Generalized Stochastic Matching

Alireza Farhadi[*]**, Jacob Gilbert**[*]**, MohammadTaghi Hajiaghayi**[*]

University of Maryland
farhadi@cs.umd.edu, jgilber8@umd.edu, hajiagha@cs.umd.edu

Abstract

In this paper, we generalize the recently studied *stochastic matching* problem to more accurately model a significant medical process, kidney exchange, and several other applications. Up until now the *stochastic matching* problem that has been studied was as follows: given a graph $G = (V, E)$, each edge is included in the *realized* sub-graph $\mathcal{G}$ of G mutually independently with probability p_e, and the goal is to find a degree-bounded sub-graph Q of G that has an expected maximum matching that approximates the expected maximum matching of $\mathcal{G}$. This model does not account for possibilities of vertex dropouts, which can be found in several applications, e.g. in kidney exchange when donors or patients opt out of the exchange process as well as in online freelancing and online dating when online profiles are found to be faked. Thus, we will study a more generalized model of *stochastic matching* in which vertices and edges are both *realized* independently with some probabilities p_v, p_e, respectively, which more accurately fits important applications than the previously studied model.

We will discuss the first algorithms and analysis for this generalization of the *stochastic matching* model and prove that they achieve good approximation ratios. In particular, we show that the approximation factor of a natural algorithm for this problem is at least 0.6568 in unweighted graphs, and $1/2 + \epsilon$ in weighted graphs for some constant $\epsilon > 0$. We further improve our result for unweighted graphs to 2/3 using edge degree constrained subgraphs (EDCS).

Introduction

The *stochastic matching* problem has been used to model kidney exchange in several research papers in recent years, and in this paper, we generalize this model to better suit the needs of kidney exchange and other applications. Kidney exchange is an important medical procedure that is utilized to increase the amount of possible successful kidney transplants between patients and donors for hundreds of donor-patient pairs in the U.S. each year. This medical process occurs when an incompatible kidney donor-patient pair matches with another incompatible pair such that the donors are swapped to become compatible pairs with the patients. Unfortunately, compatibility medical testing can require patients and donors to be hospitalized and are expensive. Thus, minimizing the amount of compatibility tests while maximizing compatible exchanges is an important problem in the medical world. Moreover, in this paper we also account for the possibility that a patient or donor may decide to drop out of the exchange due to health conditions or at their discretion at any point throughout the months' long process. Therefore, while prior papers only considered donor-patient compatibility, we will also consider potential dropouts from the exchange process on top of compatibility.

In our proposed *stochastic matching* model of kidney exchange, each donor-patient pair is represented by a vertex in the graph. Edges in this graph G represent donor-patient pairs that may be compatible for exchange. Only a subset of the edges are found to be compatible through medical records and testing, and this subset forms a realized sub-graph $\mathcal{G}$ of possible successful exchanges. We say these edges are *realized*, i.e. appear in $\mathcal{G}$, with some probability p_e. Similarly, a vertex is *realized* with probability p_v if the pair does not dropout during the exchange process. In our generalized model, an edge can only be included in $\mathcal{G}$ if both of its vertices are realized as well. A maximum matching algoritm seeks to pair vertices connected by an edge of the graph together to create the maximum amount of matches. So, a maximum matching of $\mathcal{G}$ represents maximized compatible kidney exchanges. As mentioned, medical tests for compatibility are expensive, and so *querying* the edges of G to see if they were realized should be kept to a minimum. Without knowing the sub-graph $\mathcal{G}$, the goal of the *stochastic matching* problem is to find some degree-bounded sub-graph Q with an expected maximum matching of realized

edges that has a size approximately that of the actual maximum matching of $\mathcal{G}$. We will state and prove the existence of the first bounds for the approximation ratio achieved with this model of kidney exchange in which vertices and edges may be dropped from the original graph.

Additional Applications

With our generalized *stochastic matching*, in addition to kidney exchange we can model the freelancing industry comprised of freelance workers and their potential employers. In modern freelancing, workers may have online profiles on websites that businesses can look through to find freelancers with compatibility for a job or project. Unfortunately, a large amount of fake profiles and fake job offerings plague these websites. Finding out profiles and jobs are fake costs time and money from those who hired the fake profiles or the freelancers who took up a fake job. Therefore, in the online freelancing problem the goal is to maximize matchings between jobs and freelancers while minimizing the amount of queries to freelancers and employers needed to match real freelancer profiles to real job offers.

To model online freelancing with *stochastic matching*, profiles and companies will make up the vertices of some graph G, and there is an edge between profiles and companies if a freelancer fits the qualifications for a company's job opening. Edges may be weighted by the amount a company will pay a freelancer for the job, or remain unweighted if all jobs are nearly equally valuable. Each vertex is realized with some probability p_v as long as the online profile or company is real. Each edge (u, v) is realized as long as both of the vertices u and v are realized. As in kidney exchange, *queries* of edges/vertices are expensive since they require profile reviews and lengthy communications, but in this version of the problem only vertices have a realization probability while edges are always realized if both of its vertices have been realized. Online dating is a very similar scenario with potential fake vertices, but edges between dating profiles may also drop out if a match does not lead to a relationship. In our results later in the paper, we will state and prove bounds to the approximation ratio achieved for weighted graphs for the freelancing and dating model.

Besides the aforementioned applications, the problem is significant from a computer science theory perspective as a discussion of graph sparsification. We will show that a simple, well-studied algorithm provides a sparse sub-graph with a good approximation of the expected maximum matching of the original graph for our generalized version of stochastic matching. This sub-graph will conform to a tight restriction: any vertex has at most constant $O(1)$ degree.

