Minimum Birkhoff-von Neumann Decomposition

Janardhan Kulkarni[1], Euiwoong Lee[2], and Mohit Singh[3]

[1] Microsoft Research, Redmond
[2] Carnegie Mellon University
[3] Microsoft Research, Redmond

Abstract. Motivated by the applications in routing in data centers, we study the problem of expressing an $n \times n$ doubly stochastic matrix as a linear combination using the smallest number of (sub)permutation matrices. The Birkhoff-von Neumann decomposition theorem proves that there exists such a decomposition, but does not give a representation with the smallest number of permutation matrices. In particular, we consider the case when the optimal decomposition uses a constant number of matrices. We show that the problem is not fixed parameter tractable, and design a logarithmic approximation to the problem.

1 Introduction

A non-negative $n \times n$ matrix A is called a doubly stochastic matrix if the sum of entries in every row and every column is equal to 1. A matrix P is called a permutation matrix if every row and every column has exactly one non-zero entry with value 1. A beautiful result of Birkhoff-von Nuemann (BvN Theorem) states that any doubly stochastic matrix can be presented as a linear combination of permutation matrices [13]. Formally, there exist non-negative constants $\lambda_1, \lambda_2, \ldots, \lambda_k$ for some $k > 0$ such that

$$A = \lambda_1 P_1 + \lambda_2 P_2 + \ldots \lambda_k P_k, \quad \text{and} \quad \forall i, \lambda_i > 0 \tag{1}$$

In graph theoretic language, the BvN theorem states that given a non-negative edge weighted bipartite graph, where for every vertex the total weight of edges incident on it is equal to 1 (fractional perfect matching), then it can be represented a convex combination of integral matchings. A proof of the BvN theorem follows from the fact that vertices of the doubly stochastic matrix polytope, called Birkhoff polytope, correspond to permutation matrices [13]. Note that a BvN decomposition of a doubly stochastic matrix may not be unique - both in terms of permutation matrices used and also in the number of matrices used to produce such a decomposition. In this paper, motivated by its applications to routing in data centers, we are interested in BvN representations of doubly stochastic matrices with small number of matrices. However, in our setting, the matrices in the representation can be sub-permutation matrices instead of permutation matrices. A matrix is a sub-permutation matrix if every row and every column has *at most* one non-zero entry with value 1. In graph theoretic language, we seek to represent a fractional perfect matching of a bipartite graph as a linear combination of integral matchings (not necessarily integral perfect matchings). That is,

$$A = \lambda_1 M_1 + \lambda_2 M_2 + \ldots \lambda_k M_k, \quad \text{and} \quad \forall i, \lambda_i > 0 \tag{2}$$

where each M_i is a sub-permutation matrix. We call such a representation *matching decomposition* of doubly stochastic matrix A. In this paper we study the problem of finding a minimum matching decomposition of a doubly stochastic matrix. From Carathodory's Theorem [15] we know that there is a representation with at most $n^2 + 1$ matrices, which can be tightened to $n^2 - 2n + 2$ matrices by applying Marcus-Ree theorem [14]. Yet, for a given doubly stochastic matrix A, the number of matrices needed in a matching decomposition of A can be much smaller than these bounds. Consider for example doubly stochastic matrices that lie on a line connecting two vertices of the Birkhoff polytope; such matrices can be represented as convex combination of only two permutation matrices. We say that a matching decomposition of a doubly stochastic matrix is *minimum* if there is no other matching decomposition with a smaller number of sub-permutation matrices. Our goal is to find a minimum matching decomposition of a doubly stochastic matrix.

Dufossé and Uçar [7] show that the problem of finding a minimum matching decomposition is NP-hard, building on the work of Brualdi and Gibson [3, 4]. Hence, we focus our attention on the case when the optimal representation uses a constant number of sub-permutation matrices. We ask, *is minimum matching decomposition fixed parameter tractable in k?*, where k is number of matrices used in the optimal solution. In other words, is there an algorithm that finds a matching decomposition with the minimum number of sub-permutation matrices that runs in time that is polynomial in n but can depend arbitrarily in k.

Our main motivation to study the fixed parameter tractability of the problem comes from the application of the BvN theorem in traffic routing in reconfigurable data centers, and more broadly in software defined networks. One of the emerging technologies to connect servers within a data center is to use light (laser). An advantage of such an approach is that as traffic between servers changes over time topology can be reconfigured. In such contexts, the Birkhoff-von Nuemann decomposition theorem has been extensively used to route traffic among servers [2, 4–6, 9, 10, 18, 20].

In routing applications, a doubly stochastic matrix represents traffic that needs to be routed among a set of n servers. (Although traffic matrices need not be doubly stochastic, in the applications of interest it is reduced to a doubly stochastic matrix by appropriate scaling. See [18] more details.) A routing decision at any time step is a matching between senders and receivers, a BvN decomposition of the traffic matrix gives a schedule to route traffic. Switching between matchings, however, involves reconfiguring hardware (moving laser pointers and receivers) that comes at a cost. Consequently, finding a decomposition with few permutation matrices improves performance [2, 12, 18]. Moreover, the empirical evidence shows that the number of servers that are active is small compared to the total number of servers in a data center, and hence it is observed that BvN decomposition has a small support [12]. There is a growing body of work in understanding BvN decompositions in the context of reconfigurable data center architectures and we refer the reader to [2, 10, 18] and references there in for more details. Apart from its applications in data centers, the BvN theorem has also been used in routing in wireless networks; we refer the readers to [4, 5] more details.

In this paper we show that the minimum matching decomposition problem is not fixed parameter tractable.

Theorem 1. *There exists a universal constant $k \geq 4$ for which it is NP-hard to find a minimum matching decomposition of a doubly stochastic matrix that admits a decomposition into k matchings.*

Since the optimal value k is a constant, our result implies that the problem is APX-hard and does not admit a PTAS. In addition to fixed parameter tractable algorithms, it also rules out an algorithm that runs in time $n^{f(k)}$ for any function f. Interestingly, the problem is polynomial time solvable for $k = 2, 3$ (See Theorem 8 in Appendix (B)), and we believe that it becomes NP-hard for $k = 4$. On the positive side, we show that there exists a logarithmic approximation to the problem.

Theorem 2. *There is an algorithm that is $O(\log k)$ approximation to minimum matching decomposition problem, which runs in time polynomial in n and doubly exponential in k.*

In particular, our algorithm finds a representation of A using at most $O(\log k) \cdot k$ sub-permutation matrices if there is an optimal solution with at most k sub-permutation matrices.

There is an algorithmic proof of the BvN decomposition theorem [7], and it is natural to ask what is approximation factor of that algorithm. We show an exponential lower bound on the approximation factor of the BvN decomposition algorithm. We also show that our lower bound example extends to all known variants of BvN decomposition algorithms. See Theorem 7 in Appendix (B) for a proof. Another related question, in a spirit similar to the approximate Carathéodory's Theorem, is if there is a small presentation of ϵ-close matrix of A. For any $n \times n$ doubly stochastic matrix A, call a $n \times n$ matrix A' ϵ-close if $\forall a_{ij} \in A, a'_{ij} \in A', |a_{ij} - a'_{ij}| \leq \epsilon$. We show that this problem can be solved optimally, and there exists a tight representation using at most $1/\epsilon$ matrices (Theorem 9, Appendix (B)). This is an improvement over the result of Barman [1] who showed a representation using at most $O(\log(n)/\epsilon^2)$ matrices.

