ASIAN JOURNAL OF MATHEMATICS AND APPLICATIONS
Volume 2018, Article ID ama0452, 14 pages
ISSN 2307-7743
http://scienceasia.asia

GREEDY HEURISTICS AND STOCHASTIC MATCHINGS

JOYDEEP MUKHERJEE AND C.R. SUBRAMANIAN*

The Institute of Mathematical Sciences, Taramani, Chennai - 600113, India

*Correspondence: crs@imsc.res.in

Abstract. We analyze the performance of some greedy heuristics for the weighted version of the stochastic matching (under the *probe-and-commit with patience constraints* model) as introduced by Chen et al., [5]. As input a random subgraph H of a given edge-weighted graph $(G = (V, E), \{w_e\}_e)$ (where each edge $e \in E$ is present in H independently with probability p_e) is revealed (on a *probe-and-commit* basis meaning any edge e that is probed and found to be present in H should be included irrevocably in the matching). It is also required that the the number of probes involving a vertex v cannot exceed a nonnegative parameter t_v known as v's patience. All of G, $\{w_e\}_e$, $\{p_e\}_e$ and $\{t_u\}_u$ is revealed to the algorithm before its execution. The performance of the algorithm is measured by the expected weight of the matching it produces. For approximation measures, it is compared with the expected weight of an optimal adaptive algorithm for the input instance.

We analyze a natural greedy algorithm for this problem and obtain an upper bound of $\frac{2}{p_{min}^2}$ on the approximation factor of its performance. Here, p_{min} refers to $\min_{e \in E} p_e$. No previous analysis of any greedy algorithm for the weighted stochastic matching (under the probe-and-commit model) is known. This also improves the previously best known approximation ratio of any efficeint and adaptive algorithm when $p_{min} > \frac{1}{\sqrt{2}}$. We also establish a lower bound of $\frac{2}{p_{min}}$ on the worst-case value of the approximation ratio of the greedy algorithm.

We also analyze a class of greedy heuristics and establish that the approximation ratio of each such heuristic can become arbitrarily large even if we restrict ourselves to unweighted instances.

Key words and phrases. analysis of algorithms, approximation algorithms, combinatorial problems.

A preliminary version of this work appears as part of a March 2014 dated technical report "On approximating weighted stochastic matchings" by the authors. It also presents a 5.2-factor approximation approximation algorithm for online weighted bipartite stochastic matching improving the previous 7.92-factor approximation result of [4]. The report is available at http://www.imsc.res.in/ crs/smaaim14sub.pdf

1. Introduction

The Greedy heuristic being one of the simplest algorithmic approaches has a unique place in combinatorial optimization. It is always worth looking at its performance and gather to know its power and limitations. In particular, the performance of the Greedy algorithm for computing a large matching under different settings has been studied both for arbitrary graphs (for its worst case perfomance) (see [12], [9]) and as well as for random instances (for its average case performance) (see [6], [7], [3], [8]). In this paper we study the performance of the greedy heuristics on the weighted stochastic matching problem, a natural stochastic variant of the maximum matching problem.

A typical input instance of this problem is a 4-tuple $(G = (V, E), \{t_u\}_{u \in V}, \{p_e\}_{e \in E}, \{w_e\}_{e \in E})$ where $G = (V, E)$ is an weighted graph, each t_u is a nonnegative integer (known as the patience of u). Consider a random spanning subgraph H where each $e \in E$ is present in H independently with probability p_e and where H is revealed on a probe-and-find basis. Our goal is to design an efficient algorithm (possibly adaptive, possibly randomized) to find a matching in H and which works by probing selectively edges of E for their presence in H subject to obeying the following two constraints on probing : (i) commitment : include an edge irrevocably in the matching if it is found to exist after it is probed, (ii) patience : the number of probes involving a vertex cannot exceed its patience. The performance of the algorithm is measured by the expected total weight of the matching it produces. For approximation measures, it is compared with the expected weight of an optimal adaptive algorithm for the input instance. An optimal strategy is one for which the expected weight of the solution it produces its maximum over all adaptive strategies. We use interchangeably the terms adaptive algorithm and strategy. Note that all edges of G need not be probed and hence all edges of H may not be discovered by the algorithm.

The *unweighted* stochastic matching problem (with probing commitments) models some practical optimization problems like maximizing the expected number of kidney transplants in the kidney exchange program (see [5] for details). This problem was introduced by Chen et al., [5] and they analyzed a greedy algorithm to solve it and proved that the greedy algorithm produces a solution of expected size at least a quarter of the expected size of an optimal strategy. This gives us a 4-approximate algorithm. It was also conjectured that greedy algorithm is a 2-approximate algorithm. This was later affirmatively verified by Adamczyk [1].

In this work, we study the offline, weighted version of the stochastic matching problem. In the offline version, the algorithm, after processing the entire input information ($G = (V, E)$, $\{t_u\}_u$, $\{w_e\}_e$ and $\{p_e\}_e$) that is revealed before-hand, can choose any adaptive strategy to probe the edges.

In this paper, we analyze several variants of the greedy approach to solve this problem. In Section 3, we propose and analyze a natural greedy variant which always probes an edge with the highest expected weight it contributes (if probed) and establish that its approximation ratio is at most $\frac{2}{p_{min}^2}$, where $p_{min} = \min\{p_e : e \in E\}$. This affirmatively confirms a claim presented in [5] (without details) that the approximation factor of the greedy algorithm for the weighted version can be unbounded. It also follows that approximation ratio is less than 4 on general weighted graphs if $p_{min} > \frac{1}{\sqrt{2}}$. Since this variant selects edges for probing based on their individual expected contribution, it can be thought of as being greedy edge-wise and denote it by **GRD-EW** Our result is the first analysis of a greedy heuristic for stochastic matching on weighted graphs. The precise statement of our result is as follows.