Generalized Stochastic Matching Model

As discussed, the kidney exchange problem may be modeled with our proposed generalized *stochastic matching* model. In the stochastic setting, we have a random sub-graph, the *realized* sub-graph, of some given graph, and we want to approximate some property of the realized sub-graph.

Definition 1. *Given fixed parameters $p_v, p_e \in (0, 1]$ and weighted or unweighted graph $G = (V, E)$ with vertex set*

V *and edge set $E \subset V^2$, let graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a sub-graph of G such that any vertex $v \in V$ is in $\mathcal{V}$ mutually independently randomly with probability p_v and any edge $e = (u, v) \in E$ is in $\mathcal{E}$ mutually independently randomly with probability p_e if $u, v \in \mathcal{V}$. We call $\mathcal{G}$ the realized sub-graph of G.*

Definition 2. *Given weighted graph $G = (V, E, W)$ where W is a set of edge weights, let $w_e \in W$ be the weight of edge $e \in E$. Define $M(G)$ to be the maximum weighted matching of G; furthermore, let $\mu(G) := \sum_{e \in M(G)} w_e$ be the weight of the the maximum matching of G.*

In the *stochastic matching* problem, we want to find a sparse sub-graph of a given graph such that the realized portion of the sparse sub-graph approximates the maximum weighted matching of the realized sub-graph. More formally, given a graph G with n vertices, we want to find a sub-graph $Q = (V, E_Q)$ that satisfies the following two conditions:

1. Let $\mathcal{Q} = Q \cap \mathcal{G}$, then the approximation ratio $\mathbb{E}[\mu(\mathcal{Q})]/\mathbb{E}[\mu(\mathcal{G})]$ is as large as possible.
2. The degree of Q is $O(1)$. Specifically, the maximum degree of any vertex in Q may be bounded by a constant determined by p_v, p_e but not n.

So, if we can find such a sub-graph Q, then we may query the $O(n)$ edges of Q instead of doing expensive queries to all $O(n^2)$ edges of G to find out which were realized. However, finding such a sparse sub-graph and proving it has a large approximation ratio is non-trivial.

Related Work

The less generalized version of *stochastic matching* in which all vertices are realized with probability 1 was first introduced by (Blum et al. 2015) primarily to model the kidney exchange setting. In this paper, the authors showed positive empirical results on simulated and real data from the United Network for Organ Sharing in which stochastic matching algorithms resulted in a good approximation of the optimal solution. This problem has been extensively studied since then (Assadi, Khanna, and Li 2016; Yamaguchi and Maehara 2018; Behnezhad and Reyhani 2018; Behnezhad et al. 2019a). The first discussion of this less generalized problem by (Blum et al. 2015) achieved an approximation ratio of $(1/2 - \epsilon)$ in unweighted graphs, and then (Assadi, Khanna, and Li 2017) broke the half approximation barrier with an approximation ratio of .5001. This bound was later improved by (Behnezhad et al. 2019b) to .6568 and by (Assadi and Bernstein 2019) to $(2/3 - \epsilon)$. Afterwards, (Behnezhad, Derakhshan, and Hajiaghayi 2020) and (Behnezhad and Derakhshan 2020) both built on the analysis of the algorithm proposed by (Behnezhad et al. 2019b) to further improve approximation ratios for unweighted and weighted graphs to $(1 - \epsilon)$, respectively. We adapt this same algorithm as Algorithm 1 below to fit our model.

Our Results

In the Unweighted Approximations section, we will achieve and prove a .65 approximation ratio for unweighted stochastic matching, i.e. the kidney exchange model. By adapting

Algorithm 1: An algorithm for the generalized stochastic matching problem.

Input: Input weighted graph $G = (V, E)$ and realization probabilities $p_v, p_e \in [0, 1]$.

Parameter: $R := \frac{2000 \log(1/\epsilon) \log(1/(\epsilon p_v^2 p_e))}{\epsilon^4 p_v^2 p_e}$

1: $Q \leftarrow (V, \emptyset)$
2: **for** $r = 1, \ldots, R$ **do**
3:     Construct a sample $\mathcal{G}_r = (\mathcal{V}_r, \mathcal{E}_r)$ of G, where any vertex $v \in V$ appears in $\mathcal{V}_r$ independently with probability p_v, and each edge $e \in E$ connecting vertices u, v appears in $\mathcal{E}_r$ independently with probability p_e if and only if $u, v \in \mathcal{V}_r$.
4:     Add the edges in maximum weighted matching $M(E_r)$ of $\mathcal{G}_r$ to Q.
5: **end for**
6: Query the edges in Q and report the maximum weighted matching of it.

---

the analysis techniques of (Behnezhad et al. 2019b) for our new generalization of stochastic matching, we will prove the following theorem and lower bound for the unweighted case:

Theorem 3. *For unweighted graph G, constant $\epsilon > 0$, vertex and edge realization probabilities $p_v, p_e \in (0, 1]$, there is an algorithm to find an $O_{\epsilon,p}(1)$[1]-degree subgraph Q of G such that $\mathbb{E}[\mu(\mathcal{Q})]/\mathbb{E}[\mu(\mathcal{G})] \geq .6568 - \epsilon$.*

In the Weighted Approximations section, we further consider the weighted stochastic matching problem, i.e. the freelancing model. Here, we prove the following bounds:

Theorem 4. *For weighted graph G, constant $\epsilon > 0$, vertex and edge realization probabilities $p_v, p_e \in (0, 1]$, there is an algorithm to find an $O_{\epsilon,p}(1)$-degree subgraph Q of G such that $\mathbb{E}[\mu(\mathcal{Q})]/\mathbb{E}[\mu(\mathcal{G})] \geq .501 - \epsilon$.*

One important distinction between our generalizations for freelancing and kidney exchange from prior work is that some edges are no longer realized completely independently. Specifically, if a vertex is not realized in our model, then every edge connected to it is also not realized. This dependence sets our model apart from previous *stochastic matching* papers. Thus, our techniques will not utilize independent realizations of certain edges, a property that both (Behnezhad, Derakhshan, and Hajiaghayi 2020) and (Behnezhad and Derakhshan 2020) have relied on before. We also improve our bound for unweighted graphs to $(2/3 - \epsilon)$ in the final section of the Appendix using a different algorithm that uses edge degree constraint sub-graphs (EDCS).