Discussion Our results leave open several important questions. The most interesting question is if there is a constant factor fixed parameter tractable approximation algorithm to the problem. Another interesting direction is to understand the approximability of the problem for the general case when k is not a constant.

2 Approximation Algorithm

In this section we prove Theorem 2. We design an algorithm that runs in time $f(n)g(k)$, where $f(n)$ is polynomial in n and $g(k)$ is doubly exponential.

Overview of the Algorithm Our algorithm consists of three main steps. In the first step, our algorithm finds a set of values $\lambda_1, \lambda_2, \dots \lambda_k$ such that every entry in A can be represented as sum of some subset of the values. We find such a set of λ values by a combination of a brute force search and solving a sequence of linear equations, and hence this step of our algorithm runs in time that is exponential in k. Given λ values, we reduce our problem to a combinatorial problem called *generalized bipartite edge coloring* (GBEC), which is a generalization of the bipartite edge coloring problem. Here, for each edge we are given a list S_e where each element $s \in S_e$ is a subset of $\{\lambda_1, \lambda_2, \dots \lambda_k\}$ (we allow the same number to appear multiple times). Our goal is to assign each edge e an element $s_e \in S_e$ such that the maximum degree in the induced bipartite graph corresponding to each λ_i is as small as possible. An edge e is in the induced bipartite graph for λ_i if $\lambda_i \in s_e$ and e is assigned s_e. Note that if there is a matching decomposition of A using k matchings and our "guess" of λ values was correct, then the induced bipartite graph corresponding to each λ_i would be a matching. Finally, we give a LP rounding based algorithm to get a logarithmic approximation to the generalized bipartite edge coloring problem.

2.1 Computing λ values

Let $w_1, w_2, \dots w_\ell$ be the set of distinct elements of matrix A. We say that a set of k real numbers $S = \{\lambda_1, \lambda_2, \dots \lambda_k\}$ is *feasible* if every entry w_i can be represented as a sum of subset of values from S. That is, for all w_i, $\exists s \subseteq S$, such that $w_i = \sum_{\lambda_i \in s} \lambda_i$. Our first observation is that one can find a feasible set for distinct elements of A in time that is polynomial in n and exponential in k.

Lemma 1. *A feasible set representing all distinct entries in A can be found in time $O(2^{k2^k})$.*

Proof. (sketch) First we observe that there cannot be more than 2^k distinct entries in A. This is true since every distinct entry in A needs to be represented as some subset of $[k]$, and there are at most 2^k such subsets. Now we set up a sequence of linear equations to find a feasible set.

The variables of our linear equations are $x_1, x_2, x_3 \dots \dots x_k$, which are intended to represent a feasible set of $\lambda_1, \lambda_2, \dots \lambda_k$. For every distinct entry a_{ij}, we "guess" a subset $s \subset [k]$ such that $\sum_{i \in s} x_i = a_{ij}$ and add it as a constraint in our linear equations. Therefore, our system of linear equations has k variables and a constraint for every distinct entry in A. We write a sequence of such linear equations for every possible guess of a_{ij} values. If A admits a matching decomposition using at most k distinct matrices, then at least one of 2^{k2^k} system of linear equations should have a feasible solution, which will find a feasible subset of λ values. □

We remark that although the above algorithm finds a feasible set of λ values, it is only a necessary condition for a matrix to have matching decomposition using at most k entries. For the sake of getting a good approximation algorithm, as we show in the sequel, this is also sufficient.

Given the $\lambda_1, \lambda_2, \ldots \lambda_k$ values, our next step is to find a set of sub-permutation matrices such that A can be represented as a linear combination of sub-permutation matrices. Our idea to get a good approximation algorithm is to first find an *intermediate representation* of A by a set of k incidence matrices corresponding to a set of bipartite graphs such that maximum degree of any vertex is minimized.

Formally, we first represent

$$A = \lambda_1 B_1 + \lambda_2 B_2 + \ldots \lambda_k B_k, \tag{3}$$

where B_i is 0-1 matrix; that is, all non-zero entries in B_i have value 1. Further, each B_i corresponds to a bipartite graph. We call this bipartite graph-decomposition of matrix A. Our first intermediate goal is to represent A using bipartite graphs $B_1, B_2, \ldots B_k$ such that degree of every vertex in B_i for all $i \in [k]$ is as small as possible. Let $\Delta = \max_i \{\max_{v \in B_i} \delta_i(v)\}$, where $\delta_i(v)$ is the degree of vertex v in B_i.

Lemma 2. *If A admits a bipartite graph-decomposition (3) with maximum degree Δ, then there is a matching decomposition of A with at most $k\Delta$ sub-permutation matrices.*

Proof. To prove the lemma we make use of the Vizing's theorem [19], which states that any bipartite graph with maximum degree Δ can be decomposed into Δ matchings. Therefore, by decomposing every B_i into at most Δ matchings (and replicating the λ_i coefficient) we get a matching decomposition with at most $k\Delta$ matchings. □

The rest of the section is devoted to finding a bipartite decomposition of A such that maximum degree is minimized. We formulate this problem as a linear program.

2.2 LP Formulation

For each entry $a_{ij} \in A$, let S_{ij} denote the set of all subsets of $[k]$ such that a_{ij} is equal to sum of the corresponding λ's. That is,

$$\forall s \in S_{ij}, s \subseteq [k], a_{ij} = \sum_{i \in s} \lambda_i.$$

Let $G(A) = (V, E)$ denote the bipartite graph represented by A, where the weight of an edge $e := (i, j)$ is equal to a_{ij}. For every subset $s \in S_{ij}$ and for every entry a_{ij}, we create a variable x_{es} where $e := (i, j)$. The LP relaxation we write has the following three simple constraints:

$$\sum_{s\in S_e} x_{es} = 1, \qquad \forall e \in E \tag{4}$$

$$\sum_{e:e\to v} \sum_{s:i\in s, s\in S_e} x_{es} \leq 1, \qquad v \in V, i \in [k] \tag{5}$$

$$x_{es} \geq 0 \tag{6}$$

The first set of constraints (4) ensure that every edge e is assigned some valid set $s \in S_e$. The second set of constraints (5) are matching constraints: Since A admits a matching decomposition, for every coefficient λ_i, there is a bipartite graph B_i such that the maximum degree is at most 1.

2.3 Rounding

Let $\mathbf{x}$ be a feasible solution to the above LP (4 - 6). Let x^*_{es} be the value of variable x_{es} in the solution $\mathbf{x}$. We do a randomized rounding of $\mathbf{x}$. That is, we assign edge e to subset $s \in S_e$ with probability x^*_{es}. The first set constraints of from LP (4) imply that every edge e gets a $s \in S_e$. Thus, we get an intermediate representation of A using bipartite graphs, and it remains to bound Δ of this representation.