Theorem 1. **GRD-EW** *is a* $\frac{2}{p_{min}^2}$*-approximate algorithm for the weighted stochastic matching problem.*

We also show that the inverse dependence on p_{min} cannot be completely eliminated by a more careful analysis even if we allow every vertex to probe all edges incident at it (that is $t_u \geq d_u$ for every u). Thus, we obtain a lower bound on the worst-case approximation ratio of **GRD-EW** for the weighted case. This is stated in the following lemma whose proof is provided in Section 3.

Lemma 1. *There exists an infinite and explicit family* $\{(G_n, t_n)\}_n$ *of weighted input instances (with unlimited patience values) such that the expected weight of the solution produced by* **GRD-EW** *is smaller than the expected weight of an optimal strategy by a multiplicative factor of nearly* $\frac{2}{p_{min}}$.

Since the algorithm works by probing edges, we model the execution of an algorithm as a full binary decision tree as in [5, 1]. [1] presents a very careful analysis of the decision tree to prove that the greedy algorithm is a 2-approximate algorithm for the unweighted version. Our analysis is inspired by the analysis of [1] and we borrow some of the notions and notations from this work. However, ours is not a straighforward generalization to the weighted version and some non-trivial issues (arising for the more general weighted case) have to be handled while analyzing the greedy heuristic.

In Section 4, we propose a simple variant **GRD-VW** of the greedy approach which can be thought of as being greedy vertex-wise. Here we define a notion of revenue m_u associated with a vertex u. For a given set S of l edges incident at a vertex u, an *optimal ordering* of S is any linear ordering σ over S such that if members of S are probed consecutively as per σ, then the expected contribution $E_{S,\sigma}$ from these probings maximized. It can be verified that an optimal ordering is any ordering obtained by sorting the edges in decreasing order of their weights. For a vertex u, let m_u denote the expected contribution one obtains by probing edges of S_u in an optimal order. Here, S_u is the set of t_u edges incident at u having the k largest expected contributions $w_e p_e$. The **GRD-VW** proceeds by choosing that vertex u for which the revenue m_u is maximized and then probes edges in S_u an optimal order and decreases the tolerances appropriately after each probe. We prove that the worst-case approximation ratio of **GRD-VW** can be unbounded even if we restrict ourselves to the unweigted instances (the case for which **GRD-EW** is a 2-approximation algorithm). Formally stated, we have the following result which is proved in Section 4.

Lemma 2. *There exists an infinite and explicit family* $\{(G_n, t_n)\}_{n\geq 1}$ *of unweighted input instances such that the expected size of the solution obtained by* **GRD-VW** (G_n, t_n) *is smaller than that of an optimal strategy by a multiplicative factor of nearly* $\Omega\left(\frac{1}{p_{max}}\right)$ *where* $p_{max} = \max_e p_e$.

The edge-wise and vertex-wise greedy heuristics **GRD-EW** and **GRD-VW** analyzed in Sections 3 and 4 can both be thought of as special cases of a more generalized notion of a greedy heuristic. Fix any function $k : \mathcal{N} \to \mathcal{N}$ satisfying $k(n) \leq n$ for every n. We define a variant for every fixed choice of k and denote the variant by $GRD_k(G, w, p, t)$ or shortly $GRD_k(G, t)$ if w and p are clear from the context. $GRD_k(,)$ is exactly the same as the vertex-wise variant **GRD-VW** but differs only in the definition of m_v, more precisely, in that m_v is the expected contribution one obtains by probing consecutively $\min\{k(|V|), t_v\}$ heaviest available edges incident at v, with the edges being probed in decreasing order of their weights. When $k(n) = n$ for every n, we obtain that $GRD_k()$ is the same as **GRD-VW** . When $k(n) = 1$ for every n, we obtain that $GRD_k()$ is the same as **GRD-EW** described in Section 3. The following lemma establishes that $GRD_k()$ also has unbounded worst-case approximation ratio for any fixed $k = k(n)$ such that $k \to \infty$ as $n \to \infty$ even if restricted to unweighted instances. The proof is presented in Section 5.

Lemma 3. *For any* $k = k(n)$ *such that* (*i*) $k \leq n$, (*ii*) k *divides* n *and* (*iii*) $k \to \infty$ *and for every sufficiently small* $\epsilon > 0$, *there exists an infinite and explicit family* $\{(G_n, t_n)\}_{n\geq 1}$ *of unweighted input instances such that the expected size of the solution obtained by* $GRD_k(G_n, t_n)$ *is smaller than that of an optimal strategy by a multiplicative factor which is nearly* $\Theta(k^{1-\epsilon})$.

1.1. **Motivation.** Kidney transplantation is one of the most effective treatments for kidney failure. Often it is the case that the patient donor pair is medically incompatible. *Kidney exchange* between two such pairs might make the tranplation possible as donor of one incompatible pair might be eligible for the patient in some other incompatible pair and vice versa. We consider the graph whose vertices are incompatible patient donor pairs with an edge between any pair of vertices if mutual exchange is possible. This explains the deterministic structure of the graph given as input. The randomness in the edges comes as a result of some medical test [5], [14], [16] which really decides whether an edge is present or absent. Since a patient-donor pair can only take part in a limited number of medical tests, a patience restriction is imposed. There are several medical tests that decide the presence of an edge. The decisive test viz *crossmatch test* is performed on the basis of the reports of the other tests [5], [14] (which explains the probability associated with an edge) and once the crossmatch succeeds the transplantation takes place. This is the reason why an edge is selected irrevocably when it is found to be present.

1.2. **Related Work.** The approximability of stochastic matching problem (with probing commitments and patience constraints) has been studied in [1], [4], [5]. Generalizations of the stochastic matching problem to stochastic probing (with probing commitments and patience and matroid constraints) have also been studied in [10], [2]. Other models of kidney exchange problem have also been studied. From a mechanism design point of view, the problem has been studied in [14], [16], [15], [17] and see the references therein.