Theorem 5. *For unweighted graph G, constant $\epsilon > 0$, vertex and edge realization probabilities $p_v, p_e \in (0, 1]$, there is an algorithm to find an $O_{\epsilon,p}(1)$-degree subgraph Q of G such that $\mathbb{E}[\mu(\mathcal{Q})]/\mathbb{E}[\mu(\mathcal{G})] \geq 2/3 - \epsilon$.*

While the bound of Theorem 5 dominates that of Theorem 3, we feel that the analysis techniques of the Unweighted Approximations section are of independent interest to those

---

[1]We use $O_{\epsilon,p}(.)$ to hide the dependency on $\text{poly}(\epsilon, p_v, p_e)$.

---

at the end of the Appendix and may be improved over time to achieve approximation ratios closer to 1.

Unweighted Approximations

In this section, given an unweighted graph $G = (V, E)$ with an unknown realized sub-graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and the subgraph $Q = (V, E_Q)$ returned by Algorithm 1, we will bound $\mathbb{E}[\mu(\mathcal{Q})]$, the expected size of the maximum realized matching of Q.

Fractional Matchings

In order to prove this bound, we will find a fractional matching x of $\mathcal{Q}$ that achieves a $.6568 - \epsilon$ approximation ratio. In an integral matching, each vertex can only be matched to one other vertex. Alternatively, one can think of an integral matching as assigning a value of either 1 or 0 to every edge such that no vertex has two incident edges with value 1. A fractional matching x provides more flexibility in analysis than an integral matching since it allows assigning fractional values $x_e \in [0, 1]$ to edge e such that for any vertex, $x_v := \sum_{v \in e} x_e \leq 1$. Once we have our fractional matching and prove that it achieves our target approximation ratio, we will use the following folklore lemma to claim the existence of an integral matching y that achieves the same approximation ratio to complete the proof of Theorem 3. Note that in Lemma 6, given graph $G = (V, E)$ and subset $U \subseteq V$, we use $E(U)$ to refer to the edges of the induced sub-graph on G by U which includes every edge $(u, v) \in E$ such that $u, v \in U$.

Lemma 6. *Let x be a fractional matching, $\epsilon > 0$ be a constant, and $G = (V, E)$ be an edge weighted graph where w_e is the weight of edge $e \in E$. If for all $U \subseteq V$ such that $|U| \leq 1/\epsilon$ it is true that $\sum_{e \in E(U)} x_e \leq \lfloor |U|/2 \rfloor$, then G has an integral matching y such that $\sum_{e \in E} w_e \cdot y_e \geq (1 - \epsilon) \sum_{e \in E} w_e \cdot x_e$.*

Proof of Lemma 6 and further discussion about fractional matchings can be found in (Behnezhad et al. 2019b) in section 2.2. Now we see that to prove Theorem 3, fractional matching x will need to satisfy the requirements of Lemma 6 in addition to achieving the .65 approximation ratio. To create such a matching, we will combine two smaller matchings over two disjoint sets of edges, a set of non-crucial edges and a set of crucial edges. Each edge will be classified as non-crucial or crucial based on the probability that the edge appears in the maximum matching of $\mathcal{G}$.

Definition 7. *For edge $e \in E$, we define $q_e := \Pr(e \in M(\mathcal{G}))$ as the probability that e appears in the maximum weighted matching of realized sub-graph $\mathcal{G}$, and we will refer to q_e as the* matching probability *of edge e[2]. Additionally, for vertex $v \in V$, let $q_v := \sum_{e \ni v} q_e$. For a vertex v and subset $X \subseteq E$, $q(X) := \sum_{e \in X} q_e$ and $q_v^X := \sum_{e:e \in X, v \in e} q_e$.*

Definition 8. *Let threshold $\tau = \frac{\epsilon^3 p_v^2 p_e}{20 \log(1/\epsilon)}$, then edge e is* crucial *if $q_e \geq \tau$ and* non-crucial *if $q_e < \tau$. We will use C*

---

[2]Given a realization, we can assume that edges belong to the maximum weighted matching are unique. These edges can be the edges returned by an arbitrary deterministic algorithm.

to denote the set of crucial edges and N to denote the set of non-crucial edges.

The matchings over non-crucial and crucial edges will be constructed with procedures analyzed below. When creating these procedures, we will have to keep a few things in mind about our new model. First, edges are only realized if their incident vertices are realized, and so they can be thought of as having a realization probability of not just p_e but $p_v^2 p_e$. With this in mind, when we sort edges into non-crucial and crucial sets, we make sure our threshold incorporates this realization probability $p_v^2 p_e$ in Definition 8. Furthermore, note that in Algorithm 1, the product $p_v^2 p_e$ makes an appearance in the number of iterations. One of the main reasons is to easily relate the number of rounds of the algorithm to our matching probability threshold τ.

Another quirk of the new model is that there is a correlation in realization probabilities of adjacent edges that share an incident vertex. In step 2 of the upcoming non-crucial edge procedure, we scale down our fractional matching by a factor of p_v to account for this.

The first procedure we discuss will create a near-optimal matching on the non-crucial edges using the following useful observation.