Let $\delta_i(v)$ be the degree of vertex v in the bipartite graph i after the randomized rounding. (That is, bipartite graph corresponding to coefficient λ_i). Note that $\delta_i(v)$ is a random variable and from the second set of LP constraints we have $\mathbb{E}[\delta_i(v)] \leq 1$. We now show that probability that $\delta_i(v) \geq O(\log k)$ is at most $O(1/k^2)$. Towards that we need the following version of Chernoff bound.

Theorem 3. *(Chernoff Bounds [17]) Let $X_1, X_2, \cdots X_n$ be n independent random variables with $X_i = 1$ with probability p_i and $X_i = 0$ with probability $1 - p_i$. Let $X = \sum_i X_i$. Then for any $\epsilon > 0$,*

$$\mathbb{P}(X \geq (1+\epsilon)\mathbb{E}(X)) \leq exp(-\epsilon^2/3 \cdot \mathbb{E}(X)).$$

A simple application of the above theorem gives the following result.

Lemma 3. $\mathbb{P}(\delta_i(v) \geq c\log(k)) \leq \frac{1}{k^3}$*, for* $c \geq 10$.

Proof. First we note that $\delta_i(v)$ is the sum of independent random variables X_e. Formally,

$$\delta_i(v) = \sum_{e:e\to v} \sum_{s:i\in s} \mathbf{1}(e \to s)$$

The indicator function $\mathbf{1}(e \to s)$ denotes that edge e is assigned set s in our random coloring. From the constraints of LP

$$\mathbb{E}[\delta_i(v)] = \sum_{e:e\to v} \sum_{s:i\in s} \mathbb{P}(e \to s) = \sum_{e:e\to v} \sum_{s:i\in s} x^*_{es} \leq 1$$

Now we apply the Chernoff bound by taking $\epsilon = 9\log(k)$ for the random variable $\delta_i(v)$ to complete the proof. □

However, the fact that $\mathbb{P}(\delta_i(v) \geq c\log(k)) \leq \frac{1}{k^3}$ for fixed v and i does not guarantee that $\mathbb{P}(\max_{i,v}\{\delta_i(v)\} \leq c\log(k))$ is non-zero. This is because there are nk events corresponding to every pair of (v,i) and probability of failure of each event is only polynomial in $1/k$. Since n can be much larger than k, union bound does not give a non-zero probability for $\mathbb{P}(\max_{i,v}\{\delta_i(v)\} \leq c\log(k))$. To overcome this, we apply Lovaśz Local Lemma (LLL).

Theorem 4. *(Lovaśz Local Lemma [8]) Let $T_1, T_2 \ldots T_m$ be events such that: 1) $\mathbb{P}[T_i] \leq p$ 2) each T_i depends on at most d other events. 3) $4 \cdot p \cdot d \leq 1$. Then there is a nonzero probability that none of the events occurs; $\mathbb{P}(\bigcap_{i=1}^m \overline{T_i}) > 0$.*

We now apply LLL to show that when we do randomized rounding of LP solution, with non-zero probability no vertex gets a degree more than $O(\log k)$ in any bipartite graph B_i.

Lemma 4. *With non-zero probability, no vertex $v \in B_i, i \in [k]$, gets a degree more than $c\log k$ for some constant $c \geq 10$.*

Proof. We prove the lemma by applying LLL. We define a set of bad events $T_{i,v}$ corresponding to every vertex v in the bipartite graph B_i. An event $T_{i,v}$ is bad if $\delta_i(v)$ is greater than $10\log k$. From Lemma (3), we have $\mathbb{P}(T_{i,v}) \leq \frac{1}{k^3}$.

Next, we bound dependency degree of an event $T_{i,v}$. Towards this, we define a bipartite graph with a vertex for every bad event $T_{i,v}$ in the left-hand side. The right-hand side of this bipartite graph consists of random variables X_e, where X_e represents the random variable for an edge e in $G(A)$. Now, observe that each bad event $T_{i,v}$ depends on at most k random variables X_e. This follows from the observation that at most k edges can be incident at a vertex v in $G(A)$ since A admits a BvN decomposition of size k. Further, each random variable X_e can affect at most $2k$ bad events corresponding to (i,v), where $i \in [k]$ and $e \to v$. This means that each $A_{i,v}$ depends on at most $2k^2$ other events.

With these two facts, it is easy to verify the LLL condition that $4 \cdot p \cdot d \leq 1$ since $p \leq 1/k^3$ and $d \leq 2k^2$. Thus, with non-zero probability none of the bad events occur. By applying Moser-Tardos [16] framework, we can find such an outcome in polynomial time. □

Putting all the pieces together we have the following theorem.

Theorem 5. *The randomized rounding of LP (4 - 6) is an $O(\log(k))$ approximation to the problem of finding minimum matching decomposition of a doubly stochastic matrix.*

Proof. From Lemma 4, we conclude that there is a decomposition of A into bipartite graphs such that maximum degree (Δ) is at most $O(\log(k))$. Then it follows from Lemma 2 that there is a matching decomposition with at most $O(k\log k)$ matrices. This completes the proof. □

Now we show that our rounding of LP is almost optimal by showing an instance where LP has $\Omega(\log k/\log\log k)$ integrality gap.

2.4 LP Gap

Our LP in Section 2.2 fractionally assigns a subset s_e of $[k]$ to each edge e, where s_e must belong to the given collection S_e of subsets. For each integral solution $\{s_e\}_{e\in E}$ with $s_e \in S_e$, recall that $\delta_i(v) = |\{e \in E : e \to v \text{ and } i \in s_e\}|$. Each edge e satisfies $\sum_{i\in s} \lambda_i = \sum_{i\in s'} \lambda_i$ for every $s, s' \in S_e$, so S_e cannot be an arbitrary collection of subsets. We show that if we ignore restrictions given by $\{\lambda_i\}_{i\in[k]}$ and allow S_e to be an arbitrary collection of subsets, our LP in Section 2.2 has a gap of $\Omega(\frac{\log k}{\log\log k})$.

Lemma 5. *There is a bipartite graph $G = (V, E)$, $k \in \mathbb{N}$, and $\{S_e \subseteq 2^{[k]}\}_{e\in E}$ such that the LP (4 - 6) is feasible, but for any integral solution $\{s_e\}_{e\in E}$ with $s_e \in S_e$, there exist at least $k/2$ numbers $p_1, \ldots, p_{k/2} \in [k]$ such that for each p_i, $\max_v \delta_{p_i}(v) \geq \Omega(\frac{\log k}{\log\log k})$.*

Proof. We first present an instance that is feasible for the LP and each integral solution has one $p \in [k]$ with $\max_v \delta_p(v) \geq \Omega(\frac{\log k}{\log\log k})$. Then we show how to extend this construction to achieve $p_1, \ldots, p_{k/2}$. Our instance is parameterized by integers k and d, where d divides k. It has $d+1$ vertices $\{u, v_1, \ldots, v_d\}$ and d edges $\{(u, v_i)\}_{1\leq i\leq d}$. Let $e_i := (u, v_i)$. For each i, e_i is associated with d subsets $s_{i,1}, \ldots, s_{i,d}$ in the following way: for each element $p \in [k]$, pick a random number $j \in [d]$ and put p into $s_{i,j}$. So for each i, $s_{i,1}, \ldots, s_{i,d}$ are disjoint and their union is $[k]$. Fractionally, if each e_i picks every $s_{i,j}$ with $\frac{1}{d}$, at vertex u, every $p \in [k]$ is picked exactly once.