One can view the stochastic matching problem as a generalization of the problem of finding a matching in heterogenuous random graphs (those with independent and different edge probabilities) subject to probing and patience constraints. In a related direction, there is a huge literature [6], [7], [8], [3], [13] and [11] pertaining to greedy (without commitment and patience constraints) analysis of matching in (homogenuous) random graphs. But these works have mostly concentrated on the typical or expected size of the matching produced.

2. Preliminaries

Below, we present some conventions, assumptions and models we will be employing for the rest of this work. Throughout, we consider an instance $I = (G, w, p, t)$ where $G = (V, E)$ is an undirected graph, $w : E \rightarrow \mathcal{R}^+$ is the weight function, $t : V \rightarrow \mathcal{N}$ is the patience function and $p : E \rightarrow [0, 1]$ is the edge probability function. For the sake of simplicity, we often denote this collective input by the short notation (G, t) if the additional inputs $\{p_e\}_e$, $\{w_e\}_e$ can be inferred from the context.

2.1. **Convention : rationalization of patience values.** We assume, without loss of generality, that $t_u \leq d_u$ for every $u \in V$, where d_u is the degree of u in G. Higher values of t_u are not going to lead to better solutions. Throughout the paper, we always enforce this assumption (wherever it becomes necessary), by invoking a subroutine $Rationalize(G, t)$ which, for any vertex u with $t_u > d_u$, redefines t_u to be d_u. Enforcing this assumption helps us to simplify the description of some greedy variants we will study in Sections 4 and 5.

Also, at any point, the current graph contains only those edges joining vertices with positive patience values. This can be ensured by removing edges incident at vertices whose patience has been exhausted.

2.2. **Assumption : normalization of weights.** Since multiplying each edge weight by a common factor c does not really change the outcome (except multiplying its total weight by c) of any algorithm, we can normalize all weights by replacing each w_e by $\frac{w_e}{w_{max}}$ where $w_{max} = \max_e w_e$. This normalization simplifies some of the expressions arising in the analysis. In view of this, from now on, *we assume without loss of generality that* $w_e \leq 1$ *for each* e.

2.3. **Modeling algorithms by decision trees.** Our focus is on algorithms (possibly adaptive, possibly randomized) which are based on probing edges (with a commitment to inclusion) and we analyze such algorithms using the decision tree model employed in the works [5, 1]. The model is described as follows. Any algorithm ALG can be represented by a (possibly) exponential sized full binary tree also denoted by ALG. Each internal node represents either probing an edge or tossing a (biased) coin. The coin tosses capture the randomness (possibly) employed by the algorithm. For deterministic algorithms, each internal node will correspond to only an edge probe. An internal node x probing an edge e will be labeled with e and $w_x = w_e$. An internal node x tossing a coin will be labeled by an empty string and $w_x = 0$. Conider an internal node x. If x involves probing an edge e and if the probe is successful, then the algorithm will proceed further as per the strategy specified by the left subtree of x and if it is unsuccessful, it will proceed as per the right subtree. Similarly, if x correspnds to a coin toss, then the algorithm will proceed further as per the strategy specified by the left (or the right) subtree of x depending on whether the toss is successful or not. However, only internal nodes probing edges can make a positive contribution to the weight of the solution found.

We give a recursive definition of a decision tree : The decision tree ALG corresponding to an algorithm ALG on an instance $I = (G, t)$ (ignoring the specification of w_e's and p_e's which are not going to change through the execution) is a rooted full binary tree T (with root r) where

(1) r is labelled by the emptyset if G is an empty graph having no edges.
(2) r probes an edge $e = \alpha\beta$ if G has at least one edge or r tosses a coin with bias p_r.
(3) left edge out of r is labelled by $p_{\alpha\beta}$ if r probes $\alpha\beta$ or is labelled by p_r if r tosses a coin.
(4) right edge coming out of r is labelled by $1 - p_{\alpha\beta}$ or by $1 - p_r$ depending on the case.
(5) the left subtree of r represents further execution of ALG on on the instance $I_L = (G \setminus \{\alpha, \beta\}, t)$ if r probes $\alpha\beta$. Otherwise, it represents further execution of ALG on I.
(6) the right subtree of r represents further execution of ALG on the instance $I_R = (G \setminus \{\alpha\beta\}, t^{'})$ where $t^{'}_{\alpha} = t_{\alpha} - 1$, $t^{'}_{\beta} = t_{\beta} - 1$ and $t^{'}_{\gamma} = t_{\gamma}$ for all other vertices γ if r probes $\alpha\beta$. Otherwise, it represents further execution of ALG on I.

Without loss of generality, we assume that the root r of an optimal tree OPT always probes an edge.

We make use of the following notations. For any algorithm ALG and any node x in ALG, let q_x denote the probability of reaching x in an execution of $ALG(G, t)$. Also, for a node x representing an edge e, we use w_x to denote the weight w_e. It can be verified that the performance of ALG on (G, t) can be expressed as $\mathbb{E}[ALG] = \sum_{x \in ALG} q_x p_x w_x$ where the summation is over all internal nodes.

3. Greedy heuristic for the weighted version

We focus on the offline version. This means that the input I consisting of the random model $(G = (V, E), \{p_e\}_{e\in E})$ alongwith the additional inputs $(\{w_e\}_{e\in E}, t = \{t_u\}_{u\in V})$ will be revealed to the algorithm before its execution. After some preprocessing, the algorithm can choose to select and probe the edges in any order of its choice. We analyze the following greedy algorithm for the above problem. We use Gr to denote both the greedy algorithm and the corresponding decision tree. Let $\alpha\beta$ be the first edge probed by $Gr(G, t)$. This means that $w_{\alpha\beta}p_{\alpha\beta}$ maximizes w_ep_e over all edges e. We also use OPT to denote any optimal strategy for I and also the associated decision tree. It also denotes the weight of the matching produced by OPT when executed on I.