Observation 9. $\mathbb{E}[\mu(\mathcal{G})] = \sum_{e \in E} w_e q_e.$

Essentially, we will use matching probabilities as the assigned values in our fractional matching because the expected size of the maximum matching of $\mathcal{G}$ is just the sum of edge weights times matching probabilities. Since we don't actually know the matching probabilities exactly, we will assign the value of f_e, defined as the fraction of times edge e appears in a maximum matching of an iteration of Algorithm 1 out of R, the total number of iterations.

Definition 10. *For an edge $e \in E$, let k be the number of times e appeared in a maximum weighted matching of a sampled graph $\mathcal{G}_i$ for $1 \le i \le R$ during Algorithm 1. We define $f_e := \frac{k}{R}$.*

Observe that from the above definition, $\mathbb{E}[f_e] = q_e$, which is what we wanted.

Non-crucial Edge Procedure

Given $Q = (V, E_Q)$ from Algorithm 1 with realized subgraph $\mathcal{Q} = (\mathcal{V}, \mathcal{E}_Q)$. Let $\tilde{x}_e = 0$ for all edges $e \in E$. Then,

1. For any realized edge $e \in \mathcal{E}_Q \cap N$, set $\tilde{x}_e$ to be

$$\min\{f_e/(p_v^2 p_e), 2\tau/(p_v^2 p_e)\}.$$

2. Let s_e be the *scaling-factor* of e where the default value is $s_e = 1$. For each vertex $v \in V$ and edge e incident to v, set s_e

$$s_e = \min\left\{s_e, \max\{q_v^N, \epsilon\}/(p_v \cdot \sum_{e \ni v} \tilde{x}_e)\right\}.$$

Note that this step may be done for each vertex in an arbitrary order.

3. Finally scale down the fractional matching with s_e. So for all edges e, let $x_e := \tilde{x}_e \cdot s_e$.

By definition of a fractional matching, it is required that $x_v \le 1$ for any vertex $v \in V$. So, each vertex can be thought to have a "budget" of size 1. We want to make sure the non-crucial edge procedure leaves some remaining budget for the crucial edges later. To this end, we set the scaling-factors in step 2 to have a factor of $\sum_{e \ni v} \tilde{x}_e$ in the denominator. Moreover, we place a p_v in the denominator because vertices can only be matched if they are first realized. Altogether since $q_v \le p_v$, in expectation the scaling-factor $q_v^N/(p_v \cdot \sum_{e \ni v} \tilde{x}_e)$ should keep x_e below 1. Note that in the actual definition of the scaling-factors we have a max over q_v^N and ϵ, and so, we must make sure it is small enough to stay within the vertex budget even with ϵ. So as discussed, the properties of the following lemma prove an upper bound on the size of the fractional matching per vertex, i.e. some budget remains for crucial edges. Additionally, the first property of Lemma 11 proves that the matching so far satisfies the requirements of Lemma 6, which as mentioned will be applied in our last step of this analysis to go from our fractional matching to an integral matching. The proof of Lemma 11 and other missing proofs of this section are available in the Appendix.

Lemma 11. *Given graph $G = (V, E)$, constant $\epsilon \in (0, 1]$, and fractional matching x from the non-crucial edge procedure:*

1. *$\forall U \subseteq V$ such that $|U| \le 1/\epsilon$, $\sum_{e \in E(U)} x_e \le \epsilon \lfloor |U|/2 \rfloor$.*
2. *$\forall v \in V$, $x_v \le \max\{q_v, \epsilon\}/p_v$.*

Now, we have proven that the non-crucial edge procedure is not only a fractional matching, but leaves some room to grow when we discuss crucial edges. However, as previously discussed, we must now also prove that the current matching is nearly optimal in size. The following definition will be useful to see this.

Definition 12. *For $X \subset E$, define $\varphi(X) := \sum_{e \in X} w_e \cdot q_e$ as the* expected matching weight *of X. Also, for a vertex v and subset $X \subset E$, we define $\varphi_v^X := \sum_{e \in X, v \in e} w_e \cdot q_e$.*

By Observation 9, the expected size of the maximum matching on $\mathcal{G}$ is $\varphi(E)$. So in Lemma 13, we will show that matching x over non-crucial edges of our sparse sub-graph is within a $(1 - \epsilon)$ factor of $\varphi(N)$. To begin, we will bound the size of $\tilde{x}_e$ from the first step of the non-crucial edge procedure.

Lemma 13. $\mathbb{E}\left[\sum_{e \in \mathcal{E}_Q \cap N} w_e \cdot \tilde{x}_e \right] \ge (1 - \epsilon)\varphi(N).$

We then show that the scaling factor is at least $(1 - 5\epsilon)$ with high probability. Therefore, we will know that the scaling factor does not decrease the size of the fractional matching too greatly.

Lemma 14. *For vertex v of realized edge $e \in \mathcal{E}_Q \cap N$ with probability at least $1 - 2\epsilon$, it is true that $\max\{q_v^N, \epsilon\}/(p_v \cdot \sum_{e \ni v} \tilde{x}_e) \ge 1 - 5\epsilon$.*

Proof of Lemma 14. In order to prove this lemma, we will need to relate the numerator of the scaling factor, q_v^N, to the denominator of the scaling factor, $\sum_{e \ni v} \tilde{x}_e$. Let $\tilde{x}_v := \sum_{e \ni v} \tilde{x}_e$. To create this relation, we will analyze f_e since

it is related to q_e and $\tilde{x}_e$. Note from the lemma's statement, we have already specified some incident edge $e = (u, v) \in \mathcal{E}_Q \cap N$. If e is not realized, we know that x_e will just be 0 and does not contribute to the matching. Therefore, moving forward we assume that e is realized. We will use $e_1, e_2, ..., e_k \in E_Q \cap N$ to denote the rest of the edges incident to v, realized or unrealized. Now, let $f_v^N := \sum_{e_i} f_e$; we will begin the proof by showing that f_v^N is a good approximation of q_v^N.