We want to claim that if each e_i integrally picks one s_{i,j_i}, one $p \in [k]$ is picked many times. Fix one integral solution $(j_1, \ldots, j_d)$, which represents that e_i picks s_{i,j_i}. For each $p \in [k]$, the number of occurrences of p in $s_{1,j_1}, \ldots, s_{d,j_d}$ is a random variable drawn from $B(d, \frac{1}{d})$, the binomial distribution with d trials and probability $1/d$. Note that each $p \in [k]$ is independent. Let m be an integer fixed later, and let $P_m := \Pr[X \geq m]$ where X is drawn from $B(d, \frac{1}{d})$. The probability that every $p \in [k]$ occurs strictly less than m times is $(1 - P_m)^k \leq e^{-P_m k}$.

We can lower bound P_m by

$$P_m \geq \Pr[X = m] = \binom{d}{m}(\frac{1}{d})^m(1 - \frac{1}{d})^{d-m} \geq \frac{(d-m)^m}{m^m} \cdot \frac{1}{d^m} \cdot \frac{1}{e} \geq \frac{1}{(2m)^m},$$

for $d \geq 4m$. There are d^d tuples $(j_1, \ldots, j_d)$, so as long as

$$(1 - P_m)^k \cdot d^d < 1 \Leftarrow e^{-P_m k} \cdot e^{d\log d} < 1 \Leftarrow \frac{1}{(2m)^m} > \frac{d\log d}{k},$$

there is an instance where for each integral solution $\{s_e\}_{e\in E}$ there exists $p \in [k]$ with $\max_v \delta_p(v) \geq m$. It works when $d = \sqrt{k}$ and $m = \Omega(\frac{\log k}{\log\log k})$.

To extend the above strategy to find $p_1, \ldots, p_{k/2}$ simultaneously for each integral solution, create ℓ disjoint copies of the above instance independently. As the above argument, it will be feasible for the LP if each edge picks each of d subsets with $\frac{1}{d}$. Fix an integral solution (there are now $(d^d)^\ell = d^{d\ell}$ choices).

For each $p \in [k]$, the probability that the number occurrences of p is less than m in each copy is $(1 - P_m)^\ell$. The probability that there are $k/2$ numbers that occur less than m times in each copy is at most $\binom{k}{k/2} \cdot (1 - P_m)^{\frac{\ell k}{2}} \le e^k \cdot e^{-P_m \frac{\ell k}{2}}$. Union bounding over all $d^{d\ell}$ choices, as long as (take $\ell \gg k$)

$$\begin{aligned} & e^{d\ell \log d} \cdot e^k \cdot e^{-P_m \frac{\ell k}{2}} < 1 \\ \Leftarrow\ & P_m \frac{\ell k}{2} > d\ell \log d + k \\ \Leftarrow\ & P_m k > 4d \log d \\ \Leftarrow\ & \frac{1}{(2m)^m} > \frac{d \log d}{4k}, \end{aligned}$$

there is an instance where for every integral solution, there are at least $k/2$ numbers $p_1, \ldots, p_{k/2}$ that occur more than m times in a single copy. Since all edges in one copy is incident on a single vertex, for each p_i, $\max_v \delta(v)_{p_i} \ge m$. It works again with $d = \sqrt{k}$ and $m = \Omega(\frac{\log k}{\log \log k})$. $\square$

Remark: There is a natural configuration LP for the problem which overcomes our intergrality gap example. It would be interesting to see if it can be rounded to get a constant approximation to the problem.

3 Hardness Results

In this section we show that it is NP-hard to find a minimum matching decomposition of a doubly stochastic matrix even when $k = O(1)$. To obtain the hardness for minimum matching decomposition, we show NP-hardness *generalized bipartite edge coloring* (GBEC) introduced in Section 2. Recall that this problem was formally defined as follows.

- Input: A bipartite graph $G = (V, E)$, an integer $k \in \mathbb{N}$. For each edge $e \in E$, a collection of subsets $S_e \subseteq 2^{[k]}$.
- Output: For each edge e, $s_e \in S_e$.
- Goal: Minimize $\max_{i \in [k], v \in V} \delta_i(v)$, where $\delta_i(v) := |\{e \in E : e \to v \text{ and } i \in s_e\}|$.

Our algorithm in Section 2 gives an $O(\log k)$-approximation algorithm. We complement our algorithmic result by showing that GBEC is NP-hard even when $k = 28$ in Section 3.1. We extend this result to show NP-hardness of minimum matching decomposition when k is a universal constant in Appendix (A).

3.1 Hardness of GBEC

We prove the following theorem showing NP-hardness of GBEC even when k is a constant.

Theorem 6. *Given an instance of* GBEC *with* $k = 28$*, it is NP-hard to distinguish whether the optimal value is* 1 *or higher.*

We reduce from Edge Coloring in 3-regular (general) graphs to GBEC. Vizing's theorem shows that every 3-reulgar graph can be 4-edge-colorable, but Holyer [11] shows that it is NP-hard to decide whether it is 3-edge-colorable or 4-edge-colorable.

Given a cubic graph $G = (V, E)$ for Edge Coloring, let $c^1 : V \mapsto [4]$ be a 4-vertex coloring, and $c^2 : E \mapsto [4]$ be a 4-edge coloring (both are easily computable). The instance of GBEC is defined as follows.

- $G' = (V', E')$ where $V' = V \cup E$ and $E' = \{(v, e) : v \in V, e \in E, v \in e\}$. It is clearly bipartite.
- Each edge $e = (u, v) \in E$ is divided into two edges $(u, e), (v, e)$ in G'. Arbitrarily call one of them head and the other tail.
- $k = 28$.
- Define $\{T_{i,d}\}_{i,d}$ for each $i \in [3]$ and $d \in \{\mathsf{head}, \mathsf{tail}\}$ such that
  - $T_{i,d} \subseteq [4]$ and $|T_{i,d}| = 2$ for each i, d.
  - $T_{i,\mathsf{head}} \cup T_{i,\mathsf{tail}} = [4]$ for each i.
  - $T_{i,d} \cap T_{j,d'} \neq \emptyset$ for every $i \neq j$ and d, d'.
  - Simply, $T_{1,\mathsf{head}} = \{1, 2\}, T_{1,\mathsf{tail}} = \{3, 4\}, T_{2,\mathsf{head}} = \{1, 3\}, T_{2,\mathsf{tail}} = \{2, 4\}, T_{3,\mathsf{head}} = \{1, 4\}, T_{3,\mathsf{tail}} = \{2, 3\}$ works.
- Fix an edge $e' = (e, u) \in E'$ where $e = (u, v) \in E$ and $d \in \{\mathsf{head}, \mathsf{tail}\}$ be the type of e'. $S_{e'}$ has the following three subsets. When $T \subseteq [4]$ and $j \in \mathbb{N}$, let $(T + j)$ denote $\{t + j : t \in T\}$.
  - Let $s_{e',i} := \{i + 3(c^1_u - 1)\} \cup (T_{i,d} + (4c^2_e + 8))$.
  - $S_{e'} := \{s_{e',i} : i \in [3]\}$.