Algorithm 1 Greedy Algorithm $Gr(G, t)$:

```
$E' \leftarrow E$. $M \leftarrow \emptyset$.
while $E' \neq \emptyset$ do
  Choose an arbitrary edge $e = uv \in E'$ which maximizes $w_ep_e$.
  Probe $e$ and add $e$ to $M$ if $e$ is found to be present.
  If $e \in M$, then set each of $t_u$ and $t_v$ to be zero; else decrement $t_u$ and $t_v$.
  Remove $e$ from $E'$.
  Remove any edge in $E'$ incident at $u$ ($v$) if $t_u$ ($t_v$) equals zero.
  Rationalize$(G, t)$.
endwhile
Output M.
```

To analyze the performance of $Gr(G, t)$, we study the following two algorithms ALG_L and ALG_R introduced and defined as in [1] to work on instances I_L and I_R respectively. By an $\alpha\beta$-probe (α-probe or β-probe) of $OPT(G, t)$, we mean probing edge $\alpha\beta$ (probing edge $\alpha\gamma$ for some $\gamma \neq \beta$ or probing edge $\delta\gamma$ for some $\delta \neq \alpha$).

The algorithm ALG_L mimics the execution of $OPT(G, t)$ except that it replaces each $\alpha\beta$-probe, each α-probe and each β-probe by an appropriate coin toss. That is, whenever there is such a probe (at a node x of $OPT(G, t)$) of an edge e incident at either α or β or both, a coin with bias p_e is tossed. With probability p_e, ALG_L mimics the left subtree of x and with probability $1-p_e$ it mimics the right subtree at x. Obviously, ALG_L is a valid strategy for the instance I_L. If S_L is the random variable denoting the total contribution of the omitted probes in an execution, then it is easy to see that $\mathbb{E}[OPT(G, t)] = \mathbb{E}[ALG_L] + \mathbb{E}[S_L]$. Similarly we define ALG_R. Here the algorithm ALG_R mimics the execution of $OPT(G, t)$ by replacing each $\alpha\beta$-probe, each t_α^{th} α-probe and each t_β^{th} β-probe by flipping a coin of suitable bias. As before it is easy to see that $\mathbb{E}[OPT(G, t)] = \mathbb{E}[ALG_R] + \mathbb{E}[S_R]$ where S_R is a random variable which denotes the total contribution of the probes omitted by ALG_R.

Before proceeding further, we introduce some definitions and notations. We use W_α to denote the contribution that a α-probe (if any) makes to the weight of the solution that $OPT(G,t)$ produces. We use $W_\alpha^{t_\alpha}$ to denote the contribution that a t_α^{th} α-probe (if it happens) makes. W_β and $W_\beta^{t_\beta}$ are similarly defined. We use $O_{\alpha\beta}$ to denote the event that $OPT(G,t)$ probes $\alpha\beta$. We also use $O_{\alpha\gamma}^{t_\alpha}$ ($\gamma \neq \alpha$) to denote the event that $OPT(G,t)$ probes $\alpha\gamma$ in the t_α-th α-probe. It follows that

$$\begin{aligned}\mathbb{E}OPT &= \mathbb{E}ALG_R + \mathbb{E}S_R \\ &= \mathbb{E}ALG_R + \mathbf{Pr}(O_{\alpha\beta})\mathbb{E}(S_R|O_{\alpha\beta}) \\ &+ \mathbf{Pr}(\overline{O_{\alpha\beta}})\left[\mathbb{E}(W_\alpha^{t_\alpha}|\overline{O_{\alpha\beta}}) + \mathbb{E}(W_\beta^{t_\beta}|\overline{O_{\alpha\beta}})\right]\end{aligned} \tag{1}$$

$$\begin{aligned}\mathbb{E}OPT &= \mathbb{E}ALG_L + \mathbb{E}S_L \\ &= \mathbb{E}ALG_L + \mathbf{Pr}(O_{\alpha\beta})\mathbb{E}(S_L|O_{\alpha\beta}) \\ &+ \mathbf{Pr}(\overline{O_{\alpha\beta}})\left[\mathbb{E}(W_\alpha|\overline{O_{\alpha\beta}}) + \mathbb{E}(W_\beta|\overline{O_{\alpha\beta}})\right]\end{aligned} \tag{2}$$

Multiplying (1) by $(1-p_{\alpha\beta})$ and (2) by $p_{\alpha\beta}$ we get

$$\begin{aligned}& \mathbb{E}OPT \\ = \;& p_{\alpha\beta}\mathbb{E}ALG_L + (1-p_{\alpha\beta})\mathbb{E}ALG_R + \mathbf{Pr}(O_{\alpha\beta})\left[p_{\alpha\beta}\mathbb{E}(S_L|O_{\alpha\beta}) + (1-p_{\alpha\beta})p_{\alpha\beta}w_{\alpha\beta}\right] \\ + \;& \mathbf{Pr}(\overline{O_{\alpha\beta}})\left[p_{\alpha\beta}\mathbb{E}(W_\alpha|\overline{O_{\alpha\beta}}) + (1-p_{\alpha\beta})\mathbb{E}(W_\alpha^{t_\alpha}|\overline{O_{\alpha\beta}})\right] \\ + \;& \mathbf{Pr}(\overline{O_{\alpha\beta}})\left[p_{\alpha\beta}\mathbb{E}(W_\beta|\overline{O_{\alpha\beta}}) + (1-p_{\alpha\beta})\mathbb{E}(W_\beta^{t_\beta}|\overline{O_{\alpha\beta}})\right]\end{aligned} \tag{3}$$

Auxilliary Graph J: Recall our assumption that $w_e \leq 1$ for each e. Now, for the sake of the analysis, we define an **auxiliary instance** J which is the same as the original input I except that edge weights are $z_e = 1 - x_e$ where $x_e = p_e w_e$ for each e. Define $p_{min} = \min_{e \in E} p_e$. The following observation plays a role in the lemmas that follow.