Trivially, since $\mathbb{E}[f_e] = q_e$, $\mathbb{E}[f_v^N] \leq q_v^N$. Intuitively, f_v^N is the fraction representing the number of matchings in which v is matched by some edge e_i in a round of Algorithm 1 divided by $R - 1$ since we do not count the round v is matched by e. So, by Hoeffding's inequality we have

$$\Pr(f_v^N - q_v^N \geq \epsilon^2) \leq \Pr(f_v^N - \mathbb{E}[f_v^N] \geq \epsilon^2)$$
$$\leq exp(-2(R-1)\epsilon^4) \leq \epsilon.$$

So far we have that with probability at least $1 - \epsilon$, $f_v^N - q_v^N \leq \epsilon^2$, and this inequality implies that with probability at least $1 - \epsilon$, $\max\{f_v^N, \epsilon\} - \max\{q_v^N, \epsilon\} \leq \epsilon^2$. Therefore, with probability at least $1 - \epsilon$,

$$\max\{f_v^N, \epsilon\} \leq \max\{q_v^N, \epsilon\} + \epsilon^2 \leq (1 + \epsilon) \max\{q_v^N, \epsilon\}. \tag{1}$$

So, we have shown that f_v^N is close to $(1 + \epsilon)q_v^N$. Note that in step 2 of the non-crucial edge procedure, s_e only decreases when $\tilde{x}_v$ is greater than $\max\{q_v^N, \epsilon\}$. Therefore, we will show that the probability of this happening is small, but we will use f_v^N in place of q_v^N. To do so, we define $X_1, X_2, ..., X_k$ to be the random variables conditioned on edge e already being realized where we set X_i to 0 if e_i is not realized and $(\min\{f_{e_i}/(p_v p_e), 2\tau/(p_v p_e)\})$ if e_i is realized. Note that we conditioned on the realization of edge e to keep the set of events X_i mutually independent. Since edge e is realized, we know that the incident vertex v to each edge e_i must also be realized. Furthermore, the probability that each edge e_i is realized is now only $p_v p_e$ since a factor of p_v has been removed for the incident vertex. Now, we define $X := \sum_{e_i} X_i$. Observe that $X = \tilde{x}_v - p_v \cdot \tilde{x}_e$ since step 1 of the non-crucial edge procedure sets any realized edges e_i to be $\min\{f_{e_i}/(p_v^2 p_e), 2\tau/(p_v^2 p_e)\}$. Additionally by linearity of expectation, $\mathbb{E}[X] = \sum_i \mathbb{E}[X_i] \leq f_v$.

Now we will consider two cases. First, assume $\mathbb{E}[X] \leq \epsilon/(2p_v)$, then by definition of X we also have that $\mathbb{E}[X] \leq \max\{f_v^N, \epsilon\}/(2p_v)$. From here, we can achieve the following probability bound:

$$\Pr(X > \max\{f_v^N, \epsilon\})$$
$$= \Pr(X - \max\{f_v^N, \epsilon\}/2 \geq \max\{f_v^N, \epsilon\}/2)$$
$$\leq \Pr(X - \mathbb{E}[X] \geq \max\{f_v^N, \epsilon\}/2)$$
$$= \Pr\left(X \geq \left(1 + \frac{\max\{f_v^N, \epsilon\}}{2 \cdot \mathbb{E}[X]}\right)\mathbb{E}[X]\right)$$
$$\leq \exp\left(-\frac{\max\{f_v^N, \epsilon\}/2}{6\tau/(p_v p_e)}\right), \text{ by Chernoff bound.}$$
$$\leq \exp\left(-\frac{\epsilon}{12\tau/(p_v p_e)}\right) \leq \exp(-1/\epsilon^2) \leq \epsilon.$$

For the second case, if $\mathbb{E}[X] > \epsilon/(2p_v)$:

$$\Pr(X > (1 + \epsilon) \max\{f_v^N, \epsilon\}/p_v) \leq \Pr(X > (1 + \epsilon)\mathbb{E}[X])$$
$$\leq \exp\left(-\frac{\epsilon^2 \mathbb{E}[X]}{6\tau/(p_v p_e)}\right), \text{ by Chernoff bound.}$$
$$\leq \exp\left(\frac{\epsilon^3}{12\tau/p_e}\right), \text{ since } \mathbb{E}[X] > \epsilon/(2p_v).$$
$$\leq \exp\left(-\frac{1}{\log(1/\epsilon)}\right) \leq \epsilon.$$

Thus, in either case we know that

$$\Pr(X \geq (1 + \epsilon) \max\{f_v^N, \epsilon\}) \leq \epsilon.$$

Since $X = \tilde{x}_v - \tilde{x}_e$, we also know that with probability at least $1 - \epsilon$,

$$\tilde{x}_v \leq (1 + \epsilon)(\max\{f_v^N, \epsilon\}/p_v) + p_v \cdot \tilde{x}_e$$
$$\leq (1 + \epsilon)(\max\{f_v^N, \epsilon\}/p_v) + \epsilon$$
$$\leq (1 + 2\epsilon)(\max\{f_v^N, \epsilon\}/p_v).$$

Now, we can use Inequality 1 to obtain our inequality to substitute in q_v^N. With probability at least $1 - 2\epsilon$,

$$\tilde{x}_v \leq (1 + 2\epsilon)(\max\{f_v^N, \epsilon\}/p_v)$$
$$\leq (1 + 2\epsilon)(1 + \epsilon)(\max\{q_v^N, \epsilon\}/p_v)$$
$$\leq (1 + 5\epsilon)(\max\{q_v^N, \epsilon\}/p_v).$$

Finally, with probability at least $1 - 2\epsilon$,

$$\max\{q_v^N, \epsilon\}/(\tilde{x}_v \cdot p_v) \geq \frac{1}{1 + 5\epsilon} \geq 1 - 5\epsilon.$$

