)+1$ columns.

It is important to note that $Z_0,\dots,Z_{n'}$ may not be contingency tables in $\Sigma_{r,c}$ since they need not satisfy the row constraints. Thus, we cannot use this path directly to define our flow from X to Y. Nevertheless, we may base our path on these tables. The important point is that π and, in particular, (2) and (3) will allow us to turn $Z_0,\dots,Z_{n'}$ into a path of contingency tables. In the remainder of this subsection, we will show that we do not have to change too many columns to turn Z_h into a contingency table. Subsequently in Subsection 4.1.2 we will show that if we are careful about how we map Z_h to a contingency table, the resulting collection of paths will have good congestion.

We introduce the notation $(J(X), J(Y))$ to denote a set containing columns from X and from Y: for sets of indices $J(X) \subseteq [n]$ and $J(Y) \subseteq [n]$, $(J(X), J(Y))$ contains the column X_j for each $j \in J(X)$ and the column Y_j for each $j \in J(Y)$. 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 $\sum_{j \in J(X)} X[i,j] + \sum_{j \in J(Y)} Y[i,j]$.

Defining $J_h(X) = R \cup \pi\{h+1,\dots,n'\}$ and $J_h(Y) = \pi\{1,\dots,h\}$, we see that Z_h is the set of columns $(J_h(X), J_h(Y))$. Each of the Z_h tables is a column constrained table because, as previously noted, it is possible that some rows i may have $row^i(J_h(X), J_h(Y)) \neq r_i$. However, all the column sums are satisfied by Z_h.

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 $r_i(1+1/n)$. The "adding" causes some column indices to appear in both $J(X)$ and $J(Y)$ so the resulting configuration isn't much like a contingency table, but the construction will be useful below. Let $V_{i,h}$ and $W_{i,h}$ be the sets of inequalities (2) and (3).

First, instead of considering Z_h, consider $(\pi([n'] - V_{i,h})), \pi(V_{i,h}))$. This set of columns is the result of starting with the contingency table X, removing X_j for every reserved column $j \in R$, and then adding the weights w_j for $j \in \pi(V_{i,h})$. By (2), we know that $|V_{i,h} \oplus \{1,\dots,h\}| \leq 17m^2$ and

$$
\begin{aligned}
row^i(\pi([n'] - V_{i,h}), \pi(V_{i,h})) &\leq (r_i - \sum_{j \in R} X[i,j]) + (h-1)\mu^i + 3g_m M_i \\
&= (r_i - \sum_{j \in R} X[i,j]) + 3g_m M_i \\
&\quad + \frac{h-1}{n'}(\sum_{j \in R} X[i,j] - Y[i,j]) \\
&= r_i - \frac{n'-h+1}{n'}(\sum_{j \in R} X[i,j]) \\
&\quad - \frac{h-1}{n'}(\sum_{j \in R} Y[i,j]) + 3g_m M_i \\
&\leq r_i(1 - \frac{1}{n}) \qquad\qquad (a)
\end{aligned}
$$

where the last step follows because (i) we know that $M_i \leq X[i,j]$ for every $j \in R_i^X$, and $R_i^X \subseteq R$, $|R_i^X| = 3g_m + 1$. Also $\max_{j \in R_i^X} X[i,j] \geq r_i/n$. Thus $3g_m M_i + r_i/n \leq \sum_{j \in R} X[i,j]$; (ii) similarly, $3g_mM_i+r_i/n \le \sum_{j\in R} Y[i,j]$; (iii) the convex combination $((n'-h+1)\sum_{j\in R} X[i,j]+(h-1)\sum_{j\in R} Y[i,j])/n'$ is at least $3g_mM_i + r_i/n$.

Now suppose we also "delete" $\pi\{1,\ldots,h\} \oplus \pi\{V_{i,h}\}$ from $(\pi([n'] - V_{i,h}), \pi(V_{i,h}))$.

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

Then

$$\begin{aligned} J_h(X) - B_{i,h} &= \pi\{h+1,\ldots,n'\} \cap \pi([n'] - V_{i,h}) \\ J_h(Y) - B_{i,h} &= \pi\{1,\ldots,h\} \cap \pi(V_{i,h}). \end{aligned}$$

By (a), we have $row^i(J_h(X)-B_{i,h}, J_h(Y)-B_{i,h}) \le row^i(\pi([n']-V_{i,h}), \pi(V_{i,h})) \le r_i(1-1/n)$. Also, $|B_{i,h}| \le 2m(3g_m+1)+1+17m^2$.

For the dual result, consider $(\pi([n'] - W_{i,h}) \cup R, \pi(W_{i,h}) \cup R)$. This set of columns is the result of starting with the contingency table X, adding the Y_j columns for $j \in R$, and then adding the weights w_j for $j \in \pi(W_{i,h})$. By (3), we know that $|W_{i,h} \oplus \{1,\ldots,h\}| \le 17m^2$ and

$$\begin{aligned} row^i(\pi([n'] - W_{i,h}) \cup R, \pi(W_{i,h}) \cup R) &\ge (r_i + \sum_{j\in R} Y[i,j]) + (h-1)\mu^i - 3g_mM_i \\ &= (r_i + \sum_{j\in R} Y[i,j]) - 3g_mM_i + \\ &\quad \frac{h-1}{n'}(\sum_{j\in R} X[i,j] - Y[i,j]) \\ &= r_i + \frac{n'-h+1}{n'}(\sum_{j\in R} Y[i,j]) + \\ &\quad \frac{h-1}{n'}(\sum_{j\in R} X[i,j]) - 3g_mM_i \\ &\ge r_i(1+\frac{1}{n}) \qquad\qquad (b) \end{aligned}$$

where the last step follows because (i) $3g_mM_i + r_i/n \le \sum_{j\in R} X[i,j]$; (ii) $3g_mM_i + r_i/n \le \sum_{j\in R} Y[i,j]$; (iii) convex combination.

Suppose we add $\pi\{1,\ldots,h\} \oplus \pi\{W_{i,h}\}$ to $(\pi([n'] - W_{i,h}) \cup R, \pi(W_{i,h}) \cup R)$.

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

Then

$$\begin{aligned} J_h(X) \cup C_{i,h} &= \pi\{h+1,\ldots,n'\} \cup \pi([n'] - W_{i,h}) \cup R \\ J_h(Y) \cup C_{i,h} &= \pi\{1,\ldots,h\} \cup \pi(W_{i,h}) \cup R. \end{aligned}$$

By (b), we have $row^i(J_h(X) \cup C_{i,h}, J_h(Y) \cup C_{i,h}) \ge row^i(\pi([n'] - W_{i,h}) \cup R, \pi(W_{i,h}) \cup R) \ge r_i(1+1/n)$. Also, $|C_{i,h}| \le 2m(3g_m+1)+1+17m^2$.

Finally, define D_h to be a set of d_m column indices, including any column that is in $(\cup_i B_{i,h}) \cup (\cup_i C_{i,h})$. This is possible because $d_m = 2m(3g_m+1)+1+34m^3$. In addition to the set R (of size $2m(3g_m+1)+1$), each set $B_{i,h}$ contains up to $17m^2$ indices as does each set $C_{i,h}$.

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

$$Z_h^* =_{\text{def}} (J_h(X) - D_h, J_h(Y) - D_h).$$

Since $B_{i,h} \subseteq D_h$ for every i, we have $row^i(J_h(X) - D_h, J_h(Y) - D_h) \le r_i(1 - 1/n)$ for all i. Also define

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

Since $C_{i,h} \subseteq D_h$ for every i, we have $row^i(J_h(X) \cup D_h, J_h(Y) \cup D_h) \ge r_i(1 + 1/n)$ for all i.

Note that D_h contains all of R, including the index k.

4.1.2 Going from Z_h to an element of Ω

Having defined the set D_h, we now show how to change Z_h into a contingency table. This is a crucial step, actually turning the tables $Z_0, \ldots, Z_{n'}$ into a path of elements of Ω that joins the two contingency tables X and Y. The most critical part in the construction of this multicommodity flow is to ensure that the flow on any edge $e \in G_\Omega$ is not too large. Therefore it will be important to prove that we do not map too many of the column constrained tables Z_h to any given element of Ω. We will see below that we will need to be careful in mapping column constrained tables which have large column sums for some of the "deleted" columns D_h.

Step 2: We now show how to convert Z_h into an 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 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 at most 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.

Note that $s_i \ge r_i/n$ for all i.

Let $N_h = \sum_{i=1}^m s_i = \sum_{j \in D_h} c_j$, by construction. We have two cases to consider.

Case 1: 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 [6]). 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 $1 \le i \le m$. 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.

Case 2: Alternatively, assume that $N_h \ge 2(md_m)^2$. 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 need something stronger – we show how 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. Performing the modification in this carefully structured way allows us to ensure that the congestion on edges in G_Ω in our multicommodity flow is not too large, a point we return to following the definitions in the next paragraph.

By our previous definition of D_h, we already know that k is the index of the largest column sum c_j for $j \in D_h$. Let ℓ be the index of the biggest s_i value. For every $i \ne \ell$ 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, for every $i \neq \ell$ and every $j \in D_h - \{k\}$, we can write

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

for non-negative integers $Q[i,j], R[i,j]$, where $Q[i,j] < n(d_m)^2$ and $0 \leq R[i,j] \leq a_{i,j}$. $Q[i,j]$ and $R[i,j]$ are uniquely determined by $Z_h[i,j]$. 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. As promised, however, we first give the reason why we want to view the entries of Z_h in this way, focusing on the importance of changing only the $Q[i,j]$ terms to obtain the subtable Z'_h.

Our analysis of the flow we construct will rely on bounding the number of column constrained tables Z_h that can be transformed into the contingency table Z'_h. In particular, for each particular choice of D_h, we bound the number of tables Z_h that can be transformed into Z'_h. If the subtable sum N_h is less than $2(md_m)^2$, then $r_i < 2n(md_m)^2$ for every i and therefore, for every $j \in D_h$ and every row i, there are at most $2n(md_m)^2$ possible original values for $Z_h[i,j]$. On the other hand, when $N_h \geq 2(md_m)^2$, we do not have an upper bound on the original entries of the subtable. However, the method above for constructing the $Z'_h[i,j]$ values using $a_{i,j}$ gives an indirect upper bound for the number of Z_h that could be converted to Z'_h. For any true contingency table Z'_h and any selection of D_h columns which gives $N_h \geq 2(md_m)^2$, we can calculate $a_{i,j}$ for all $i \neq \ell$ and all $j \in D_h - \{k\}$. Then, for every row $i \neq \ell$ and every column j in $D_h - \{k\}$, we can find the original value of $R[i,j]$ using that $R[i,j] = Z'_h[i,j] \bmod (a_{i,j}+1)$. By the definition of $a_{i,j}$, there are at most $n(d_m)^2$ possible values for the original $Q[i,j]$, so there are only $(n(d_m)^2)^{(m-1)(d_m-1)}$ different ways of filling in the entire subtable on the $D_h - \{k\}$ columns. The fact that there are at most $poly(n)$ possible values for the original $Q[i,j]$ cells is the key that makes our analysis work in Section 4.2.