Observation 1. *For any edge $e \in E$, $w_e + \frac{z_e}{p_{min}} \leq \frac{1}{p_{min}}$.*

First, we obtain the following lemmas whose proofs are provided later.

Lemma 4.

$$\left(\frac{1-x_{\alpha\beta}}{x_{\alpha\beta}}\right)\mathbb{E}(W_\alpha^{t_\alpha}(I)|\overline{O_{\alpha\beta}}) \;\leq\; \frac{\mathbb{E}(W_\alpha(J)|\overline{O_{\alpha\beta}})}{p_{min}}$$

Lemma 5.

$$p_{\alpha\beta}\mathbb{E}(W_\alpha(I)|\overline{O_{\alpha\beta}}) + (1-p_{\alpha\beta})\mathbb{E}(W_\alpha^{t_\alpha}(I)|\overline{O_{\alpha\beta}}) \;\leq\; \frac{p_{\alpha\beta}}{p_{min}} \tag{4}$$

An analogous inequality involving vertex β also holds.

$$p_{\alpha\beta}\mathbb{E}(W_\beta(I)|\overline{O_{\alpha\beta}}) + (1-p_{\alpha\beta})\mathbb{E}(W_\beta^{t_\beta}(I)|\overline{O_{\alpha\beta}}) \;\leq\; \frac{p_{\alpha\beta}}{p_{min}} \tag{5}$$

We observe that $w_{\alpha\beta}p_{\alpha\beta} \geq p_{min}$. Also we have $\mathbb{E}(S_L|O_{\alpha\beta}) \leq 2 \;\leq\; \frac{2w_{\alpha\beta}p_{\alpha\beta}}{p_{min}}$.

Theorem 2. *The greedy algorithm is a $\frac{2}{p_{min}^2}$-approximation algorithm.*

Proof : We prove the theorem by induction on the number of edges in G. The base cases of induction would be all those graphs G with $\mu(G) \leq 1$ where $\mu(G)$ is the maximum size (= the number of edges) of any matching in G. It is easy to verify that for each of the base cases, $Gr(G,t)$ is itself an optimal strategy. Our inductive hypothesis is that the greedy algorithm is a $\frac{2}{p_{min}^2}$-approximation to the optimal strategy for all graphs on lesser number of edges. Using (3), (4) and (5), we have

$$\begin{aligned}
\mathbb{E}OPT(I) &\leq p_{\alpha\beta}\mathbb{E}ALG_L + (1-p_{\alpha\beta})\mathbb{E}ALG_R \\
&+ \mathbf{Pr}(O_{\alpha\beta})\left[p_{\alpha\beta}\mathbb{E}(S_L|O_{\alpha\beta}) + (1-p_{\alpha\beta})p_{\alpha\beta}w_{\alpha\beta}\right] + \mathbf{Pr}(\overline{O_{\alpha\beta}})\frac{2p_{\alpha\beta}^2 w_{\alpha\beta}}{p_{min}^2} \\
&\leq p_{\alpha\beta}\mathbb{E}ALG_L + (1-p_{\alpha\beta})\mathbb{E}ALG_R \\
&+ \mathbf{Pr}(O_{\alpha\beta})\left[\frac{2p_{\alpha\beta}^2 w_{\alpha\beta}}{p_{min}^2} + \frac{(1-p_{\alpha\beta})p_{\alpha\beta}w_{\alpha\beta}}{p_{min}^2}\right] + \mathbf{Pr}(\overline{O_{\alpha\beta}})\frac{2p_{\alpha\beta}^2 w_{\alpha\beta}}{p_{min}^2} \\
&\leq p_{\alpha\beta}\mathbb{E}ALG_L + (1-p_{\alpha\beta})\mathbb{E}ALG_R + \frac{2p_{\alpha\beta}^2 w_{\alpha\beta}}{p_{min}^2} + \frac{(1-p_{\alpha\beta})p_{\alpha\beta}w_{\alpha\beta}}{p_{min}^2} \\
&\leq p_{\alpha\beta}\mathbb{E}ALG_L + (1-p_{\alpha\beta})\mathbb{E}ALG_R + \frac{2p_{\alpha\beta}w_{\alpha\beta}}{p_{min}^2}
\end{aligned}$$

Using the last inequality and applying the inductive hypothesis to the smaller graphs, it follows that (with $OPT(I_L)$ ($OPT(I_R)$) representing the weight of the matching produced by an optimal strategy for I_L (I_R))

$$\begin{aligned}
\mathbb{E}OPT(I) &\leq p_{\alpha\beta}\mathbb{E}OPT(I_L) + (1-p_{\alpha\beta})\mathbb{E}OPT(I_R) + \frac{2p_{\alpha\beta}w_{\alpha\beta}}{p_{min}^2} \\
&\leq \frac{2p_{\alpha\beta}}{p_{min}^2}\mathbb{E}[Gr(I_L)] + \frac{2(1-p_{\alpha\beta})}{p_{min}^2}\mathbb{E}[Gr(I_R)] + \frac{2p_{\alpha\beta}w_{\alpha\beta}}{p_{min}^2} \\
&\leq \frac{2}{p_{min}^2}\left[p_{\alpha\beta}\mathbb{E}[Gr(I_L)] + (1-p_{\alpha\beta})\mathbb{E}[Gr(I_R)] + p_{\alpha\beta}w_{\alpha\beta}\right] \quad = \quad \frac{2}{p_{min}^2}\mathbb{E}Gr(I)
\end{aligned}$$

It now follows from the recursive definition of the performance of a strategy that the greedy strategy is a $\frac{2}{p_{min}^2}$ approximation to the optimal strategy. ■

It only remains to prove Lemmas 4 and 5 and the proofs are presented below.