Intuition For any of subset $s_{e',i} = \{i + 3(c^1_u - 1)\} \cup (T_{i,d} + (4c^2_e + 8))$, note that $\{i + 3(c^1_u - 1)\} \subseteq \{1, \ldots, 12\}$ and $(T_{i,d} + (4c^2_e + 8)) \subseteq \{13, \ldots, 28\}$. Out of 28 colors for GBEC, $\{13, \ldots, 28\}$ ensures that for each $e \in E$, its head and tail get the same color in [3] for Edge Coloring. The subset $\{1, \ldots, 12\}$ checks that for each vertex $v \in V$, all incident edges have different colors. We use many ($k = 28$) colors to perform these checks because we want these checks only between desired edges (decided by c^1 and c^2).

Lemma 6 (Completeness). *If G is* 3*-edge colorable, the optimum of* GBEC *for G' is* 1.

Proof. Let $c^* : E \mapsto [3]$ be an edge 3-coloring of G. For edge $e = (u, v) \in E$ with head (e, u) and tail (e, v), let

$$s_{(e,u)} \leftarrow s_{(e,u),c^*_e} = \{c^*_e + 3(c^1_u - 1)\} \cup (T_{c^*_e,\mathsf{head}} + (4c^2_e + 8)),$$
$$s_{(e,v)} \leftarrow s_{(e,u),c^*_v} = \{c^*_e + 3(c^1_v - 1)\} \cup (T_{c^*_e,\mathsf{tail}} + (4c^2_e + 8)).$$

Let $e'_1, e'_2 \in E'$ be adjacent edges. There are two cases.

- $e'_1 = (e, u)$ and $e'_2 = (e, v)$ for some edge $e = (u, v) \in E$: One of them is head, and the other is tail. In $\{13, \ldots, 28\}$, they are disjoint since $T_{c^*_e,\mathsf{head}}$ and $T_{c^*_e,\mathsf{tail}}$ are disjoint. In $\{1, \ldots, 12\}$, they are disjoint $c^1_u \neq c^1_v$.
- $e'_1 = (e, u)$ and $e'_2 = (f, u)$ for $e \neq f \in E$: In $\{13, \ldots, 28\}$, they are disjoint since $c^2_e \neq c^2_f$. In $\{1, \ldots, 12\}$, they are disjoint $c^*_e \neq c^*_f$.

Therefore, the optimum of GBEC for G' is 1. □

Lemma 7 (Soundness). *If the optimum of* GBEC *for* G' *is* 1*,* G *is* 3*-edge colorable.*

Proof. Let $c : E' \to [3]$ be a solution of GBEC where each $e' \in E'$ chooses $s_{e',c(e')}$ and the optimum is 1. The fact that the optimum is 1 means that for every adjacent $e', f' \in E'$, $s_{e',c(e')}$ and $s_{f',c(f')}$ are disjoint.

For each edge $e = (u, v) \in E$, let $e'_1 = (e, u)$ be its head and $e'_2 = (e, v)$ be its tail. We must have $c(e'_1) = c(e'_2)$ to have the optimum at most 1 since otherwise $(T_{c(e'_1),\mathsf{head}} + (4c^2_e + 8)) \subseteq s_{e'_1,c(e'_1)}$, $(T_{c(e'_2),\mathsf{tail}} + (4c^2_e + 8)) \subseteq s_{e'_2,c(e'_2)}$, and $T_{c(e'_1),\mathsf{head}}$ and $T_{c(e'_2),\mathsf{tail}}$ intersect if $c(e'_1) \neq c(e'_2)$.

For any $e = (u, v) \in E$ and $f = (u, w) \in E$ that meet at vertex u, let $e' = (u, e) \in E'$ and $f' = (u, f) \in E'$. We must have $c(e') \neq c(f')$ to have the optimum at most 1 since otherwise $c(e') + 3(c^1_u - 1)$ is contained in both $s_{e',c(e')}$ and $s_{f',c(f')}$.

Therefore, c gives the same color to head and tail of the same edge of E, and adjacent edges must have different colors. Therefore, it gives a proper 3-edge coloring of G. □

References

1. Siddharth Barman. Approximating nash equilibria and dense bipartite subgraphs via an approximate version of caratheodory's theorem. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing*, pages 361–369. ACM, 2015.
2. Shaileshh Bojja Venkatakrishnan, Mohammad Alizadeh, and Pramod Viswanath. Costly circuits, submodular schedules and approximate carathéodory theorems. In *Proceedings of the 2016 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Science*, pages 75–88. ACM, 2016.
3. Richard A Brualdi. Notes on the birkhoff algorithm for doubly stochastic matrices. *Canadian Mathematical Bulletin*, 25(2):191–199, 1982.
4. Richard A Brualdi and Peter M Gibson. Convex polyhedra of doubly stochastic matrices. i. applications of the permanent function. *Journal of Combinatorial Theory, Series A*, 22(2):194–230, 1977.
5. Cheng-Shang Chang, Wen-Jyh Chen, and Hsiang-Yi Huang. On service guarantees for input-buffered crossbar switches: a capacity decomposition approach by birkhoff and von neumann. In *Quality of Service, 1999. IWQoS'99. 1999 Seventh International Workshop on*, pages 79–86. IEEE, 1999.
6. Kai Chen, Ankit Singla, Atul Singh, Kishore Ramachandran, Lei Xu, Yueping Zhang, Xitao Wen, and Yan Chen. Osa: an optical switching architecture for data center networks with unprecedented flexibility. *IEEE/ACM Transactions on Networking*, 22(2):498–511, 2014.