We now return to our proof, since we still need to show that we can choose values $Q'[i,j]$ which ensure that the new $Z'_h[i,j]$ values will satisfy row sums s_i and column sums c_j. Recall our assumption that $N_h \geq 2(md_m)^2$. 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,j, where the $Z'[i,j]$ also satisfy the following inequalities:

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

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

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

Now define $Q'[i,j]$ 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 $j \in D_h - \{k\}$. For those values of i and j, we set $Z'_h[i,j] = Q'[i,j](a_{i,j}+1) + R[i,j]$. Note that $Q'[i,j] \geq 0$ for all i,j. We now prove that inequalities (4), (5), and (6) are satisfied for these $Z'[i,j]$.

Inequality (4):

$$\begin{aligned}
\sum_{i\neq\ell} Z'_h[i,j] &= \sum_{i\neq\ell} Q'[i,j](a_{i,j}+1) + \sum_{i\neq\ell} R[i,j] \\
&= \sum_{i\neq\ell} \lfloor s_i c_j / N_h(a_{ij}+1) \rfloor (a_{i,j}+1) + \sum_{i\neq\ell} R[i,j] \\
&\leq \sum_{i\neq\ell} s_i c_j / N_h + \sum_{i\neq\ell} R[i,j] \\
&\leq \sum_{i\neq\ell} s_i c_j / N_h + \sum_{i\neq\ell} a_{i,j} \\
&= c_j(1 - s_\ell/N_h) + \sum_{i\neq\ell} \lfloor \min\{r_i, c_j\}/n(d_m)^2 \rfloor \\
&\leq c_j(1 - s_\ell/N_h) + c_j(m-1)/n(d_m)^2 \\
&\leq c_j(1 - 1/m + 1/nd_m) \\
&\leq c_j.
\end{aligned}$$

The last two steps use the facts that (i) $s_\ell/N_h \geq 1/m$ (because s_ℓ is the largest row sum) and (ii) $d_m > m$.

Inequality (5): The early steps are similar to the proof for inequality (4). We get

$$\begin{aligned}
\sum_{j\in D_h-\{k\}} Z'_h[i,j] &\leq s_i(1 - c_k/N_h) + \sum_{j\in D_h-\{k\}} \lfloor \min\{r_i, c_j\}/n(d_m)^2 \rfloor \\
&\leq s_i(1 - c_k/N_h) + (d_m - 1)r_i/n(d_m)^2 \\
&\leq s_i(1 - 1/d_m) + s_i/d_m \\
&\leq s_i
\end{aligned}$$

where the second last step uses the facts that (i) $c_k/N_h \geq 1/d_m$ (because c_k is the largest column sum in the subtable) and (ii) $s_i \geq r_i/n$.

Inequality (6):

$$\begin{aligned}
\sum_{i\neq\ell} \sum_{j\in D_h-\{k\}} Z'_h[i,j] &= \sum_{i\neq\ell} \sum_{j\in D_h-\{k\}} Q'[i,j](a_{i,j}+1) + \sum_{i\neq\ell} \sum_{j\in D_h-\{k\}} R[i,j] \\
&= \sum_{i\neq\ell} \sum_{j\in D_h-\{k\}} \lfloor s_i c_j / N_h(a_{i,j}+1) \rfloor (a_{i,j}+1) + \sum_{i\neq\ell} \sum_{j\in D_h-\{k\}} R[i,j] \\
&\geq \sum_{i\neq\ell} \sum_{j\in D_h-\{k\}} s_i c_j / N_h - \sum_{i\neq\ell} \sum_{j\in D_h-\{k\}} (a_{i,j}+1) + \sum_{i\neq\ell} \sum_{j\in D_h-\{k\}} 0 \\
&= (N_h - s_\ell)(N_h - c_k)/N_h - \sum_{i\neq\ell} \sum_{j\in D_h-\{k\}} (a_{i,j}+1) \\
&= (N_h - s_\ell - c_k) + s_\ell c_k / N_h - \sum_{i\neq\ell} \sum_{j\in D_h-\{k\}} (a_{i,j}+1).
\end{aligned}$$

Next, note that

$$\begin{aligned}
s_\ell c_k / N_h &\geq N_h/(md_m) \\
\sum_{i\neq\ell} \sum_{j\in D_h-\{k\}} (a_{i,j}+1) &\leq md_m + m(N_h - c_k)/n(d_m)^2 \leq md_m + N_h/(nd_m m),
\end{aligned}$$

by definition of the $a_{i,j}$ and because $d_m \geq m^2$. Therefore, to show that inequality (6) holds, it suffices to show

$$N_h/(md_m) \geq md_m + N_h/(nd_m m). \tag{7}$$

We note the following statements:

$$\begin{aligned} N_h/(md_m) &\geq md_m + N_h/(nd_m m) \\ \iff N_h(1-1/n)/(md_m) &\geq md_m \\ \Longleftarrow N_h/2 &\geq (md_m)^2 \quad \text{(using that } n \geq 2) \\ \iff N_h &\geq 2(md_m)^2. \end{aligned}$$

This last statement holds by our previous assumption on N_h, therefore inequality (7) holds, and hence (6) is true, as required.

Therefore, in parallel with our path of column constrained tables, we have a path $X = Z'_0, \ldots, Z'_h, \ldots, Z'_{n'}$ 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 a final 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 is proportional to the probability that π is chosen. Since $n' \leq n$, we see that the length of this path from X to Y is at most $n+1$.

Remember that the column constrained table Z_h contains exactly n columns and that $X_j \in Z_h \Leftrightarrow Y_j \notin Z_h$. Also, Z_h is the pair $(J_h(X), J_h(Y))$. Then if we define $\bar{Z}_h$ by the pair of sets $(J_h(Y), J_h(X))$, $\bar{Z}_h$ also contains exactly n columns and $X_j \in \bar{Z}_h \Leftrightarrow X_j \notin Z_h$. Now to calculate the row sums of $\bar{Z}_h$, consider $\bar{Z}_h$ with the columns of D_h removed. Then

$$\begin{aligned} row^i(J_h(Y) - D_h, J_h(X) - D_h) &= row^i(([n] - J_h(X)) - D_h), ([n] - J_h(Y)) - D_h) \\ &= row^i([n] - (J_h(X) \cup D_h), [n] - (J_h(Y) \cup D_h) \\ &= 2r_i - row^i(J_h(X) \cup D_h, J_h(Y) \cup D_h) \\ &\leq r_i(1 - 1/n) \end{aligned}$$

because we proved that $row^i(J_h(X) \cup D_h, J_h(Y) \cup D_h) \geq r_i(1+1/n)$ in Step 1.

Therefore, for $\bar{Z}_h$, there also exists a set of d_m columns (the same set of indices D_h) such that we can obtain a true contingency table $\bar{Z}'_h \in \Sigma_{r,c}$ by modifying these columns in the structured way described above.

4.2 Analysis of the multicommodity flow

Next we show that the flow through any state $Z' \in \Omega$ is at most $poly(n)\,|\Omega|$. (Remark: We actually bound all of the flow *except* that due to pairs (X, Y) where $Z' = Y$. The total flow for all such pairs is only $|\Omega|$. Hence showing a bound of the form $poly(n)\,|\Omega|$ for the flow between remaining pairs where $Z' \neq Y$ provides us a similar bound for the total flow through Z'.) We assume Z' occurs as Z'_h for some pairs of contingency tables (X, Y) and some values of h and $\pi\{1, \ldots, h\}$. Again, we write $\bar{Z}_h$ for the column constrained table with X_j for $j \in J_h(Y)$ and Y_j for $j \in J_h(X)$. We claim that if we are given Z'_h and

(1) $\bar{Z}'_h \in \Omega$, a true contingency table obtained by changing d_m of the columns of $\bar{Z}_h$;

(2) the value of h and the set $\pi\{1, \ldots, h\}$ of columns already changed from X to Y in Z_h;

(3) The set D_h of d_m columns of Z_h which were modified to obtain Z'_h (note that the columns of $\bar{Z}_h$ that were modified are also the columns in D_h);

(4) The index ℓ of the largest induced row sum s_ℓ for the subtable of Z_h on D_h and the index ℓ' of the largest induced row sum for the subtable of $\bar{Z}_h$ on D_h;

(5) Two possibilities.
(i) If $N_h < 2(md_m)^2$, then we are given the original values of $Z_h[i,j]$ for all i, for all $j \in D_h$. Note we have $Z_h[i,j] \leq 2(md_m)^2$;
(ii) Otherwise if $N_h \geq 2(md_m)^2$, we are given the original integers $Q[i,j]$ obtained from $Z_h[i,j]$ for every $i \neq \ell$ and every $j \in D_h - \{k\}$. Note we have $0 \leq Q[i,j] < n(d_m)^2$;

(5′) Two possibilities.
(i) If $N_h < 2(md_m)^2$, then we are given the original values of $\bar{Z}_h[i,j]$ for all i, for all $j \in D_h$. Note we have $\bar{Z}_h[i,j] \leq 2(md_m)^2$;
(ii) Otherwise if $N_h \geq 2(md_m)^2$, we are given the original integers $\bar{Q}[i,j]$ obtained from $\bar{Z}_h[i,j]$ for every $i \neq \ell'$ and every $j \in D_h - \{k\}$. Note we have $0 \leq \bar{Q}[i,j] < n(d_m)^2$;

then we can construct X and Y. We refer to the information in (1)-(5′) as an "encoding" (for X and Y at the element $Z' \in \Omega$).