$\square$

Lemma 15. *Given a non-crucial edge matching x from Procedure 1,*

$$\mathbb{E}\left[\sum_{e \in N} w_e \cdot x_e\right] \geq (1 - 10\epsilon)\varphi(N).$$

Proof. First, note that the non-crucial edge procedure only assigns non-zero values to edges in the realized sparse subgraph $\mathcal{Q}$. So, with this fact,

$$\mathbb{E}\left[\sum_{e \in N} w_e \cdot x_e\right] = \mathbb{E}\left[\sum_{e \in \mathcal{E}_Q \cap N} w_e \cdot x_e\right]$$
$$= \mathbb{E}\left[\sum_{e \in \mathcal{E}_Q \cap N} w_e \cdot s_e \cdot \tilde{x}_e\right].$$

From Lemma 14, we know that for any edge $e \in \mathcal{E}_Q \cap N$, the probability that the scaling factor s_e from steps 2 and 3 of the non-crucial edge procedure is less than $1 - 5\epsilon$ is at most 2ϵ given a vertex $v \in e$. Since there are two incident vertices

for edge e, $s_e \geq 1 - 5\epsilon$ with probability at least $1 - 4\epsilon$, and thus,

$$\mathbb{E}\left[\sum_{e \in \mathcal{E}_Q \cap N} w_e \cdot s_e \cdot \tilde{x}_e\right]$$

$$\geq (1 - 4\epsilon)(1 - 5\epsilon)\mathbb{E}\left[\sum_{e \in \mathcal{E}_Q \cap N} w_e \cdot \tilde{x}_e\right] \quad \text{By Lemma 14.}$$

$$\geq (1 - 9\epsilon)\mathbb{E}\left[\sum_{e \in \mathcal{E}_Q \cap N} w_e \cdot \tilde{x}_e\right]$$

$$\geq (1 - 9\epsilon)(1 - \epsilon)\varphi(N) \quad \text{By Lemma 13.}$$

$$\geq (1 - 10\epsilon)\varphi(N).$$

$\square$

Crucial Edges

Now we will augment the previous fractional matching from the non-crucial edge procedure with a second procedure on crucial edges. In our new stochastic matching model, it is important to emphasize that the realization of non-crucial edges actually gives some information about the realization of crucial edges. In fact, because some non-crucial edges share an incident vertex with crucial edges, the probability of a crucial edge being realized loses a factor of p_v when we know an adjacent non-crucial edge already has been realized as that implies the shared incident vertex has been realized as well. In order to avoid this further complexity to the problem, this crucial edge procedure will assign remaining budget to a crucial edge based on the incident vertices' non-crucial edge budget such that no vertex contributes more than size 1 to the whole matching. Specifically, we will set x_e of edge $e = (u, v)$ for crucial edges equal to approximately $1 - q_v^N$ or $1 - q_u^N$ (with an additional $(1 - \epsilon)$ factor) in the crucial edge procedure. Additionally, we utilize a probability distribution defined below.

Definition 16. *Take any matching M of the realized crucial edges in sub-graph $Q = (V, E_Q)$ from Algorithm 1. Given $\mathcal{G}$, the* appearance probability *of M is the probability that M is the exact set of crucial edges that is in both E_Q and the maximum matching of $\mathcal{G}$.*

Crucial Edge Procedure

1. Draw a matching M_C of $\mathcal{E}_Q \cap C$ based on the appearance probabilities of matchings given $\mathcal{G} \cap C$.
2. For $e = (u, v) \in M_C$, let $x_e = (1 - \epsilon)\min\{1 - q_u^N, 1 - q_v^N\}$.

Trivially, this procedure makes sure that all vertices $v \in V$ do not exceed their budget, i.e. $x_v \leq 1$. So, first, we will prove that this procedure still allows fractional matching x to satisfy the constraints of Lemma 6.

Lemma 17. $\forall U \subseteq V$ with $|U| \leq 1/\epsilon$, $\sum_{e \in E} x_e \leq \lfloor |U|/2 \rfloor$.

Second, we show that the size of the fractional matching after the crucial edge procedure is the size we wanted. The proof of the following lemma largely relies on the optimal

matching on non-crucial edges from the non-crucial edge procedures as well as a bit of algebra involving matching probabilities and the additional value assignments from the crucial edge procedure.

Lemma 18. *Given fractional matching x from the non-crucial and crucial edge procedures, $\mathbb{E}[\sum_{e \in \mathcal{E}_Q} x_e]/\mathbb{E}[\mu(G)] \geq (1 - 2\epsilon)(4\sqrt{2} - 5)$.*

Finally, we may prove our first theorem, Theorem 3.

Proof of Theorem 3. Take fractional matching x on $\mathcal{Q} = (\mathcal{V}, \mathcal{E}_Q)$ as constructed by the non-crucial edge and crucial edge procedures. By Lemma 17, x satisfies the condition for Lemma 6. Thus, by Lemma 6, there exists an integral matching y on $\mathcal{Q}$ with size at least $(1 - \epsilon)$ times that of x. So, we have that

$$\mathbb{E}[\mu(\mathcal{Q})] \geq \mathbb{E}\left[\sum_{e \in \mathcal{E}_Q} w_e \cdot y_e\right]$$

$$\geq (1 - \epsilon)\mathbb{E}\left[\sum_{e \in \mathcal{E}_Q} w_e \cdot x_e\right] \quad \text{By Lemma 6.}$$

$$\geq (1 - 3\epsilon)(4\sqrt{2} - 5)\mathbb{E}[\mu(G)] \quad \text{By Lemma 18.}$$

$$\geq (.6568 - \epsilon_0)\mathbb{E}[\mu(G)]$$

where $\epsilon_0 = 3(.6568)\epsilon$. $\square$

Unfortunately, it can be shown that this bound is tight for our analysis using the non-crucial and crucial edge procedures as there are examples of graphs where the best approximation using these procedures has ratio $(4\sqrt{2} - 5)$ (see (Behnezhad et al. 2019b))

Weighted Approximations

For weighted graphs, we will begin similarly to the unweighted graph analysis. The end goal will be to create a fractional matching of non-crucial edges and crucial edges to prove our desired .501 bound. We have slightly changed and improved the results of this section compared to (Behnezhad et al. 2019b) by working with a smaller constant δ in Definition 20. From this update, the approximation ratio is a bit better for the following weighted case analysis.