3.1. **Proof of Lemma 4.**

$$\begin{aligned}
\frac{1-x_{\alpha\beta}}{x_{\alpha\beta}}\mathbb{E}(W_\alpha^{t_\alpha}(I)|\overline{O_{\alpha\beta}}) &= \sum_{\gamma\neq\beta}\frac{1-x_{\alpha\beta}}{x_{\alpha\beta}}w_{\alpha\gamma}p_{\alpha\gamma}\mathbf{Pr}(O_{\alpha\gamma}^{t_\alpha}|\overline{O_{\alpha\beta}}) \\
&\leq \sum_{\gamma\neq\beta}\frac{1-x_{\alpha\gamma}}{x_{\alpha\gamma}}w_{\alpha\gamma}p_{\alpha\gamma}\mathbf{Pr}(O_{\alpha\gamma}^{t_\alpha}|\overline{O_{\alpha\beta}}) \\
&\leq \frac{1}{p_{min}}\sum_{\gamma\neq\beta}(1-x_{\alpha\gamma})p_{\alpha\gamma}\mathbf{Pr}(O_{\alpha\gamma}^{t_\alpha}|\overline{O_{\alpha\beta}}) \\
&= \frac{\mathbb{E}(W_\alpha^{t_\alpha}(J)|\overline{O_{\alpha\beta}})}{p_{min}} \leq \frac{\mathbb{E}(W_\alpha(J)|\overline{O_{\alpha\beta}})}{p_{min}}
\end{aligned}$$

The first inequality follows since $\frac{1-x}{x}$ is a decreasing function of x in $(0,1]$ and $x_{\alpha\beta}$ is the highest. ■

3.2. **Proof of Lemma 5.** For each $\gamma \neq \beta$, let $E_{\alpha\gamma}$ denote the event that $\alpha\gamma$ is probed and the outcome is successful.

We have

$$\begin{aligned}
& p_{\alpha\beta}\mathbb{E}(W_\alpha(I)|\overline{O_{\alpha\beta}}) + (1-p_{\alpha\beta})\mathbb{E}(W_\alpha^{t_\alpha}(I)|\overline{O_{\alpha\beta}}) \\
\leq\ & p_{\alpha\beta}\mathbb{E}(W_\alpha(I)|\overline{O_{\alpha\beta}}) + (1-x_{\alpha\beta})\mathbb{E}(W_\alpha^{t_\alpha}(I)|\overline{O_{\alpha\beta}}) \\
\leq\ & p_{\alpha\beta}\mathbb{E}(W_\alpha(I)|\overline{O_{\alpha\beta}}) + \frac{x_{\alpha\beta}}{p_{min}}\mathbb{E}(W_\alpha(J)|\overline{O_{\alpha\beta}}) \text{ from Lemma 4} \\
\leq\ & p_{\alpha\beta}\mathbb{E}(W_\alpha(I)|\overline{O_{\alpha\beta}}) + \frac{p_{\alpha\beta}}{p_{min}}\mathbb{E}(W_\alpha(J)|\overline{O_{\alpha\beta}}) \\
=\ & p_{\alpha\beta}\left(\sum_{\gamma\neq\beta}\left(w_{\alpha\gamma}+\frac{1-x_{\alpha\gamma}}{p_{\min}}\right)\mathbf{Pr}\left(E_{\alpha\gamma}|\overline{O_{\alpha\beta}}\right)\right) \\
\leq\ & p_{\alpha\beta}\left(\sum_{\gamma\neq\beta}\frac{\mathbf{Pr}\left(E_{\alpha\gamma}|\overline{O_{\alpha\beta}}\right)}{p_{min}}\right) \leq \frac{p_{\alpha\beta}}{p_{min}}
\end{aligned}$$

The second last inequality follows from Observation 1. ■

3.3. **Proof of Observation 1.** We have

$$w_e + \frac{1-x_e}{p_{min}} \leq \frac{1-x_e+w_ep_{min}}{p_{min}} \leq \frac{1+w_ep_{min}-w_ep_e}{p_{min}} \leq \frac{1}{p_{min}}$$

The last inequality follows as $p_{min} \leq p_e$. ■

3.4. **Proof of Lemma 1.** For each n, let G_n denote the graph $G=(V,E)$ where $V = \{u,v,a_1,\ldots,a_n,b_1,\ldots,b_n\}$ and $E = \{(u,v)\} \cup \{(u,a_i) : 1 \leq i \leq n\} \cup \{(v,b_i) : 1 \leq i \leq n\}$. Let $w_{uv} = W$ and $p_{uv} = 1-\frac{1}{n}$. Let $p = p_{min} = \frac{1}{\sqrt{n}}$ and define W' by $W'p = W(1-1/n)^2$. Let $p_e = p$ and $w_e = W'$ for every $e \neq uv$. Let u and v be both have a patience parameter of $n+1$ and let each of a_i's and b_i's have a patience parameter of 1. The expected weight of the solution produced by the greedy algorithm can be shown to be at most $W(1-\frac{1}{n}) + 2W'(1-(1-p)^n)/n \leq W\left(1-\frac{1}{n}+\frac{2}{\sqrt{n}}\right) = W[1+o(1)]$. Now consider the strategy which first probes each of the n edges (u, a_i) and then probes each of the n edges (v, b_i) and then probes uv. The expected weight of the solution of this strategy is at least $2W'(1-(1-p)^n) = 2W\sqrt{n}[1-o(1)]$. ■

4. A vertex-wise greedy variant

GRD-VW is one variant that naturally comes to one's mind and this also does not possess a good approximation ratio. This variant tries to be greedy vertex-wise. That is, it first computes for each vertex v a value m_v which is computed as follows. Let $\sigma = (e_1, \ldots, e_{t_v})$ be an optimal ordering (sorted in decreasing weights w_e) of the t_v heaviest (in terms of expected individual contributions $w_e p_e$ one obtains if probed) edges incident and available (for probing) at v. m_v denotes the expected contribution one obtains by probing edges as per σ. It can be easily computed using the expression provided below. $\mathbf{GRD-VW}$ then chooses a vertex u for which $m_u = \max_v m_v$ for probing incident edges. Here, t_v and d_v are the current values of v's patience and its degree. It can be verified that $m_v = \sum_{i \leq t_v} w_i p_i \left(\prod_{j<i} 1 - p_j\right)$. A formal description of the algorithm is presented below. As before, the graph contains only edges joining vertices with positive patience values.