First we concentrate on recovering Z_h. From (3) we know the submatrix D_h which has to be modified to obtain Z_h from Z'_h. Since we know the D_h columns, we can calculate N_h. If $N_h < 2(md_m)^2$, then the information in (5) tells us the original $Z_h[i,j]$ values for $j \in D_h$ and this gives us the entire column constrained table Z_h. If $N_h \geq 2(md_m)^2$ then we use (4) to identify ℓ. For every $i \neq \ell$ and $j \in D_h - \{k\}$ we calculate $a_{i,j}$ from r_i and c_j and we calculate $R[i,j] = Z'_h[i,j] \bmod (a_{i,j}+1)$. Finally, by (5) we have the original values of $Q[i,j]$ for all $i \neq \ell$ and all $j \in D_h - \{k\}$. Therefore we calculate $Z_h[i,j] = Q[i,j](a_{i,j}+1) + R[i,j]$. We can also calculate $Z_h[\ell,j]$ for $j \in D_h - \{k\}$ by subtracting the other values of $Z_h[i,j]$ (which we just calculated) from c_j. We still need to find the values $Z_h[i,k]$, but we defer this for the moment.

In a similar way, we use (3), (4) and (5′) to obtain all columns except column k of $\bar{Z}_h$ from $\bar{Z}'_h$ (given to us in (1)).

We now have all the columns except for column k of Z_h and $\bar{Z}_h$. Z_h contains column X_j for every $j \in J_h(X) = R \cup \pi\{h+1,\ldots,n'\}$ and contains column Y_j for every $j \in J_h(Y) = \pi\{1,\ldots,h\}$. $\bar{Z}_h$ contains column X_j for every $j \in J_h(Y) = [n] - J_h(X)$ and column Y_j for every $j \in J_h(X) = [n] - J_h(Y)$. By (2), we are given $\pi\{1,\ldots,h\}$. Then the contingency table X is the set of columns where

$$X_j = \begin{cases} \text{column } j \text{ of } Z_h \text{ for } j \in [n] - \pi\{1,\ldots,h\} \\ \text{column } j \text{ of } \bar{Z}_h \text{ for } j \in \pi\{1,\ldots,h\}. \end{cases}$$

The contingency table Y is the set of columns where

$$Y_j = \begin{cases} \text{column } j \text{ of } Z_h \text{ for } j \in \pi\{1,\ldots,h\} \\ \text{column } j \text{ of } \bar{Z}_h \text{ for } j \in [n] - \pi\{1,\ldots,h\}. \end{cases}$$

Thus for any $Z'_h \in \Omega$, we can construct all columns of X and Y except column k. Column k of each of these can then be recovered using the original r_i values. Thus, for any $Z'_h \in \Omega$, we can construct X and Y for any pair (X,Y) whose path passes through Z'_h, given the encoding (1)-(5′).

Now we bound the flow through our given Z'_h. Note that the flow through Z'_h is (bounded by) the number of possible choices for (1) and for (3)-(5′), times the amount of flow given by all possible choices for $\pi\{1,\ldots,h\}$ (given by (2)).

Since $\bar{Z}'_h \in \Omega$, there are $|\Omega|$ choices for (1).

There are $\binom{n}{d_m-1}$ possible D_h sets (choices for (3)), since D_h always contains k.

There are m^2 choices for (4).

Depending on the value of N_h, either there are at most $n(d_m)^2$ possible values for the original value of $Z_h[i,j]$ for $j \in D_h$ (obtained via $Q[i,j]$), or there are at most $2(md_m)^2$ possible values for the original value of $Z_h[i,j]$ for $j \in D_h$. Therefore there are at most $(2n(md_m)^2)^{md_m}$ possible sets of $Z_h[i,j]$ for (5). The same upper bound holds for (5′).

The total number of choices for (4)-(5′) is $m^2(2n(md_m)^2)^{2md_m}$. We will write this as $C_m n^{2md_m}$, where C_m is the constant $2^{2m(d_m)}m^2(md_m)^{4md_m}$.

Recall that we are shipping one unit of flow from X to Y for all ordered pairs $(X,Y) \in \Omega^2$, and we want to find an upper bound on the amount of flow that passes through a given element $Z' \in \Omega$. The portion of flow from X to Y passing through Z' is determined by the distribution on the choices of permutations in (2). The permutation π defines the sequence $Z_0,\ldots,Z_{n'}$ which, in turn, defines $Z'_0,\ldots,Z'_{n'}$. We want to know how much flow passes through Z'_h for the choices (1),(3)-(5′) of the encoding. Therefore, we see the amount passing through Z'_h is bounded above by

$$|\Omega| C_m \binom{n}{d_m-1} n^{2md_m} \sum_h \sum_{U:\ |U|=h} \Pr[\pi\{1,\ldots,h\} = U]$$

where $\Pr[\pi\{1,...,h\} = U]$ is the probability that U is the set of the first h columns to be changed.

Therefore (as in [20]), we need to bound $\sum_h \sum_{U:\ |U|=h} \Pr[\pi\{1,\ldots,h\} = U]$ for each particular possibility for parts 1,3,4,5, and 5' of the encoding. As explained before, these parts of the encoding determine Z_h and $\bar{Z}_h$ (apart from column k). Z_h and $\bar{Z}_h$ together give us the set $\{X_j, Y_j\}$ for any $j \neq k$, but they don't tell us which of the two columns is X_j and which is Y_j. The relevant parts of the encoding also determine D_h. For the given value of D_h, there are at most 2^{d_m-1} possibilities for R (which must include column k), and we shall sum over all of them. We then upper bound the flow coming from all permutations $\pi\{1,\ldots,h\}$ on the columns of $[n]-R$. There are at most $(2(n-|R|))^m \leq (2n)^m$ possibilities for the vector M giving the M_i values and we like likewise sum over all such choices. For each choice of (possible) values $\{M_i\}$ we can compute from Z_h and $\bar{Z}_h$ the set $L_i = \{j : j \notin R, X[i,j] > M_i\} \cup \{j : j \notin R, Y[i,j] > M_i\}$. Note that this gives us $S = [n] - (L \cup R)$ as well. By our remark on page 9, we know that each L_i consists solely of columns of X, or solely of columns of Y. Therefore there are only two possibilities for assigning all the L_i columns to X or Y (2^m choices for all of $L = \cup_i L_i$).

Let $h_2 = h - |\pi\{1,\ldots,h\} \cap L|$. We bound the flow passing through Z'_h below. Note that the first summation is over all $(2(n-|R|))^m$ possibilities for M and all 2^m possible assignments of L.

$$\begin{aligned}
&|\Omega| C_m \binom{n}{d_m - 1} n^{2md_m} \sum_{R,M,L,h_2} \sum_{U \subseteq S:\ |U|=h_2} \Pr[\pi_2\{1,\ldots,h_2\} = U] \\
&\le\ |\Omega| C_m \binom{n}{d_m - 1} n^{2md_m} 2^{d_m-1} (2n)^m 2^m n \sum_{U \subseteq S:\ |U|=h_2} \Pr[\pi_2\{1,\ldots,h_2\} = U] \\
&\le\ |\Omega| C_m \binom{n}{d_m - 1} n^{2md_m} 2^{d_m-1} (2n)^m 2^m n \sum_{U \subseteq S:\ |U|=h_2} p_m(|S|) / \binom{|S|}{h_2} \\
&\le\ |\Omega| C_m \binom{n}{d_m - 1} n^{2md_m} 2^{d_m-1} (2n)^m 2^m n \binom{|S|}{h_2} p_m(|S|) / \binom{|S|}{h_2} \\
&\le\ |\Omega| C_m \binom{n}{d_m - 1} n^{2md_m} 2^{d_m-1} (2n)^m 2^m n p_m(|S|),
\end{aligned}$$

which is $poly(n)\,|\Omega|$.

Thus we know that the flow through any *state* is bounded by a quantity f which is at most $poly(n)\,|\Omega|$. In the application of Morris and Sinclair [20] 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}, this term may be exponentially small, and a 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 $\mathcal{M}_{\mathrm{HB}}$, with transition probability $P_{\mathrm{HB}}(e)$. Suppose that f_e units of flow are shipped along e in the multi-commodity flow $\mathcal{F}^*$ defined above. We will construct a new flow $\mathcal{F}$ in which these f_e units are dispersed, travelling 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 $\omega'' \in \Omega''$, the flow $\mathcal{F}$ 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. Hence, the length of the longest path in the new flow $\mathcal{F}$ is at most $2(n+1)$.

The quantity shipped through (ω, ω'') in $\mathcal{F}$ from the original transition e in the multi-commodity flow $\mathcal{F}^*$ is $f_e/|\Omega''|$, which is at most $4 f_e n^{2d_m+1} P_{\mathrm{HB}}(\omega, \omega'')$. To see this, let K be the (non-empty) set of columns on which ω and ω'' differ. For every heat-bath window U, let $\Omega_\omega(U)$ denote the set of contingency tables which agree with ω, except possibly on window U

and write

$$
\begin{aligned}
P_{\mathrm{HB}}(\omega,\omega'') &= \frac{1}{4}\sum_{U\supseteq K}\frac{1}{\binom{n}{2d_m+1}}\frac{1}{|\Omega_\omega(U)|}\\
&\geq \frac{1}{4}\sum_{U\supseteq K, U\in W}\frac{1}{\binom{n}{2d_m+1}}\frac{1}{|\Omega_\omega(U)|}\\
&\geq \frac{1}{4}\sum_{U\supseteq K, U\in W}\frac{1}{\binom{n}{2d_m+1}}\frac{1}{|\Omega''|}\\
&\geq \frac{1}{4}\frac{1}{\binom{n}{2d_m+1}}\frac{1}{|\Omega''|},
\end{aligned}
$$

where the last inequality follows from the fact that $\omega'' \in \Omega''$, so there is at least one U in the summation.

We call the transition (ω, ω'') a *type 1* transition and a transition (ω'', ω') a *type 2* transition.

We can now give an upper-bound for the total type 1 flow along the transition (ω, ω''). For each $e = (\omega, \omega')$, we ship at most $4f_e n^{2d_m+1} P_{\mathrm{HB}}(\omega, \omega'')$ flow. Let f be the bound from above on the *total* flow that leaves node ω in our original multicommodity flow $\mathcal{F}^*$ (so $f = \sum_e f_e$ where the sum is over transitions e which start at ω). Then the total amount routed via ω'' in $\mathcal{F}$ is at most $4fn^{2d_m+1}P_{\mathrm{HB}}(\omega, \omega'')$.

Using a symmetric argument, we can show that the total type 2 flow along the transition (ω'', ω') in $\mathcal{F}$ is at most $4fn^{2d_m+1}P_{\mathrm{HB}}(\omega'', \omega')$.

Thus, the total flow in $\mathcal{F}$ along transition (ω, ω'') is at most $8fn^{2d_m+1}P_{\mathrm{HB}}(\omega, \omega'')$. Using the fact that the longest flow-carrying path length is at most $2(n + 1)$, 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 7.