Due the new model, we must again work around the correlation between realization probabilities of adjacent edges in our new model. First, the non-crucial edge procedure will create a near-perfect fractional matching x on non-crucial edges as before. However, a new procedure for crucial edges will be used that will not only assign fractional matching values to crucial edges but may also modify the values in the fractional matching given to non-crucial edges. By lowering the previously assigned non-crucial edge matching values, there will be more budget for crucial edges to take. In weighted graphs, it is possible that crucial edges have high weights and significantly contribute to the maximum matching of $\mathcal{G}$. For this reason, increasing the remaining vertex budget accordingly for crucial edges is necessary. So, the following procedure considers matching probabilities and edge weights when assigning fractional matching values to

the crucial edges, and then it modifies the non-crucial fractional matching if needed.

Weighted Crucial Edge Procedure

1. As in the previous crucial edge procedure, draw some matching M_C of $\mathcal{E}_Q \cap C$ according to appearance-probabilities. For vertex v and constant α, define

$$g(v, \alpha) := \frac{\min\{q_v^N, 1 - \alpha\}}{q_v^N} \varphi_v^N.$$

2. For $e = (u, v) \in M_C$, set

$$x_e := (1 - \epsilon) \arg\max_{0 \leq \alpha \leq 1} (g(u, \alpha) + g(v, \alpha) + \alpha \cdot w_e).$$

3. For any vertex v, if $x_v > 1$, scale down fractional matching of incident non-crucial edges until $x_v \leq 1$.

Before analyzing this new procedure, we want to bound $\varphi(N)$ and $\varphi(C)$ so that they are easier to work with. To do so, we utilize the following lemma discussing the relation of $\varphi(C)$ to Algorithm 1's expected matching size. The proof of the lemma and other missing proofs of this section can be found in Appendix.

Lemma 19. *Given sub-graph $Q = (V, E_Q)$ from Algorithm 1, $\mathbb{E}[\mu(Q)] \geq (1 - \epsilon)\varphi(C)$.*

Recall that from Procedure and Lemma 15, there is an expected matching of Q of size at least $(1 - 10\epsilon)\varphi(N)$. From Lemma 19, we know that there is also an expected matching size of Q with size at least $(1 - \epsilon)\varphi(C)$. Thus, if either $\varphi(C)$ or $\varphi(N)$ is at least $.501 \cdot \mathbb{E}[\mu(\mathcal{G})]$, we have our approximation ratio from Theorem 4. We may now focus on when this is not the case. Since $\mathbb{E}[\mu(\mathcal{G})] = \varphi(C) + \varphi(N)$, we will analyze the case when

$$.499 \cdot \mathbb{E}[\mu(\mathcal{G})] \leq \varphi(C), \varphi(N) \leq .501 \cdot \mathbb{E}[\mu(\mathcal{G})].$$

For this analysis, we will utilize the matching constructed by the weighted crucial edge procedure and further classify crucial edges by edge weights to make claims about this matching's expected weight.

Definition 20. *Let $\delta = .09$, then a crucial edge $e = (u, v) \in C$ is* heavy *if $w_e \geq (1 + \delta)(\varphi_u^N + \varphi_v^N)$, and we denote the set of heavy edges with H.*

A crucial edge is semi-heavy *if e is not heavy, $w_e \geq 2(1 + \delta)\varphi_v^N$, and $q_u^N \leq (1 - \delta)$ where $q_v^N \geq q_u^N$. We denote the set of semi-heavy edges with H^*.*

Observe that by definition, the weight of a heavy edge $e = (u, v) \in H$ is larger than the expected fractional matching of adjacent non-crucial edges. As such, from step 2 of the weighted crucial edge procedure, x_e will be maximized when $\alpha = 0$ and $x_e = (1 - \epsilon)$. Furthermore, step 3 will reduce the fractional matching of adjacent non-crucial edges to 0 so that $x_u, x_v \leq 1$. Similarly, for semi-heavy edge $e = (u, v) \in H^*$, by definition the weight of e is greater than the expected non-crucial fractional matching of one of its vertices. Specifically, from step 2 of the procedure we have that $x_e \geq (1 - \epsilon)(1 - q_u^N) \geq (1 - \epsilon)\delta$. With these crucial edge bounds, we can prove that if a graph has enough heavy and semi-heavy edges, then will will reach at least the desired .501 approximation ratio.

Lemma 21. *If $\varphi(H) + \varphi(H*) \geq 0.074\varphi(C)$, then $\mathbb{E}[\mu(Q)]/\mathbb{E}[\mu(\mathcal{G})] \geq .501 - 11\epsilon$.*

Now, we must look at when $\varphi(H) + \varphi(H*) < 0.074\varphi(C)$, i.e. when the heavy and semi-heavy edges do not constitute a large part of the expected maximum matching on crucial edges. We will use C^* to denote the set of crucial edges that are not heavy nor semi-heavy. We will *direct* edges of C^* based on their adjacent non-crucial edges' contribution to the maximum matching of $\mathcal{G}$.