Algorithm 2 GRD-VW $MG_r(G,t)$:

```
$E' \leftarrow E$. $M \leftarrow \emptyset$.
while $E' \neq \emptyset$ do
    Choose any vertex $u$ which maximizes $m_v$
    Let $\sigma_u = (e_1, \ldots, e_{t_u})$, $e_j = (uv_j)$, denote an optimal order of edges available for probing.
    $j \leftarrow 1$.
    while $j \leq t_u$ and $t_u > 0$ do
        Probe $e_j$ and add $e_j$ to $M$ if $e_j$ is found to be present.
        If $e_j \in M$, then set each of $t_u$ and $t_{v_j}$ to be zero; else decrement $t_u$ and $t_{v_j}$.
        Remove $e_j$ from $E'$. Increment $j$.
        Remove any edge in $E'$ incident at $u$ ($v_j$) if $t_u$ ($t_{v_j}$) equals zero.
        Rationalize$(G,t)$.
    endwhile
endwhile
Output $M$.
```

The following theorem establishes a lower bound on the worst-case approximation ratio of the greedy variant $MGr(G,t)$ thereby establishing that the approximation ratio can become unbounded even if we restrict ourselves only to unweighted instances. This is in contrast to the edge-wise greedy heuristic which was shown to have an approximation ratio of 2 for unweighted instances.

Lemma 6. *There exists an infinite and explicit family $\{(G_n, t_n)\}_{n\geq 1}$ of unweighted input instances such that the expected size of the solution obtained by $MGr(G_n, t_n)$ is smaller than that of an optimal strategy by a multiplicative factor of nearly $\Omega\left(\frac{1}{p_{max}}\right)$ where $p_{max} = \max_e p_e$.*

Proof of Lemma 6 : For each n, let G_n denote the graph $G = (V, E)$ where

$$V = \{u, a_1, \ldots, a_n, b_1, \ldots, b_n\}; \quad E = \{(u, a_i) : 1 \leq i \leq n\} \cup \{(a_i, b_i) : 1 \leq i \leq n\}.$$

Let $p = p(n)$ be any function such that $p \to 0$ and $p = \omega(\frac{1}{n})$. Define $q = q(n) := \frac{2p}{n}$. Also, let $p_{(u,a_i)} = q$ for each i and $p_{(a_i,b_i)} = p$ for each i. We note that $p_{max} = p$. Let u have a patience parameter of n and let each of a_i's and b_i's have a patience parameter of 1. Consider the strategy which probes each of the n edges (a_i, b_i) and outputs the resulting matching. The expected size of the solution of this strategy is exactly np. Hence the expected size of any optimal strategy is at least np.

We now analyze $MGr(,)$. Notice that

$$m_u = 1 - (1-q)^n = nq - \Theta((nq)^2) = 2p - \Theta(p^2)$$

and $m_{a_i} = m_{b_i} = p$ for each i. Hence $m_u > m_v$ for each $v \neq u$. Without loss of generality, assume that $MGr(,)$ probes edges in the order $(ua_1, \ldots, ua_n)$. Using MGr to denote the size of the solution produced by $MGr(G, t)$, we have

$$\begin{aligned}
E[MGr] &= \sum_{j=0}^{n-1}(1-q)^j q\,(1 + (n-j-1)p) \\
&= 1 - (1-q)^n + \sum_{j=0}^{n-1}(n-j-1)(1-q)^j pq \\
&= 1 - (1-q)^n + pq(1-q)^{n-1}\left(\sum_{j=0}^{n-1} j(1-q)^{-j}\right) \\
&= 1 - (1-q)^n + pq(1-q)^{n-1}\left(\frac{(1-q)^{-1} - n(1-q)^{-n} + (n-1)(1-q)^{-n-1}}{q^2(1-q)^{-2}}\right) \\
&= 1 - (1-q)^n + p\left(\frac{(1-q)^n - n(1-q) + (n-1)}{q}\right) \\
&= 1 - (1-q)^n + p\left(\frac{1 - nq + \Theta((nq)^2) - n + nq + n - 1}{q}\right) \\
&= 2p - \Theta(p^2) + \frac{n}{2}\cdot\Theta(p^2) \;=\; \Theta(np^2)
\end{aligned}$$

Hence the ratio $\frac{E[OPT(G,t)]}{E[MGr]} = \Omega(p^{-1})$ where $p = p_{max}$. This establishes the lemma. ■

5. A generalized greedy variant

Proof of Lemma 3 : For each n, let G_n denote the graph defined in the proof of Lemma 6 with the same patience values and edge probabilities except that we redefine p and q as follows. Define $p = p(n) := \frac{k^\epsilon}{n}$. It follows that $p \to 0$ and $p = \omega(\frac{1}{n})$. Define $q = q(n) := \frac{2p}{k}$.

It follows that $nq \to 0$. As shown before, the expected size of any optimal strategy is at least np.