5 Mixing of the 2×2 chain

Theorem 7 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 [7] to show that the 2×2 chain $\mathcal{M}_{2\times 2}$ is also rapidly mixing. We briefly describe the comparison method, in the context of contingency tables, adapted from [7]. For more details and other examples of applications of this method, see [7], Randall and Tetali [23], and Vigoda [27].

5.1 Setting up the comparison

Recall that P_{HB} denotes the transition matrix of the Markov chain $\mathcal{M}_{\mathrm{HB}}$ which was analyzed in Section 4. Let $E(P_{\mathrm{HB}})$ be the set of edges (excluding loops) in the Markov kernel of that chain. That is, $E(P_{\mathrm{HB}}) = \{(X, Y) : X \neq Y \text{ and } P_{\mathrm{HB}}(X, Y) > 0\}$. Let $P_{2\times 2}$ denote the transition matrix of $\mathcal{M}_{2\times 2}$ and let $E(P_{2\times 2})$ denote the edge-set of its Markov kernel.

For every $(X, Y) \in E(P_{\mathrm{HB}})$ we define a set of paths $\Gamma_{X,Y}$ where each $\gamma \in \Gamma_{X,Y}$ is a path $X = \eta_0, \eta_1, \ldots, \eta_k = Y$, such that $(\eta_i, \eta_{i+1}) \in E(P_{2\times 2})$ for all $i \in \{0, \ldots, k-1\}$. Let $|\gamma|$ denote the length (number of edges) of the path. We also define a *flow* $f_{X,Y}$, which is a function from $\Gamma_{X,Y}$ to the positive reals satisfying the condition

$$\sum_{\gamma \in \Gamma_{X,Y}} f_{X,Y}(\gamma) = 1. \tag{8}$$

It is important to note that this flow need only be defined for pairs $(X,Y) \in E(P_{\mathrm{HB}})$, not for all pairs (X,Y).

For each $(Z,Z') \in E(P_{2\times 2})$, define the quantity

$$A_{Z,Z'} = \frac{1}{P_{2\times 2}(Z,Z')} \sum_{(X,Y)\in E(P_{\mathrm{HB}})} \sum_{\substack{\gamma\in\Gamma_{X,Y} \text{ such} \\ \text{that } (Z,Z')\in\gamma}} |\gamma| f_{X,Y}(\gamma) P_{\mathrm{HB}}(X,Y).$$

We use the comparison theorem of Diaconis and Saloff-Coste, which says[1] that

$$\tau_{2\times 2}(\epsilon) \in O(\tau_{HB}(\epsilon)\log(|\Sigma_{r,c}|) \max_{(Z,Z')\in E(P_{2\times 2})} A_{Z,Z'}).$$

In our construction of the flow, we ensure that the length of each path $\gamma \in \Gamma_{X,Y}$ is bounded by a constant. Thus, the theorem of Diaconis and Saloff-Coste tells us that to establish rapid mixing, we need only define $f_{X,Y}$ for every $(X,Y) \in E(P_{\mathrm{HB}})$ such that Equation (8) is satisfied and also, for all $(Z,Z') \in E(P_{2\times 2})$, the following is satisfied:

$$\frac{1}{P_{2\times 2}(Z,Z')} \sum_{(X,Y)\in E(P_{\mathrm{HB}})} \sum_{\substack{\gamma\in\Gamma_{X,Y} \text{ such} \\ \text{that } (Z,Z')\in\gamma}} f_{X,Y}(\gamma) P_{\mathrm{HB}}(X,Y) \le poly(n). \tag{9}$$

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

$$P_{\mathrm{HB}}(X,Y) = \frac{1}{4} \sum_{W\in\mathcal{W}(X,Y)} \frac{1}{\binom{n}{2d_m+1}} \frac{1}{|\Omega_X(W)|},$$

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 (a multiple of) the average of the quantities $1/|\Omega_X(W)|$ over all windows $W \in \mathcal{W}(X,Y)$. Therefore, there is some $W(X,Y) \in \mathcal{W}(X,Y)$ such that

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

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 2×2 part of) this specially chosen window $W(X,Y)$ that satisfies (10).

For each $m \times (2d_m+1)$ window W, let $E_W = \{(X,Y) : (X,Y) \in E(P_{HB}) \text{ and } W(X,Y) = W\}$. Later, when we define our flows, we do the following for every fixed window W: For

[1] The statement of the theorem in this form is from Vigoda [27] and Randall and Tetali [23]. The derivation of Proposition 4 in [23] required the eigenvalues of P_{HB} to be at least 1/2, which is why we added the self-loops to $\mathcal{M}_{\mathrm{HB}}$. (Actually, bounding the eigenvalues above zero by any amount suffices.) The comparison uses the fact that the eigenvalues of $P_{2\times 2}$ are positive since this method provides a lower bound for $1 - \lambda_1(P_{2\times 2})$ in terms of $1 - \lambda_1(P_{\mathrm{HB}})$, and in these situations those differences directly relate to mixing times.

every $(X, Y) \in E_W$, we define a flow $f_{X,Y}$ such that Equation (8) is satisfied. We also ensure that for all $(Z, Z') \in E(P_{2\times2})$, the following is satisfied:

$$\frac{1}{P_{2\times2}(Z,Z')} \sum_{(X,Y)\in E_W} \sum_{\substack{\gamma\in\Gamma_{X,Y} \text{ such} \\ \text{that } (Z,Z')\in\gamma}} f_{X,Y}(\gamma) P_{\mathrm{HB}}(X,Y) \le \mathit{poly}(n). \tag{11}$$

Since there are only polynomially-many windows W, by summing over all of them we see that Equation (11) implies Equation (9), and ensures rapid mixing of $\mathcal{M}_{2\times2}$.

For each window W, Section 5.2 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 $O(|\Omega_X(W)|)$. At the end of Section 5.1 we will 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 $f^*_{X,Y}$ will ensure that for every $(Z, Z') \in E(P_{2\times2})$, we have

$$f(Z,Z') \le 4 f^*(Z) P_{2\times2}(Z,Z') \binom{n}{2}\binom{m}{2}. \tag{12}$$

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

$$\begin{aligned}
&\frac{1}{P_{2\times2}(Z,Z')} \sum_{(X,Y)\in E_W} \sum_{\substack{\gamma\in\Gamma_{X,Y} \text{ such} \\ \text{that } (Z,Z')\in\gamma}} f_{X,Y}(\gamma) P_{\mathrm{HB}}(X,Y) \\
\le\ & \frac{1}{P_{2\times2}(Z,Z')} \sum_{(X,Y)\in E_W} \sum_{\substack{\gamma\in\Gamma_{X,Y} \text{ such} \\ \text{that } (Z,Z')\in\gamma}} f_{X,Y}(\gamma) \frac{1}{|\Omega_X(W)|} \qquad (13)\\
=\ & \frac{1}{P_{2\times2}(Z,Z')} \cdot \frac{1}{|\Omega_X(W)|} \sum_{(X,Y)\in E_W} f_{X,Y}(Z,Z') \\
=\ & \frac{f(Z,Z')}{P_{2\times2}(Z,Z')} \cdot \frac{1}{|\Omega_X(W)|} \\
\le\ & \frac{4f^*(Z)\binom{n}{2}\binom{m}{2}}{|\Omega_X(W)|} \ \le\ \mathit{poly}(n). \qquad (14)
\end{aligned}$$

Inequality (13) comes from (10), where $\Omega_X(W) = \Omega_X(W(X, Y))$, and from our definition of E_W. The first inequality in (14) comes from Equation (12) which we establish shortly, and the second inequality in (14) comes from the fact that $f^*(Z) \in O(|\Omega_X(W)|)$, which is established in Section 5.2. We will then have shown that Equation (11) is satisfied, as required, so $\mathcal{M}_{2\times2}$ is rapidly mixing on $\Sigma_{r,c}$.

We finish this section by showing how to construct $f_{X,Y}$ given the flow $f^*_{X,Y}$, thereby establishing (12). The method is similar to (but simpler than) the one that we used at the end of Section 4.2.

Recall that $E(P_{2\times2})$ excludes self-transitions of the form (Z,Z). Thus, for each $(Z,Z'')\in E(P_{2\times2})$, there is a *unique* 2×2 window $U(Z,Z'')$ on which Z and Z'' disagree. Let $\Omega_Z(U(Z,Z''))$ denote the set of all contingency tables that agree with Z (and Z'') everywhere outside of the 2×2 window $U(Z,Z'')$. Let $f^*_{X,Y}(Z,Z'')$ be the flow that passes through (Z,Z'') in $f^*_{X,Y}$. For each $Z'\in\Omega_Z(U(Z,Z''))$, some of this flow is allocated to the path Z,Z',Z''. In particular, $f^*_{X,Y}(Z,Z'')/|\Omega_Z(U(Z,Z''))|$ flow is sent this way.

Now $f_{X,Y}(Z,Z')\le 2\sum_{Z''\in U(Z,Z')}\frac{f^*_{X,Y}(Z,Z'')}{|\Omega_Z(U(Z,Z''))|}$, where the first "2" comes from the fact that we must consider paths Z,Z',Z'' above, and also paths that end in edge Z,Z'. The right-hand side is at most

$$2\frac{f^*_{X,Y}(Z)}{|\Omega_Z(U(Z,Z'))|}\le 4f^*_{X,Y}(Z)P_{2\times2}(Z,Z')\binom{n}{2}\binom{m}{2},$$

so (12) holds.

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 $(Z,Z')\in E(P_{2\times2})$, we have

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

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 8

$$\tau_{\mathcal{M}_{2\times2}}(\epsilon)\in O(\tau_{\mathcal{M}_{\mathrm{HB}}}(\epsilon)\log(|\Sigma_{r,c}|)n^{2d_m+1}).$$

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

In this section we 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, due to pairs in E_W, is in $O(|\Omega_X(W)|)$.