7. Fanny Dufossé and Bora Uçar. Notes on birkhoff–von neumann decomposition of doubly stochastic matrices. *Linear Algebra and its Applications*, 497:108–115, 2016.
8. Paul Erdos and László Lovász. Problems and results on 3-chromatic hypergraphs and some related questions. *Infinite and finite sets*, 10(2):609–627, 1975.
9. Nathan Farrington, George Porter, Sivasankar Radhakrishnan, Hamid Hajabdolali Bazzaz, Vikram Subramanya, Yeshaiahu Fainman, George Papen, and Amin Vahdat. Helios: a hybrid electrical/optical switch architecture for modular data centers. *ACM SIGCOMM Computer Communication Review*, 40(4):339–350, 2010.
10. Monia Ghobadi, Ratul Mahajan, Amar Phanishayee, Nikhil Devanur, Janardhan Kulkarni, Gireeja Ranade, Pierre-Alexandre Blanche, Houman Rastegarfar, Madeleine Glick, and Daniel Kilper. Projector: Agile reconfigurable data center interconnect. In *Proceedings of the 2016 conference on ACM SIGCOMM 2016 Conference*, pages 216–229. ACM, 2016.
11. Ian Holyer. The np-completeness of edge-coloring. *SIAM Journal on computing*, 10(4):718–720, 1981.
12. He Liu, Matthew K Mukerjee, Conglong Li, Nicolas Feltman, George Papen, Stefan Savage, Srinivasan Seshan, Geoffrey M Voelker, David G Andersen, Michael Kaminsky, et al. Scheduling techniques for hybrid circuit/packet networks. *ACM CoNEXT*, 2015.
13. László Lovász and Michael D Plummer. *Matching theory*, volume 367. American Mathematical Soc., 2009.
14. Marvin Marcus and Rimhak Ree. Diagonals of doubly stochastic matrices. *The Quarterly Journal of Mathematics*, 10(1):296–302, 1959.
15. Jiří Matoušek. *Lectures on discrete geometry*, volume 108. Springer New York, 2002.
16. Robin A Moser and Gábor Tardos. A constructive proof of the general lovász local lemma. *Journal of the ACM (JACM)*, 57(2):11, 2010.
17. Rajeev Motwani and Prabhakar Raghavan. *Randomized algorithms*. Chapman & Hall/CRC, 2010.
18. George Porter, Richard Strong, Nathan Farrington, Alex Forencich, Pang Chen-Sun, Tajana Rosing, Yeshaiahu Fainman, George Papen, and Amin Vahdat. *Integrating microsecond circuit switching into the data center*, volume 43. ACM, 2013.
19. Vadim G Vizing. On an estimate of the chromatic class of a p-graph. *Diskret. Analiz*, 3(7):25–30, 1964.
20. Guohui Wang, David G Andersen, Michael Kaminsky, Konstantina Papagiannaki, TS Ng, Michael Kozuch, and Michael Ryan. c-through: Part-time optics in data centers. In *ACM SIGCOMM Computer Communication Review*, volume 40, pages 327–338. ACM, 2010.

A Hardness of Matching Decomposition

Based on the hardness of GBEC, we finally show the hardness of minimum matching decomposition, even when the optimal decomposition has a constant number of permutation matrices in the support, proving Theorem 1.

We will first show NP-hardness of minimum matching decomposition when the matrix A is non-negative, but not doubly stochastic. In this case, the problem can be formulated as follows: given a bipartite graph $G = (V, E)$ and a weight vector $x \in (\mathbb{R}^+)^E$, find a decomposition $x = \lambda_1 M_1 + \dots \lambda_k M_k$ with minimum k where each λ_i is non-negative and each M_i is an indicator vector of a matching in G. At the end of the section, we will complete the proof of Theorem 1 by showing that the same hardness holds even when A is promised to be doubly stochastic.

Let $G = (V, E)$, $k \in \mathbb{N}$, and $\{S_e\}_{e \in E}$ be an instance of GBEC. We will construct the graph $G' = (V', E')$ with edge weights $X = \{x_e\}_{e \in E'} \in [0,1]^{E'}$, and an integer k' such that the optimal value of the GBEC instance is 1 if and only if X can be written as

$$X = \lambda_1 M_1 + \cdots + \lambda_{k'} M_{k'},$$

where M_i is the indicator vector of a matching in G' and $\lambda_i \geq 0$. In particular, we do not require each M_i to be a perfect matching.

Arbitrarily order S_e's to $S_1, \dots, S_T \subseteq 2^{[k]}$ so that (1) they are pairwise distinct, and (2) for any $e \in E$, $S_e = S_i$ for some $i \in [T]$. Call such i as the *type* of e, denoted by $t(e)$. There are $T \leq 2^{2^k}$ types. Also, let $R := \sum_{i=1}^{T} |S_i|$. Since each S_i is a collection of subsets of $[k]$, R is the number of type-subset pairs (S_i, s) such that $s \subseteq [k]$ and $s \in S_i$. Order such R type-subset pairs arbitrarily. Let $l \in \mathbb{N}$ be such that $10^l > \max(2k, R)$.

Let $k' := R + k$. The reduced instance will have the optimum at least k' by design. If the instance of GBEC has the optimum 1, the reduced instance of minimum matching decomposition will have the optimum exactly k'. Our instance also will *reveal* $\lambda_1, \dots, \lambda_{k'}$ in a sense that if there exists a decomposition into k' matchings, their weights must be $\lambda_1, \dots, \lambda_{k'}$. They are defined as follows.

- For $R < i \leq k'$, $\lambda_i := 10^{-l(i-R+T)}$. They represent k colors of GBEC.
- For $1 \leq j \leq [R]$, consider the jth type-subset pair (S_i, s) where $i \in [T]$, $s \subseteq [k]$, and $s \in S_i$. Let $\lambda_j := 10^{-li} - (\sum_{p \in s} 10^{-l(T+p)}) + 10^{-l(T+k+1)}$.

The reduction is the following.

- The bipartite graph $G' = (V', E')$ is the disjoint union of G and $K_{k',k'}$.
- Assume that the left (resp. right) vertices of $K_{k',k'}$ are indexed by $\{u_0, \dots, u_{k'-1}\}$ (resp. $\{v_0, \dots, v_{k'-1}\}$). Each edge $e = (u_i, v_j)$ gets $x_e \leftarrow \lambda_{((j-i) \mod k')+1}$.
  - Since $K_{k',k'}$ is k'-regular, the optimum must be at least k'. Since each vertex is incident to edges of weights $\lambda_1, \dots, \lambda_{k'}$, if a decomposition into k' matchings exists, the weights must be $\lambda_1, \dots, \lambda_{k'}$.
- For each edge $e \in E$ with its type $t(e) \in [T]$, let $x_e := 10^{-lt(e)} + 10^{-l(T+k+1)}$.

The following completeness and soundness lemmas prove the NP-hardness of minimum matching decomposition.

Lemma 8 (Completeness). *If the instance of* GBEC *has the optimum* 1*, x admits a decomposition into k' matchings.*

Proof. The $K_{k',k'}$ part is easily decomposed into k' matchings as $\sum_{i=1}^{k'} \lambda_i M_i$ where for each $j \in \{0, \ldots, k'-1\}$, M_i matches u_j to $v_{j+(i-1) \mod k'}$. Let $\{s_e^*\}_{e \in E}$ be a solution of GBEC such that $s_e^* \in S_e$ for each e, and for any adjacent $e, f \in E$, s_e^* and s_f^* are disjoint. For each edge $e \in E$, let $i := t(e)$ and $j \in [R]$ be such that the jth type-subset pair is (S_i, s_e^*). Add e to M_j and M_{p+R} for all $s \in s_e^*$. Since

$$\lambda_j + \sum_{p \in s_e^*} \lambda_{p+R} = \left(10^{-li} - (\sum_{p \in s_e^*} 10^{-l(T+p)}) + 10^{-l(T+k+1)}\right) + \sum_{p \in s_e^*} 10^{-l(T+p)}$$
$$= 10^{-li} + 10^{-l(T+k+1)} = x_e,$$

every edge is correctly decomposed. It remains to show that each M_i is a matching. If two adjacent edges e and f are both contained in M_j for some $j \in [k']$,

- If $j \in \{R+1, \ldots, k'\}$, it means that s_e^* and s_f^* intersect, leading to contradiction.
- If $j \in [R]$, it means that $t(e) = t(f)$ and $s_e^* = s_f^*$, again leading to contradiction.