We now analyze $Gr_k(,)$. Recall our assumption that k divides n. Notice that

$$m_u = 1-(1-q)^k = kq - \Theta((kq)^2) = 2p - \Theta(p^2)$$

and $m_{a_i} = m_{b_i} = p$ for each i. Hence $m_u > m_v$ for each $v \neq u$, as long as u has at least k un-probed edges incident at it and hence $Gr_k()$ will pick k of these edges and probe them consecutively. Since k divides n, this means that $Gr_k(,)$ will probe all edges incident at u and stop with that. Without loss of generality, assume that $Gr_k(,)$ probes edges in the order $(ua_1, \ldots, ua_n)$. Using Gr_k to denote the size of the solution produced by $Gr_k(G,t)$, we have (as shown before)

$$\begin{aligned} E[Gr_k] &= 1-(1-q)^n + p\left(\frac{(1-q)^n - n(1-q) + (n-1)}{q}\right) \\ &= 1-(1-q)^n + p\left(\frac{1-nq+\Theta((nq)^2) - n + nq + n - 1}{q}\right) \\ &= nq - \Theta((nq)^2) + \frac{k}{2}\cdot\Theta(k^{-2+2\epsilon}) \;=\; \Theta(k^{-1+2\epsilon}) \end{aligned}$$

Hence the ratio $\frac{E[OPT(G,t)]}{E[Gr_k]} = \Theta(\frac{np}{k^{-1+2\epsilon}}) = \Theta(k^{1-\epsilon}) \to \infty$ as $n \to \infty$. This establishes the lemma. ■

6. Remarks

We analyzed some variants of greedy heuristic for both weighted and unweighted stochastic matching instances. The following observations are relevant in this context and the last question should be addressed to gather a better comprehension of greedy heuristics.

- For the greedy heuristic $Gr(,)$ applied to weighted instances, the upper and lower bounds on the worst-case approximation ratio still differ by a multiplicative factor of $\frac{1}{p_{min}}$. It would be interesting to reduce this gap and obtain a tight upper bound on the worst-case ratio.
- The assumption that k divides n can be weakened to $n \pmod k = 0$ or $n \pmod k \geq (\frac{1}{2}+\delta)k$ for some fixed $\delta > 0$.
- The multiplicative factor $\Theta(k^{1-\epsilon})$ in the statement of Lemma 3 can be improved to $\Theta(\frac{k}{\omega})$ where $\omega = \omega(n)$ is any sufficiently slow-growing function satisfying $\omega = o(k)$.
- The assumption of $k \to \infty$ in the statement of Lemma 3 can be removed with a corresponding replacement of the term $\Theta(k^{1-\epsilon})$ by a suitable function $f(k)$ (where $f(k) \to \infty$ obviously if $k \to \infty$). This establishes that Gr_k is worse than an optimal strategy by a factor of at least $f(k)$.
- Does there exist (for every fixed $k(n)$), a function $g(k)$ such that $Gr_k(,)$ produces a solution whose expected size is within a multiplicative factor of $g(k)$ from that of an optimal solution (for all instances). In particular, we conjecture that for every

$c \geq 1$, there exists a value $g(c)$ such that $Gr_c(,)$ is a $g(c)$-approximation algorithm for unweighted instances. We know that $g(1) = 2$ from the work of [1].

References

[1] Marek Adamczyk. Improved analysis of the greedy algorithm for stochastic matching. *Information Processing Letters*, 111(15):731–737, 2011.

[2] Marek Adamczyk, Maxim Sviridenko, and Justin Ward. Submodular stochastic probing on matroids. *arXiv preprint arXiv:1310.4415*, 2013.

[3] Jonathan Aronson, Alan Frieze, and Boris G Pittel. Maximum matchings in sparse random graphs: Karp-sipser revisited. *Random Structures and Algorithms*, 12(2):111–177, 1998.

[4] Nikhil Bansal, Anupam Gupta, Jian Li, Julián Mestre, Viswanath Nagarajan, and Atri Rudra. When lp is the cure for your matching woes: Improved bounds for stochastic matchings. *Algorithmica*, 63(4):733–762, 2012.

[5] Ning Chen, Nicole Immorlica, Anna R Karlin, Mohammad Mahdian, and Atri Rudra. Approximating matches made in heaven. In *Automata, Languages and Programming*, pages 266–278. Springer, 2009.

[6] Martin Dyer and Alan Frieze. Randomized greedy matching. *Random Structures & Algorithms*, 2(1):29–45, 1991.

[7] Martin Dyer, Alan Frieze, and Boris Pittel. The average performance of the greedy matching algorithm. *The Annals of Applied Probability*, 3(2):526–552, 1993.

[8] Alan Frieze, AJ Radcliffe, and Stephen Suen. Analysis of a simple greedy matching algorithm on random cubic graphs. In *Proceedings of the fourth annual ACM-SIAM Symposium on Discrete algorithms*, pages 341–351. Society for Industrial and Applied Mathematics, 1993.

[9] David Gale and Lloyd S Shapley. College admissions and the stability of marriage. *American Mathematical Monthly*, 69(1):9–15, 1962.

[10] Anupam Gupta and Viswanath Nagarajan. A stochastic probing problem with applications. In *Integer Programming and Combinatorial Optimization*, pages 205–216. Springer, 2013.

[11] Richard M Karp and Michael Sipser. Maximum matching in sparse random graphs. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 364–375. IEEE, 1981.

[12] Bernhard Korte and Dirk Hausmann. An analysis of the greedy heuristic for independence systems. *Annals of Discrete Mathematics*, 2:65–74, 1978.

[13] Andrew Mastin and Patrick Jaillet. Greedy online bipartite matching on random graphs. *arXiv preprint arXiv:1307.2536*, 2013.

[14] Alvin E Roth, Tayfun Sonmez, and M Utku Unver. Kidney exchange. Technical report, National Bureau of Economic Research, 2003.

[15] Alvin E Roth, Tayfun Sönmez, and M Utku Ünver. Efficient kidney exchange: Coincidence of wants in markets with compatibility-based preferences. *The American economic review*, pages 828–851, 2007.

[16] Alvin E Roth, Tayfun Sönmez, and M Utku Ünver. Pairwise kidney exchange. *Journal of Economic Theory*, 125(2):151–188, 2005.

[17] Özgür Yılmaz. Kidney exchange: An egalitarian mechanism. *Journal of Economic Theory*, 146(2):592–618, 2011.