Throughout this entire section, we therefore 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 $\rho=(\rho_1,\dots,\rho_m)$, $\zeta=(\zeta_1,\dots,\zeta_\delta)$ be the lists of induced row and column sums. Let $\Sigma_{\rho,\zeta}$ denote the set of $m\times\delta$ contingency tables with rows 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 moves of $\mathcal{M}_{2\times2}$. 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 does not 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. This is similar to, but simpler than, what we do later in Section 5.2.3. 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 4.1.2, 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 the following inequalities (see Dyer, Kannan and Mount [12]):

$$\sum_{i=1}^{m-1} \Phi[i,j] \leq \zeta_j \quad \text{for every } j \in [\delta-1] \tag{15}$$

$$\sum_{j=1}^{\delta-1} \Phi[i,j] \leq \rho_i \quad \text{for every } i \in [m-1] \tag{16}$$

$$\sum_{i=1}^{m-1}\sum_{j=1}^{\delta-1} \Phi[i,j] \geq N_W - \rho_m - \zeta_\delta. \tag{17}$$

Let

$$\alpha_{i,j} = \left\lfloor \frac{\min\{\rho_i, \zeta_j\}}{m^2\delta^2} \right\rfloor \tag{18}$$

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

Note that

$$\begin{aligned} \max_{j\in[\delta-1]} \alpha_{i,j} &= \alpha_{i,\delta-1} \text{ for all } i; \\ \max_{i\in[m-1]} \alpha_{i,j} &= \alpha_{m-1,j} \text{ for all } j; \\ \max_{i\in[m-1],j\in[\delta-1]} \alpha_{i,j} &= \alpha_{m-1,\delta-1}. \end{aligned}$$

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 $Q[i,j]$.

For all $i \in [m-1]$, $j \in [\delta-1]$ we define

$$Q^*[i,j] = \left\lfloor \frac{\rho_i\zeta_j}{N_W(\alpha_{i,j}+1)} \right\rfloor. \tag{19}$$

Let $\Upsilon, \Psi \in \Sigma_{\rho,\zeta}$. We are almost ready to start building a path from Υ to Ψ that uses transitions of $\mathcal{M}_{2\times 2}$. We use the following lemma:

Lemma 9 *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 $R[i,j] \le \alpha_{i,j}$, for all $i \in [m-1]$, $j \in [\delta-1]$. Also, we set

$$\begin{aligned}
\Phi^*[i,\delta] &= \rho_i - \sum_{j=1}^{\delta-1} \Phi^*[i,j] \qquad \text{for } i \in [m-1],\\
\Phi^*[m,j] &= \zeta_j - \sum_{i=1}^{m-1} \Phi^*[i,j] \qquad \text{for } j \in [\delta-1],\\
\text{and} \quad \Phi^*[m,\delta] &= \zeta_\delta - \sum_{i=1}^{m-1} \Phi^*[i,\delta].
\end{aligned}$$

Then

(i) $\Phi^* \in \Sigma_{\rho,\zeta}$.

(ii) Under the assumption that $N_W \ge (2m\delta)^2$, we have the following:

$$\begin{aligned}
\Phi^*[i,\delta] &\ge (\alpha_{i,\delta-1}+1) \text{ for all } i \in [m-1]\\
\Phi^*[m,j] &\ge (\alpha_{m-1,j}+1) \text{ for all } j \in [\delta-1]\\
\Phi^*[m,\delta] &\ge 2(\alpha_{m-1,\delta-1}+1).
\end{aligned}$$

Proof: (i) It suffices to show that the inequalities (15), (16) and (17) hold for the defined values of $\Phi^*[i,j]$ with $i \in [m-1]$ and $j \in [\delta-1]$. (Then the definitions of $\Phi^*[i,\delta], \Phi^*[m,j]$, and $\Phi^*[m,\delta]$ are such that the entire $m \times \delta$ table satisfies the row and column sums ρ and ζ, respectively.)

This proof is analogous to that given in Case 2 in Section 4.1.2 when we showed that Inequalities (4), (5), and (6) held there, so we do not repeat that proof here.

(ii) By definition,

$$\begin{aligned}
\Phi^*[i,\delta] &= \rho_i - \sum_{j=1}^{\delta-1} \Phi^*[i,j]\\
&= \rho_i - \sum_{j=1}^{\delta-1} \left(\left\lfloor \frac{\rho_i \zeta_j}{N_W(\alpha_{i,j}+1)} \right\rfloor (\alpha_{i,j}+1) + R[i,j] \right)\\
&\ge \rho_i - \sum_{j=1}^{\delta-1} \left(\frac{\rho_i \zeta_j}{N_W} + \alpha_{i,j} \right)\\
&= \frac{\rho_i \zeta_\delta}{N_W} - \sum_{j=1}^{\delta-1} \alpha_{i,j}\\
&\ge \frac{\rho_i \zeta_\delta}{N_W} - \frac{\delta \rho_i}{m^2 \delta^2}\\
&\ge \frac{\rho_i}{\delta} - \frac{\rho_i}{m^2 \delta}\\
&\ge \frac{\rho_i}{m\delta}\\
&\ge 2\alpha_{i,\delta-1}.
\end{aligned}$$

Then, if $\alpha_{i,\delta-1} \ge 1$ we automatically have $\Phi^*[i,\delta] \ge (\alpha_{i,\delta-1}+1)$. However, if $\alpha_{i,\delta-1} = 0$ then $\alpha_{i,j} = 0$ for all $j \in [\delta-1]$, and there are two subcases to consider. The first subcase is when

$\rho_i\zeta_\delta < N_W$. Then $\rho_i\zeta_j < N_W$ for all $j \in [\delta-1]$, and $\Phi^*[i,j] = 0$ for all $j \in [\delta-1]$. Therefore $\Phi^*[i,\delta] = \rho_i \geq 1$. If $\rho_i\zeta_\delta \geq N_W$, then the fourth line of the derivation above (with $\alpha_{i,j} = 0$) gives $\Phi^*[i,\delta] \geq \rho_i\zeta_\delta/N_W \geq 1$. So in either subcase we have $\Phi^*[i,\delta] \geq 1 = (\alpha_{i,\delta-1}+1)$.

Using a similar argument we conclude that

$$\Phi^*[m,j] \quad \geq \quad (\alpha_{m-1,j}+1).$$

By definition

$$\begin{aligned}
\Phi^*[m,\delta] &= \zeta_\delta - \sum_{i=1}^{m-1}\Phi^*[i,\delta] \\
&= \zeta_\delta - \sum_{i=1}^{m-1}\left(\rho_i - \sum_{j=1}^{\delta-1}\Phi^*[i,j]\right) \\
&= (\rho_m + \zeta_\delta - N_W) + \sum_{i=1}^{m-1}\sum_{j=1}^{\delta-1}\left(\left\lfloor\frac{\rho_i\zeta_j}{N_W(\alpha_{i,j}+1)}\right\rfloor(\alpha_{i,j}+1) + R[i,j]\right) \\
&\geq (\rho_m + \zeta_\delta - N_W) + \sum_{i=1}^{m-1}\sum_{j=1}^{\delta-1}\left(\left\lfloor\frac{\rho_i\zeta_j}{N_W(\alpha_{i,j}+1)}\right\rfloor(\alpha_{i,j}+1)\right) \\
&\geq (\rho_m + \zeta_\delta - N_W) + \sum_{i=1}^{m-1}\sum_{j=1}^{\delta-1}\left(\frac{\rho_i\zeta_j}{N_W} - (\alpha_{i,j}+1)\right) \\
&= \frac{\rho_m\zeta_\delta}{N_W} - \sum_{i=1}^{m-1}\sum_{j=1}^{\delta-1}(\alpha_{i,j}+1) \\
&\geq \frac{\rho_m\zeta_\delta}{N_W} - \frac{m(N_W-\zeta_\delta)}{m^2\delta^2} - (m-1)(\delta-1) \\
&\geq \frac{N_W}{m\delta} - \frac{N_W}{m\delta^2} - (m-1)(\delta-1) \\
&\geq \frac{N_W}{4m\delta} + \left(\frac{N_W}{2m\delta} - \frac{N_W}{m\delta^2}\right) + \left(\frac{N_W}{4m\delta} - (m-1)(\delta-1)\right) \\
&\geq \frac{2\zeta_{\delta-1}}{4m\delta} + 0 + (\delta+m-1) \\
&\geq 2\alpha_{m-1,\delta-1} + 2,
\end{aligned}$$

as required. $\square$

Definition 10 *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 $\Sigma_{\rho,\zeta}$.*

Now we consider a pair $\Upsilon, \Psi \in \Sigma_{\rho,\delta}$. For every $i \in [m-1]$, $j \in [\delta-1]$, we write

$$\Upsilon[i,j] = Q_\Upsilon[i,j](\alpha_{i,j}+1) + R_\Upsilon[i,j],$$

for the unique integer $R_\Upsilon[i,j]$ that satisfies $0 \leq R_\Upsilon[i,j] \leq \alpha_{i,j}$.

Similarly, for every $i \in [m-1]$, $j \in [\delta-1]$, we write

$$\Psi[i,j] = Q_\Psi[i,j](\alpha_{i,j}+1) + R_\Psi[i,j],$$

for the unique $R_\Psi[i,j]$ satisfying $0 \le R_\Psi[i,j] \le \alpha_{i,j}$.

The routing from Υ to Ψ proceeds in two stages. In the first phase, we route flow from Υ to the table Υ^* in the inner domain of $\Sigma_{\rho,\zeta}$ such that $\Upsilon^*[i,j] = Q^*[i,j](\alpha_{i,j}+1) + R_\Upsilon[i,j]$ holds for every $i \in [m-1]$, $j \in [\delta-1]$. Note that $\Upsilon^* \in \Sigma_{\rho,\zeta}$ using the previous lemma.

By defining a similar path between Ψ and Ψ^* and then reversing all the edges, we can route flow from some Ψ^* in the inner domain of $\Sigma_{\rho,\zeta}$ to Ψ such that $\Psi^*[i,j] = Q^*[i,j](\alpha_{i,j} + 1) + R_\Psi[i,j]$ holds for every $i \in [m-1]$, $j \in [\delta-1]$. Similarly, we also have $\Psi^* \in \Sigma_{\rho,\zeta}$.

In the second phase of the routing we show how to route flow from Υ^* to Ψ^* by changing the $R_\Upsilon[i,j]$ to the $R_\Psi[i,j]$ values.

5.2.1 Phase 1

We show how to route Υ to Υ^* in the inner domain of $\Sigma_{\rho,\zeta}$, by only changing $Q[i,j]$ values at any step. For our analysis, we define the following metric on pairs $\Phi, \Phi' \in \Sigma_{\rho,\zeta}$:

$$\begin{aligned}
d(\Phi,\Phi') &= d_1(\Phi,\Phi') + d_2(\Phi,\Phi') + d_3(\Phi,\Phi') + d_4(\Phi,\Phi') \qquad \text{where} \\
d_1(\Phi,\Phi') &= \sum_{i=1}^{m-1}\sum_{j=1}^{\delta-1} |\Phi[i,j] - \Phi'[i,j]| \\
d_2(\Phi,\Phi') &= \frac{N_W+1}{3N_W}\left(\sum_{i=1}^{m-1} |\Phi[i,\delta] - \Phi'[i,\delta]|\right) \\
d_3(\Phi,\Phi') &= \frac{N_W+1}{3N_W}\left(\sum_{j=1}^{\delta-1} |\Phi[m,j] - \Phi'[m,j]|\right) \\
d_4(\Phi,\Phi') &= \frac{N_W+1}{3N_W}|\Phi[m,\delta] - \Phi'[m,\delta]|.
\end{aligned}$$

This metric is used to show the path we construct is moving us closer to Υ^*, and that the length of this path from Υ to Υ^* is bounded by a constant.