Therefore, $x = \sum_{i=1}^{k'} \lambda_i M_i$ where each M_i is a matching. □

Lemma 9 (Soundness). *If x admits a decomposition into k' matchings, the instance of* GBEC *has the optimum* 1.

Proof. We saw that if x admits a decomposition into k' matchings, the weights must be $\lambda_1, \ldots, \lambda_{k'}$. Suppose $x = \sum_{i=1}^{k'} \lambda_i M_i$ for some matchings M_i. Fix an edge $e \in E$. Let $U \subseteq [k']$ be the set of the indices that e is assigned to. It implies that

$$x_e = \sum_{i \in U} \lambda_i \tag{7}$$

Proposition 1. $|U \cap [R]| = 1$.

Proof. Note that every x_e is of the form $c \cdot 10^{-l(T+k)} + 10^{-l(T+k+1)}$ for some integer c. The same is true for $\lambda_1, \ldots, \lambda_R$. On the other hand, $\lambda_{R+1}, \ldots, \lambda_{k'}$ are of the form $c \cdot 10^{-l(T+k)}$ for some integer c. We have

$$\sum_{j \in U} \lambda_j = c \cdot 10^{-l(T+k)} + |U \cap [R]| \cdot 10^{-l(T+k+1)} \text{ for some integer } c.$$

Since $10^l > R$, $x_e = \sum_{j \in U} \lambda_j$ implies that $|U \cap [R]| = 1$. □

Proposition 2. *$U \cap [R] = \{j\}$, where the jth type-subset pair is $(S_{t(e)}, s)$ for some $s \in S_{t(e)}$.*

Proof. We saw that $|U \cap [R]| = 1$. Let j be such that $U \cap [R] = \{j\}$. Let the jth pair be (S_i, s) for some $i \in [T]$ and $s \in S_i$. Recall that $x_e = 10^{-lt(e)} + 10^{-l(T+k+1)}$ and $\lambda_j := 10^{-li} - (\sum_{p \in s} 10^{-l(T+p)}) + 10^{-l(T+k+1)}$.

- If $i < t(e)$, $\lambda_j > x_e$, contradicting (7).
- If $i > t(e)$, $x_e - \lambda_j = 10^{-lt(e)} - 10^{-li} + (\sum_{p \in s} 10^{-l(T+p)}) \geq \frac{10^{-lt(e)}}{2} \geq \frac{10^{-lT}}{2}$. For all $j' \in U \setminus \{j\}$, $j' > R$ and $\lambda_j \leq 10^{-l(T+1)}$. Since $10^l > 2k$, $x_e - \lambda_j > \sum_{j' \in U \setminus \{j\}} \lambda'_j$, contradicting (7).

Therefore, $i = t(e)$ as desired. □

Finally, we show that $U \setminus \{j\}$ is a translation of s. Recall that for $s \subseteq [k]$ and $R \in \mathbb{N}$, $s + R := \{p + R : p \in s\}$.

Proposition 3. *$s + R = U \setminus \{j\}$.*

Proof. We saw that $U \cap [R] = \{j\}$, and the jth pair is $(S_{t(e)}, s)$ for some $s \in S_{t(e)}$. By (7),

$$x_e - \lambda_j = \sum_{p \in s} 10^{-l(T+p)} = \sum_{u \in U \setminus \{j\}} \lambda_u = \sum_{u \in U \setminus \{j\}} 10^{-l(u-R+T)}.$$

The claim directly follows by comparing the second and the final expression. □

An optimal solution to GBEC is constructed by computing above s for every $e \in E$ and assigning s to e. This solution gives value 1 because for every adjacent edges $e, f \in E$, the corresponding U's (thus s's) are disjoint. □

This establishes the NP-hardness of minimum matching decomposition for a matrix A that is non-negative but not necessarily doubly stochastic. We complete the proof of Theorem 1 by showing that the same hardness holds even when A is doubly stochastic.

Proof. (Theorem 1) Let a bipartite graph $G = (U \cup V, E)$ with edge weights $x \in (\mathbb{R}^+)^E$ be the hard instance of minimum matching decomposition constructed above such that it is NP-hard to determine whether x can be decomposed into a non-negative linear combination of k integral matchings (i.e., ignore the GBEC instance and let $G \leftarrow G'$ and $k \leftarrow k'$ from above).

Let $A(G) \in \mathbb{R}^{U \times V}$ be the bipartite adjacency matrix such that $A[u, v] = x_{(u,v)}$ for $(u, v) \in E$ and $A[u, v] = 0$ for $(u, v) \notin E$. For each vertex $v \in U \cup V$, let $\deg(v) := \sum_{e \in E : e \ni v} x_e$. Our construction above fixes k values $\lambda_1, \ldots, \lambda_k \geq 0$ such that if x can be decomposed into k integral matchings, it has to be of the form $x = \sum_{i=1}^{k} \lambda_i M_i$ for some matchings $M_1, \ldots, M_k$. One can scale x without changing the optimal value, so assume without loss of generality that $\sum_{i=1}^{k} \lambda_i = 1$. This ensure that $\deg(v) \leq 1$ for every $v \in V$.

From G and x, construct $H = (U_1 \cup V_1 \cup U_2 \cup V_2, E')$ and weights $x' \in (\mathbb{R}^+ \cup \{0\})^{E'}$ as follows:

1. U_1, U_2 are copies of U and V_1, V_2 are copies of V. For a vertex $u \in U \cup V$ and $i \in \{1, 2\}$, let u_i be its copy in $U_i \cup V_i$.
2. Copy G between U_i and V_i for $i = 1, 2$. Formally, for every pair $(u, v) \in E$, for $i = 1, 2$, add (u_i, v_i) to E' with $x'_{(u_i, v_i)} = x_{(u,v)}$.

3. For each $v \in U \cup V$, add (v_1, v_2) to E' with $x_{(v_1,v_2)} = 1 - \deg(v)$.

Note that H is bipartite, since $V_1 \cup U_2$ and $V_2 \cup U_1$ do not have any induced edge. By construction, every vertex of H has $\deg(v) = 1$, so $A(H) \in \mathbb{R}^{(V_1 \cup U_2) \times (V_2 \cup U_1)}$ is doubly stochastic. We claim that $A(H)$ admits a matching decomposition of support k if and only if x does, which completes the proof.

If $x = \sum_{i=1}^{k} \lambda_i M_i$ where each M_i is the indicator vector of an integral matching, then $A(H)$ can be represented as $\sum_{i=1}^{k} \lambda_i A(N_i)$, where N_i is a perfect matching in H where

- If (u, v) are matched in M_i, match (u_1, v_1) and (u_2, v_2) in N_i.
- If $v \in U \cup V$ is unmatched in N_i, match (v_1, v_2).