Definition 22. *We will classify edges $e = (u, v) \in C^*$ into the following three types of edges:*

1. *If $\varphi_v^N \geq \varphi_u^N$, direct e towards v.*
2. *If $\varphi_v^N < \varphi_u^N$ and $w_e \leq 2(1 + \delta)\varphi_v^N$, direct e towards v.*
3. *Else, direct e towards u. Note, in this case $\varphi_v^N < \varphi_u^N$ and $w_e > 2(1 + \delta)\varphi_v^N$.*

Note that edge types 1-3 partition C^*. Thus, using these definitions and directed edge types, we may prove properties of C^* and our desired bound for the approximation ratio of sub-graph Q from Algorithm 1.

Theorem 23. *Given sub-graph Q from Algorithm 1, $\mathbb{E}[\mu(Q)]/\mathbb{E}[\mu(\mathcal{G})] \geq .501 - 11\epsilon$.*

Conclusion

We have now proven bounds on the approximation ratio of Algorithm 1 breaking a half-approximation. The natural next step is to improve the ratio up to $(1 - \epsilon)$. For the old model of stochastic matching, (Behnezhad, Derakhshan, and Hajiaghayi 2020) and (Behnezhad and Derakhshan 2020) achieved $(1 - \epsilon)$ approximations using a slightly modified version of the algorithm in this paper for the unweighted and weighted cases, respectively. Unfortunately, differences in the models bar us from adapting their analysis techniques directly.

The main analysis issue is that in the new model, we do not have complete mutual independence of edge realizations. The realization of an edge requires the realization of each of its incident vertices. If we know one edge is realized, then any adjacent edge is not realized independently since we know the shared incident vertex has been realized. Previous analysis techniques relied on the complete independence of edge realization, and it seems to us to be non-trivial to overcome this difference.

References

Assadi, S.; and Bernstein, A. 2019. Towards a Unified Theory of Sparsification for Matching Problems. In Fineman, J. T.; and Mitzenmacher, M., eds., *2nd Symposium on Simplicity in Algorithms, SOSA@SODA 2019, January 8-9, 2019 - San Diego, CA, USA*, volume 69 of *OASICS*, 11:1–11:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Assadi, S.; Khanna, S.; and Li, Y. 2016. The Stochastic Matching Problem with (Very) Few Queries. In Conitzer, V.; Bergemann, D.; and Chen, Y., eds., *Proceedings of the 2016 ACM Conference on Economics and Computation, EC '16, Maastricht, The Netherlands, July 24-28, 2016*, 43–60. ACM.

Assadi, S.; Khanna, S.; and Li, Y. 2017. The Stochastic Matching Problem: Beating Half with a Non-Adaptive Algorithm. In Daskalakis, C.; Babaioff, M.; and Moulin, H., eds., *Proceedings of the 2017 ACM Conference on Economics and Computation, EC '17, Cambridge, MA, USA, June 26-30, 2017*, 99–116. ACM.

Behnezhad, S.; and Derakhshan, M. 2020. Stochastic Weighted Matching: $(1-\epsilon)$ Approximation. *CoRR*, abs/2004.08703.

Behnezhad, S.; Derakhshan, M.; Farhadi, A.; Hajiaghayi, M.; and Reyhani, N. 2019a. Stochastic Matching on Uniformly Sparse Graphs. In Fotakis, D.; and Markakis, E., eds., *Algorithmic Game Theory - 12th International Symposium, SAGT 2019, Athens, Greece, September 30 - October 3, 2019, Proceedings*, volume 11801 of *Lecture Notes in Computer Science*, 357–373. Springer.

Behnezhad, S.; Derakhshan, M.; and Hajiaghayi, M. 2020. Stochastic matching with few queries: $(1-\epsilon)$ approximation. In Makarychev, K.; Makarychev, Y.; Tulsiani, M.; Kamath, G.; and Chuzhoy, J., eds., *Proccedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22-26, 2020*, 1111–1124. ACM.

Behnezhad, S.; Farhadi, A.; Hajiaghayi, M.; and Reyhani, N. 2019b. Stochastic Matching with Few Queries: New Algorithms and Tools. In Chan, T. M., ed., *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, 2855–2874. SIAM.

Behnezhad, S.; and Reyhani, N. 2018. Almost Optimal Stochastic Weighted Matching with Few Queries. In Tardos, É.; Elkind, E.; and Vohra, R., eds., *Proceedings of the 2018 ACM Conference on Economics and Computation, Ithaca, NY, USA, June 18-22, 2018*, 235–249. ACM.

Bernstein, A.; and Stein, C. 2015. Fully Dynamic Matching in Bipartite Graphs. In Halldórsson, M. M.; Iwama, K.; Kobayashi, N.; and Speckmann, B., eds., *Automata, Languages, and Programming - 42nd International Colloquium, ICALP 2015, Kyoto, Japan, July 6-10, 2015, Proceedings, Part I*, volume 9134 of *Lecture Notes in Computer Science*, 167–179. Springer.

Bernstein, A.; and Stein, C. 2016. Faster Fully Dynamic Matchings with Small Approximation Ratios. In Krauthgamer, R., ed., *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, 692–711. SIAM.

Blum, A.; Dickerson, J. P.; Haghtalab, N.; Procaccia, A. D.; Sandholm, T.; and Sharma, A. 2015. Ignorance is Almost Bliss: Near-Optimal Stochastic Matching With Few Queries. In Roughgarden, T.; Feldman, M.; and Schwarz, M., eds., *Proceedings of the Sixteenth ACM Conference on Economics and Computation, EC '15, Portland, OR, USA, June 15-19, 2015*, 325–342. ACM.

Yamaguchi, Y.; and Maehara, T. 2018. Stochastic Packing Integer Programs with Few Queries. In Czumaj, A., ed., *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018*, 293–310. SIAM.