We route Υ to Υ^* as a series of moves. Let Υ' denote some interim contingency table on the path from Υ to Υ^*. We choose our next move on this path from amongst four cases.

Case (a): Suppose that $\Upsilon'[m,\delta] < (\alpha_{m-1,\delta-1}+1)$. Then we perform a move to make $\Upsilon'[m,\delta]$ bigger (because we need to "leave room" for our other cases).

We now show there is at least one $\ell \in [m-1]$ such that $\Upsilon'[\ell,\delta] \ge \Upsilon^*[\ell,\delta] + (\alpha_{\ell,\delta-1}+1)$.

Note that by Lemma 9, in this case $\Upsilon'[m,\delta] < \Upsilon^*[m,\delta] - (\alpha_{m-1,\delta-1}+1)$. Suppose (for contradiction) that there is no ℓ as described above, so that

$$\Upsilon'[i,\delta] \quad < \quad \Upsilon^*[i,\delta] + (\alpha_{i,\delta-1}+1) \ \ \text{for } i \in [m-1].$$

By definition

$$\begin{aligned}
\Upsilon^*[i,\delta] &= \rho_i - \Big(\sum_{j=1}^{\delta-1} \Upsilon^*[i,j]\Big) \\
&= \rho_i - \left(\sum_{j=1}^{\delta-1}\left(\left\lfloor \frac{\rho_i\zeta_j}{N_W(\alpha_{i,j}+1)}\right\rfloor (\alpha_{i,j}+1) + R_\Upsilon[i,j]\right)\right).
\end{aligned}$$

Now

$$
\begin{aligned}
\Upsilon'[m,\delta] &= \zeta_\delta - (\sum_{i=1}^{m-1} \Upsilon'[i,\delta]) \\
&> \zeta_\delta - \sum_{i=1}^{m-1} \left(\rho_i - \left(\sum_{j=1}^{\delta-1} \Upsilon^*[i,j] \right) + (\alpha_{i,\delta-1}+1) \right) \\
&= (\zeta_\delta + \rho_m - N_W) + \sum_{i=1}^{m-1} \left(\left(\sum_{j=1}^{\delta-1} \Upsilon^*[i,j] \right) - (\alpha_{i,\delta-1}+1) \right). \qquad (20)
\end{aligned}
$$

Expanding $\sum_{i=1}^{m-1}((\sum_{j=1}^{\delta-1} \Upsilon^*[i,j]) - (\alpha_{i,\delta-1}+1))$

$$
\begin{aligned}
&= \sum_{i=1}^{m-1} \left(\sum_{j=1}^{\delta-1} \left(\left\lfloor \frac{\rho_i \zeta_j}{N_W(\alpha_{i,j}+1)} \right\rfloor (\alpha_{i,j}+1) + R_\Upsilon[i,j] \right) - (\alpha_{i,\delta-1}+1) \right) \\
&\geq \sum_{i=1}^{m-1} \left(\sum_{j=1}^{\delta-1} \left(\frac{\rho_i \zeta_j}{N_W} - (\alpha_{i,j}+1) \right) - (\alpha_{i,\delta-1}+1) \right) \\
&= (N_W - \rho_m - \zeta_\delta) + \frac{\rho_m \zeta_\delta}{N_W} - \sum_{i=1}^{m-1} \sum_{j=1}^{\delta-1} (\alpha_{i,j}+1) - \sum_{i=1}^{m-1} (\alpha_{i,\delta-1}+1) \\
&\geq (N_W - \rho_m - \zeta_\delta) + \frac{\rho_m \zeta_\delta}{N_W} - \frac{m(N_W - \zeta_\delta)}{m^2\delta^2} - (m-1)(\delta-1) - \frac{m\zeta_{\delta-1}}{m^2\delta^2} - (m-1) \\
&\geq (N_W - \rho_m - \zeta_\delta) + \frac{N_W}{m\delta} - \frac{N_W}{m\delta^2} - m\delta + \delta \\
&\geq (N_W - \rho_m - \zeta_\delta) + \frac{N_W}{2m\delta} - m\delta + \delta \\
&\geq (N_W - \rho_m - \zeta_\delta) + \frac{N_W}{4m\delta} + \delta \\
&\geq (N_W - \rho_m - \zeta_\delta) + (\alpha_{m-1,\delta-1}+1) \qquad (21)
\end{aligned}
$$

where the second last line follows since $N_W \geq (2m\delta)^2$, and the last line follows from the definition of $\alpha_{i,j}$ and because $\delta \geq m \geq 2$.

Combining (20) and (21), we have a contradiction to our original assumption that $\Upsilon'[m,\delta] < (\alpha_{m-1,\delta-1}+1)$. Therefore it must be that $\Upsilon'[\ell,\delta] \geq \Upsilon^*[\ell,\delta] + (\alpha_{\ell,\delta-1}+1)$ for some $\ell \in [m-1]$.

Similarly, there must be some $k \in [\delta-1]$ such that $\Upsilon'[m,k] \geq \Upsilon^*[m,k] + (\alpha_{m-1,k}+1)$.

In this case we add the transition $(\Upsilon' \rightarrow \Upsilon'')$ to our path from Υ to Υ^*, where Υ'' is identical to Υ' except for the following entries:

$$
\begin{array}{cc}
\Upsilon''[\ell,k] = \Upsilon'[\ell,k] + (\alpha_{\ell,k}+1) & \Upsilon''[\ell,\delta] = \Upsilon'[\ell,\delta] - (\alpha_{\ell,k}+1) \\
\Upsilon''[m,k] = \Upsilon'[m,k] - (\alpha_{\ell,k}+1) & \Upsilon''[m,\delta] = \Upsilon'[m,\delta] + (\alpha_{\ell,k}+1)
\end{array}
$$

Now we show that $d(\Upsilon'',\Upsilon^*) < d(\Upsilon',\Upsilon^*)$.

$$
\begin{array}{rclcl}
d_1(\Upsilon'', \Upsilon^*) & = & \sum_{i=1}^{m-1} \sum_{j=1}^{\delta-1} |\Upsilon''[i,j] - \Upsilon^*[i,j]| & \leq & d_1(\Upsilon', \Upsilon^*) + (\alpha_{\ell,k}+1) \\
d_2(\Upsilon'', \Upsilon^*) & = & \frac{N_W+1}{3N_W} \left(\sum_{i=1}^{m-1} |\Upsilon''[i,\delta] - \Upsilon^*[i,\delta]| \right) & = & d_2(\Upsilon', \Upsilon^*) - \frac{N_W+1}{3N_W}(\alpha_{\ell,k}+1) \\
d_3(\Upsilon'', \Upsilon^*) & = & \frac{N_W+1}{3N_W} \left(\sum_{j=1}^{\delta-1} |\Upsilon''[m,j] - \Upsilon^*[m,j]| \right) & = & d_3(\Upsilon', \Upsilon^*) - \frac{N_W+1}{3N_W}(\alpha_{\ell,k}+1) \\
d_4(\Upsilon'', \Upsilon^*) & = & \frac{N_W+1}{3N_W} |\Upsilon''[m,\delta] - \Upsilon^*[m,\delta]| & = & d_4(\Upsilon', \Upsilon^*) - \frac{N_W+1}{3N_W}(\alpha_{\ell,k}+1)
\end{array}
$$

The equation for d_4 follows from Lemma 9.

So $d(\Upsilon'', \Upsilon^*) \leq d(\Upsilon', \Upsilon^*) + (\alpha_{\ell,k}+1) - (\alpha_{\ell,k}+1)(N_W+1)/N_W \leq d(\Upsilon', \Upsilon^*) - (\alpha_{\ell,k}+1)/N_W < d(\Upsilon', \Upsilon^*)$.

Case (b): We execute (b) whenever Case (a) does not hold and when *either* $\Upsilon'[\ell,\delta] < \Upsilon^*[\ell,\delta]$ for some $\ell \in [m-1]$ *or* $\Upsilon'[m,j] < \Upsilon^*[m,j]$ for some $j \in [\delta-1]$. Without loss of generality, assume the former. If this is the case, then there must be at least one $k \in [\delta-1]$ such that $\Upsilon'[\ell,k] > \Upsilon^*[\ell,k]$. Since we only change $Q[i,j]$ values during our routing, we know that $\Upsilon'[\ell,k] \geq \Upsilon^*[\ell,k] + (\alpha_{\ell,k}+1)$. Also, since we are not in Case (a), we know $\Upsilon'[m,\delta] \geq (\alpha_{m-1,\delta-1}+1) \geq (\alpha_{\ell,k}+1)$.

In this case we add the transition $(\Upsilon' \to \Upsilon'')$ in our path from Υ to Υ^*, where Υ'' is identical to Υ' except for the following entries:

$$
\begin{array}{rr}
\Upsilon''[\ell,k] = \Upsilon'[\ell,k] - (\alpha_{\ell,k}+1) & \Upsilon''[\ell,\delta] = \Upsilon'[\ell,\delta] + (\alpha_{\ell,k}+1) \\
\Upsilon''[m,k] = \Upsilon'[m,k] + (\alpha_{\ell,k}+1) & \Upsilon''[m,\delta] = \Upsilon'[m,\delta] - (\alpha_{\ell,k}+1)
\end{array}
$$

Now we show $d(\Upsilon'', \Upsilon^*) < d(\Upsilon', \Upsilon^*)$.

$$
\begin{array}{rclcl}
d_1(\Upsilon'', \Upsilon^*) & = & \sum_{i=1}^{m-1} \sum_{j=1}^{\delta-1} |\Upsilon''[i,j] - \Upsilon^*[i,j]| & = & d_1(\Upsilon', \Upsilon^*) - (\alpha_{\ell,k}+1) \\
d_2(\Upsilon'', \Upsilon^*) & = & \frac{N_W+1}{3N_W} \left(\sum_{i=1}^{m-1} |\Upsilon''[i,\delta] - \Upsilon^*[i,\delta]| \right) & \leq & d_2(\Upsilon', \Upsilon^*) + \frac{N_W+1}{3N_W}(\alpha_{\ell,k}-1) \\
d_3(\Upsilon'', \Upsilon^*) & = & \frac{N_W+1}{3N_W} \left(\sum_{j=1}^{\delta-1} |\Upsilon''[m,j] - \Upsilon^*[m,j]| \right) & \leq & d_3(\Upsilon', \Upsilon^*) + \frac{N_W+1}{3N_W}(\alpha_{\ell,k}+1) \\
d_4(\Upsilon'', \Upsilon^*) & = & \frac{N_W+1}{3N_W} |\Upsilon''[m,\delta] - \Upsilon^*[m,\delta]| & \leq & d_4(\Upsilon', \Upsilon^*) + \frac{N_W+1}{3N_W}(\alpha_{\ell,k}+1)
\end{array}
$$

where the expression for $d_2(\Upsilon'', \Upsilon^*)$ follows because $\Upsilon'[\ell,\delta] \leq \Upsilon^*[\ell,\delta] - 1$.