Therefore, $A(H)$ admits a matching decomposition with support k.

For the other direction, let $A(H) = \sum_{i=1}^{k} \mu_i B_i$ for some sub-permutation matrices B_i and $\mu_i \geq 0$. Since H contains a vertex with k incident edges with weights $\lambda_1, \ldots, \lambda_k$, up to reordering $\mu_i = \lambda_i$ for every i. Furthermore, since A is doubly stochastic and $\sum_{i=1}^{k} \lambda_i = 1$, indeed every B_i must be a permutation matrix. Let M_i be the matching of G that contains every edge (u, v) where $B[v_1, u_1] = 1$. Then $x = \sum_{i=1}^{k} \lambda_i M_i$, by considering the submatrix of A whose rows are indexed by V_1 and columns are indexed by U_1. □

B Special Cases

B.1 Lowerbound For BvN Algorithm

Recall that a BvN decomposition of a doubly stochastic matrix can be obtained using the following simple algorithm. To describe the algorithm, it is easier to think of a doubly stochastic matrix as a weighted bipartite graph where for each vertex v the total weight of edges incident is at most 1. Let e be the minimum weight edge in the bipartite graph. Find a perfect matching in the bipartite graph that contains e, and take it as one permutation matrix in the final representation with λ value equal to the minimum weight edge. Now, decrease the weight of every by edge in the matching by w_e. This implies that weight of the at least one edge drops to zero, and we repeat this process.

Theorem 7. *The approximation factor of BvN decomposition algorithm is at least $2^k/k$, where k is the value of optimal solution.*

Proof. Consider the following edge weighted bipartite graph with n vertices on the left-hand side and n vertices on the right hand side. Let $k > 0$ be some positive integer. The vertex $i \in [n]$ has $1, 2, \ldots \log(k) + 1$ edges incident on with it. The jth edge incident on i has weight $1/2^j$ for $j = 1, 2, \ldots \log(k)$. The $(\log(k) + 1)$ edge has weight $1/k$. The jth edge connects vertex i on the left-hand side with the vertex $(i + j) \mod n$ on the right-hand side. It is easy to verify that total weight edges incident on each vertex is at most 1. Moreover, it is easy to verify that the optimum minimum matching decomposition has size $\log(k) + 1$. Consider the following sequence of matchings produced by BvN algorithm. In the first matchings, it picks an edge with weight $1/k$, and the remaining $n - 1$ edges in the matching come from the edges with weight $1/2$. Hence, after removing the first matching, only 1 edge disappears and the weight of the edges decreases by $1/k$. This process repeats at each step and in every iteration each vertex looses $1/k$ weight, and hence it leads to a decomposition of size k. □

Another variant of BvN algorithm finds a matching which maximimzes the edge with minimum weight. A simple extension of the above example shows that this algorithm also has a bad approximation factor. We augment the bipartite graph constructed above with following $\log(k)$ additional sets of vertices. A set $s \in [\log k]$ consists of s-regular bipartite graph with edge weights $1/2^s$. Now it is easy to verify optimal decomposition is still $\log k + 1$, where as BvN algorithm might need $O(k)$.

B.2 Optimal Decomposition For $k = 2$ and $k = 3$

We now show that if a doubly stochastic matrix A can be decomposed using 3 perfect matchings, then we can find such a decomposition in polynomial time.

Theorem 8. *There is a polynomial time algorithm that finds an optimal matching decomposition for $k = 2, 3$.*

Proof. (sketch) We sketch the proof for $k = 3$ as it subsumes the proof for case $k = 2$. As earlier, we assume that we know the values of $\lambda_1, \lambda_2, \lambda_3$. Lets consider various cases.

Case 1: All λ's are equal In this case, we convert bipartite graph A into a 3 regular graph by splitting edges with weight 1 and 2λ and then use the edge coloring result for 3 regular graphs.

Case 2: All λ's are distinct If all λ's are distinct, and sum of $\lambda_1 + \lambda_2 \neq \lambda_3$, then again we can easily solve this case as we can easily read where each edge goes in the matching. So, lets consider the more tricky case where $\lambda_1 + \lambda_2 = \lambda_3$. This implies that $\lambda_3 = 0.5$. Note that the vertices with degree 1 and 3 are easy to handle. So, we focus on vertices with degree 2. Main difficult comes with edges with w_e equal to 0.5. Now, if a vertex of degree two is adjacent to vertex of degree 3 (or 1 but a vertex of degree 1 cannot be adjacent to vertex of degree 2), we are again done. Hence, vertices of degree two form a 2 regular connected graph. Therefore, we can again decompose them easily.

Case 3: $\lambda_1 = \lambda_2$, and $\lambda_1 + \lambda_2 = \lambda_3$ Same as above.

Case 4: $\lambda_1 = \lambda_2$, but $\lambda_1 + \lambda_2 \neq \lambda_3$ As above, we can easily color edges incident on vertices with degree 1 and 3. Therefore, it remains to color edges incident on vertices with degree 2. As argued above, if a vertex of degree 2 is adjcent to vertex of degree 3 then it fixes the color of that edge, and hence other edge incident on the vertex. Thus, vertices of degree 2 form a connected 2 regular graph. This 2-regular graph has the following structure. Consider a vertex with degree 2. Suppose its edges have weight $\lambda_1 + \lambda_2, \lambda_3$. Then, all the vertices in connected component it is in should have same weight structure on their edges. In particular, we cannot have a vertex $\lambda_1, \lambda_2 + \lambda_3$ or $\lambda_1 + \lambda_3, \lambda_2$. Thus, each 2-regular connected component is consistent and hence can be colored using 2-edge coloring.

Case 5: $\lambda_2 = \lambda_3$, but $\lambda_2 + \lambda_3 \neq \lambda_1$ This case cannot happen and the argument is similar to case 4. □

B.3 Approximate Decomposition

For any doubly stochastic matrix A, call A' ϵ-close if $\forall a_{ij} \in A, a'_{ij} \in A', |a_{ij} - a'_{ij}| \leq \epsilon$. That is, entry by entry A and A' differ by at most ϵ. Now we show that A' can be decomposed using at most $1/\epsilon$ matchings.

Theorem 9. *For any doubly stochastic matrix A, there is a ϵ-close matrix A' that has a matching decomposition of size $\frac{1}{\epsilon}$*

Proof. (sketch) The idea is as follows. If any entry in A is less than ϵ then round down to 0. For any other value $a_{ij} \in [k\epsilon, (k+1)\epsilon]$ for some $k \geq 1$, round-down a_{ij} to $k\epsilon$. Clearly, this produces a A' that is ϵ-close. Now, observe that degree of any vertex in A' is at most $1/\epsilon$. Furthermore, every edge is a multiple of ϵ, hence there is a decomposition using $1/\epsilon$ matrices where each $\lambda_i = \epsilon$. The proof can be even extended to perfect matching decomposition. □