Therefore $d(\Upsilon'', \Upsilon^*) \leq d(\Upsilon', \Upsilon^*) - (\alpha_{\ell,k}+1) + (\alpha_{\ell,k}+1)(N_W+1)/N_W - 2(N_W+1)/3N_W$. This is at most $d(\Upsilon', \Upsilon^*) + (\alpha_{\ell,k}+1)/N_W - 2(N_W+1)/3N_W$. Then because $\alpha_{\ell,k} \leq N_W/(\delta m)^2$, we have $d(\Upsilon'', \Upsilon^*) \leq d(\Upsilon', \Upsilon^*) + 1/(m\delta)^2 + 1/3N_W - 2/3$, and using $N_W \geq (2\delta m)^2$, we obtain $d(\Upsilon'', \Upsilon^*) < d(\Upsilon', \Upsilon^*)$.

Case (c): We execute (c) when Cases (a)-(b) do not hold and *either* $\Upsilon'[\ell,\delta] > \Upsilon^*[\ell,\delta]$ for some $\ell \in [m-1]$ *or* $\Upsilon'[m,j] > \Upsilon^*[m,j]$ for some $j \in [\delta-1]$. Assume the former without loss of generality. Then there must exist some $k \in [\delta-1]$ such that $\Upsilon'[\ell,k] < \Upsilon^*[\ell,k]$, and since we can only change the $\Upsilon[\ell,k]$ values by factors of $(\alpha_{\ell,k}+1)$, we have $\Upsilon'[\ell,k] \leq \Upsilon^*[\ell,k] + (\alpha_{\ell,k}+1)$. Note that since $\Upsilon'[\ell,\delta] > \Upsilon^*[\ell,\delta]$ and using part (ii) of Lemma 9, we know $\Upsilon'[\ell,\delta] \geq (\alpha_{\ell,\delta-1}+1) \geq (\alpha_{\ell,k}+1)$. Because Case (b) does not hold, we know $\Upsilon'[m,k] \geq \Upsilon^*[m,k]$, and, from part (ii) of Lemma 9, this is at least $(\alpha_{m-1,k}+1) \geq (\alpha_{\ell,k}+1)$.

Therefore we can perform the transition ($\Upsilon' \to \Upsilon''$), where Υ'' is identical to Υ' except for the following entries:

$$\begin{array}{rr}\Upsilon''[\ell,k] = \Upsilon'[\ell,k] + (\alpha_{\ell,k}+1) & \Upsilon''[\ell,\delta] = \Upsilon'[\ell,\delta] - (\alpha_{\ell,k}+1) \\ \Upsilon''[m,k] = \Upsilon'[m,k] - (\alpha_{\ell,k}+1) & \Upsilon''[m,\delta] = \Upsilon'[m,\delta] + (\alpha_{\ell,k}+1)\end{array}$$

As before, we now show $d(\Upsilon'', \Upsilon^*) < d(\Upsilon', \Upsilon^*)$.

$$\begin{array}{rclcl} d_1(\Upsilon'',\Upsilon^*) &=& \sum_{i=1}^{m-1}\sum_{j=1}^{\delta-1} |\Upsilon''[i,j] - \Upsilon^*[i,j]| &=& d_1(\Upsilon',\Upsilon^*) - (\alpha_{\ell,k}+1) \\ d_2(\Upsilon'',\Upsilon^*) &=& \frac{N_W+1}{3N_W}\sum_{i=1}^{m-1} |\Upsilon''[i,\delta] - \Upsilon^*[i,\delta]| &\leq& d_2(\Upsilon',\Upsilon^*) + \frac{N_W+1}{3N_W}(\alpha_{\ell,k}-1) \\ d_3(\Upsilon'',\Upsilon^*) &=& \frac{N_W+1}{3N_W}\sum_{j=1}^{\delta-1} |\Upsilon''[m,j] - \Upsilon^*[m,j]| &\leq& d_3(\Upsilon',\Upsilon^*) + \frac{N_W+1}{3N_W}(\alpha_{\ell,k}+1) \\ d_4(\Upsilon'',\Upsilon^*) &=& \frac{N_W+1}{3N_W}|\Upsilon''[m,\delta] - \Upsilon^*[m,\delta]| &\leq& d_4(\Upsilon',\Upsilon^*) + \frac{N_W+1}{3N_W}(\alpha_{\ell,k}+1) \end{array}$$

where the bound on $d_2(\Upsilon'', \Upsilon^*)$ follows using that $\Upsilon'[\ell,\delta] \geq \Upsilon^*[\ell,\delta] + 1$.

Case (d): This is the case when $\Upsilon'[i,\delta] = \Upsilon^*[i,\delta]$ for all $i \in [m-1]$ and $\Upsilon'[m,j] = \Upsilon^*[m,j]$ for all $j \in [\delta-1]$ (so neither Case (b) nor (c) holds), but $\Upsilon'[\ell,k] \neq \Upsilon^*[\ell,k]$ for some $\ell \in [m-1], k \in [\delta-1]$. We also assume Case (a) does not hold, otherwise we would not consider Case (d). In this case we will specify *two* transitions of $\mathcal{M}_{2\times 2}$, $\Upsilon' \to \Upsilon''$ and $\Upsilon'' \to \Upsilon'''$, so that $d(\Upsilon''', \Upsilon^*) < d(\Upsilon', \Upsilon^*)$.

Assume without loss of generality that $\Upsilon'[\ell,k] > \Upsilon^*[\ell,k]$. Hence, there must be some $k' \in [\delta-1]$ such that $\Upsilon'[\ell,k'] < \Upsilon^*[\ell,k']$.

Now because we only change $Q[i,j]$ values on the path from Υ to Υ^*, $\Upsilon'[\ell,k] > \Upsilon^*[\ell,k]$ implies $\Upsilon'[\ell,k] \geq \Upsilon^*[\ell,k] + (\alpha_{\ell,k}+1)$, and $\Upsilon'[\ell,k'] \leq \Upsilon^*[\ell,k'] + (\alpha_{\ell,k'}+1)$.

By Lemma 9 and $\Upsilon'[m,\delta] = \Upsilon^*[m,\delta]$ we know $\Upsilon'[m,\delta] \geq (\alpha_{\ell,k}+1)$. Therefore we can perform the transition ($\Upsilon' \to \Upsilon''$), where Υ'' is identical to Υ' except for the following entries:

$$\begin{array}{rr}\Upsilon''[\ell,k] = \Upsilon'[\ell,k] - (\alpha_{\ell,k}+1) & \Upsilon''[\ell,\delta] = \Upsilon'[\ell,\delta] + (\alpha_{\ell,k}+1) \\ \Upsilon''[m,k] = \Upsilon'[m,k] + (\alpha_{\ell,k}+1) & \Upsilon''[m,\delta] = \Upsilon'[m,\delta] - (\alpha_{\ell,k}+1)\end{array}$$

By Lemma 9 we also know $\Upsilon'[\ell,\delta] \geq (\alpha_{\ell,k'}+1)$ and $\Upsilon'[m,k'] \geq (\alpha_{\ell,k'}+1)$. Then we can perform the transition ($\Upsilon'' \to \Upsilon'''$) by changing the following entries:

$$\begin{array}{rr}\Upsilon'''[\ell,k'] = \Upsilon''[\ell,k'] + (\alpha_{\ell,k'}+1) & \Upsilon'''[\ell,\delta] = \Upsilon''[\ell,\delta] - (\alpha_{\ell,k'}+1) \\ \Upsilon'''[m,k'] = \Upsilon''[m,k'] - (\alpha_{\ell,k'}+1) & \Upsilon'''[m,\delta] = \Upsilon''[m,\delta] + (\alpha_{\ell,k'}+1)\end{array}$$

Finally we show $d(\Upsilon''', \Upsilon^*) < d(\Upsilon', \Upsilon^*)$.

$$\begin{array}{rcl} d_1(\Upsilon''',\Upsilon^*) &=& d_1(\Upsilon',\Upsilon^*) - (\alpha_{\ell,k}+\alpha_{\ell,k'}+2) \\ d_2(\Upsilon''',\Upsilon^*) &=& d_2(\Upsilon',\Upsilon^*) + \frac{N_W+1}{3N_W}|\alpha_{\ell,k}-\alpha_{\ell,k'}| \\ d_3(\Upsilon''',\Upsilon^*) &=& d_3(\Upsilon',\Upsilon^*) + \frac{N_W+1}{3N_W}(\alpha_{\ell,k}+\alpha_{\ell,k'}+2) \\ d_4(\Upsilon''',\Upsilon^*) &=& d_4(\Upsilon',\Upsilon^*) + \frac{N_W+1}{3N_W}|\alpha_{\ell,k}-\alpha_{\ell,k'}|. \end{array}$$

Therefore

$$\begin{array}{rcl} d(\Upsilon''',\Upsilon^*) &=& d(\Upsilon',\Upsilon^*) - (2/3 - 1/3N_W)(\alpha_{\ell,k}+\alpha_{\ell,k'}+2) + (2/3+2/3N_W)|\alpha_{\ell,k}-\alpha_{\ell,k'}| \\ &\leq& d(\Upsilon',\Upsilon^*) - (2/3 - 1/3N_W)(\alpha_{\ell,k}+\alpha_{\ell,k'}+2) + (2/3+2/3N_W)(\alpha_{\ell,k}+\alpha_{\ell,k'}) \\ &\leq& d(\Upsilon',\Upsilon^*) + (\alpha_{\ell,k}+\alpha_{\ell,k'})/N_W - (2-1/N_W)(2/3) \\ &\leq& d(\Upsilon',\Upsilon^*) + 2/m^2\delta^2 - (2/3) \\ &<& d(\Upsilon',\Upsilon^*). \end{array}$$

By a repeated application of these cases, we construct a path joining Υ to some Υ^* that is in the inner domain of $\Sigma_{\rho,\zeta}$. As mentioned, we can also construct such a path joining Ψ to some Ψ^* in the inner domain, and then reverse all of the edges. Following a brief analysis of the flow for this first phase in the next section, we show how to join pairs of elements in the inner domain.

5.2.2 Analysis of flow for Phase 1

The definition of $\alpha_{i,j}$ defines an equivalence class on the set $\Sigma_{\rho,\zeta}$, where $\Phi \equiv \Phi'$ if and only if $\Phi[i,j] = \Phi'[i,j] \bmod (\alpha_{i,j}+1)$ for every $i \in [m-1], j \in [\delta-1]$ (i.e. all the remainders $R[i,j]$ and $R'[i,j]$ are the same).

Note that by definition of the $\alpha_{i,j}$ values, and since $\Phi[i,j] \leq \{\rho_i, \zeta_j\}$ for all $i \in [m-1], j \in [\delta-1]$ for every $\Phi \in \Sigma_{\rho,\zeta}$, any equivalence class contains at most $(m^2\delta^2)^{m\delta}$ contingency tables (there are at most $m^2\delta^2$ choices for each $Q[i,j]$ with $i \in [m-1], j \in [\delta-1]$). Therefore each equivalence class contains a constant number of contingency tables.

The routing scheme given in Cases (a)-(d) defines a path $\Upsilon = \Phi_0, \Phi_1, \ldots, \Phi_t = \Upsilon^*$ from Υ to Υ^*, for every $\Upsilon \in \Sigma_{\rho,\zeta}$. We know Φ_h lies in the same equivalence class as Υ and Υ^* for every h. By our analysis of Cases (a)-(d), we know that for every $h \geq 0$, either $d(\Phi_{h+1}, \Upsilon^*) < d(\Phi_h, \Upsilon^*)$ or $d(\Phi_{h+2}, \Upsilon^*) < d(\Phi_h, \Upsilon^*)$. This means we can define a subsequence of the path $\{\Phi_h\}$ such that (i) the subsequence contains at least every second element of $\{\Phi_h\}$, and (ii) no contingency table ever appears twice in the subsequence. Thus the length of the path from Υ to Υ^* is at most $2(m^2\delta^2)^{m\delta}$.

To analyse the amount of flow from Phase 1 that may pass through $\Phi \in \Sigma_{\rho,\zeta}$, we rely on the fact that for any $\Upsilon \in \Sigma_{\rho,\zeta}$ the path from Υ to Υ^* lies in the equivalence class of Υ. Therefore, for any fixed Ψ, there are at most $(m^2\delta^2)^{m\delta}$ different contingency tables Υ that may pass through Φ on the way to Υ^*. Also, by our bound on the length of the path, we know that for any fixed Ψ and Υ, Φ may occur at most $(m^2\delta^2)^{m\delta}$ times on the path from Υ to Υ^*.

Putting all of this information together, we see that for any fixed Ψ, the flow through any Φ during Phase 1 is at most $(m^2\delta^2)^{2m\delta}$. This implies the total flow through Φ during Phase 1 is at most $|\Sigma_{\rho,\zeta}|\,(m^2\delta^2)^{2m\delta}$.

5.2.3 Phase 2

In this phase we describe how to route Υ^* to Ψ^*, i.e. route flow between any pair of elements in the inner domain of $\Sigma_{\rho,\zeta}$. We know that

$$\begin{aligned} \Upsilon^*[i,j] &= Q^*[i,j](\alpha_{i,j}+1) + R_\Upsilon[i,j] \\ \Psi^*[i,j] &= Q^*[i,j](\alpha_{i,j}+1) + R_\Psi[i,j] \end{aligned}$$

for all $i \in [m-1], j \in [m-1]$, where $Q^*[i,j]$ was defined in (19).

In this phase we route Υ^* to Ψ^* in $(m-1)(\delta-1)$ steps, by "fixing" one remainder at a time. The key to this approach is part (i) of Lemma 9, which shows that for *any* remainders $R[i,j]$ satisfying $R[i,j] \leq \alpha_{i,j}$ for $i \in [m-1], j \in [\delta-1]$, if Φ is defined by

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

for $i \in [m-1], j \in [\delta-1]$, then $\Phi \in \Sigma_{\rho,\zeta}$ (where $\Phi[m,j]$ and $\Phi[i,\delta]$ are defined as in Lemma 9 to satisfy the row and column sums).

Suppose we order the "boxes" of the $m \times \delta$ contingency tables in lexicographic order: $(1,1),(1,2),\ldots,(1,\delta-1),(2,1),\ldots,(m-1,1),\ldots,(m-1,\delta-1)$. Let $h=(h_1,h_2)$ denote any point in this lexicographic order, and we use h^+ to denote the successor to h in this ordering.

Then we can define a series of tables

$$\Upsilon^* = \Phi_{(1,1)}, \Phi_{(1,2)}, \ldots, \Phi_h, \ldots, \Phi_{(m-1,\delta-1)} = \Psi^*$$

by

$$\Phi_h[i,j] = \begin{cases} \Psi^*[i,j] & \text{if } (i,j) \text{ is less than or equal to } h \text{ (and } i \neq m \text{ and } j \neq \delta) \\ \Upsilon^*[i,j] & \text{if } (i,j) \geq h^+ \text{ (and } i \neq m \text{ and } j \neq \delta) \\ \rho_i - \sum_{j=1}^{\delta-1} \Phi_h[i,j] & \text{if } j = \delta \\ \zeta_j - \sum_{i=1}^{m-1} \Phi_h[i,j] & \text{if } i = m. \end{cases}$$

By part (i) of Lemma 9 we know that $\Phi_h \in \Sigma_{\rho,\zeta}$ for all h, and therefore $\Phi_h \to \Phi_{h^+}$ is a transition of $\mathcal{M}_{2\times 2}$. ("Fixing" the (i,j) remainder, i.e. changing $R_\Upsilon[i,j]$ into $R_\Psi[i,j]$, uses a transition of $\mathcal{M}_{2\times 2}$ that involves the four "boxes" $(i,j),(i,\delta),(m,j)$, and (m,δ).)

Note that if we define a dual table $\bar{\Phi}_h$ by

$$\bar{\Phi}_h[i,j] = \begin{cases} \Upsilon^*[i,j] & \text{if } (i,j) \text{ is less than or equal to } h \text{ (and } i \neq m \text{ and } j \neq \delta) \\ \Psi^*[i,j] & \text{if } (i,j) \geq h^+ \text{ (and } i \neq m \text{ and } j \neq \delta) \\ \rho_i - \sum_{j=1}^{\delta-1} \bar{\Phi}_h[i,j] & \text{if } j = \delta \\ \zeta_j - \sum_{i=1}^{m-1} \bar{\Phi}_h[i,j] & \text{if } i = m. \end{cases}$$

then Lemma 9(i) also tells us that $\bar{\Phi}_h \in \Sigma_{\rho,\zeta}$.

Therefore we have a path of length $(m-1)(\delta-1)$ connecting Υ^* to Ψ^* in $\Sigma_{\rho,\zeta}$.

5.2.4 Analysis of flow for Phase 2

We bound the amount of flow from Phase 2 that can pass through any $\Phi \in \Sigma_{\rho,\zeta}$. Similar to Section 4.2, we do this using an encoding. Suppose that we are given

(1) A pair of indices $h=(h_1,h_2)$ specifying that Φ occurs as Φ_h on the path from Υ^* to Ψ^* during phase 2;

(2) The dual contingency table $\bar{\Phi}_h$;

(3) The integers $Q_\Upsilon[i,j]$ for all $i \in [m], j \in [\delta]$;

(4) The integers $Q_\Psi[i,j]$ for all $i \in [m], j \in [\delta]$.

Then we can construct the original pair of tables Υ and Ψ exactly. The information in (1) and (2) first allows us to reconstruct Υ^* and Ψ^*. Then using Υ^*, we may reconstruct $R_\Upsilon[i,j]$ for all i,j since we know (or may compute) $Q^*[i,j]$. Knowing $R_\Upsilon[i,j]$, together with the information in (3), we can find Υ exactly, and in a like manner we can reconstruct Ψ. There are at most $(m^2\delta^2)^{m\delta}$ possible values for the $Q_\Upsilon[i,j]$, and the same number for the $Q_\Psi[i,j]$.

Therefore, for any $\Phi \in \Sigma_{\rho,\zeta}$, the total amount of flow that may pass through Φ during Phase 2 is at most $(m\delta)(m^2\delta^2)^{2m\delta}|\Sigma_{\rho,\zeta}|$.

5.2.5 Finishing up

Combining Phases 1 and 2, the length of any (Υ, Ψ)-path is at most $4(m\delta)^{2m\delta} + (m-1)(\delta-1)$. The total amount of flow that can pass through any $\Phi \in \Sigma_{\rho,\zeta}$ is at most $|\Sigma_{\rho,\zeta}|(2(m\delta)^{4m\delta} + m\delta(m^2\delta^2)^{2m\delta})$.

This establishes the condition on the flow f^* that was cited in Section 5.1. As explained in that section we can alter this flow to get the new flow f, letting us establish Theorem 8, proving rapid mixing of $\mathcal{M}_{2\times 2}$ on the set of $m \times n$ contingency tables $\Sigma_{r,c}$.

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), 1994, pp. 769–779.

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

[3] M. Cryan and M. Dyer, A polynomial time algorithm to approximately count contingency tables when the number of rows is constant. *Journal of Computer and System Sciences*, **67**(2), 2003, pp. 291–310.

[4] J.A. De Loera and B. Sturmfels, Algebraic unimodular counting. *Mathematical Programming, Series B*, **96**, 2003, pp. 183–203.

[5] 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**, 1995, pp. 845–913.

[6] 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, 1995, pp. 15–41.

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

[8] 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.

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

[10] M. Dyer, Approximate counting by dynamic programming. *Proceedings of the 35th ACM Symposium on Theory of Computing*, San Diego, 2003, pp. 693–699.

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

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

[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), 1980, pp. 56–57.

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

[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, 1997, pp. 482–520.

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

[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*, 1999, pp. 121–129.

[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. *SIAM Journal on Computing*, **34**(1), 2004, pp. 195–226 (electronic).

[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. *Foundations of Computational Mathematics (Proceedings of SMALEFEST 2000)*, 2002, pp. 319–324.

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

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

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

[26] V.V. Vazirani, *Approximation Algorithms,* Springer-Verlag, New York, 1999.

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